NATIONAL TECHNICAL UNIVERSITY OF ATHENS (NTUA)
SCHOOL OF CIVIL ENGINEERING
INSTITUTE OF STRUCTURAL ANALYSIS AND SEISMIC RESEARCH

ADVANCED COMPUTATIONAL TECHNIQUES IN STRUCTURAL DYNAMICS

PhD Dissertation

by
George Papazafeiropoulos

Advisor:
Professor Manolis Papadrakakis

December 2021

Final version

Advanced Computational Techniques in Structural Dynamics

A dissertation submitted to the School of Civil Engineering of National Technical University of Athens in partial fulfillment of the requirements for the degree of Doctor of Philosophy

by George Papazafeiropoulos

Advisor:
Professor Manolis Papadrakakis

Athens,
December 2021

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Πολιτικών Μηχανικών
Εργαστήριο Στατικής και Αντισεισμικών Ερευνών

Προηγμένες Υπολογιστικές Τεχνικές στη Δυναμική των Κατασκευών

Η διατριβή αυτή υποβλήθηκε στη Σχολή Πολιτικών Μηχανικών του Εθνικού Μετσοβίου Πολυτεχνείου προς μερική εκπλήρωση των απαιτήσεων για την απόκτηση Διδακτορικού τίτλου σπουδών

από τον Γεώργιο Παπαζαφειρόπουλο

Επιβλέπων:
Καθηγητής Μανόλης Παπαδρακάκης

Αθήνα,
Δεκέμβριος 2021

Dedicated to my beloved ones

and all those people struggling persistently

for a better future of humanity

PhD Examination Committee

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Manolis Papadrakakis
Emeritus Professor
(Advisor)
School of Civil Engineering
National Technical University of Athens

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Yannis F. Dafalias
Emeritus Professor
(Member of the Advisor Committee)
School of Applied Mathematical and Physical Sciences
National Technical University of Athens

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Nikos D. Lagaros
Professor
(Member of the Advisor Committee)
School of Civil Engineering
National Technical University of Athens

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Christoforos Provatidis
Professor
(Member of the Examination Committee)
School of Mechanical Engineering
National Technical University of Athens

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Vagelis Plevris
Professor
(Member of the Examination Committee)
Department of Civil Engineering and Energy Technology
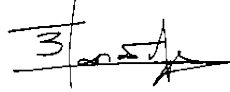OsloMet—Oslo Metropolitan University

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Charmpis Dimos
Associate Professor
(Member of the Examination Committee)
Department of Civil and Environmental Engineering
University of Cyprus

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Vissarion Papadopoulos
Professor
(Member of the Examination Committee)
School of Civil Engineering
National Technical University of Athens

Abstract

In order to properly design new structures or assess the safety and reliability of existing structures and their mechanical components, it is important to study the way in which structures respond to external loads. The vast majority of the external loads that a structure can experience is of dynamic nature: for example earthquakes, external impacts, explosions, vibrations induced by vehicles or machine equipment inside or nearby the structure, etc. The development of efficient dynamic analysis methods over the last decades has stimulated the interest for considering also dynamic response in the formulation of structural design optimization problems. In order to account for these issues, accurate and computationally affordable computational techniques, which include dynamic analysis methods and optimization techniques, are needed.

The goal of the thesis is to develop new computational techniques for the optimum design of structures based on their dynamic response, with emphasis on seismic design, and provide the necessary numerical tools for their implementation. This goal is addressed by developing algorithms for (a) solving the dynamic equilibrium differential equations in the time domain, (b) processing of strong ground motion data for the generation of various elastic and inelastic spectra, (c) optimizing the distribution of the seismic energy absorbed by MDOF shear buildings and (d) optimizing the ground motion acceleration time histories used for the dynamic analysis of structures in the framework of their seismic design. In order to deal with these problems efficiently, various algorithms and methodologies have to be used, such as efficient deterministic and stochastic optimizers, constitutive model formulations for the estimation of the nonlinear dynamic response, and a novel spectra-matching framework which employs a linear combination of raw ground motion records to generate artificial acceleration time histories taking into account both acceleration and seismic input energy equivalent velocity spectra.

The dissertation consists of seven chapters in total, plus Appendix A. It is organized as follows: following the introduction of Chapter 1, Chapter 2 introduces a generalized dynamic time – integration algorithm framework for non-linear structural dynamics. Chapter 3 presents the development of OpenSeismoMatlab, which is an innovative open-source software for strong ground motion data processing, written in MATLAB, and is based on the family of the dynamic time integration algorithms presented in Chapter 2. In Chapter 4 a new optimization concept is introduced which involves the optimization of nonlinear planar shear buildings by using a gradient method based on equivalent linear structures, instead of the traditional practice of calculating the gradients from the nonlinear objective function. Chapter 5 introduces a novel spectra-

matching framework, which employs a linear combination of raw ground motion records to generate artificial acceleration time histories perfectly matched a target spectrum, taking into account not only the acceleration but also the seismic input energy equivalent velocity. The optimization procedures employed in Chapter 5 use solvers that involve the use of OpenSeismoMatlab, among others. Chapter 6 introduces a new integrated optimization framework for engineering applications, Abaqus2Matlab. This is a tool which connects Abaqus, a sophisticated finite element package, with Matlab, the most comprehensive program for mathematical analysis. Using Abaqus2Matlab, an Abaqus analysis can be conducted through Matlab, without interacting with Abaqus/CAE interface, or even Abaqus/Command. Abaqus2Matlab transfers data between Abaqus and Matlab in a form that enables the user to easily manipulate it for further postprocessing, and also in a way that enables the performance of complex types of analyses (e.g. inverse optimization, training artificial neural networks, etc.). Chapter 7 contains the conclusions, the original contribution of the thesis, and directions for future research. Finally, Appendix A is presented, which contains a listing of publications by the author. Each Chapter is accompanied by conclusions and the corresponding bibliography and notation.

Εκτενής Περίληψη

0.1. Εισαγωγή

Ο σχεδιασμός οποιασδήποτε κατασκευής επιβάλλει την ταυτόχρονη ελαχιστοποίηση του κόστους κατασκευής και λειτουργίας της, με παράλληλη βελτιστοποίηση της συμπεριφοράς της έναντι διαφόρων εξωτερικών παραγόντων. Τα ανωτέρω επιτυγχάνονται μέσω χρήσης αλγορίθμων βελτιστοποίησης είτε ντετερμινιστικών, είτε στοχαστικών. Η παρούσα διατριβή αποσκοπεί στην ανάπτυξη κοινοτόμων υπολογιστικών τεχνικών για το βέλτιστο σχεδιασμό των κατασκευών οι οποίες υπόκεινται σε δυναμική φόρτιση, καθώς και των εργαλείων που αυτές απαιτούν. Αυτό επιτυγχάνεται με: (α) ανάπτυξη μεθοδολογιών για την άμεση βήμα προς βήμα εν χρόνω ολοκλήρωση των διαφορικών εξισώσεων δυναμικής ισορροπίας, (β) την ανάπτυξη ενός υπολογιστικού εργαλείου για την επεξεργασία σεισμικών επιταχυνσιογραφημάτων για την παραγωγή συμβατών φασμάτων ελαστικής και ανελαστικής απόκρισης μετατόπισης, ταχύτητας και επιτάχυνσης, καθώς και φασμάτων Fourier και συνήθων δεικτών για τη μέτρηση συγκεκριμένων χαρακτηριστικών των σεισμικών καταγραφών, (γ) ανάπτυξη ενός νέου επαναληπτικού αλγορίθμου βελτιστοποίησης τύπου Newton με δυνατότητες επιπλέον γραμμικής αναζήτησης, ειδικά σχεδιασμένο για γραμμικά ελαστικά και ελαστοπλαστικά διατμητικά κτίρια, ο οποίος βρίσκει τη βέλτιστη κατανομή της δυσκαμψίας και της αντοχής καθ' ύψος του κτιρίου για δεδομένη θεμελιώδη (ελαστική) ιδιοπερίοδο του κτιρίου, ούτως ώστε η σεισμική ενέργεια που αποσβέννεται να είναι σταθερή καθ' ύψος του κτιρίου, (δ) την ανάπτυξη ενός μικτού γενετικού αλγορίθμου με κατάλληλους τελεστές για την ανεύρεση βελτιωμένων επιταχυνσιογραφημάτων για τη δυναμική ανάλυση κατασκευών στο πλαίσιο του αντισεισμικού σχεδιασμού τους, καθιστώντας με αυτό τον τρόπο πιο ρεαλιστικό τον αντισεισμικό σχεδιασμό τους και τέλος (ε) την ανάπτυξη ενός νέου λογισμικού, του Abaqus2Matlab, το οποίο χρησιμοποιείται για την ενοποίηση του κώδκα πεπερασμένων στοιχείων Abaqus με τη γλώσσα προγραμματισμού Matlab, σε διάφορες διαδικασίες βελτιστοποίησης συμπεριλαμβανομένων, αλλά όχι περιοριζόμενων σε, βέλτιστο σχεδιασμό των κατασκευών με βάση τις δυναμικές ιδιότητες και τη δυναμική απόκρισή τους. Το συγκεκριμένο υπολογιστικό περιβάλλον μεταξύ των δυο γνωστών προαναφερθέντων λογισμικών όχι μόνο συνδυάζει το εξελιγμένο γραφικό περιβάλλον και χαρακτηριστικά παρουσίασης δδεδομένων του Matlab, αλλά ανοίγει νέους δρόμους στον τρόπο μετεπεξεργασίας, στατιστικής ανάλυσης και βελτιστοποίησης των αποτελεσμάτων της ανάλυσης πεπερασμένων στοιχείων του Abaqus, και παρέχει επίσης πολλές άλλες δυνατότητες.

0.2. Ένας γενικευμένος αλγόριθμος μη γραμμικής δυναμικής των κατασκευών

0.2.1. Ο γραμμικός γενικευμένος αλγόριθμος εν χρόνω ολοκλήρωσης απλού βήματος απλής λύσης

Η εξίσωση κίνησης ενός γραμμικού μονοβάθμιου ταλαντωτή δίνεται από τη σχέση:

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = f(t) \tag{0.1}$$

με αρχικές συνθήκες:

$$u(0) = u_0, \quad \dot{u}(0) = \dot{u}_0 \tag{0.2}$$

Η εξίσωση (2.10) μπορεί να εφαρμοστεί και σε πολυβάθμια συστήματα, με την προυπόθεση ότι τα τελευταία μπορούν να αναχθούν σε ένα πεπερασμένο αριθμό μονοβάθμιων συστημάτων, χρησιμοποιώντας διάφορες μεθόδους γραμμικής επαλληλίας. Οι Zhou & Tamma (2004) παρουσίασαν μια οικογένεια αλγορίθμων εν χρόνω ολοκλήρωσης απλού βήματος απλής λύσης, ήτοι αλγορίθμων που δεν περιλαμβάνουν πολλαπλασιασμό μητρώων, αλλά μόνο μια επίλυση γραμμικού συστήματος εξισώσεων για καθε χρονικό βήμα. Για τους αλγορίθμους αυτούς ισχύει το θεώρημα Dahlquist (Dahlquist, 1963) το οποίο αναφέρει ότι ενας αλγόριθμος απλού βήματος απλής λύσης ο οποίος είναι απόλυτα ευσταθής (unconditionally stable) μπορεί να έχει ακρίβεια το πολύ δευτέρας τάξης (second order accurate).

Σύμφωνα με τη θεωρία που παρουσίασαν οι Zhou & Tamma, (2004), η εξίσωση (2.10) μπορεί να αναπαρασταθεί ως χρονικά σταθμισμένο υπόλοιπο ως εξής:

$$\int_{t_n}^{t_{n+1}} W\left(M\ddot{u} + C\dot{u} + Ku - f\right)dt \tag{0.3}$$

όπου η στάθμιση του χρόνου υποτίθεται ότι είναι της μορφής

$$W = 1 + w_1\Gamma + w_2\Gamma^2 + w_3\Gamma^3 \tag{0.4}$$

και:

$$\Gamma = \tau/\Delta t, \quad \tau = t - t_n, \quad \Delta t = t_{n+1} - t_n \tag{0.5}$$

Οι εξαρτώμενες μεταβλητές πεδίου (u, $\dot{u}$, $\ddot{u}$) μπορούν να προσεγγιστούν από τα ακόλουθα αναπτύγματα ασυμπτωτικών σειρών:

$$u = u_n + \Lambda_1\dot{u}_n\tau + \Lambda_2\ddot{u}_n\tau^2 + \Lambda_3\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}\tau^3 \tag{0.6}$$

$$\dot{u} = \dot{u}_n + \Lambda_4\ddot{u}_n\tau + \Lambda_5\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}\tau^2 \tag{0.7}$$

$$\ddot{u} = \ddot{u}_n + \Lambda_6\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}\tau \tag{0.8}$$

και το διάνυσμα της φόρτισης αναπτύσσεται σε σειρά πρώτης τάξης μεσω αναπτύγματος Taylor:

$$f = f_n + \frac{f_{n+1} - f_n}{\Delta t}\tau \tag{0.9}$$

Οι τιμές της μετατόπισης και της ταχύτητας στο επόμενο βήμα δίνονται από τις ακόλουθες σχέσεις:

$$u_{n+1} = u_n + \lambda_1\dot{u}_n\Delta t + \lambda_2\ddot{u}_n\Delta t^2 + \lambda_3\left(\ddot{u}_{n+1} - \ddot{u}_n\right)\Delta t^2 \tag{0.10}$$

$$\dot{u}_{n+1} = \dot{u}_n + \lambda_4\ddot{u}_n\Delta t + \lambda_5\left(\ddot{u}_{n+1} - \ddot{u}_n\right)\Delta t \tag{0.11}$$

Η τιμή της επιτάχυνσης στο επόμενο βήμα προκύπτει με αντικατάσταση των εξισώσεων από (2.13) εως (2.20) στην εξίσωση (2.12) ως εξής:

$$\begin{aligned}
\left(\mu_6 M + \mu_5 C\Delta t + \mu_3 K\Delta t^2\right)\ddot{u}_{n+1} = \\
-M\left(\ddot{u}_n - \mu_6\ddot{u}_n\right) \\
-C\left(\dot{u}_n + \mu_4\ddot{u}_n\Delta t - \mu_5\ddot{u}_n\Delta t\right) \\
-K\left(u_n + \mu_1\dot{u}_n\Delta t + \mu_2\ddot{u}_n\Delta t^2 - \mu_3\ddot{u}_n\Delta t^2\right) \\
+\left(1 - W_1\right)f_n + W_1 f_{n+1}
\end{aligned} \tag{0.12}$$

ή σε απλοποιημένη μορφή:

$$\tilde{M}\ddot{u}_{n+1} = \tilde{F}_n \tag{0.13}$$

όπου

$$\begin{aligned}
\tilde{F}_n\left(K, C, f\right) = -M\left(\ddot{u}_n - \mu_6\ddot{u}_n\right) - C\left(\dot{u}_n + \mu_4\ddot{u}_n\Delta t - \mu_5\ddot{u}_n\Delta t\right) \\
-K\left(u_n + \mu_1\dot{u}_n\Delta t + \mu_2\ddot{u}_n\Delta t^2 - \mu_3\ddot{u}_n\Delta t^2\right) + \left(1 - W_1\right)f_n + W_1 f_{n+1}
\end{aligned} \tag{0.14}$$

και

$$\tilde{M}\left(K, C\right) = \mu_6 M + \mu_5 C\Delta t + \mu_3 K\Delta t^2 \tag{0.15}$$

Ο δείκτης της ποσότητας $\tilde{F}_n$ υποδηλώνει το χρονικό βήμα στο οποίο χρησιμοποιούνται οι ποσότητες u, $\dot{u}$, $\ddot{u}$ για τον υπολογισμό του. Οι σταθερές W_i δίνονται από τη σχέση:

$$W_i = \frac{\dfrac{1}{1+i} + \displaystyle\sum_{j=1}^{3}\dfrac{w_j}{1+i+j}}{1 + \displaystyle\sum_{j=1}^{3}\dfrac{w_j}{1+j}}, \quad i = 1, 2, 3 \tag{0.16}$$

Υπάρχουν 12 ανεξάρτητες σταθερές ολοκλήρωσης που απαιτούνται προκειμένου να γίνει εφαρμογή των εξισώσεων (2.21), (2.19) και (2.20) προκειμένου ο αλγόριθμος να προχωρήσει στο επόμενο βήμα. Αυτές είναι οι W_1, μ_1, μ_2, μ_3, μ_4, μ_5, μ_6, λ_1, λ_2, λ_3, λ_4, λ_5. Καθε συνδυασμός των παραπάνω παραμέτρων αντιστοιχεί και σε ένα μοναδικό αλγόριθμο εν χρόνω ολοκλήρωσης και μπορεί να θεωρηθεί κατα κάποιο τρόπο ως η ταυτότητα του αλγορίθμου αυτού. Πολλοί γνωστοί αλγόριθμοι εν χρόνω ολοκλήρωσης, που παρουσιάζονται εντός της διατριβής, προκύπτουν από κατάλληλη επιλογή των ανωτέρω σταθερών ολοκλήρωσης. Στην εργασία των Zhou & Tamma (2004), οι σταθερές ολοκλήρωσης υπολογίζονται με επιβολή διαφορετικών περιορισμών στον αλγόριθμο, σχετικά με την τάξη της ακρίβειάς του, την υπερακόντιση

(overshooting), ιδιάζουσες ρίζες στα κατωτερα και ανώτερα συχνοτικά όρια, τις ιδιότητες σκεδασμού και διασποράς, διακλάδωση των πρωτευουσών ριζών, κλπ. Με τον τρόπο αυτό εξάγονται εννέα διαφορετικοί αλγόριθμοι.

0.2.2. Τροποποίηση του γραμμικού αλγορίθμου για υπολογισμό της μη γραμμικής δυναμικής απόκρισης

Η οικογένεια των αλγορίθμων εν χρόνω ολοκλήρωσης που παρουσιάστηκε παραπάνω τροποποιείται για να λάβει υπόψη της τη μη γραμμική δυναμική απόκριση που προέρχεται από μη γραμμικότητα υλικού. Στη γενική περίπτωση, για να προχωρήσει ο αλγόριθμος από το τρέχον βήμα (u_n , $\dot{u}_n$, $\ddot{u}_n$) στο επόμενο (u_{n+1} , $\dot{u}_{n+1}$, $\ddot{u}_{n+1}$), απαιτούνται τα μητρώα της τέμνουσας δυσκαμψίας και της απόσβεσης, τα οποία συνήθως εξαρτώνται από τα u_{n+1} και $\dot{u}_{n+1}$. Δεδομένου ότι τα τελευταία είναι άγνωστα, τα εν λόγω μητρώα υπολογίζονται με επαναληπτικό τρόπο ούτως ώστε ο αλγόριθμος να συγκλίνει στη λύση. Η σύγκλιση επιτυγχάνεται μεσω μια επαναληπτικής διαδικασίας τύπου Newton-Raphson. Σε μερικούς αλγορίθμους εν χρόνω ολοκλήρωσης η ανωτέρω επαναληπτική διαδικασία αποφεύγεται με τη χρήση των αρχικών εφαπτομενικών μητρώων δυσκαμψίας και απόσβεσης, ωστόσο η προσέγγιση αυτή δεν είναι απόλυτα θεωρητικά σωστή. Το διαγραμμα ροής του μη γραμμικού αλγορίθμου εν χρόνο ολοκλήρωσης απλού βήματος απλής λύσης που αναπτύχθηκε στην παρούσα διατριβή παρουσιάζεται στην εικόνα 0.1.

0.2.3. Αποτελέσματα σχετικά με την αποτελεσματικότητα των μη γραμμικών αλγορίθμων χρονικής ολοκλήρωσης

Συγκρίνονται δεκατρείς διαφορετικοί αλγόριθμοι εν χρόνω ολοκλήρωσης μεσω εφαρμογής τους για την επιλυση έξι προβλημάτων αναφοράς. Οι αλγόριθμοι που συγκρίνονται είναι οι εξής:
- Newmark Average Constant Acceleration (Newmark, 1959) <1>
- Newmark Linear Acceleration (Newmark, 1959) <2>
- Newmark Backward Acceleration (Ascher & Petzold, 1998) <3>
- Fox-Goodwin (Fox & Goodwin, 1949) <4>
- U0-V1-Opt <5>
- U0-V1-CA <6>
- U0-V1-DA <7>
- U0-V0-Opt <8>
- U0-V0-CA <9>
- U0-V0-DA <10>
- U1-V0-Opt <11>
- U1-V0-CA <12>
- U1-V0-DA <13>

Ο δείκτης που υπάρχει κατω από τα σύμβολα U (μετατόπιση) και V (ταχύτητα) δίνει την τάξη υπερακόντισης του καθε αλγορίθμου ως προς τη μετατόπιση και την ταχύτητα αντίστοιχα, ενώ η κατάληξη «-Opt» υποδηλώνει ότι ο αλγόριθμος διαθέτει βέλτιστο αριθμητικό σκεδασμό και διασπορά, και οι καταλήξεις «-CA» και «-DA» υποδηλώνουν αντιστοιχα συνεχή και ασυνεχή μεταβολή της επιτάχυνσης σε καθε χρονικό βήμα.

Τα έξι προβλήματα αναφοράς περιλαμβάνουν τη δυναμική ανάλυση μη γραμμικών μονοβάθμιων συστημάτων με διάφορα καταστατικά προσομοιώματα και τύπους απόσβεσης και είναι τα εξής:

- Μονοβάθμιος ταλαντωτής χωρίς απόσβεση με σκλήρυνση
- Μονοβάθμιος ταλαντωτής χωρίς απόσβεση με κράτυνση
- Μονοβάθμιος γραμμικά ελαστικός ταλαντωτής χωρίς απόσβεση
- Μονοβάθμιος γραμμικά ελαστικός ταλαντωτής με ιξωδοελαστική απόσβεση
- Μονοβάθμιος γραμμικά ελαστικός ταλαντωτής με απόσβεση Coulomb
- Μονοβάθμιος γραμμικά ελαστικός ταλαντωτής με υστερητική απόσβεση

Στόχος των παραπάνω εφαρμογών είναι η αξιολόγηση της απόδοσης των διαφόρων αλγορίθμων εν χρόνω ολοκλήρωσης.

Set $w_1, w_2, w_3, W_1\Lambda_1, W_2\Lambda_2, W_3\Lambda_3, W_1\Lambda_4, W_2\Lambda_5, W_1\Lambda_6, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$

Find W_i from eq. (2.25) for $i=1$

Initialize $u_n = u_0$, $\dot{u}_n = \dot{u}_0$

Find $K_0 = \bar{K}(u_0, \dot{u}_0)$, $C_0 = \bar{C}(u_0, \dot{u}_0)$, $p_0 = \bar{p}(u_0, \dot{u}_0)$

Find $\ddot{u}_0 = (f_0 - p_0)/M$

Set $K_n = K_0$, $C_n = C_0$, $p_n = p_0$, $u_n = u_0$, $\dot{u}_n = \dot{u}_0$, $\ddot{u}_n = \ddot{u}_0$

for n = 1:length(**f**)-1

    Initialize k = 1

    Initialize tol = tol$_{max}$

    Find $\ddot{u}_{n+1}^1 = \tilde{M}^{-1}(K_n, C_n)\tilde{F}_n(K_n, C_n, p_n)$ from eq. (2.22)

    while tol $\geq$ tol$_{max}$ & k $\leq$ k$_{max}$

        Iteration k of increment n+1:

        Set $\ddot{u}_{n+1}^k = \ddot{u}_n + \ddot{u}_{n+1}^1$

        Find u_{n+1}^k and $\dot{u}_{n+1}^k$ according to (2.19) and (2.20) respectively

        Find $K_{n+1}^k = \bar{K}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $C_{n+1}^k = \bar{C}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $p_{n+1}^k = \bar{p}(u_{n+1}^k, \dot{u}_{n+1}^k)$

        Find the residual $R_{n+1}^k = \tilde{F}_{n+1}^k(K_{n+1}^k, C_{n+1}^k, p_{n+1}^k) - \tilde{M}(K_{n+1}^k, C_{n+1}^k)\ddot{u}_{n+1}^1$

        Set $d\ddot{u}_{n+1}^k = \ddot{u}_n + (\ddot{u}_{n+1}^1 + da)$, where da is a constant infinitesimal variation of acceleration

        Find du_{n+1}^k and $d\dot{u}_{n+1}^k$ from (2.19) and (2.20) respectively

        Find $dK_{n+1}^k = \bar{K}(du_{n+1}^k, d\dot{u}_{n+1}^k)$, $dC_{n+1}^k = \bar{C}(du_{n+1}^k, d\dot{u}_{n+1}^k)$, $dp_{n+1}^k = \bar{p}(du_{n+1}^k, d\dot{u}_{n+1}^k)$

        Find the residual $dR_{n+1}^k = d\tilde{F}_{n+1}^k(dK_{n+1}^k, dC_{n+1}^k, dp_{n+1}^k) - d\tilde{M}(dK_{n+1}^k, dC_{n+1}^k)d\ddot{u}_{n+1}^k$

        Find $qda = [R_{n+1}^k ./\ddot{u}_{n+1}^1] ./ [(dR_{n+1}^k - R_{n+1}^k) ./da]$

        Update $\ddot{u}_{n+1}^1 = (1 - qda)\ddot{u}_{n+1}^1$

        Update k = k+1

    end

    Set $\ddot{u}_{n+1}^k = \ddot{u}_n + \ddot{u}_{n+1}^1$

    Find u_{n+1}^k and $\dot{u}_{n+1}^k$ according to (2.19) and (2.20) respectively

    Find $K_{n+1}^k = \bar{K}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $C_{n+1}^k = \bar{C}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $p_{n+1}^k = \bar{p}(u_{n+1}^k, \dot{u}_{n+1}^k)$

    Assign for next increment: $K_{n+1} = K_{n+1}^k$, $C_{n+1} = C_{n+1}^k$, $p_{n+1} = p_{n+1}^k$, $u_{n+1} = u_{n+1}^k$, $\dot{u}_{n+1} = \dot{u}_{n+1}^k$, $\ddot{u}_{n+1} = \ddot{u}_{n+1}^k$

end

Εικόνα 0.1: Ψευδοκώδικας του αλγορίθμου μη γραμμικής εν χρόνω ολοκλήρωσης που αναπτύχθηκε στην παρούσα διατριβή.

Η μάζα όλων των μονοβάθμιων ταλαντωτών που θεωρούνται στο παρόν κεφάλαιο θεωρείται ίση με τη μονάδα, χωρίς βλάβη της γενικότητας. Το χρονικό βήμα ισούται με $\Delta t = 0.01$ (επαρκώς μεγάλο για να προκύπτει υπολογίσιμο σφάλμα κατα τη χρονική ολοκλήρωση) και η διάρκεια της δυναμικής απόκρισης είναι ίση με 100 βήματα (1 sec) για όλα τα προβλήματα αναφοράς. Ένας αποτελεσματικός
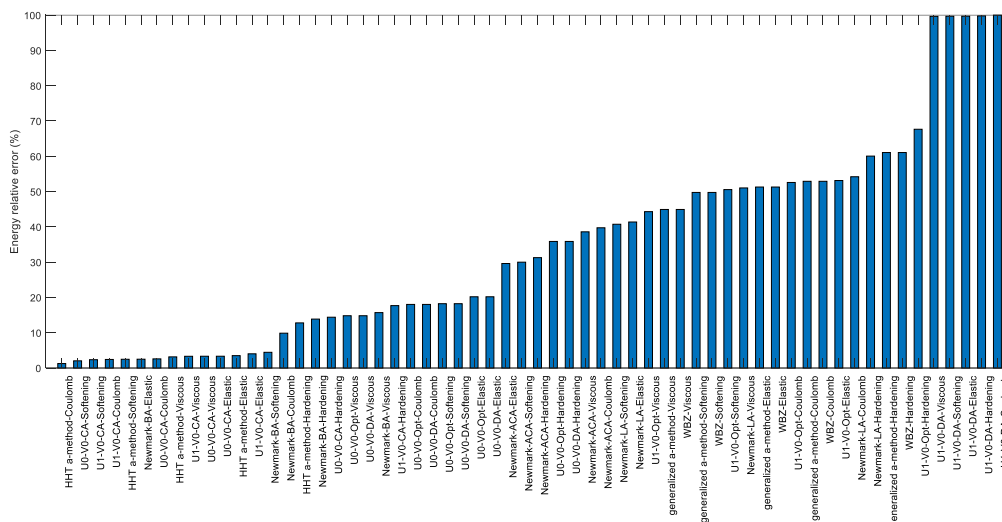
αλγόριθμος χρονικής ολοκλήρωσης θα πρεπει να κάνει ακριβή υπολογισμό της ενέργειας των μονοβάθμιων συστημάτων. Ως «ακριβής» τιμή της ενέργειας του κάθε ταλαντωτή, θεωρείται η ενέργεια που υπολογίζεται με τον εκάστοτε αλγόριθμο χρονικής ολοκλήρωσης χρησιμοποιώντας ένα πολύ μικρότερο χρονικό βήμα (και άρα πολύ ακριβέστερο υπολογισμό). Το εν λόγω χρονικό βήμα λαμβάνεται ίσο με $\Delta t = 0.0001$, που αντιστοιχεί σε 10000 βήματα για δυναμική απόκριση διάρκειας 1 sec. Δεδομένου ότι το χρονικό βήμα η χρονική διάρκεια, η δυσκαμψία (σε μικρές παραμορφώσεις) οι αρχικές συνθήκες και οι σταθερές ολοκλήρωσης είναι ίδιες για όλα τα προβλήματα αναφοράς, οι διαφορές που εμφανίζονται στους ενεργειακούς υπολογισμούς οφείλονται αποκλειστικά στις διαφορετικές ποιότητες των εμπλεκόμενων αλγορίθμων χρονικής ολοκλήρωσης. Στην Εικόνα 0.2 παρουσιάζονται αποτελέσματα για τη χρονική εξέλιξη του σχετικού σφάλματος στη συνολική ενέργεια κάθε ταλαντωτή, για τρεις αλγορίθμους με βέλτιστο αριθμητικό σκεδασμό και διασπορά. Παρατηρείται ότι ο αλγόριθμος Uo-Vo-Opt δίνει το μικρότερο σφάλμα σε σχέση με τους υπόλοιπους αλγορίθμους, για όλα τα προβλήματα αναφοράς. Με δεδομένο ότι όλοι οι συγκρινόμενοι αλγόριθμοι είναι κατηγορίας βέλτιστου αριθμητικού σκεδασμού και διασποράς, τα αποτελέσματα δείχνουν ότι είναι επιθυμητή η χρήση αλγορίθμων ολοκλήρωσης με μηδενική τάξη υπερακόντισης τόσο ως προς την μετατόπιση όσο και ως προς την ταχύτητα, για αυξημένη ακρίβεια στα αποτελέσματα.
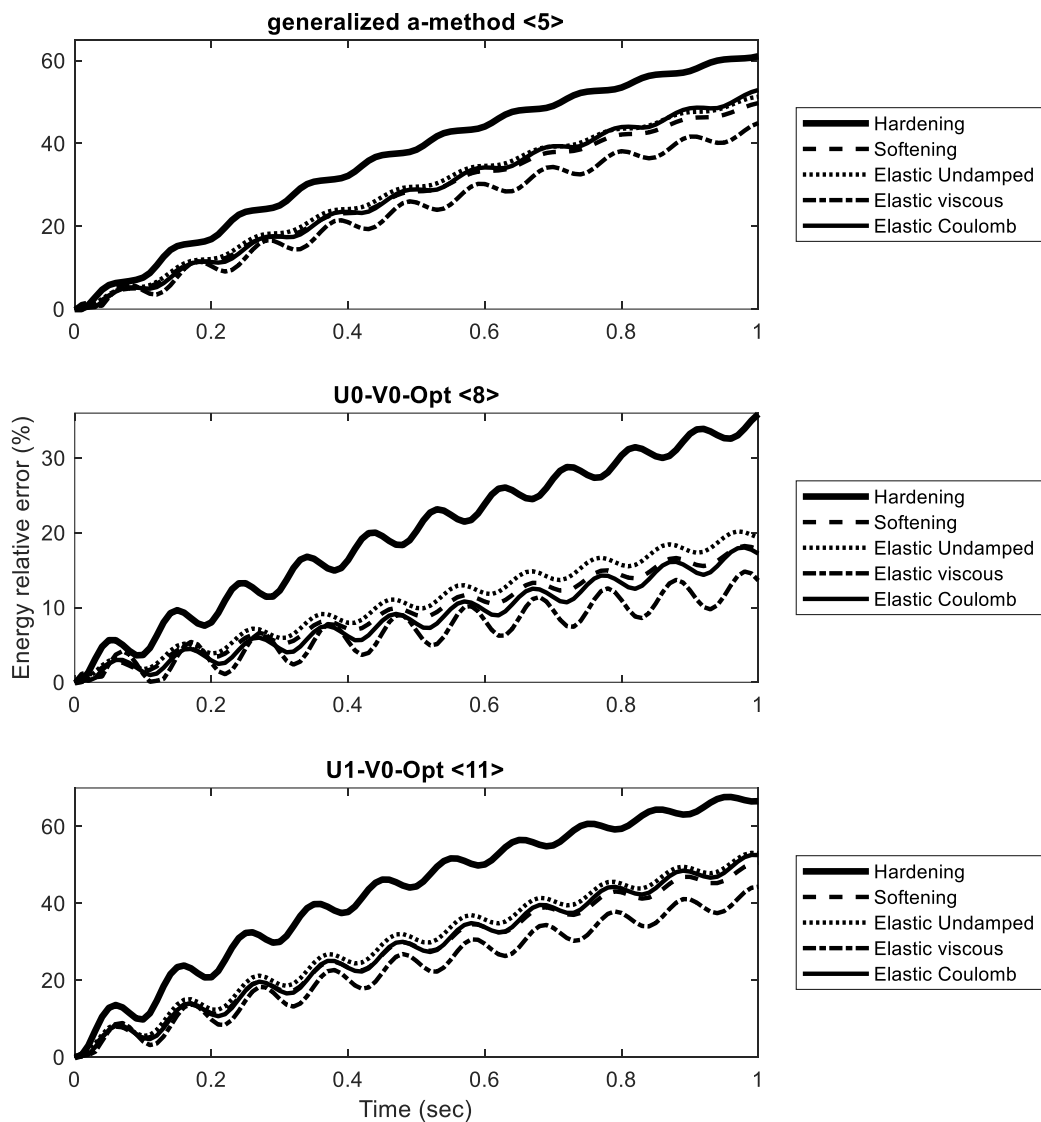
Εικόνα 0.2: Χρονοιστορία του σχετικού σφάλματος της ολικής ενέργειας των μονοβάθμιων ταλαντωτών των προβλημάτων αναφοράς 1-5, με χρήση αλγορίθμων χρονικής ολοκληρωσης κατηγορίας βέλτιστου αριθμητικού σκεδασμού και διασποράς (-Opt).

Το μέγιστο σχετικό σφάλμα της χρονοιστορίας της ολικής ενέργειας για όλα τα ζεύγη αλγορίθμων – προβλημάτων αναφοράς παρουσιάζονται στην εικόνα 0.3, όπου στον κατακόρυφο άξονα παρουσιάζεται το μέγιστο σχετικό σφάλμα της ολικής ενέργειας και στον οριζόντιο άξονα παρουσιάζονται τα ζεύγη αλγορίθμου – προβλήματος αναφοράς. Το ελάχιστο σφάλμα παρατηρείται για τους αλγορίθμους συνεχούς επιτάχυνσης (συμπεριλαμβανομένου και του αλγορίθμου Hilber-Hughes-Taylor – HHT) για τον γραμμικά ελαστικό ταλαντωτή με απόσβεση Coulomb, τον ταλαντωτή χωρίς απόσβεση με κρατυνόμενη συμπεριφορά, τον γραμμικά ελαστικό ταλαντωτή χωρίς και με ιξωδοελαστική απόσβεση. Από την άλλη μεριά, στην περίπτωση του ταλαντωτή χωρίς απόσβεση

με σκλήρυνση όπου εφαρμόζονται αλγόριθμοι χρονικής ολοκλήρωσης ασυνεχούς επιτάχυνσης, παρατηρείται το μεγαλύτερο σφάλμα.

Εικόνα 0.3: Μέγιστο σχετικό σφάλμα της συνολικής ενέργειας των ταλαντωτών των προβλημάτων αναφοράς 1-5 υπολογιζόμενης με τους αλγορίθμους χρονικής ολοκλήρωσης που αναπτύχθηκαν στο παρόν κεφάλαιο.

### 0.2.4. Γραμμικά ελαστικό σύστημα 3 βαθμών ελευθερίας χωρίς απόσβεση

Πέραν των προαναφερθέντων προβλημάτων αναφοράς, το πρόβλημα των Bathe & Noh (2012) επιλύθηκε με τους αλγορίθμους χρονικής ολοκλήρωσης που παρουσιάζονται στο κεφάλαιο αυτό. Το πρόβλημα αφορά ένα σύστημα 3 βαθμών ελευθερίας που φαίνεται στην Εικόνα 0.4, και αντιπροσωπεύει μοντέλα πεπερασμένων στοιχείων που περιλαμβάνουν στοιχεία τόσο κανονικής, όσο και σχετικά μεγάλης δυσκαμψίας. Τα στοιχεία μεγάλης δυσκαμψίας μπορεί να αντιπροσωπεύουν π.χ. άκαμπτες συνδέσεις, στηρίξεις, πακτώσεις, κλπ, ή ακόμα και συντελεστές ποινής (penalty factors) για διάφορους τύπους περιορισμών. Οι αλγόριθμοι που παρουσιάζονται στο κεφάλαιο αυτό έλυσαν το εν λόγω πρόβλημα ακριβέστερα σε σχέση με τους στοιχειώδεις γνωστούς αλγορίθμους χρονικής ολοκλήρωσης.
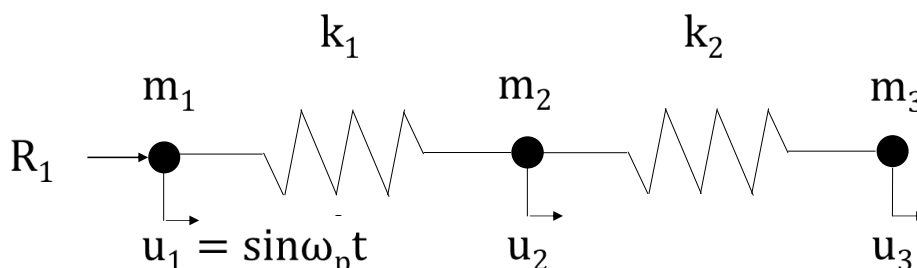
Εικόνα 0.4: Προσομοίωμα αναφοράς 3 βαθμών ελευθερίας (Bathe & Noh, 2012).

0.2.5. Ταλάντωση απλού εκκρεμούς χωρίς απόσβεση με μεγάλη στροφή

Ένα ευρέως διαδεδομένο πρόβλημα αναφοράς που χρησιμοποιείται για την εκτίμηση της αποτελεσματικότητας αλγορίθμων χρονικής ολοκλήρωσης είναι η ταλάντωση απλού εκκρεμούς χωρίς απόσβεση με μεγάλη στροφή. Αυτό αποτελείται από μια σημειακή μάζα συνδεδεμένη σε μια αβαρή και άκαμπτη ράβδο, εντός βαρυτικού πεδίου. Η ακριβής λύση της μη γραμμικής ελεύθερης ταλάντωσης του εκκρεμούς χρησιμοποιείται ως λύση αναφοράς. Το απλό εκκρεμές εικονίσεται στην Εικόνα 0.5.
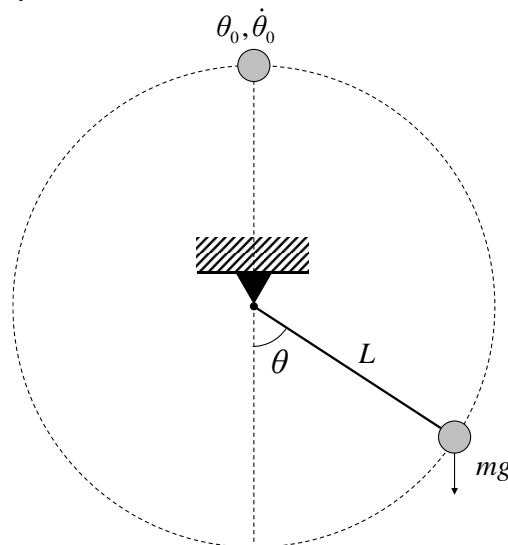
Εικόνα 0.5: Απλό εκκρεμές χωρίς απόσβεση με μεγάλη στροφή.

Όπως και στο προηγούμενο πρόβλημα αναφοράς, αποδεικνύεται ότι οι αλγόριθμοι που παρουσιάζονται στο παρόν κεφάλαιο δίνουν ακριβέστερες λύσεις σε σχεση με τους στοιχειώδεις αλγορίθμους.

0.2.6. Συμπεράσματα

Η οικογένεια των αλγορίθμων χρονικής ολοκλήρωσης απλού βήματος απλής λύσης που περιλαμβάνει τους πιο γνωστούς αλγορίθμους χρονικής ολοκλήρωσης ως ειδικές περιπτώσεις, μπορεί να επεκταθεί για την επίλυση μη γραμμικών προβλημάτων δυναμικής απόκρισης λόγω μη γραμμικότητας περοερχόμενης τόσο από το υλικό όσο και από τη γεωμετρία, μεσω μιας επαναληπτικής διαδικασίας τύπου Newton – Raphson. Ακόμα και με σχετικά μεγάλο μέγεθος χρονικού βήματος, οι ανωτέρω αλγόριθμοι είναι αποτελεσματικοί, με παραδεκτή ακρίβεια και ευστάθεια, ενώ είναι πολύ ανώτεροι από τους στοιχειώδεις αλγορίθμους, οι οποίοι αποτυγχάνουν να υπολογίσουν τη δυναμική απόκριση χρησιμοποιώντας το ίδιο μέγεθος χρονικού βήματος.
Επισημαίνεται ότι, απόλυτα ευσταθείς αλγόριθμοι για γραμμικά προβλήματα, μπορεί να χάσουν την ευστάθειά τους στη μη γραμμική περιοχή. Για αυξημένο

χρονικό βήμα (και συνεπώς μικρότερο υπολογιστικό φόρτο) οι αλγόριθμοι σταθερής επιτάχυνσης, στους οποίους περιλαμβάνεται ο αλγόριθμος HHT ως ειδική περίπτωση, δίνουν το πιο ακριβές αποτέλεσμα υπολογισμού της δυναμικής απόκρισης για τις περισσότερες από τις περιπτώσεις που μελετήθηκαν. Αποτελούν για το λόγο αυτό την βέλτιστη επιλογή, ως προς τη γενική τους απόδοση για τον υπολογισμό της δυναμικής απόκρισης. Περαν αυτού, ο μεγάλος αριθμός των σταθερών ολοκλήρωσης των αλγορίθμων απλού βήματος απλής λύσης, δίνει την ευελιξία της επιλογής των τιμών τους λαμβάνοντας υπόψη τις ιδιαιτερότητες του προβλήματος ή/και της κατασκευής που πρόκειται να επιλυθεί.

0.3. OpenSeismoMatlab: Ένα νέο λογισμικό ανοικτού κώδικα για την επεξεργασία δεδομένων ισχυρών εδαφικών κινήσεων

0.3.1. Χαρακτηριστικά, δυνατότητες και εφαρμογές

Ένα νέο λογισμικό ανοικτού κώδικα εισάγεται στην παρούσα διατριβή το οποίο μπορεί να επεξεργάζεται δεδομένα ισχυρών σεισμικών καταγραφών, και είναι γραμμένο σε γλώσσα προγραμματισμού MATLAB. Το εν λόγω λογισμικό χρησιμοποιεί ένα ελαστοπλαστικό διγραμμικό καταστατικό προσομοίωμα με κρατυνόμενη συμπεριφορά κινηματικού τύπου, το οποίο ενσωματώνει σε ένα αλγόριθμο εν χρόνω ολοκλήρωσης, απλού βήματος, απλής λύσης, ο οποίος παρουσιάζεται στο κεφάλαιο 2 της παρούσας διατριβής. Το OpenSeismoMatlab μπορεί να υπολογίζει χρονοιστορίες, μέγιστες τιμές, ένταση Arias και τη χρονοιστορία αυτής, την ενεργό διάρκεια ενός σεισμού, διάφορα γραμμικά ελαστικά φάσματα και ψευδοφάσματα απόκρισης, διάφορα ανελαστικά φάσματα απόκρισης σταθερής πλαστιμότητας, και επίσης φάσματα πλατους Fourier και τη μεση περίοδο ενός σεισμογραφήματος. Δεδομένης της ανοικτής φύσης του, το OpenSeismoMatlab μπορεί να επεκταθεί ή/και να τροποποιηθεί εύκολα, πράγμα που το καθιστά ένα λογισμικό μεγάλης εκπειδευτικής ή/και ερευνητικής αξίας για την επαγγελματική και ερευνητική κοινότητα.

Στο 3ο κεφάλαιο της παρούσας διατριβής παρουσιάζονται λεπτομερώς η δομή, οι αλγόριθμοι καθώς και οι κύριες υπορουτίνες του ανωτέρω λογισμικού. Επίσης, γίνεται εφαρμογή του σε μια γκαμα ισχυρών σεισμικών εδαφικών κινήσεων και υπολογίζονται διάφορα φάσματα, τα οποία συγκρίνονται με αντίστοιχα αποτελέσματα από άλλα κατοχυρωμένα λογισμικά.

Η σημαντικότητα του λογισμικού καθώς και των αυξημένης ακριβείας αποτελεσμάτων του διαφαίνεται από το γεγονός ότι σχεδόν όλοι οι αντισεισμικοί κώδικες και κανονισμοί διεθνώς απαιτούν την επιλογή αντιπροσωπευτικών ισχυρών σεισμικών καταγραφών οι οποίες χρησιμοποιούνται ως αρχικά δεδομένα για το σχεδιασμό των κατασκευών, με βαση την απόκριση των τελευταίων στις καταγραφές αυτές. Είναι επομένως σημαντικό να γίνεται ρεαλιστική επιλογή και επεξεργασία των ακατέργαστων ισχυρών καταγραφών, με σκοπό τον υπολογισμό εκείνων των σεισμικών παραμέτρων που χρησιμεύουν στον υπολογισμό της δυναμικής απόκρισης της κατασκευής που σχεδιάζεται. Διάφορα λογισμικά έχουν αναπτυχθεί για την επιλογή ισχυρών σεισμικών εδαφικών κινήσεων, που

χρησιμοποιούνται για τη δυναμική ανάλυση και τον αντισεισμικό σχεδιασμό των κατασκευών (Katsanos & Sextos, 2013; Macedo & Castro, 2017). Μεταξύ των πιο σημαντικών σεισμικών παραμέτρων μις ισχυρής εδαφικής κίνησης είναι οι διάφοροι τύποι φασμάτων (ελαστικά φάσματα απόκρισης, φάσματα σταθερής πλαστιμότητας, φάσματα σταθερής αντοχής διαρροής, φάσματα Fourier, κλπ) τα οποία προκύπτουν από την ελεξεργασία των ακατέργαστων σεισμικών καταγραφών και τα οποία χρησιμοποιούνται σε διάφορες μεθόδους αντισεισμικού σχεδιασμού, όπως η δυναμική φασματική ανάλυση, η ασύζευκτη ιδιομορφική ανάλυση χρονοιστορίας (Uncoupled Modal Response History Analysis, UMRHA), η ιδιομορφική ανάλυση pushover (MPA), κλπ (Chopra, 2012). Επίσης, με κατάλληλη προσαρμογή του φάσματος Fourier μιας ισχυρής σεισμικής καταγραφής, είναι δυνατός ο έλεγχος του συχνοτικού περιεχομένου της. Όλα τα παραπάνω δείχνουν ότι ένα λογισμικό ακριβείας για την επεξεργασία των καταγραφών είναι απαραίτητο για τον σωστό αντισεισμικό σχεδιασμό των κατασκευών, συμπεριλαμβανομένων στρατηγικών για διασφάλιση της υγείας και της ασφάλειας των ενοίκων και των περιουσιακών τους στοιχείων κατα τη διάρκεια ζωής της κατασκευής. Στο 5ο κεφάλαιο της παρούσα διατριβής αναπτύσσεται μεθοδολογία για την επιλογή σεισμικών καταγραφών με σκοπό τη δυναμική ανάλυση και τον αντισεισμικό σχεδιασμό των κατασκευών, η οποία χρησιμοποιεί με επιτυχία το OpenSeismoMatlab, όπως φαίνεται από την ποιότητα και αντιπροσωπευτικότητα των παραγόμενων τεχνητών σεισμικών καταγραφών

Το OpenSeismoMatlab έχει τα ακόλουθα πλεονεκτήματα και μοναδικά χαρακτηριστικά, έναντι των υπόλοιπων λογισμικών επεξεργασίας ισχυρών σεισμικών εδαφικών κινήσεων:

- Χρησιμοποιεί αλγόριθμους εν χρόνω ολοκλήρωσης τελευταίας τεχνολογίας (απλού βήματος, απλής λύσης) οι οποίοι είναι πιο εύρωστοι και ακριβείς, όπως αποδείχθηκε στο κεφάλαιο 2 της παρούσας διατριβής (Papazafeiropoulos et al., 2017a; Papazafeiropoulos et al., 2017b) σε σχέση με τους συμβατικούς αλγορίθμους εν χρόνω ολοκλήρωσης (Newmark, κλπ.) που χρησιμοποιούνται ευρέως από τα άλλα λογισμικά. Οι πρώτοι ανήκουν σε μια οικογένεια αλγορίθμων απλού βήματος απλής λύσης και, μεσω των 14 ανεξάρτητων σταθερών ολοκλήρωσης που διαθέτουν, μπορούν να προσαρμοστούν εύκολα για τον έλεγχο της αριθμητικής απόσβεσης και διασποράς, της συνέχειας της επιτάχυνσης εντός του χρονικού βήματος, καθώς και της τάξης της υπερακόντισης στη μετακίνηση και την ταχύτητα. Επιλέγοντας τις τιμές των σταθερών ολοκλήρωσης, ο χρήστης μπορεί να διαλέξει από μια μεγάλη ποικιλία αλγορίθμων εν χρόνω ολοκλήρωσης, σύμφωνα με τους Papazafeiropoulos et al. (2017a) και αυτό επιτρέπει την βέλτιστη ποιότητα των παραγόμενων αποτελεσμάτων (χρονοιστοριών, φασμάτων, κλπ).
- Είναι τελείως δωρεάν και παρέχεται σε μορφή ανοικτού κώδικα, χαρακτηριστικά που το καθιστούν μεγάλης εκπαιδευτικής και ερευνητικής αξίας. Περιέχει κώδικα MATLAB γραμμένο με απλό τρόπο, συνοδευόμενο από σχόλια και συνεπώς είναι εύκολα κατανοήσιμο από το χρήστη. Η λογική των χρησιμοποιούμενων μεθόδων εξηγείται λεπτομερώς με μορφή σολίων εντός του κώδικα. Περαν αυτών, ο ανοικτός κώδικας παρέχει την

δυνατότητα επέκτασης, αναβάθμισης ή ουσιώδους τροποποίησής του με κάθε επιθυμητό τρόπο.

- Επιπλέον, το ελαστοπλαστικό διγραμμικό καταστατικό προσομοίωμα με κρατυνόμενη σκλήρυνση που περιλαμβάνει το OpenSeismoMatlab, το οποίο αποτελεί βασικό κομμάτι για τον υπολογισμό των μη γραμμικών φασμάτων, είναι διατυπωμένο με τρόπο ταυτόχρονα απλό και ακριβή. Το OpenSeismoMatlab δεν περιλαμβάνει απλοϊκές μορφές του εν λόγω καταστατικού προσομοιώματος υλικού, όπως συμβαίνει συχνά στη βιβλιογραφία (Newmark & Hall, 1982; Krawinkler & Nassar, 1992; Miranda & Bertero, 1994).

Ο πλήρης κώδικας του OpenSeismoMatlab έχει ανέβει σε δυο διαφορετικές διαδικτυακές πλατφόρμες διανομής: (i) την υπηρεσία File Exchange του MATLAB central (Papazafeiropoulos, 2018) και (ii) στο ResearchGate (Papazafeiropoulos & Plevris, 2018), ούτως ώστε να είναι δημόσια διαθέσιμο.

0.3.2. Δομή και κώδικας του OpenSeismoMatlab

Στην Εικόνα 0.6 φαίνεται το διάγραμμα εξάρτησης μεταξύ των διαφόρων υπορουτίνων που χρησιμοποιούνται από το OpenSeismoMatlab Οι τέσσερις κύριες συναρτήσεις είναι οι LEReSp για τον υπολογισμό του φάσματος γραμμικά ελαστικής απόκρισης, CDReSp για τον υπολογισμό του φάσματος σταθερής πλαστιμότητας, FASp για τον υπολογισμό των φασμάτων Fourier, και baselineCorr για τη διόρθωση της γραμμής βάσης (baseline correction) της καταγραφής εισόδου. Οι υπορουτίνες DRHA, NLIDABLKIN και HalfStep καλούνται άμεσα από τη συνάρτηση CDReSp και χρησιμοποιούνται για τη δυναμική ανάλυση χρονοϊστορίας, το μοντέλο μη γραμμικής δυναμικής ανάλυσης με διγραμμικό κινηματικής κράτυνσης υλικό, και διαίρεση μιας σεισμικής καταγραφής στο μισό του αρχικού χρονικού βήματος, αντίστοιχα. Η συνάρτηση LIDA χρησιμοποιείται για τη γραμμική δυναμική ανάλυση και καλείται από τις συναρτήσεις LEReSp και DRHA, όπου η υπορουτίνα BLKIN καλείται από τη συνάρτηση NLIDABLKIN.
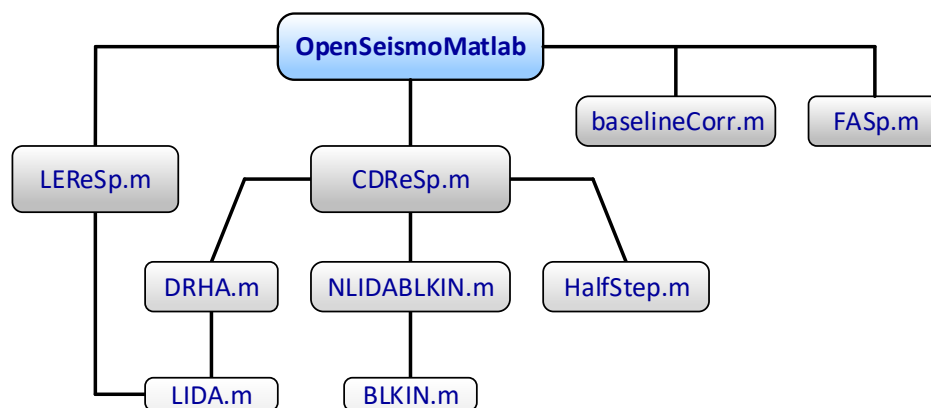
Εικόνα 0.6: Σχηματικό διάγραμμα εξάρτησης μεταξύ των διαφόρων υπορουτίνων που περιλαμβάνονται στο λογισμικό OpenSeismoMatlab.

Οι σχετικοί κώδικες των ανωτέρω συναρτήσεων παρουσιάζονται αναλυτικά στο κεφάλαιο 3 της διατριβής. Για παράδειγμα, ένα διάγραμμα ροής για τον υπολογισμό του φάσματος ελαστικής απόκρισης μιας ισχυρής σεισμικής καταγραφής φαίνεται στην Εικόνα 0.7

Input: $\ddot{u}_g$, ω, ξ

Initialize SD, SV SA

Set u_0 and $\dot{u}_0$

for each SDOF i with eigenfrequency ω_i

   if $\omega_i \Delta t / (2\pi) > 0.02$

      Reproduce $\ddot{u}_g$ with half time step (from Δt to $\Delta t/2$)

      Set $\Delta t = \Delta t/2$

   end

   Perform dynamic analysis of SDOF with input ($\ddot{u}_g$, ξ, u_0, $\dot{u}_0$)

   Assign $\max(\|u(t)\|)$ to $SD(i)$

   Assign $\max(\|\dot{u}(t)\|)$ to $SV(i)$

   Assign $\max(\|\ddot{u}(t)\|)$ to $SA(i)$

end

Calculate $PSV = \omega SD$ and $PSA = \omega^2 SD$

Output: SD, SV, SA, PSV, PSA

Εικόνα 0.7: Διάγραμμα ροής για τον υπολογισμό του φάσματος ελαστικής απόκρισης, που χρησιμοποιείται στο OpenSeismoMatlab.

Ενδεικτικά παρουσιάζεται στην Εικόνα 0.8 (και στην Εικόνα 0.9, ως συνέχεια της 0.8) ο κώδικας της συνάρτησης LEReSp.m, η οποία χρησιμοποιείται για τον υπολογισμό του φάσματος γραμμικά ελαστικής απόκρισης.

```matlab
function [PSa,PSv,Sd,Sv,Sa]=LEReSp(dt,xgtt,T,varargin)
% set defaults for optional inputs
optargs = {0.05,0.01,'U0-V0-CA',0};
% skip any new inputs if they are empty
newVals = cellfun(@(x) ~isempty(x), varargin);
% overwrite the default values by those specified in varargin
optargs(newVals) = varargin(newVals);
% place optional args in memorable variable names
[ksi,dtTol,AlgID,rinf] = optargs{:};
% initialize
NumSDOF=length(T);
Sd=zeros(NumSDOF,1);
Sv=zeros(NumSDOF,1);
Sa=zeros(NumSDOF,1);
% Set the eigenfrequencies of the SDOF population
omega=2*pi./T;
% Flip eigenfrequency vector in order for the half-stepping algorithm
% (HalfStep function) to work from large to small eigenperiods
omega=omega(end:-1:1);
% set initial conditions
u0=0;
ut0=0;
% zero-order displacement & velocity overshooting behavior and
% optimal numerical dissipation and dispersion
rinf=1; % mid-point rule a-form algorithm
for j=1:length(T)
    omegaj=omega(j);
    % Check if dt/T>dtTol. If yes, then reproduce the time history
    % with the half step
    if dt*omegaj/(2*pi)>dtTol
        xgtt=HalfStep(xgtt);
        dt=dt/2;
    end
    [u,ut,utt] = LIDA(dt,xgtt,omegaj,ksi,u0,ut0,AlgID,rinf);
    % output
    Sd(j)=max(abs(u));
    Sv(j)=max(abs(ut));
    Sa(j)=max(abs(utt));
end
```

Εικόνα ο.8: Πηγαίος κώδικας της συνάρτησης LEReSp.

```matlab
% Flip output quantities to be compatible with omega
omega=omega(end:-1:1);
Sd=Sd(end:-1:1);
Sv=Sv(end:-1:1);
Sa=Sa(end:-1:1);
% Calculate pseudovelocity and pseudoacceleration
PSv=Sd.*omega;
PSa=Sd.*omega.^2;
end
```

Εικόνα 0.9: Πηγαίος κώδικας της συνάρτησης LEReSp (συνέχεια της Εικόνας 0.8).

0.3.3. Αποτελέσματα - συγκρίσεις

Με στόχο την επαλήθευση της ορθότητας των αποτελεσμάτων του OpenSeismoMatlab, αυτά συγκρίνονται με αντίστοιχα αποτελέσματα ενός εμπορικού λογισμικού επεξεργασίας δεδομένων ισχυρών σεισμικών καταγραφών, του SeismoSignal. Λεπτομερής περιγραφή του λογισμικού αυτού δίνεται στην παράγραφο 3.1 της παρούσας διατριβής. Το SeismoSignal επιλέχθηκε λόγω του ότι είναι εύκολο στη χρήση, έχει σχετικά λεπτομερή βιβλιογραφία υψηλής ποιότητας και είναι γενικά αποδεκτό ως ένα αξιόπιστο εργαλείο παγκοσμίως, δεδομένου ότι χρησιμοποιείται εδώ και μερικά χρόνια από ερευνητές και επαγγελματίες.

Ωστόσο, το SeismoSignal χρησιμοποιεί συμβατικούς αλγορίθμους εν χρόνω ολοκλήρωσης, και σε ορισμένες περιπτώσεις μπορεί να είναι επιρρεπές σε σφάλματα, ιδίως όταν απαιτείται χρήση αλγορίθμων με ανώτερης ποιότητας ιδίότητες αριθμητικής απόσβεσης και υπερακοντισμού σε σχέση με αυτές των αλγορίθμων Newmark. Το OpenSeismoMatlab ερχεται για να βελτιώσει αυτήν την αδυναμία. Για λόγους σύγκρισης, επιλέχθηκε ένας αριθμός ισχυρών σεισμικών καταγραφών σε όρους χρονοιστορίας επιτάχυνσης τα στοιχειά των οποίων φαίνονται στον Πίνακα 0.1

Earthquake	Year	Station
Imperial Valley	1979	El Centro Array Sta 8, CA, 95 E Cruickshank Rd
Izmit-Kocaeli	1999	Yarimca Petkim
Loma Prieta	1989	Gilroy Array Sta 3, CA, Sewage Plant
Northridge	1994	090 CDMG Station 24278
San Fernando	1971	Castaic, CA, Old Ridge Route
Spitak	1988	Gukasyan
Cape Mendocino	1992	Cape Mendocino, CA, Petrolia

Chi-Chi	1999	Nantou - Hsinjie School, WNT
El Centro	1940	El Centro Terminal Substation Building
Hollister	1961	USGS Station 1028
Kobe	1995	Takarazuka

Πίνακας 0.1: Σεισμοί ισχυρές καταγραφές των οποίων χρησιμοποιήθηκαν για τη σύγκριση των αποτελεσμάτων των SeismoSignal και OpenSeismoMatlab.

Στην εικόνα 0.10 φαίνεται η φασματική μετατόπιση των ανελαστικών φασμάτων σταθερής πλαστιμότητας των έντεκα ισχυρών σεισμικών καταγραφών που παρουσιάζονται στον πίνακα 0.1, για στοχευόμενη πλαστιμότητα ίση με 2. Οι διαφορές μεταξύ των καμπυλών του OpenSeismoMatlab και SeismoSignal αποδίδονται στους διαφορετικούς αλγόριθμους εν χρόνω ολοκλήρωσης που χρησιμοποιούνται από τα δυο προαναφερόμενα λογισμικά, την ανωτερότητα του αλγορίθμου εν χρόνω ολοκλήρωσης που χρησιμοποιεί το OpenSeismoMatlab και άλλους παράγοντες σχετιζόμενους με την αποτελεσματικότητα της υλοποίησης των διαφόρων αλγοριθμικών διαδικασιών στον κώδικα των δυο λογισμικών. Παρά τα παραπάνω, γενικά τα αντίστοιχα αποτελέσματα των δυο λογισμικών είναι λογικά κοντά μεταξύ τους τόσο στη γραμμική όσο και στη μη γραμμική περιοχή. Η συμφωνία μεταξύ των δυο λογισμικών είναι πολύ μεγαλύτερη στη γραμμική περιοχή, σε σχέση με τη μη γραμμική.
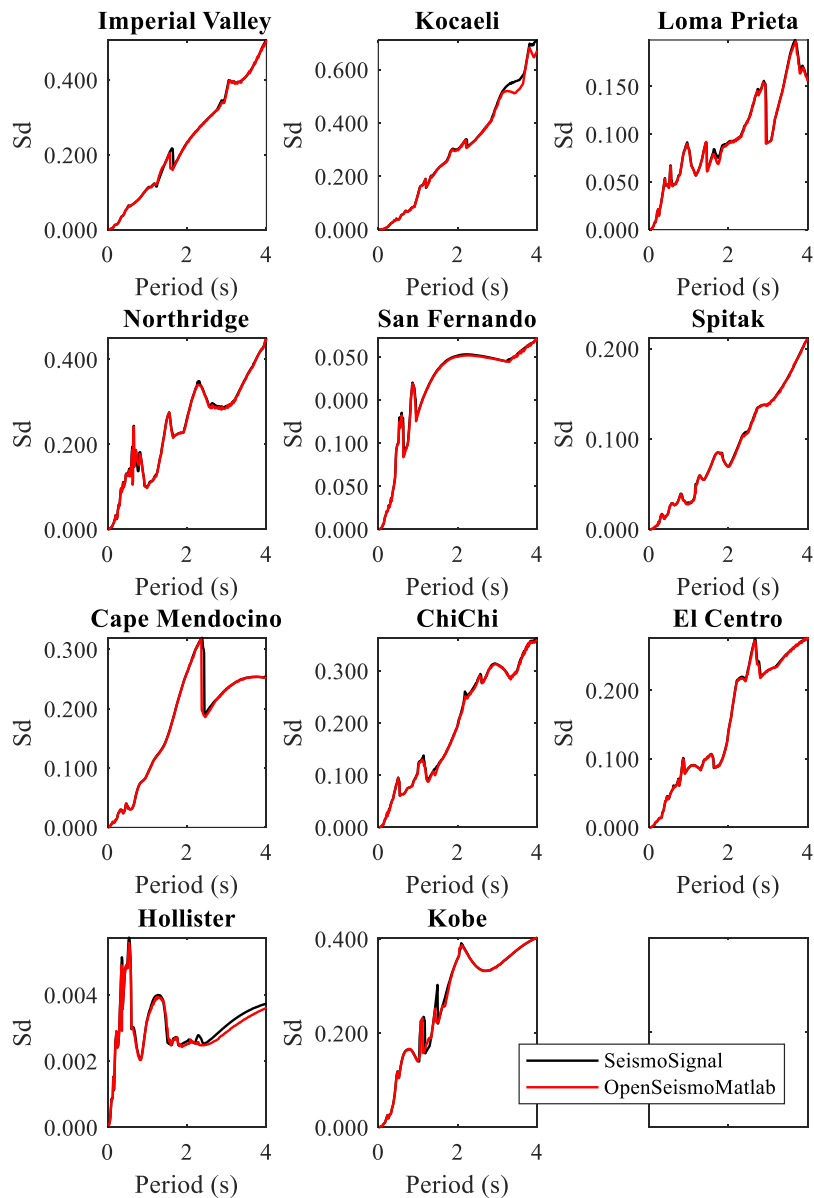
Εικόνα 0.10: Ανελαστικά φάσματα μετατόπισης σταθερής πλαστιμότητας για τις 11 ισχυρές σεισμικές καταγραφές υπολογισμένα με OpenSeismoMatlab και SeismoSignal.

## 0.3.4. Επίδραση του χρονικού βήματος στην ακρίβεια των αποτελεσμάτων

Θεωρείται το φάσμα της ψευδοεπιτάχυνσης (PSa) της χρονοιστορίας επιτάχυνσης που αντιστοιχεί στη συνάρτηση $\ddot{u}_g = \sin(20\pi t)$ με ποσοστό κρίσιμης απόσβεσης ξ ίσο με 5%. Η διέγερση είναι μια αρμονική (ημιτονοειδής) κίνηση με κυκλική συχνότητα ίση με 20π (δηλ. συχνότητα 10 Hz και περίοδο 0.1 sec) και συνολική διάρκεια 2 sec, ενώ έχει ψηφιοποιηθεί σε επαρκώς μικρά χρονικά βήματα (

$\Delta t = 0.0001s$). Το φάσμα PSa υπολογίζεται ξεχωριστά με το OpenSeismoMatlab και το SeismoSignal και αρχικά γίνεται σύγκριση μεταξύ των δυο λύσεων. Η σύγκριση αυτή φαίνεται στην Εικόνα 0.11 όπου ο δεκαδικός λογάριθμος του φάσματος PSa παριστάνεται γραφικά σαν συνάρτηση της ιδιοπεριόδου.

Είναι προφανές ότι οι δυο καμπύλες σχεδόν συμπίπτουν και επίσης συμπίπτουν με την ακριβή λύση, δεδομένου ότι το χρονικό βήμα είναι αρκούντως μικρό. Οι διαφορές μεταξύ των καμπυλών είναι της τάξης του $5^{ου}$ δεκαδικού ψηφίου. Ορίζουμε τη λύση αυτή ως λύση αναφοράς για καθε λογισμικό, PSa_0, η οποία από εδω και εφεξής θεωρείται ως η ακριβής λύση. Όσο το μέγεθος του χρονικού βήματος αυξάνει, το φάσμα PSa εμφανίζει αριθμητικό σφάλμα. Ένα μέτρο αυτού του σφάλματος μπορεί να είναι η τετραγωνική ρίζα του μέσου όρου των τετραγώνων των αποκλίσεων μεταξύ του φάσματος PSa για μια αυθαίρετη τιμή του Δt και του ακριβούς φάσματος PSa_0 που παρουσιάζεται στην εικόνα 0.11. Το σφάλμα υπολογίζεται από την εξίσωση (3.22) ως ακολούθως:

$$RMSD = \sqrt{\frac{\sum_{i=1}^{n}\left(PSa_{\Delta t}^{i} - PSa_{0}^{i}\right)^{2}}{n}} \tag{0.17}$$

όπου $PSa_{\Delta t}$ είναι το φάσμα PSa που υπολογίζεται για χρονικό βήμα ίσο με Δt και n είναι ο αριθμός των τιμών ιδιοπεριόδου που περιέχονται στην καμπύλη του φάσματος PSa (n=400 στην παρούσα διερεύνηση). Οι διαφορετικές τιμές του Δt που θεωρούνται είναι 3x10-4 s, 1x10-3 s και 3x10-3 s. Για κάθε μια από αυτές τις τιμές υπολογίζονται δύο φάσματα $PSa_{\Delta t}$, ένα από το OpenSeismoMatlab και ένα από το SeismoSignal. Μετά, εφαρμόζεται η εξίσωση (3.22) για τα δυο λογισμικά ξεχωριστά, όπου υπολογίζονται δυο ξεχωριστές καμπύλες RMSD και παριστάνονται στην Εικόνα 0.12 για λόγους σύγκρισης. Είναι προφανές ότι οι λύσεις που παρέχονται από το OpenSeismoMatlab έχουν μικρότερο σφάλμα από τις αντίστοιχες λύσεις που παρέχοναι από το SeismoSignal, για τα διάφορα μεγέθη του χρονικού βήματος. Συνεπώς, η ποιότητα των αποτελεσμάτων του OpenSeismoMatlab είναι ανώτερη από αυτήν των αποτελεσμάτων του SeismoSignal, τουλάχιστον υπό συγκεκριμένες συνθήκες. Το γεγονός αυτό αποδίδεται στο ότι το πρώτο χρησιμοποιεί προχωρημένους αλγορίθμους εν χρόνω ολοκλήρωσης, οι οποίοι παρουσιάστηκαν εκτενώς στο 2° κεφάλαιο της παρούσας διατριβής.
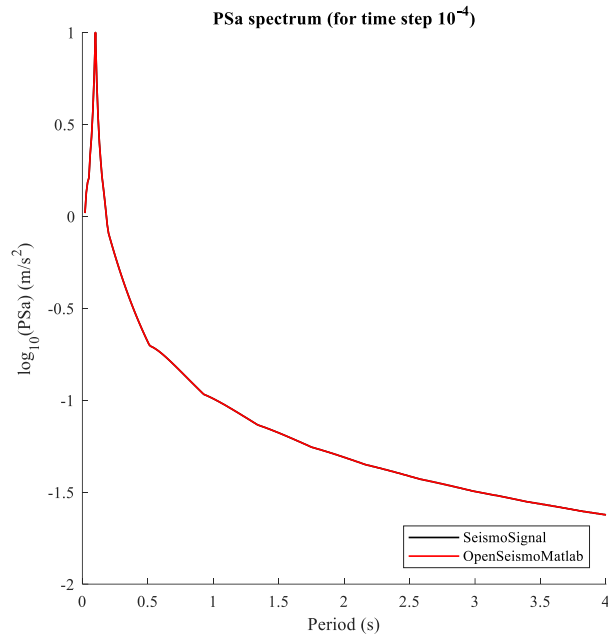
Εικόνα 0.11: Σύγκριση του φάσματος ψευδοεπιτάχυνσης PSa για σχετικά μικρό μέγεθος χρονικού βήματος (Δt=10⁻⁴ s) μεταξύ του OpenSeismoMatlab και του SeismoSignal.
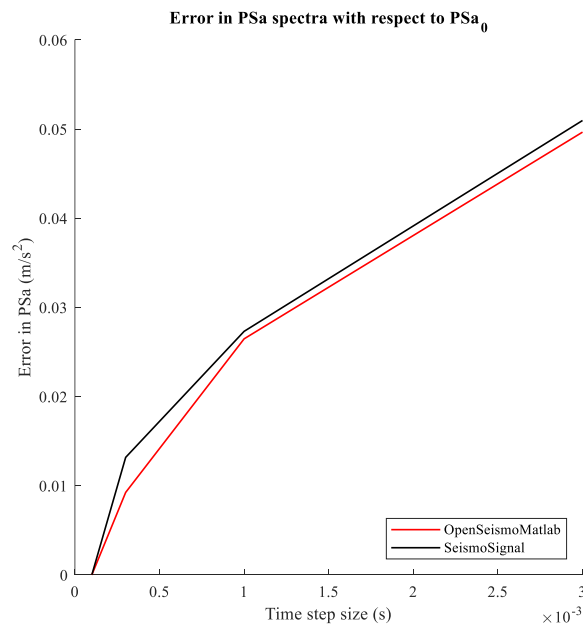
Εικόνα 0.12: Σύγκριση του σφάλματος του φάσματος ψευδοεπιτάχυνσης (PSa) σε σχέση με το PSa_0, μεταξύ του OpenSeismoMatlab και του SeismoSignal.

0.4. Μια νέα διαδικασία βέλτιστου σχεδιασμού των κατασκευών εναντι σεισμικών φόρτισεων με βαση τη σεισμική ενέργεια

0.4.1. Εισαγωγή

Στο κεφάλαιο αυτό εισάγεται μια διαδικασία για το βέλτιστο σχεδιασμό των μη γραμμικών επίπεδων διατμητικών κτιρίων με τη χρήση μιας μεθόδου κλίσης βασισμένης σε ισοδύναμες γραμμικές κατασκευές, αντί για τη συνηθισμένη μέθοδο υπολογισμού της κλίσης από την αντικειμενική συνάρτηση που απαιτεί την επίλυση των μη γραμμικών κτιρίων. Το πρόβλημα της βελτιστοποίησης διατυπώνεται με τη μορφή ενός ισοδύναμου γραμμικού συστήματος εξισώσεων στο οποίο οι συνιστώσες της αντικειμενικής συνάρτησης είναι μια στοχευόμενη θεμελιώδ ιδιοσυχνότητα και μια ομοιόμορφη κατανομή της ενέργειας απόσβεσης καθ' ύψος του κτιρίου. Η ως άνω διαδικασία εφαρμόζεται με τη μέθοδο Newton Raphson για την εύρεση της βέλτιστης κατανομής δυσκαμψίας καθ' ύψος για δυο αντιπροσωπευτικά πολυβάθμια διατμητικά κτίρια, τόσο γραμμικά όσο και μη γραμμικά, ώστε η αποσβεννύμενη ενέργεια κατα τη διάρκεια μιας σεισμικής διέγερσης (τόσο ιξωδοελαστικής μορφής όσο και υστερητικής μορφής) να είναι ομοιόμορφη κατα το ύψος της κατασκευής. Εξετάζεται η επιρροή της σεισμικής διέγερσης, του ποσοστού της κρίσιμης απόσβεσης και του κανονικοποιημένου ορίου διαρροής της σχετικής μετακίνησης μεταξύ διαδοχικών ορόφων στα αποτελέσματα της διαδικασίας βελτιστοποίησης. Ο σχεδιασμός με βάση την προτεινόμενη μέθοδο βελτιστοποίησης είναι περισσότερο λογικός και κατασκευαστικά εφαρμόσιμος σε σύγκριση με άλλες στρατηγικές βελτιστοποίησης (π.χ. βελτιστος σχεδιασμός με βαση την ομοιόμορφη πλαστιμότητα καθ' ύψος), ενώ αναμένεται να οδηγήσει σε αυξημένη προστασία της κατασκευή έναντι ολικής κατάρρευσης και της συνεπαγόμενης απώλειας έμψυχου και άψυχου υλικού κατα τη διάρκεια ισχυρών σεισμικών εδαφικών κινήσεων. Τέλος, περαν των ανωτέρω, αποδεικνύεται ότι η νέα μέθοδος βελτιστοποίησης όχι μόνο μειώνει τον απαιτούμενο χρόνο επίλυσης έως και 91% σε σύγκριση με την κλασσική μέθοδο Newton Raphson, αλλά μπορεί επίσης να εφαρμοστεί και σε άλλα προβλήματα βελτιστοποίησης, όπου η προχώρηση σε κάθε βήμα γίνεται με υπολογισμό της κλίσης της αντικειμενικής συνάρτησης.

Κατα το σύγχρονο σχεδιασμό των κατασκευών έναντι στατικής ή/και δυναμικής φόρτισης είναι επιθυμητή η απόκριση της κατασκευής στην πλαστική περιοχή, καθότι η θεώρηση αυτή οδηγεί σε πιο οικονομικό σχεδιασμό. Ειδικότερα στον αντισεισμικό σχεδιασμό, η ανελαστική συμπεριφορά είναι παραδεκτή εντός συγκεκριμένων ορίων, που καθορίζονται από μια ισορροπία μεταξύ ασφάλειας και οικονομίας. Στο παρελθόν πολλές κατασκευές έχουν αντέξει δυνάμεις μεγαλύτερες από εκείνες που υπολογίζονται με παραδοχή γραμμικά ελαστικής απόκρισης. Η πλαστιμότητα είναι εκείνη η ποσότητα που ρυθμίζει το πρώτο και δικαιολογεί το δεύτερο. Ωστόσο, η χρονοιστορία της φόρτισης μιας κατασκευής παίζει σημαντικό ρόλο, πέραν της πλαστιμότητάς της, και θα πρέπει να

λαμβάνεται υπόψη. Υπάρχουν διάφορες ποσότητες που αυξάνουν αθροιστικά στην κατασκευή κατα τη διάρκεια μιας σεισμικής διέγερσης, και παρέχουν πληροφορίες για το πως συμπεριφέρθηκε η κατασκευή, όπως π.χ. η πλαστική ενέργεια που απορροφάται από τα δομικά στοιχεία. Η τελευταία είναι ένα χρήσιμο μέτρο της βλάβης που υφίσταται μια κατασκευή, και ειδικά για τις κατασκευές από ωπλισμένο σκυρόδεμα. Συμπερασματικά, θα πρέπει να γίνει κατανοητό ότι ο αντισεισμικός σχεδιασμός πρεπει να εξαρτάται όχι μόνο από μέγιστα σε συγκεκριμένες χρονικές στιγμές κατα τη διάρκεια της σεισμικής φόρτισης (π.χ. μέγιστη επιτάχυνση) αλλά θα πρέπει επίσης να εξαρτάται και από τη χρονοιστορία της φόρτισης. Στα ανωτέρω βασίζεται ο σχεδιασμός βασιζόμενος στην ενέργεια απόσβεσης (Energy Based Design, EBD). Σύμφωνα με αυτόν, μια κατασκευή πρέπει να σχεδιάζεται ώστε να έχει την ικανότητα να απορροφά μεγαλύτερες ποσότητες σεισμικής ενέργειας από αυτές που θα εισηχθούν σε αυτή μέσω του σεισμού, ώστε να μπορεί να αποκριθεί αποτελεσματικά σε σεισμικές φορτίσεις.

Εκτός από την πλαστιμότητα του υλικού κατασκευής, η σεισμική απόκριση μιας κατασκευής επηρεάζεται και από τη μορφολογία της καθώς και από την κατανομή της δυσκαμψίας και της αντοχής σε αυτή. Οι περισσότερες καταρρεύσεις κατασκευών στο παρελθόν έχουν συμβεί εως ένα βαθμό εξαιτίας εσφαλμένης κατασκευαστικής διαμόρφωσης. Ο σχηματισμός μαλακών ορόφων είναι χαρακτηριστικό παράδειγμα ανεπαρκούς απόκρισης της κατασκευής, όπου σε έναν όροφο παρατηρούνται υπερβολική πλαστιμότητα και σχετική μετακίνηση, οι οποίες οδηγούν σε τοπική κατάρρευση. Τα περισσότερα κτίρια σχεδιάζονται σύμφωνα με την έννοια των ισοδύναμων στατικών δυνάμεων που προδιαγράφονται από αντισεισμικούς κανονισμούς. Η καθ᾽ ύψος κατανομή αυτών των δυνάμεων προκύπτει από την υπόθεση ότι οι ιδιομορφές ταλάντωσης της κατασκευής είναι γραμμικά ελαστικές. Αφετέρου, σύμφωνα με το σχεδιασμό βασιζόμενο στην ενέργεια απόσβεσης (EBD) υποτίθεται ότι η κατασκευή αποκρίνεται μη γραμμικά, συνεπώς ο υπολογισμός των ιδιομορφών που βασίζεται σε γραμμικά ελαστικό φορέα δεν οδηγεί σε ρεαλιστικό υπολογισμό των ισοδύναμων στατικών σεισμικών δυνάμεων, και συνεπώς δεν διασφαλίζει τη βέλτιστη σεισμική απόκριση, ούτε καν πολλές φορές την ασφάλεια.

Στο κεφάλαιο 4 παρουσιάζεται ένας νέος επαναληπτικός αλγόριθμος βελτιστοποίησης με δυνατότητα γραμμικής αναζήτησης, ειδικά σχεδιασμένος για βελτιστοποίηση σχεδιασμού κατασκευών, ο οποίος χρησιμοποιείται για το βέλτιστο σχεδιασμό με όρους απορροφώμενης ενέργειας από την κατασκευή για έναν αριθμό επιταχυνσιογραφημάτων. Ο εν λόγω αλγόριθμος βρίσκει τη βέλτιστη κατανομή της δυσκαμψίας καθ᾽ υψος ενός κτιρίου ούτως ώστε να ελαχιστοποιείται η διακύμανση της ενέργειας απόσβεσης καθ᾽ υψος του κτιρίου (ήτοι αυτή να είναι ομοιόμορφη με τιμή ίση με το μέσο όρο της). Η διαδικασία βελτιστοποίησης εφαρμόζεται τόσο σε γραμμικά όσο και σε μη γραμμικά κτίρια. Διερευνάται η επιρροή της θεμελιώδους ιδιοπεριόδου, του αριθμού των ορόφων και της κατανομής ιξωδοελαστικής απόσβεσης στην βέλτιστη κατανομή της αντοχής καθ᾽ ύψος του κτιρίου.

0.4.2. Προτεινόμενος αλγόριθμος βέλτιστου σχεδιασμού

Ο προτεινόμενος αλγόριθμος βελτιστοποίησης εφαρμόζει μια τροποποιημένη μέθοδο για την εύρεση του βέλτιστου σημείου που ελαχιστοποιεί μια αντικειμενική συνάρτηση (εν προκειμένω την τυπική απόκλιση της κατανομής της ενέργειας απόσβεσης καθ' ύψος του κτιρίου). Οι μέθοδοι βελτιστοποίησης που βασίζονται σε κατεύθυνση με βαση τις παραγώγους της αντικειμενικής συνάρτησης σε κάθε βήμα, προσεγγίζουν το βέλτιστο σημείο επαναληπτικά. Σε κάθε βήμα, προκειμένου να γίνει η μετάβαση στο επόμενο σημείο από το τρέχον σημείο, χρειάζεται να υπολογιστεί τόσο η κατεύθυνση προς την οποία βρίσκεται το επόμενο σημείο, όσο και η απόστασή του από το τρέχον. Στην περίπτωση του αλγορίθμου NR, η κατεύθυνση υπολογίζεται με βάση τις μερικές παραγώγους της αντικειμενικής συνάρτησης στο τρέχον σημείο, ως προς τις μεταβλητές σχεδιασμού. Στις επριπτώσεις βελτιστοποίησης μη γραμμικών κατασκευών, ο υπολογισμός των παραγώγων απαιτεί την εκτέλεση πολλαπλών μη γραμμικών αναλύσεων, που είναι γενικά υπολογιστικά απαιτητικά. Για το λόγο αυτό προτείνεται ο υπολογισμός των παραγώγων να γίνεται με βάση την αντικειμενική συνάρτηση εφαρμοζόμενη σε μια ισοδύναμη γραμμικά ελαστική κατασκευή, που απαιτεί πολύ λιγότερο υπολογιστικό φόρτο σε σχέση με τη μη γραμμική. Η ισοδύναμη γραμμικά ελαστική κατασκευή προκύπτει αν τεθεί το όριο διαρροής της μη γραμμικής κατασκευής ίσο με μια πολύ μεγάλη τιμή (ήτοι πρακτικά άπειρη), και αν στον υπολογισμό της αντικειμενικής συνάρτησης αντί για την υστερητική ενέργεια απόσβεσης ληφθεί υπόψη η ιξωδοελαστική ενέργεια απόσβεσης που προκύπτει από τον ισοδύναμα γραμμικά ελαστικό φορέα. Παρόλη τη μικρή απώλεια σε ακρίβεια που υπάρχει με τον ανωτέρω προτεινόμενο αλγόριθμο, αποδεικνύεται ότι η συγκεκριμένη πρακτική επιταχύνει κατα πολύ τη διαδικασία βελτιστοποίησης και τείνει στο ίδιο βέλτιστο σημείο.

0.4.3. Τυπικές κατανομές απόσβεσης υστερητικής ενέργειας σε διατμητικά κτίρια

Τυπικές κατανομές της ενέργειας που αποσβέννεται εξαιτίας υστερητικής συμπεριφοράς ενός 5-όροφου και ενός 10-όροφου κτιρίου κατα τη διάρκεια ελαστοπλαστικής απόκρισης, φαίνονται στην εικόνα 0.13. Υποτίθεται ότι αμφότερα τα κτίρια έχουν ομοιόμορφες κατανομές δυσκαμψίας καθ' ύψος, οι οποίες είναι κανονικοποιημένες ώστε τα δυο κτίρια να έχουν θεμελιώδεις ιδιοσυχνότητες ίσες με 2 Hz και 1 Hz αντίστοιχα. Όπως παρατηρείται και την πράξη, οι μεγαλύτερες ποσότητες ενέργειας απόσβεσης εμφανίζονται στους κατώτερους ορόφους των κτιρίων, για όλες τις περιπτώσεις σεισμικών καταγραφών. Οι εν λόγω κατανομές υποδεικνύουν σαφώς τους λόγους που οι βλάβες κατα τη διάρκεια ενός σεισμού συσσωρεύονται στους κατώτερους ορόφους, και εξηγούν το λόγο που οι μηχανισμοί μαλακού ορόφου αναπτύσσονται τις πιο πολλές φορές σε αυτές τις περιοχές. Το φαινόμενο αυτό είναι ανεπιθύμητο, και ως εκ τούτου υπάρχει η ανάγκη για ισοκατανομή της ενέργειας απόσβεσης σε

όλους τους ορόφους, ούτως ώστε να υπάρχει όσο το δυνατόν περισσότερη εκμετάλλευση του κατασκευαστικού υλικού και να μεγιστοποιείται έτσι η ασφάλεια του εκάστοτε κτιρίου.
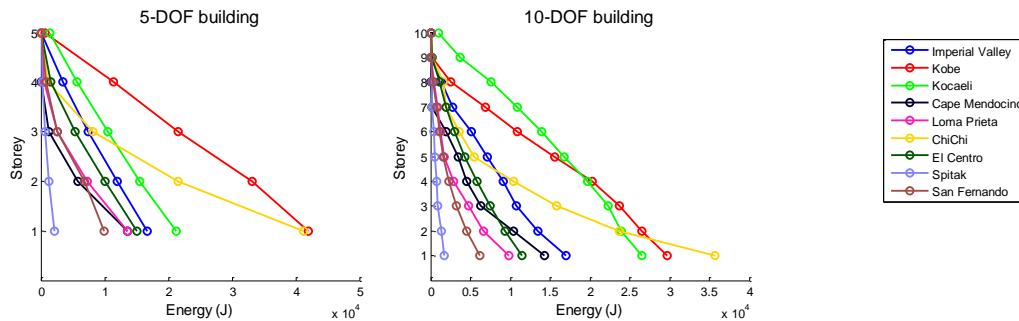

Εικόνα 0.13: Κατανομές της ενέργειας που αποσβέννεται λόγω υστέρησης για το 5- και 10-όροφο κτίριο με ομοιόμορφη κατανομή δυσκαμψίας καθ' ύψος, ξ=5%, ūγ=0.01 και θεμελιώδεις ιδιοσυχνότητες 2Hz και 1Hz αντίστοιχα, για διάφορες σεισμικές καταγραφές.

0.4.4. Βέλτιστες κατανομές δυσκαμψίας για γραμμικά ελαστικά κτίρια

Οι βέλτιστες κατανομές δυσκαμψίας για τις περιπτώσεις του 5-όροφου κτιρίου και 10-όροφου επίπεδου διατμητικού κτιρίου φαίνονται στις εικόνες 0.14 και 0.15. Αυτές αναφέρονται στις επιθυμητές θεμελιώδεις ιδιοσυχνότητες που έχουν προσδιοριστεί για κάθε κτίριο (2 Hz και 1 Hz αντίστοιχα) και διάφορες σεισμικές διεγέρσεις. Είναι εμφανές ότι οι κατανομές δυσκαμψίας είναι ομαλά μεταβαλλόμενες καθ' ύψος και έχουν τη μέγιστη τιμή τους στον 1ο όροφο και την ελάχιστη στον τελευταίο όροφο καθε κτιρίου. Επίσης, η κατανομή της δυσκαμψίας είναι σχεδόν ανεξάρτητη από την εκάστοτε σεισμική διέγερση, και έχει μεγαλύτερες τιμές για το 5-όροφο κτίριο σε σχέση με το 10-όροφο.
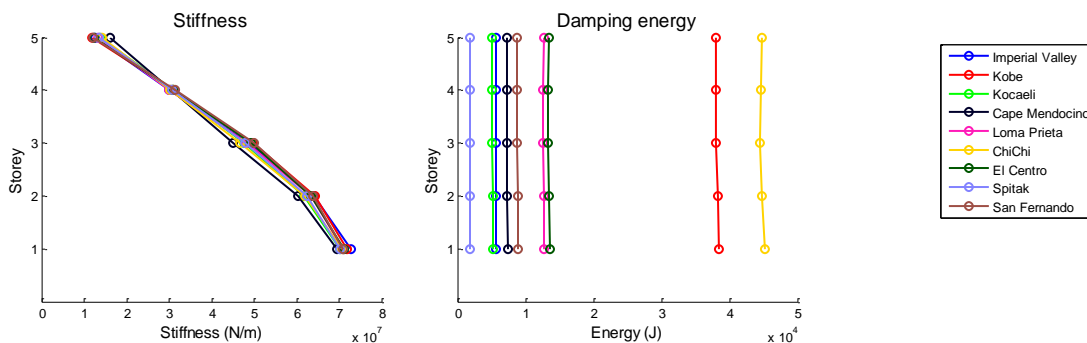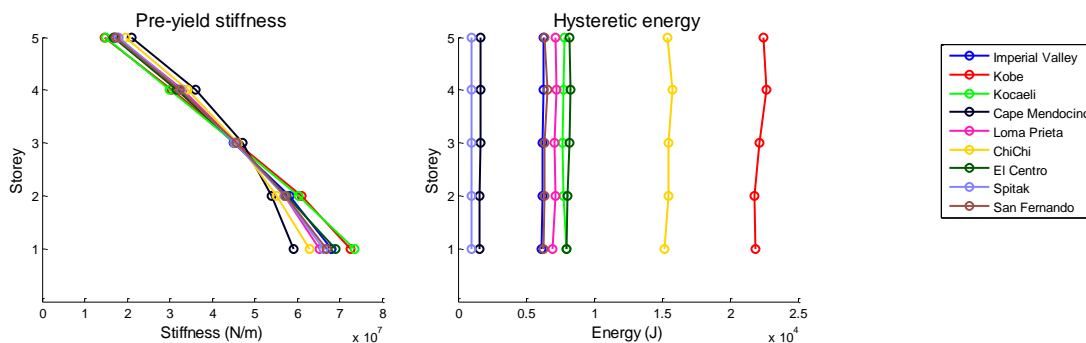

Εικόνα 0.14: Βέλτιστες κατανομές δυσκαμψίας σε γραμμικά ελαστικό 5-όροφο κτίριο και αντίστοιχες κατανομές ενέργειας απόσβεσης με ξ=5% για διάφορες σεισμικές διεγέρσεις.
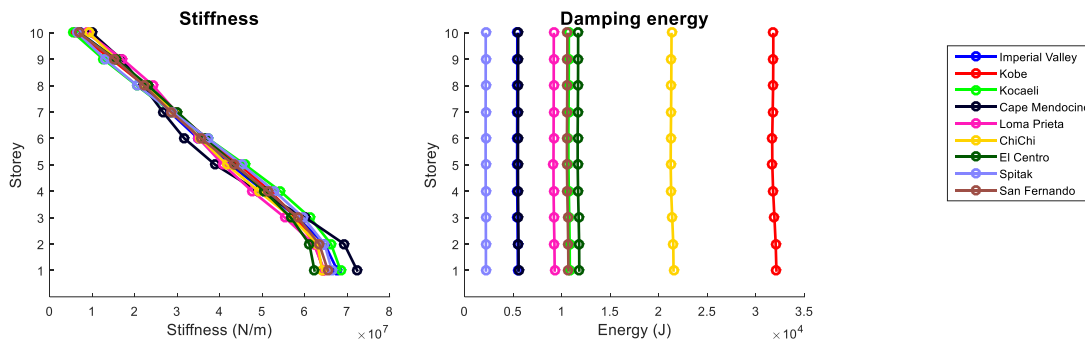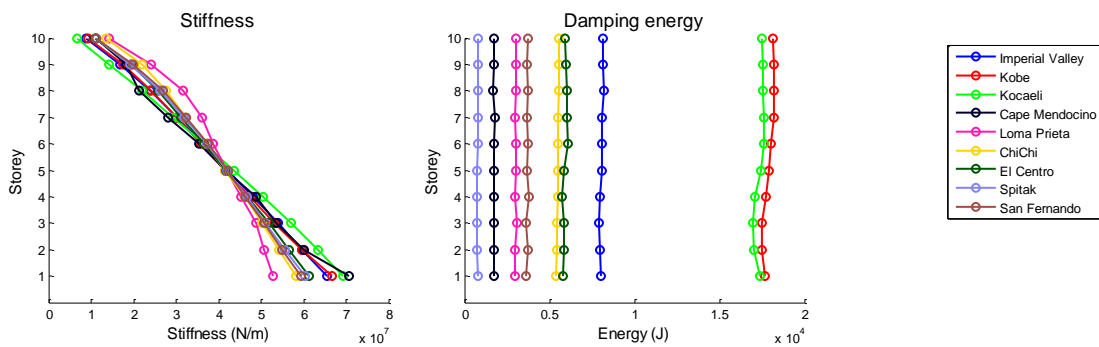
Εικόνα 0.15: Βέλτιστες κατανομές δυσκαμψίας σε γραμμικά ελαστικό 10-όροφο κτίριο και αντίστοιχες κατανομές ενέργειας απόσβεσης με ξ=5% για διάφορες σεισμικές διεγέρσεις.

0.4.5. Βέλτιστες κατανομές δυσκαμψίας για ελαστοπλαστικά κτίρια

Στην περίπτωση ελαστοπλαστικών κτιρίων, λαμβάνεται υπόψη μια επιπρόσθετη παράμετρος, που είναι το κανονικοποιημένο όριο διαρροής σε όρους μετατόπισης μεταξύ δυο διαδοχικών ορόφων, που ορίζεται στην εξίσωση (4.16). Σημειώνεται ότι η σεισμική δύναμη για την οποία σχεδιάζεται η κατασκευή υπολογίζεται από τη βέλτιστη κατανομή της δυσκαμψίας πολλαπλασιασμένη με την ομοιόμορφη κατανομή του συντελεστή διαρροής μεταξύ διαδοχικών ορόφων. Οι θεμελιώδεις ιδιοσυχνότητες μικρών παραμορφώσεων των δυο κτιρίων είναι ίδιες με αυτές των αντίστοιχων γραμμικά ελαστικών. Οι βέλτιστες κατανομές δυσκαμψίας μικρών παραμορφώσεων και οι αντίστοιχες κατανομές ενέργειας υστερητικής απόσβεσης φαίνονται στις εικόνες 0.16 και 0.17 για το 5-όροφο και το 10-όροφο κτίριο αντίστοιχα. Παρατηρείται ότι γενικά οι κατανομές δυσκαμψίας μειώνονται όσο αυξάνει το ύψος του κτιρίου, πράγμα που παρατηρήθηκε επίσης και στην περίπτωση των γραμμικά ελαστικών κτιρίων. Είναι αξιοσημείωτο ότι υπάρχει μια γενική μορφή οιονεί γραμμικής κατανομή δυσκαμψίας, την οποία ακολουθούν όλες οι κατανομές δυσκαμψίας για τις διάφορες περιπτώσεις σεισμικών καταγραφών, για αμφότερα τα δυο κτιρια που μελετωνται.

Εικόνα 0.16: Βέλτιστες κατανομές δυσκαμψίας μικρών παραμορφώσεων και ενέργεια υστερητικής απόσβεσης για το 5-όροφο διατμητικό κτίριο με ξ=5%, ūy=0.1 για διάφορες σισμικές καταγραφές.

Εικόνα 0.17: Βέλτιστες κατανομές δυσκαμψίας μικρών παραμορφώσεων και ενέργεια υστερητικής απόσβεσης για το 10-όροφο διατμητικό κτίριο με ξ=5%, ūy=0.04 για διάφορες σισμικές καταγραφές.

0.4.6. Απόδοση της νέας μεθόδου βελτιστου σχεδιασμού

Για κάθε καινούριο αλγόριθμο βελτιστοποίησης, τίθεται το θέμα της απόδοσής του, της ταχύτητάς του κλπ, σε σχέση με τους υπόλοιπους γνωστούς αλγορίθμους που χρησιμοποιούνται για ενα συγκεκριμένο πρόβλημα βελτιστοποίησης. Επισημαίνεται ότι η νεα μέθδοος βελτιστοποίησης μπορεί να εφαρμοστεί για οποιοδήποτε μη γραμμικό πρόβλημα, αλλά για τα δεδομένα της παρούσης διατριβής θα περιοριστούμε στο πρόβλημα του βέλτιστου σχεδιασμού επίπεδων διατμητικών κτιρίων, αυτών που εξετάστηκαν παραπάνω. Στην Εικόνα 0.18 φαίνεται η εξέλιξη της τυπικής απόκλισης της κατανομής ενέργειας υστερητικής απόσβεσης ως συνάρτηση του κανονικοποιημένου χρόνου εκτέλεσης του αλγορίθμου για το 5-όροφο και το 10-όροφο διατμητικό κτίριο που μελετήθηκαν παραπάνω, με fo=2Hz, ξ=0.05, ūy=0.1 και fo=1Hz, ξ=0.05, ūy=0.04 αντίστοιχα, τα οποία διεγείρονται από καταγραφή του σεισμού El Centro. Παρατηρείται ότι οι χρόνοι εκτέλεσης των αλγορίθμων Newton Raphson (NR) με γραμμικές παραγώγους είναι πολύ μικρότεροι από αυτούς με μη γραμμικές παραγώγους. Η κανονικοποίηση των χρόνων εκτέελσης έγινε με βάση το χρόνο εκτέλεσης του αντίστοιχου αλγορίθμου με μη γρμαμικές παραγώγους, και ως εκ τούτου οι κανονικοποιημένοι χρόνοι των αλγορίθμων με μη γραμμικές παραγώγους είναι αμφότεροι ίσοι με 100%.

Φαίνεται ξεκάθαρα ότι ο προτεινόμενος αλγόριθμος είναι έως και 11 φορές πιο γρήγορος από τον παραδοσιακό αλγόριθμο NR για το 10-όροφο κτίριο και έως και 7 φορές πιο γρήγορος για το 5-όροφο κτίριο. Για μεγαλύτερο αριθμό ορόφων, ο προτεινόμενος αλγόριθμος αναμένεται να είναι ακόμα πιο γρήγορος σε σχέση με τον συνηθισμένο NR, κάνοντας έτσι οικονομία υπολογιστικού φόρτου. Σημειώνεται εδώ ότι για να απομονωθεί η επιρροή του αρχικού σημείου κατα τη σύγκριση των δυο αλγορίθμων, το αρχικό σημείο είναι ίσο με τη βέλτιστη κατανομή δυσκαμψίας για το γραμμικά ελαστικό κτίριο για ξ=0.05. Με τον τρόπο αυτό, δεδομένου ότι οι αλγόριθμοι εκκινούν από το ίσιο αρχικό σημείο για να

επιλύσουν το ίδιο πρόβλημα (σε όρους σεισμικής καταγραφής που εξετάζεται και των διαφόρων ιδιοτήτων της κατασκευής), η διαφορά του χρόνου εκτέλεσης οφείλεται μόνον από τη φύση του αλγορίθμου και τις ιδιότητές του. Τα αποτελέσματα των διαδικασιών βελτιστοποίησης που συγκρίθηκαν φαίνονται στην Εικόνα 0.18 και στον Πίνακα 0.2.

Παρατηρείται ότι ο προτεινόμενος αλγόριθμος NR, ενώ χρειάζεται σχεδόν τον ίδιο αριθμό επαναλήψεων με τον κλασσικό NR, μπορεί να μειώσει το χρόνο εκτέλεσης κατα 85% στην περίπτωση του 5-όροφο κτιρίου και κατα 91% στην περίπτωση του 10-όροφο κτιρίου. Η μείωση στο χρόνο εκτέλεσης αναμένεται να είναι μεγαλύτερη για κτίρια με μεγαλύτερο αριθμό ορόφων, ή κατασκευές με μεγαλύτερο αριθμό βαθμών ελευθερίας γενικά. Ως αποτέλεσμα αυτού, για πολυπλοκότερες κατασκευές, η χρήση του ρποτεινόμενου αλγορίθμου θα οδηγήσει σε μεγαλύτερη οικονομία χρόνου. Σημειώνεται αναφορικά με τις παραπάνω συγκρίσεις, ότι οι τελικές βέλτιστες κατανομές δυσκαμψίας μικρών παραμορφώσεων είναι ίδιες τόσο για τον κλασσικό αλγόριθμο NR, όσο και για τον προτεινόμενο αλγόριθμο NR.
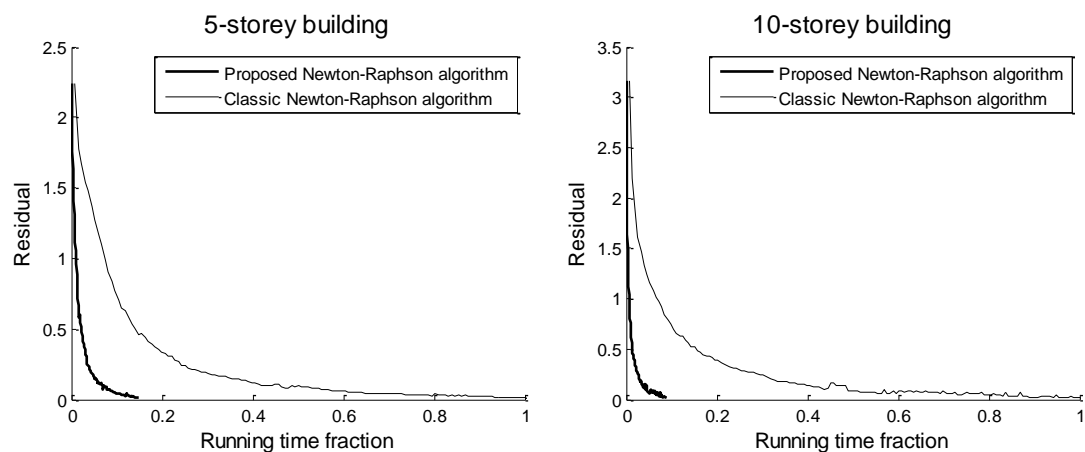
Εικόνα 0.18: Εξέλιξη της τυπικής απόκλισης της κατανομής ενέργειας υστερητικής απόσβεσης για διάφορες διαδικασίες βελτιστοποίησης για τον κλασσικό αλγόριθμο NR και τον προτεινόμενο αλγόριθμο NR, για 5- και 10-όροφα διατμηατικά κτίρια με fo=2 Hz, ξ=5%, ūy=0.1 και fo=1 Hz, ξ=5%, ūy=0.04 αντίστοιχα, διεγειρόμενα από σεισμική καταγραφή του σεισμού El Centro.

Περίπτωση	Κανονικοποιημένος χρόνος εκτέλεσης	Μείωση	Επαναλήψεις
5-όροφο, Newton-Raphson	100%	-	136
5-όροφο, proposed algorithm	14.9%	85.1%	135
10-όροφο, Newton-Raphson	100%	-	153
10-όροφο, proposed algorithm	8.8%	91.2%	143

Πίνακας 0.2: Numerical results of the optimization processes the evolution of which is presented in Figure 4-14

0.4.7. Συμπεράσματα

Τα κυριότερα συμπεράσματα είναι τα ακόλουθα:

- Αποδεικνύεται ότι υπάρχουν μοναδικές κατανομές δυσκαμψίας που αντιστοιχούν σε ομοιόμορφα κατανεμημένη ενέργεια ιξωδολεαστικής και υστερητικής απόσβεσης για γραμμικά ελαστικά και ελαστοπλαστικά επίπεδα διατμητικά κτίρια αντίστοιχα.

- Η βέλτιστη κατανομή δυσκαμψίας τόσο για ελαστικά όσο και για ελαστοπλαστικά κτίρια εμφανίζεται ως οιονεί γραμμική (με ελάχιστη καμπυλότητα), με τη μέγιστη τιμή στον πρώτο όροφο και την ελάχιστη τιμή στον τελευταίο όροφο του εκάστοτε κτιρίου. Το σχήμα αυτό των βέλτιστων κατανομών είναι γενικά ανεξάρτητο της σεισμικής διέγερσης και προσφέρει τη δυνατότητα υπολογισμού της με απλές μεθόδους.

- Ο σχεδιασμός των κατασκευών με βάση την προτεινόμενη μεθοδολογία είναι πιο λογικός και τεχνικά εφικτός σε σύγκριση με το σχεδιασμό που έχει στόχο την ομοιόμορφη κατανομή πλαστιμότητας, ενώ αναμένεται να οδηγήσει σε κατασκευές με μεγαλύτερη ασφάλεια και προστασία έναντι ολικής κατάρρευσης και απωλειών κατα τη διάρκεια ισχυρών σεισμών.

Τέλος, αποδεικνύεται ότι ο νέος προτεινόμενος αλγόριθμος που ακολουθεί κατεύθυνσεις για την προσέγγιση του βέλτιστου σημείου με βαση τις παραγώγους της ισοδύναμης γραμμικής κατασκευής, εφοδιασμένος με ένα σταθεροποιητή με σκοπό τη βελτιστοποίηση μη γραμμικών προβλημάτων επιτυγχάνει ουσιαστικό κέρδος σε υπολογιστικό φόρτο, μολονότι απαιτεί τον ίδιο σχεδον αριθμό επαναλήψεων προκειμένου να συγκλίνει. Η νέα μεθοδολογία βελτιστοποίησης που παρουσιάζεται στην παρούσα διατριβή μπορεί να εφαρμοστεί για τη βελτιστοποίηση οποιασδήποτε μη γραμμικής κατασκευής, καθώς και να εισαχθεί σαν έννοια σε άλλους γνωστούς αλγορίθμους, πέραν του NR, όπως π.χ. αλγορίθμους γραμμικής αναζήτησης, περιοχής εμπιστοσύνης, μεθόδους απότομης κατάβασης, συζυγών κλίσεων, Broyden, κλπ.

0.5. Αναπτυξη βελτιωμένων ενεργειακά συμβατών τεχνητών καταγραφών με μεθόδους βέλτιστου γραμμικού συνδυασμού

0.5.1. Εισαγωγή

Προτείνεται μια νέα μεθοδολογία για την παραγωγή τεχνητών σεισμογραφημάτων τα οποία συμβαδίζουν με δεδομένα φάσματα ελαστικής απόκρισης και ενέργειας απόσβεσης. Με τη μεθοδολογία αυτή χρησιμοποιείται ένας γραμμμικός συνδυασμός ακατέργαστων καταγραφών σεισμικής εδαφικής κίνησης, προκειμένου να παραχθεί ένα τεχνητό επιταχυνσιογράφημα το οποίο θα συμμορφώνεται με έναν αριθμό στοχευόμενων φασμάτων, που λαμβάνουν υπόψη όχι μόνο τη φασματική επιτάχυνση, αλλά και τη φασματική ισοδύναμη ταχύτητα σεισμικής ενέργειας. Η θεώρηση του φάσματος ισοδύναμης ταχύτητας σεισμικής

ενέργειας οδηγεί στην ανάπτυξη τεχνητών επιταχυνσιογραφημάτων που είναι πιο ρεαλιστικά σε σχέση με αυτά που λαμβάνουν υπόψη τους μόνο το φάσμα ελαστικής απόκρισης. Η διαδικασία γραμμικού συνδυασμού καταγραφών από μια δεδομένη δεξαμενή προκειμένου η παραγόμενη καταγραφή να συμμορφώνεται με δεδομένα στοχευόμενα φάσματα διατυπώνεται σε όρους προβλήματος βελτιστοποίησης. Για την επαλήθευση της αποτελεσματικότητας του αλγορίθμου, επιλέχθηκαν χαρακτηριστικές εδαφικές κινήσεις με διαφορετικά χαρακτηριστικά και ιδιότητες, τα φάσματα των οποίων θεωρήθηκαν ως στοχευόμενα φάσματα. Πέραν αυτού, η διερεύνηση της αποτελεσματικότητας της προτεινόμενης μεθοδολογίας έγινε και αναφορικά με τη σεισμική απόκριση μονοβάθμιων και πολυβάθμιων συστημάτων, μεσω του ποσοστού της σεισμικής ενέργειας που αποσβέννεται εντός της κατασκευής. Αποδεικνύεται ότι υπάρχει άριστη συμφωνία μεταξύ αφενός των φασμάτων των τεχνητών καταγραφών που προκύπτουν από βέλτιστο γραμμικό συνδυασμό ανεπεξέργαστων σεισμικών καταγραφών και αφετέρου των στοχευόμενων φασμάτων, πράγμα που επαληθεύει την αξιοπιστία της προτεινόμενης μεθοδολογίας.

Παραδοσιακά, ο αντισεισμικός σχεδιασμός των κατασκευών βασίζεται σε μεθόδους δυνάμεων ή/και μετατοπίσεων, στις οποίες η επιρροή της σεισμικής φόρτισης ποσοτικοποιείται χρησιμοποιώντας τη μέγιστη επιτάχυνση χρονοιστορίας ή τη μέγιστη φασματική επιτάχυνση της εκάστοτε σεισμικής καταγραφής. Με την επιλογή αυτή δεν λαμβάνεται υπόψη η επίδραση της χρονοιστορίας φόρτισης της κατασκευής, παρα μόνο ένα σημείο αυτής, που αντιστοιχεί στο μέγιστο. Ετσι σημαντικό μέρος της πληροφορίας της σεισμικής απόκρισης της κατασκευής χάνεται κατα την ανωτέρω διαδικασία σχεδιασμού. Η ανωτέρω απώλεια πληροφορίας αντανακλάται στο σφάλμα που ενυπάρχει συχνά κατα την παραγωγή τεχνητών σεισμογραφημάτων τα οποία παραδοσιακά συμμορφώνονται μόνο με φασματα επιτάχυνσης. Η παραγωγή σεισμογραφημάτων τα οποία, περαν των ανωτέρω φασμάτων, θα συμμορφώνονται επιπλέον με φάσματα ισοδυναμης ταχύτητας σεισμικής ενέργειας θα λαμβάνει υπόψη της την επιρροή της χρονοιστορίας φόρτισης μεσω των τελευταίων, και οδηγεί έτσι σε πιο ρεαλιστικά σεισμογραφήματα.

0.5.2. Αλγόριθμος δημιουργίας τεχνητών επιταχυνσιογραφημάτων

0.5.2.1. Επεξεργασία ακατέργαστων σεισμικών εδαφικών κινήσεων

Ένας γραμμικός συνδυασμός πραγματικών επιταχυνσιογραφημάτων απαιτεί μόνο την επιλογή και στάθμιση των τελευταίων, ενώ δεν αλλοιώνει τα εσωτερικά τους χαρακτηριστικά, όπως π.χ. μη στασιμότητα, υστερούσες κυματοσειρές (coda), φασικό περιεχομενο, κλπ. τα οποία θα πρέπει να διατηρούνται όσο το δυνατόν καλύτερα με στόχο τη δημιουργία όσο το δυνατόν πιο ρεαλιστικών τεχνητών καταγραφών, ως αποτέλεσμα γραμμικού συνδυασμού. Δεδομένου ότι οι πραγματικές καταγραφές δεν έχουν πάντα την ίδια διάρκεια, η διαδικασία του

γραμμικού συνδυασμού δεν μπορεί να εφαρμοστεί άμεσα σε αυτές. Ωστόσο, μπορεί να εφαρμοστεί στα φάσματα Fourier που προκύπτου από τον ομώνυμο μετασχηματισμό τους, τα οποία έχουν το ίδιο μήκος (διάρκεια). Η γραμμικά συνδυασμένη τεχνητή καταγραφή μπορεί να ληφθεί από τον αντίστροφο μετασχηματισμό Fourier του γραμμικού συνδυασμού των φασμάτων Fourier των επιμέρους πραγματικών καταγραφών (οι οποίες λαμβάνονται από δεδομένη «δεξαμενή»). Είναι προφανές ότι ένας αντίστροφος μετασχηματισμός Fourier απαιτεί την τιμή του χρονικού βήματος με βαση το οποίο έγινε ο μετασχηματισμός Fourier των πραγματικών καταγραφών πριν το γραμμικό συνδυασμό τους, όπως παρουσιάζεται παραπάνω. Το μέγεθος αυτό του χρονικού βήματος πρεπει να ειναι ίδιο με αυτό που χρησιμοποιείται για το μετασχηματισμό Fourier των καταγραφών που επιλέγονται για το γραμμικό συνδυασμό. Για το λόγο αυτό, κάθε καταγραφή υφίσταται επεξεργασία κατα την οποία γίνεται επαναδειγματισμός της, με το χρονικό βήμα που χρησιμοποιείται για τους μετασχηματισμούς Fourier. Συνεπώς, πριν το γραμμικό συνδυασμό τους, οι καταγραφές επαναδειγματίζονται και εν συνεχεία μετασχηματίζονται κατα Fourier, προκειμένου να εξασφαλιστεί η ομοιογένεια τους κατα το γραμμικό συνδυασμό τους.

0.5.2.2. Προσδιορισμός βέλτιστου γραμμικού συνδυασμού για παραγωγή τεχνητών επιταχυνσιογραφημάτων

Το πρόβλημα της παραγωγής ενός τεχνητού επιταχυνσιογραφήματος που συμμορφώνεται με δεδομένα στοχευόμενα φάσματα, διατυπώνεται ως πρόβλημα βελτιστοποίησης, στο οποίο η αντικειμενική συνάρτηση που ελαχιστοποιείται είναι το σφάλμα μεταξύ των στοχευόμενων φασμάτων και των αντίστοιχων φασμάτων τα οποία επιτυγχάνει ένας οποιοσδήποτε γραμμικός συνδυασμός καταγραφών. Η αντικειμενική συνάρτηση που χρησιμοποιείται είναι της μορφής:

$$f_{Sa-Siev} = \int_{T_1}^{T_2} \left(\left| \frac{Sa_c(T) - Sa_t(T)}{Sa_t(T)} \right| + \left| \frac{Siev_c^{ABS}(T) - Siev_t^{ABS}(T)}{Siev_t^{ABS}(T)} \right| + \left| \frac{Siev_c^{REL}(T) - Siev_t^{REL}(T)}{Siev_t^{REL}(T)} \right| \right) p(T) dT$$

και αποτελείται από ένα άθροισμα της επιφάνειας μεταξύ των καμπυλών του στοχευόμενου (Sa_t) και επιτυγχανόμενου (Sa_c) φασματος ελαστικής απόκρισης επιτάχυνσης, στοχευόμενου ($Siev_t^{ABS}$) και επιτυγχανόμενου ($Siev_c^{ABS}$) φασματος ισοδύναμης απόλυτης ταχύτητας της σεισμικής ενέργειας εισόδου, και στοχευόμενου ($Siev_t^{REL}$) και επιτυγχανόμενου ($Siev_c^{REL}$) φασματος ισοδύναμης σχετικής ταχύτητας της σεισμικής ενέργειας εισόδου. Η ποσότητα $p(T)$ είναι ένας γραμμικός συντελεστής ποινής, με αυξημένη βαρύτητα στην πειροχή των μικρών ιδιοπεριόδων. Με αυτό τον τρόπο δίνεται έμφαση στην ανωτέρω περιοχή, όπου ενδιαφέρει η ρεαλιστικότητα των παραγόμενων τεχνητών καταγραφών.

Οι μεταβλητές σχεδιασμού του προβλήματος βελτιστοποίησης είναι 2m, όπου m είναι ο αριθμός των πραγματικών καταγραφών που υπάρχουν στη δεξαμενή και συνδυάζονται γραμμικά. Οι πρώτες m μεταβλητές είναι οι συντελεστές στάθμισης των επιλεγόμενων καταγραφών, ενώ οι τελευταίες m μεταβλητές είναι ακέραιοι αριθμοί που δηλώνουν τις «ετικέτες» των καταγραφών που συνδυάζονται, όπως αυτές είναι αποθηκευμένες μέσα στη δεξαμενή.

Για την επίλυση του προβλήματος βελτιστοποίησης επιλέχθηκε ένας γενετικός αλγόριθμος κατάλληλος για μεικτά προβλήματα (πραγματικών – ακεραίων), δεδομένης της φύσης των μεταβλητών σχεδιασμού. Ο γενετικός αλγόριθμος εφαρμόζει σε έναν πληθυσμό υποψήφιων λύσεων την αρχή της εξελικτικής θεωρίας, προκειμένου να παράγει καλύτερες προσεγγίσεις προς τη βέλτιστη λύση. Σε κάθε γενιά, ένας νέος πληθυσμός δημιουργείται από τον τρέχοντα, μεσα από την επιλογή των υποψηφίων απογόνων σύμφωνα με τα επίπεδα καταλληλότητάς τους, και εν συνεχεία τηνα ναπαραγωγή τους χρησιμοποιώντας τελεστές εμπνευσμένους από τις αντίστοιχες φυσικές διαδικασίες της εξέλιξης (επιλογή, διασταύρωση και μετάλλαξη). Η διαδικασία αυτή οδηγεί στην εξέλιξη του πληθυσμού ούτως ώστε τα άτομα να γίνονται όλο και «καλύτερα» προς την κατεύθυνση του βέλτιστου σημείου του προβλήματος. Ο αλγόριθμος τερματίζει όταν ικανοποιείται ένα κατάλληλο προς το σκοπό αυτό κριτήριο, π.χ. μέγιστος αριθμός γενεών, MAXGEN. Ένας ψευδοκώδικας του γενετικού αλγορίθμου που χρησιμοποιήθηκε στην παρούσα διατριβή περιγράφεται στην Εικόνα 0.19.

| **Pseudocode of the GA**	
1	Set parameters
2	Generate the initial population
3	**while** GEN < MAXGEN **do**
4	Fitness calculation
5	Selection
6	Crossover
7	Mutation
8	**end while**
9	Obtain the individual with maximum fitness
10	**return** the best solution

Εικόνα 0.19: Ο ψευδοκώδικας ενός γενετικού αλγορίθμου

0.5.3. Σύγκλιση προτεινόμενου αλγορίθμου

Η αποτελεσματικότητα του προτεινόμενου αλγόριθμου εξετάζεται μεσω της παραγωγής τεχνητών επιταχυνσιογραφημάτων τα οποία συμμορφώνονται σε διάφορες ειδικές περιπτώσεις όσον αφορά τη φύση των στοχευόμενων φασμάτων και η διασφάλιση της ανεξαρτησίας της απόδοσης του αλγορίθμου από το είδος του στοχευόμενου φάσματος. Τρεις διαφορετικές σεισμικές καταγραφές επιλέχθηκαν για την προδιαγραφή στοχευόμενων φασμάτων: α) καταγραφή El

Centro Terminal Substation Building του σεισμού στο Imperial Valley (1940), β) καταγραφή Rinaldi του σεισμού του Northridge (1994) και γ) καταγραφή Sakarya – SKR του σεισμού του Kocaeli (1999). Τα τρία παραπάνω επιταχυνσιογραφήματα εκφράζουν μια καταγραφή μακρινού πεδίου, μια καταγραφή κοντικού πεδίου που περιέχει φαινόμενα εμπρόσθιας κατευθυντικότητας (forward directivity) και μια καταγραφή κοντινού πεδίου που περιέχει φαινόμενα μόνιμης στατικής παραμόρφωσης (fling step) αντίστοιχα. Εξετάζονται δυο σενάρια: i) Σενάριο 1, όπου το τεχνητό επιταχυνσιογράφημα συμμορφώνεται μόνο με το στοχευόμενο φάσμα ελαστικής απόκρισης επιτάχυνσης, και ii) Σενάριο 2, όπου το τεχνητό επιταχυνσιογράφημα συμμορφώνεται, όχι μόνο με το στοχευόμενο φάσμα ελαστικής απόκρισης επιτάχυνσης, αλλά και με το στοχευόμενο φάσμα ισοδύναμης ταχύτητας σεισμικής ενέργειας (απόλυτης και σχετικής).

Τα αποτελέσματα για το σενάριο 1 φαίνονται στην εικόνα 0.20. Αντίστοιχα αποτελέσματα για το σενάριο 2 φαίνονται στην εικόνα 0.21. Για κάθε στοχευόμενο φάσμα (με μαύρη γραμμή) εμφανίζονται, μεταξύ των 30 ανεξάρτητων αποτελεσμάτων του γενετικού αλγορίθμου (ο αλγόριθμος έτρεξε 30 φορές για να διασφαλιστεί ότι το αποτέλεσμα είναι ανεξάρτητο από τις παραμέτρους του αλγορίθμου, από τις οποίες παρήχθησαν 30 καμπύλες που φαίνονται με γκρι χρώμα), το αποτέλεσμα με το ελάχιστο σφάλμα (κόκκινη γραμμή) και το μέγιστο σφάλμα (μπλέ γραμμή). Με πράσινη γραμμή απεικονίζεται ο συντελεστής μεταβλητότητας των 30 καμπυλών κάθε στοχευόμενου φάσματος. Παρατηρείται ότι υπάρχει πολύ καλή συμφωνία μεταξύ της καμπύλης με το ελάχιστο σφάλμα και του στοχευόμενου φάσματος σε όλες τις περιπτώσεις.


(α)   (β)

(γ)
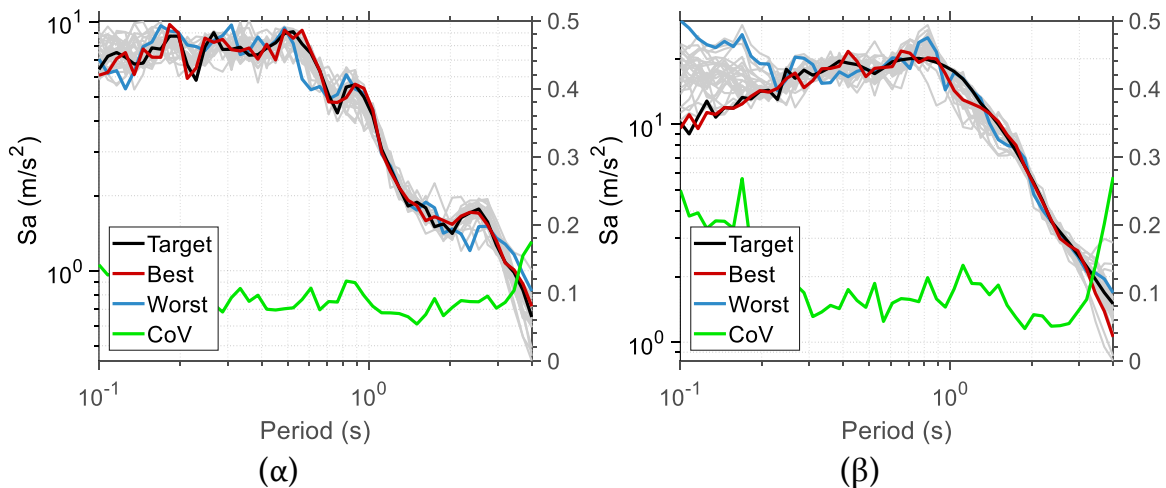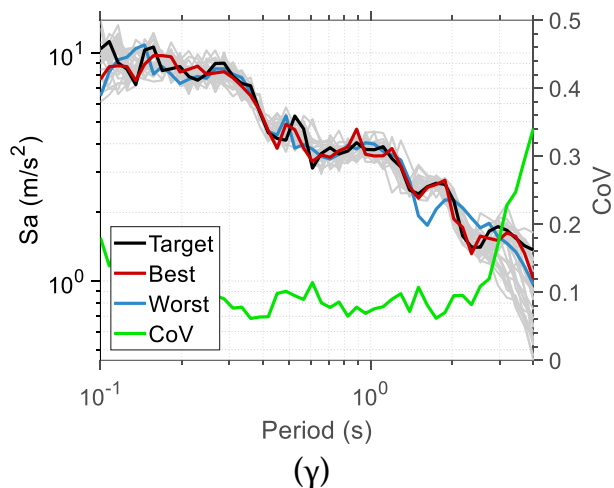
Εικόνα 0.20: Αποτελέσματα βελτιστοποίησης για το σενάριο 1, για κάθε στοχευόμενο φάσμα: (α) El Centro, (β) Northridge και (γ) Sakarya.


(α)

(β)

(γ)
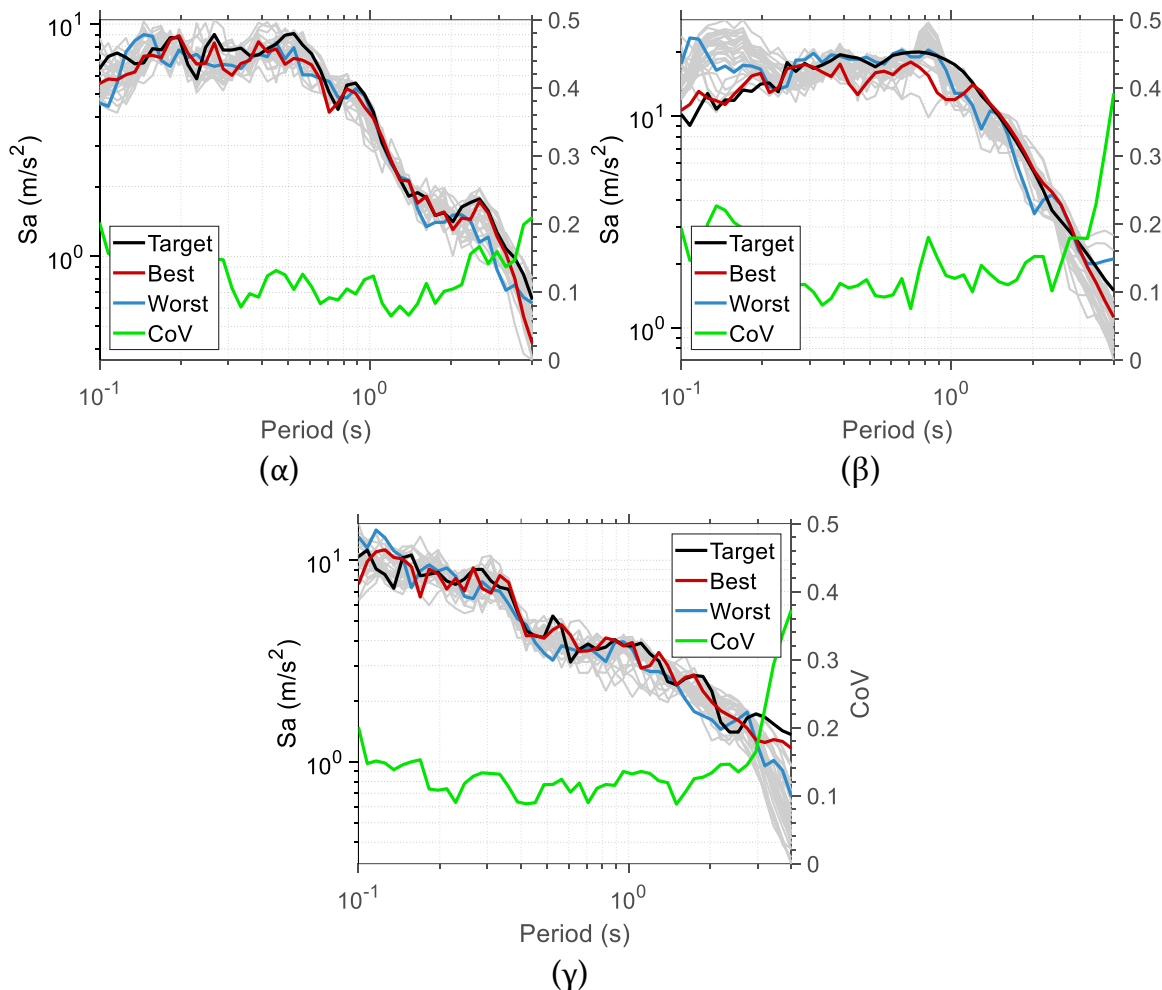
Εικόνα 0.21: Αποτελέσματα βελτιστοποίησης για το σενάριο 2, για το στοχευόμενο φάσμα ελαστικής απόκρισης επιτάχυνσης για κάθε στοχευόμενο φάσμα: (α) El Centro, (β) Northridge and (γ) Sakarya.

0.5.4. Σεισμική ενέργεια απόσβεσης

Για την αξιολόγηση της ευρωστίας και της ακρίβειας της προτεινόμενης μεθοδολογίας εξετάστηκε η σεισμική απόκριση μονοβάθμιων και πολυβάθμιων συστημάτων. Έγιναν μη γραμμικές εν χρόνω αναλύσεις των εν λόγω συστημάτων για τα βέλτιστα τεχνητά επιταχυνσιογραφήματα που έδωσαν τα δυο σενάρια και έγινε σύγκριση μεταξύ τους, και με το αντίστοιχο αποτέλεσμα της εν χρόνω ολοκλήρωσης των ανωτέρω συστημάτων με τα επιταχυνσιογραφήματα που χρησιμοποιήθηκαν για τον καθορισμό των στοχευόμενων φασμάτων. Για τη σύγκριση χρησιμοποιείται η σεισμική ενέργεια που αποσβέννεται στις κατασκευές λόγω ιξωδοελαστικής απόσβεσης. Στην Εικόνα 0.22 φαίνεται η χρονική μεταβολή της ενέργειας ιξωδοελαστικής απόσβεσης στην κατασκευή ανα μονάδα μάζας για κάθε περίπτωση στοχευόμενου φάσματος, στην περίπτωση ενός μονοβάθμιου ταλαντωτή. Παρατηρείται καλή συμφωνία μεταξύ της ενέργειας απόσβεσης του ταλαντωτή όταν διεγείρεται με το τεχνητό επιταχυνσιογράφημα που παρήχθη από τον προτεινόμενο αλγόριθμο (κόκκινη γραμμή) και όταν διεγείρεται με το επιταχυνσιογράφημα με βαση το οποίο παρήχθη το στοχευόμενο φάσμα (μαύρη γραμμή). Η συμφωνία αυτή παρατηρείται για όλες τις περιπτώσεις στοχευόμενων φασμάτων και είναι σε όλες τις περιπτώσεις καλύτερη για το σενάριο 2 σε σχέση με αυτή για το σενάριο 1. Αυτό σημαίνει ότι η μεθοδολογία του σεναρίου 2 (ήτοι συμμόρφωση τόσο με το φάσμα ελαστικής απόκρισης επιτάχυνσης όσο και με τα φάσματα ισοδύναμης ταχύτητας σεισμικής ενέργειας) δίνει πιο ρεαλιστικά τεχνητά επιταχυνσιογραφήματα σε σχέση με τη μεθοδολογία του σεναρίου 1 (ήτοι συμμόρφωση μόνο με το φάσμα ελαστικής απόκρισης επιτάχυνσης)

(α)   (β)

(γ)
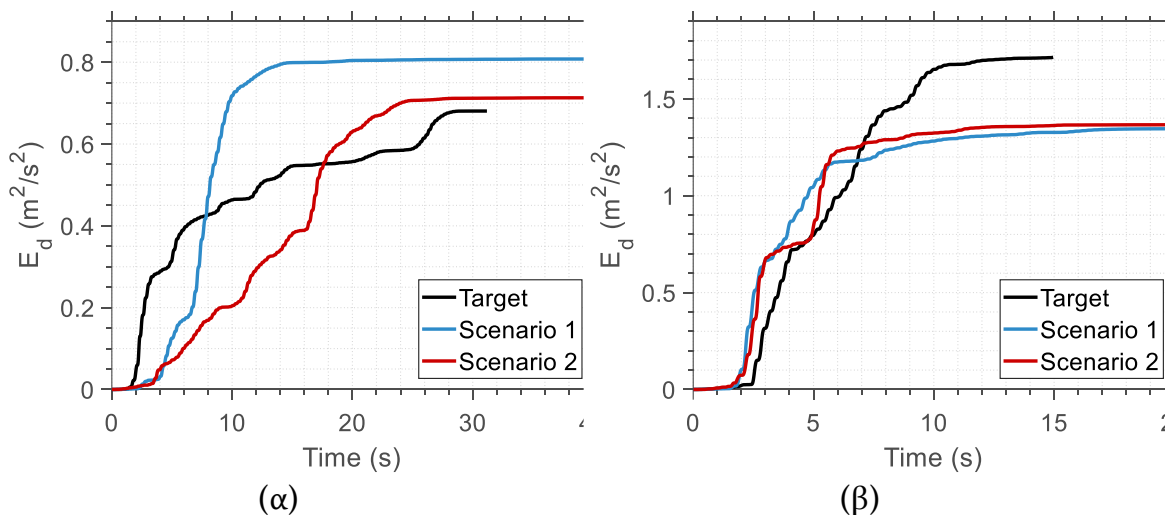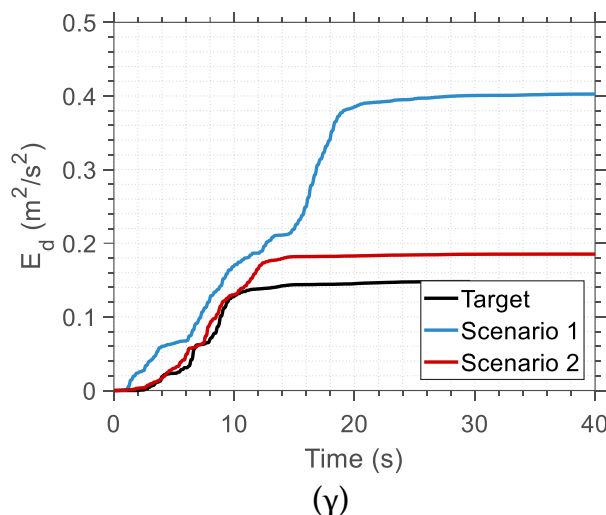
Εικόνα 0.22: Χρονική εξέλιξη της ενέργειας ιξωδοελαστικής απόσβεσης ανα μονάδα μάζας για τα τεχνητά επιταχυνσιογραφήματα των δυο σεναρίων και για κάθε περίπτωση στοχευόμενου φάσματος (α) El Centro, (β) Northridge και (γ) Sakarya.

0.5.5. Συμπεράσματα

Αναπτύχθηκε μια νεα μεθοδολογία παραγωγής τεχνητών επιταχυνσιογραφημάτων με γραμμικό συνδυασμό ακατέργαστων καταγραφών από μια βαση δεδομένων. Κατα τη δημιουργία των τεχνητών επιταχυνσιογραφημάτων λαμβάνεται υπόψη όχι μόνο η συμμόρφωσή τους με το φάσμα ελαστικής απόκρισης επιτάχυνσης που πραδιαγράφεται από τους κανονισμούς κατα το συνήθη αντισεισμικό σχεδιασμό, αλλά και τα φάσματα της ισοδύναμης ταχύτητας σεισμικής ενέργειας. Η θεώρηση αυτή οδηγεί στην παραγωγή βελτιωμένων τεχνητών καταγραφών που είναι αρκετά πιο ρεαλιστικές Το πρόβλημα παραγωγής τεχνητών επιταχυνσιογραφημάτων ανάγεται σε πρόβλημα βελτιστοποίησης, για την επίλυση του οποίου χρησιμοποιείται γενετικός αλγόριθμος μεικτής φύσης (πραγματικών – ακεραίων). Αποδεικνύεται ότι οι τεχνητές καταγραφές που παράγονται με βαση την προτεινόμενη μεθοδολογία οδηγούν σε ακριβέστερη εκτίμηση της σεισμικής ενέργειας απόσβεσης στις κατασκευές, σε σχέση με τις συμβατικές μεθόδους παραγωγής τεχνητών καταγραφών, και συνεπώς σε ακριβέστερη εκτίμηση της καταπόνησης των κατασκευών κατα τη διάρκεια σεισμικών διεγέρσεων. Επίσης, αποδεικνύεται η ευρωστία και ακρίβεια του γενετικού αλγορίθμου που χρησιμοποιήθηκε στην ανωτέρω μεθοδολογία.

0.6. Abaqus2Matlab: Ένα Ολοκληρωμένο Λογισμικό Βέλτιστου Σχεδιασμού για Εφαρμογές Μηχανικού

0.6.1. Εισαγωγή

Οι πρακτικές εφαρμογές της μεθόδου των πεπερασμένων στοιχείων συνήθως εντάσσονται σε ένα πλαίσιο βελτιστοποίησης, όπου επιδιώκεται καθε φορά μεσω της μεθόδου των πεπερασμένων στοιχείων η επίτευξη ενός βέλτιστου σχεδιασμού.

Κύριος λόγος είναι η μεγάλη πολυπλοκότητα των αντικειμενικών στόχων του βέλτιστου κατασκευαστικού σχεδιασμού, που δεν επιτρέπει την ενσωμάτωσή τους στην μέθοδο των πεπερασμένων στοιχείων. Τις περισσότερες φορές απαιτείται μια αντίστροφη ανάλυση, όπου είναι γνωστά τα αποτελέσματα της μεθόδου των πεπερασμενων στοιχείων και είναι άγνωστες οι παράμετροι του προσομοιώματος. Συνήθως αντίστροφες αναλύσεις τέτοιου είδους εφαρμόζονται για την εκτίμηση παραμέτρων σε καταστατικά προσομοιώματα, και όχι μόνο, και μπορούν να δώσουν αποτελέσματα που είναι σχεδόν αδύνατο να ανακτηθούν με άλλο τρόπο. Για τους παραπάνω λόγους, και για την επίλυση των ανωτέρω προβλημάτων όπου για τη μέθοδο των πεπερασμένων στοιχείων χρησιμοποιείται το Abaqus και για τη διαδικασία βελτιστοποίησης χρησιμοποιείται η γλώσσα προγραμματισμού Matlab, αναπτύχθηκε ένα νέο λογισμικό, το Abaqus2Matlab (Papazafeiropoulos et al., 2017), το οποίο έχει τα ακόλουθα κύρια χαρακτηριστικά:

• Παρέχει μια σύνδεση με την οποία μπορεί να γίνει μεταφορά δεδομένων από Abaqus σε Matlab και αντιστρόφως. Η ανάλυση σε Abaqus μπορεί να εκτελεστεί μέσω Matlab, χωρίς να απαιτείται η εκδήλωση ενεργειών εκ μέρους του χρήστη, είτε στο γραφικό περιβάλλον είτε στη γραμμή εντολών του Abaqus (Abaqus/CAE και Abaqus /Command αντίστοιχα).

• Μεταφέρει αποτελεσματικά τα αποτελέσματα από το Abaqus στο Matlab, με διαδικασίες απαλλαγμένες από σφάλματα, δεδομένου και ότι όλες οι υπορουτίνες του ανωτέρω λογισμικού έχουν επαληθευτεί μέσω ανάλογων αναλύσεων. Τα αποτελέσματα των επαληθεύσεων αυτών καθώς και όλα τα συνοδευτικά αρχεία παρουσιάζονται στη βιβλιογραφία του Abaqus2Matlab, και είναι πλήρως αναπαράξιμα από τον ενδιαφερόμενο χρήστη.

• Παρέχει τα αποτελέσματα του Abaqus στο Matlab ως μεταβλητές έτοιμες προς χειρισμό από το χρήστη μέσα στο περιβάλλον της Matlab, δίνοντας έτσι ευκολία για την περαιτέρω μετεπεξεργασία τους.

• Μπορεί να διαβάσει πολλά διαφορετικά είδη αποτελεσμάτων (σε κόμβους, σε σημεία ολοκλήρωσης πεπερασμένων στοιχείων και γενικά αποτελέσματα, όπως π.χ. συνδεσιμότητα δικτύου πεπερασμένων στοιχείων, ιδιοτιμές, κλπ).

• Παρέχεται πλήρης οδηγός χρήσης του Abaqus2Matlab καθώς και ανοικτός κώδικας.

• Συμπεριλαμβάνεται επαρκής αριθμός υπορουτίνων στις βιβλιοθήκες του ανωτέρω λογισμικού για την επεξεργασία των συνηθέστερων τύπων αποτελεσμάτων του Abaqus.

• Το λογισμικό, η βιβλιογραφία του καθώς και άλλο συνοδευτικό υλικό είναι προσβάσιμα μεσω της ιστοσελίδας www.abaqus2matlab.com.

Το Abaqus2Matlab διανέμεται προς τους χρήστες ως ανοικτός πηγαίος κώδικας με στόχο τη διευκόλυνση και επιτάχυνση της επιστήμης του μηχανικού. Τα αποτελέσματα του Abaqus διαβάζονται από το Abaqus2Matlab μέσω αρχείων της μορφής ASCII, που λέγονται results files. Τα εν λόγω αρχεία μπορούν να είναι είτε σε δυαδική μορφή (binary) είστε σε αναγνώσιμη μορφή (ASCII). Ο χειρισμός και επεξεργασία των αρχείων σε μορφή ASCII είναι ευκολότερος σε σχέση με το

χειρισμό των αρχείων δυαδικής μορφής, αφού τα αρχεία μορφής ASCII μπορούν να διαβαστούν από διαφορετικούς υπολογιστές, με διαφορετική κωδικοποίηση, με χρήση διαφορετικών λογισμικών μετεπεξεργασίας, χωρίς την ανάγκη ειδικών ρυθμίσεων που απαιτούν χρόνο και πολύτιμους πόρους. Κύριο μειονέκτημα των αρχείων ASCII είναι ότι είναι μεγαλύτερα σε μέγεθος από τα αρχεία τύπου binary. Τόσο στην παρούσα διατριβή όσο και στο εγχειρίδιο χρήσης του λογισμικού (βιλβιογραφία) αναφέρεται λεπτομερώς ο τρόπος εξαγωγής των αποτελεσμάτων από τα εν λόγω αρχεία, τα οποία δημιουργούνται μετά από κάθε ανάλυση του Abaqus, ύστερα από κατάλληλη παραμετροποίηση των αντίστοιχων αρχείων εισόδου (Abaqus input files). Για περισσότερες πληροφορίες όσοαν αφορά το λογισμικό, ο αναγνώστης παραπέμπεται στο σχετικός άρθρο (Papazafeiropoulos et al., 2017).

0.6.2. Οργάνωση του κώδικα του Abaqus2Matlab

Το Abaqus2Matlab αποτελείται από τις εξής υπορουτίνες που κατανέμονται σε υποφακέλους:

- Μια συνάρτηση που μετατρέπει τα περιεχόμενα του αρχείου αποτελεσμάτων (ή ισοδύναμα, *.fil), σε ένα διανυσμα χαρακτήρων που εκτείνεται σε μια γραμμή, από το οποίο ανακτώνται τα αποτελέσματα για να φορτωθούν εν συνεχεία στο περιβάλλον του Matlab.

- Ένας υποφάκελος με το όνομα «OutputAnalysis» που περιέχει όλες τις υπορουτίνες για την εξαγωγή των αποτελεσμάτων τύπου «ανάλυσης» (ήτοι αυτών που δεν αναφέρονται σε κόμβους ή στοιχεία), όπως για παράδειγμα ορισμούς κόμβων, στοιχείων, συνδεσιμότητα δικτύου, κλπ. Οι σχετικές μεταβλητές φαίνονται στον πίνακα 0.3. Η πρώτη στήλη (με τίτλο "record type") περιγράφει τη μεταβλητή τα αποτελέσματα της οποίας καταγράφονται στο αρχείο *.fil, για το αντίστοιχο κλειδί εγγραφής που αναγράφεται στη δεύτερη στήλη. Στην τρίτη στήλη αναγράφεται το κλειδί της μεταβλητής εξόδου, που χρησιμοποιείται στο αρχείο εισόδου στο Abaqus, προκειμένου να εκτυπωθούν τα αποτελέσμτα της συγκεκριμένης εμταβλητής στο αρχείο *.fil κατα τη διάρκεια της ανάλυσης. Τέλος, στην τέταρτη στήλη φαίνεται η υπορουτίνα του Abaqus2Matlab που είναι υπεύθυνη για την εξεγωγή των αποτελεσμάτων της εκάστοτε μεταβλητής.

- Ένας υποφάκελος με το όνομα «OutputNodes» που περιέχει τις υπορουτίνες για την επεξεργασία αποτελεσμάτων που αναφέρονται σε κόμβους. Οι διαθεσιμες μεταβλητές για το σκοπό αυτό φαίνονται στον πίνακα 0.4.

- Ένας υποφάκελος με το όνομα «OutputElements» που περιέχει τις υπορουτίνες για την επεξεργασία αποτελεσμάτων που αναφέροντσι σε πεπερασμένα στοιχεία (είτε στα σημεία ολοκλήρωσης αυτών είτε στο σύνολο του στοιχείου). Οι διαθέσιμες μεταβλητές για το σκοπό αυτό φαίνονται στον Πίνακα 0.5.

- Ένας υποφάκελος με το όνομα «Verification», που περιέχει κώδικες σε Matlab για την επαλήθευση των διαφόρων υπορουτίνων.

- Ένας υποφάκελος με το όνομα «ABAQUSInputFiles» που περιέχει όλα τα αρχεία εισόδου του Abaqus που χρησιμοποιούνται για την επαλήθευση.
- Ένας υποφάκελος με το όνομα «help» που περιέχει όλα τα αρχεία σχετικά με τη βιβλιογραφία του Abaqus2Matlab.

ANALYSIS RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Element definitions	1900	-	Rec1900.m
Node definitions	1901	-	Rec1901.m
Modal	1980	-	Rec1980.m

Πίνακας 0.3: Λίστα μεταβλητών για αποτελέσματα τύπου «ανάλυσης» που εξάγονται με το Abaqus2Matlab.

NODAL RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Nodal Acceleration	103	A	Rec103.m
Concentrated Electrical Nodal Charge	120	CECHG	Rec120.m
Concentrated Electrical Nodal Current	139	CECUR	Rec139.m
Nodal Point Load	106	CF	Rec106.m
Concentrated Flux	206	CFL	Rec206.m
Nodal Coordinate	107	COORD	Rec107.m
Fluid Cavity Volume	137	CVOL	Rec137.m
Electrical Potential	105	EPOT	Rec105.m
Motions (in Cavity Radiation Analysis)	237	MOT	Rec237.m
Normalized Concentration (Mass Diffusion Analysis)	221	NNC	Rec221.m
Temperature	201	NT	Rec201.m
Fluid Cavity Pressure	136	PCAV	Rec136.m
Pore or Acoustic Pressure	108	POR	Rec108.m
Electrical Reaction Charge	119	RCHG	Rec119.m
Electrical Reaction Current	138	RECUR	Rec138.m
Nodal Reaction Force	104	RF	Rec104.m
Residual Flux	204	RFL	Rec204.m
Internal Flux	214	RFLE	Rec214.m
Reactive Fluid Volume Flux	109	RVF	Rec109.m
Reactive Fluid Total Volume	110	RVT	Rec110.m
Total Force	146	TF	Rec146.m

Nodal Displacement	101	U	Rec101.m
Nodal Velocity	102	V	Rec102.m
Viscous Forces Due to Static Stabilization	145	VF	Rec145.m

Πίνακας 0.4: Λίστα μεταβλητών για αποτελέσματα τύπου «κόμβου» που εξάγονται με το Abaqus2Matlab.

ELEMENT RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Creep Strain (Including Swelling)	23	CE	Rec23.m
Mass Concentration (Mass Diffusion Analysis)	38	CONC	Rec38.m
Concrete Failure	31	CONF	Rec31.m
Coordinates	8	COORD	Rec8.m
Unit Normal to Crack in Concrete	26	CRACK	Rec26.m
Total Strain	21	E	Rec21.m
Total Elastic Strain	25	EE	Rec25.m
Energy (Summed over Element)	19	ELEN	Rec19.m
Energy Density	14	ENER	Rec14.m
Mechanical Strain Rate	91	ER	Rec91.m
Whole element volume	78	EVOL	Rec78.m
Film	33	FILM	Rec33.m
Total Fluid Volume Ratio	43	FLUVR	Rec43.m
Pore Fluid Effective Velocity Vector	97	FLVEL	Rec97.m
Gel (Pore Pressure Analysis)	40	GELVR	Rec40.m
Heat Flux Vector	28	HFL	Rec28.m
Total Inelastic Strain	24	IE	Rec24.m
Logarithmic Strain	89	LE	Rec89.m
Nominal Strain	90	NE	Rec90.m
Nodal Flux Caused by Heat	10	NFLUX	Rec10.m
Plastic Strain	22	PE	Rec22.m
Pore or Acoustic Pressure	18	POR	Rec18.m
Radiation	34	RAD	Rec34.m
Stress	11	S	Rec11.m
Saturation (Pore Pressure Analysis)	35	SAT	Rec35.m
Section Strain and Curvature	29	SE	Rec29.m
Section Force and Moment	13	SF	Rec13.m
Stress Invariant	12	SINV	Rec12.m

| Strain Jump at Nodes	32	SJP	Rec32.m
Principal stresses	401	SP	Rec401.m
Average Shell Section Stress	83	SSAVG	Rec83.m
Element Status	61	STATUS	Rec61.m
Section Thickness	27	STH	Rec27.m
Thermal Strain	88	THE	Rec88.m

Πίνακας 0.5: Λίστα μεταβλητών για αποτελέσματα τύπου «στοιχείου» που εξάγονται με το Abaqus2Matlab.

0.6.3. Επαλήθευση του προβλήματος βελτιστου σχεδιασμού δικτυώματος 10 κόμβων

Για την επαλήθευση του Abaqus2Matlab χρησιμοποιήθηκε μια σειρά από προβλήματα βελτιστοποίησης, όπου συγκρίνονται τα αποτελέσματα της βιβλιογραφίας με αυτά που δίνει το Abaqus2Matlab. Θεωρείται ένα επίπεδο δικτύωμα 10 κόμβων το οποίο φαίνεται στην Εικόνα 0.23, με τα ακόλουθα δομικά χαρακτηριστικά:

- Μέτρο ελαστικότητας E = 10,000 ksi
- Πυκνότητα rho = 0.1 lb/in³
- Μήκος L = 360 in
- Φορτίο P = 100 kip

Οι ράβδοι του δικτυώματος διαχωρίζονται σε 10 ομάδες. Οι μεταβλητές σχεδιασμού είναι τα εμβαδά διατομής των ράβδων που ανήκουν σε κάθε ομάδα. και τα οποία ανήκουν στο διάστημα [0.1, 35] (in²). Οι περιορισμοί επιβάλλονται τόσο στις τάσεις όσο και στις παραμορφώσεις των δομικών στοιχείων. Η μέγιστη επιτρεπόμενη μετατόπιση στις διευθύνσεις ±x και ±y για κάθε κόμβο είναι d_max= 2 in, ενώ η μέγιστη επιτρεπόμενη τάση (απόλυτη τιμή) είναι sigma_allow= 25 ksi τόσο σε εφελκυσμό όσο και σε θλίψη, και η αντικειμενική συνάρτηση είναι να ελαχιστοποιηθεί το βάρος του δικτυώματος, υπό τους ανωτέρω περιορισμούς, με απώτερο στόχο την οικονομικότερη κατασκευή.
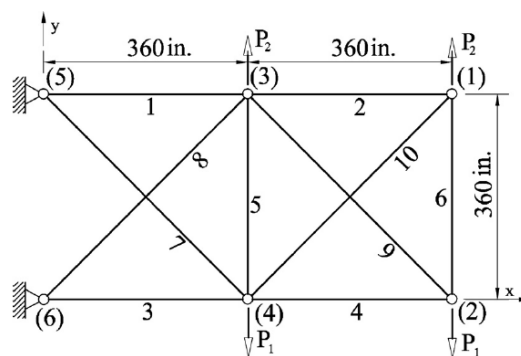
Εικόνα 0.23: Γεωμετρία και φόρτιση του δικτυώματος 10 κόμβων.

Ο Πίνακας 0.6 παρουσιάζει τα βέλτιστα αποτελέσματα τα οποία υπολογίστηκαν με την προτεινόμενη διαδικασία βελτιστοποίησης και επίσης τον αντίστοιχο αριθμό υπολογισμών αντικειμενικής συνάρτησης. Τα αποτελέσματα της προτεινόμενης διαδικασίας συγκρίνονται με αντίστοιχα αποτελέσματα ορισμένων μελετών που υπάρχουν στη βιβλιογραφία. Πρατηρείται ότι το βέλτιστο βάρος και τιμές των μεταβλητών σχεδιασμού που υπολογίστηκαν με το Abaqus2Matlab βρίσκονται πολύ κοντά με τα αντίστοιχα αποτελέσματα που υπάρχουν στη βιβλιογραφία. Ωστόσο, είναι προφανές ότι η προτεινόμενη διαδικασία απαιτεί πολύ μικρότερο αριθμό στατικών αναλύσεων σε σχέση με άλλες μεθόδους, προκειμένου ο αλγόριθμος να φτάσει τους βέλστιστους σχεδιασμούς.

Variables	Optimal cross section area (in²)						
Design name	M. Sonmez (2011)	Wu & Tseng (2010)	Li et al. (2007)	Degertekin &Hayalioglu (2013)	Degertekin (2012)	Kaveh et al. (2014)	Present thesis
A1	30.548	30.378	30.704	30.429	30.394	30.208	30.5218
A2	0.1	0.1	0.1	0.1	0.1	0.1	0.1
A3	23.18	23.468	23.167	23.244	23.098	22.698	23.1999
A4	15.218	15.196	15.183	15.368	15.491	15.275	15.2229
A5	0.1	0.1	0.1	0.1	0.1	0.1	0.1
A6	0.551	0.533	0.551	0.575	0.529	0.529	0.5514
A7	7.463	7.437	7.46	7.440	7.488	7.558	7.4572
A8	21.058	21.084	20.978	20.967	21.189	21.559	21.0364
A9	21.501	21.433	21.508	21.533	21.342	21.491	21.5284
A10	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Weight (lb)	5060.88	5060.45	5060.92	5060.96	5061.42	5062.39	5060.9
Number of function evaluations	500,000	32,100	125,000	16,872	7,081	9,791	347

Πίνακας 0.6: Αποτελέσματα βελτιστου σχεδιασμού του δικτυώματος 10 κόμβων.

0.6.4. Βέλτιστος σχεδιασμός πρόβολων τοίχων αντιστήριξης γραμμικά ελαστικού εδάφους με χρήση γενετικού αλγορίθμου

Στην ενότητα αυτή εξετάζεται ο βέλτιστος σχεδιασμός πρόβολων τοίχων αντιστήριξης που υπόκεινται σε σεισμική φόρτιση και αποκρίνονται με γραμμικά ελαστική συμπεριφορά. Η αντικειμενική συνάρτηση που ελαχιστοποιείται είναι το βάρος του τοίχου αντιστήριξης. Αυτό είναι περίπου ανάλογο του κόστους κατασκευής του, εφόσον το τελευταίο είναι γενικά μια αύξουσα συνάρτηση του βάρους του υλικού κατασκευής. Η αντικειμενική συνάρτηση ελαχιστοποιείται υποκείμενη σε σχεδιαστικούς περιορισμούς. Εκτός από τους συνήθεις περιορισμούς που επιβάλλονται στις περισσότερες περιπτώσεις βελτιστοποίησης

παρόμοιου τύπου, επιβάλλεται ένας πρόσθετος περιορισμός που αφορά την περιστροφική απόκριση του τοίχου. Η βελτιστοποίηση πραγματοποιείται μεσω ενός γενετικού αλγορίθμου, ο οποίους έχει ήδη χρησιμοποιηθεί για βελτιστοποίηση προβλημάτων δομοστατικού σχεδιασμού πολλών μεταβλητών και πολύπλοκων περιορισμών (Pei & Xia, 2012). Παρουσιάζονται δυο περιπτώσεις διαδικασίας βέλτιστου σχεδιασμού, που αντιστοιχούν σε δύο τιμές της υψομετρικής διαφοράς του εδαφικού στρώματος που πρόκειται να αντιστηριχθεί: 8m (περίπτωση 1) και 12m (περίπτωση 2). Το γενικό προσομοίωμα για τις δυο περιπτώσεις αντιστηρίξεων φαίνεται σην Εικόνα 0.24. Αποτελείται από ένα εδαφικό στρώμα που εκτείνεται στο άπειρο προς αμφότερες τις δυο οριζόντιες κατευθύνσεις, έχει οριζόνται βάση, και είναι ψηλότερο στα ανάντη προς την θετική κατεύθυνση, σε σχέση με την αρνητική κατεύθυνση. Το «σκαλοπάτι» που δημιουργείται αντιστηρίζεται από έναν τοίχο αντιστήριξης, που θεμελιώνεται σε πέδιλο, αποτελούμενο από τη «μύτη» (τμήμα κατάντη) και το «τακούνι» (τμήμα ανάντη). Η απόσταση από τον τοίχο μέχρι το μακρινό πεδίο (κατάντη) είναι ίση με 10 φορές το αντιστηριζόμενο ύψος. Ομοίως προς την ανάντη κατεύθυνση.

Τα αποτελέσματα των δυο διαδικασιών βελτιστοποίησης φαίνονται στον Πίνακα 0.6. Παρατηρείται γενικά ότι το βάθος και το πλάτος της θεμελίωσης του τοίχου είναι αυξημένα στην περίπτωση 2 σε σχεση με την περίπτωση 1. Επίσης το ίδιο ισχύει για τη βέλτιστη τιμή της αντικειμενικής συνάρτησης. Οι παρατηρήσεις αυτές συμφωνούν με τη συνήθη διαίσθηση. Το μήκος του τακουνιού του τοίχου (heel) είναι ίσο με το ελάχιστο όριο σε αμφότερες τις περιπτώσεις.
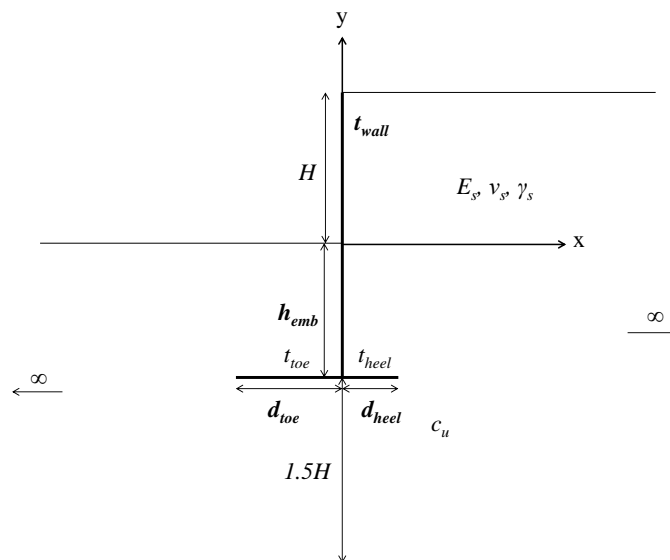
Εικόνα 0.24: Προσομοίωμα τοίχου αντιστήριξης από ωπλισμένο σκυρόδεμα.

	Case 1 (*H*=8m)	Case 2 (*H*=12m)
Design variables		
h_{emb} (m)	7.76	7.16
d_{toe} (m)	4.57	6.57
d_{heel} (m)	2.00	2.00
t_{wall} (m)	0.20	0.22
t_{toe} (m)	0.20	0.22
t_{heel} (m)	0.20	0.20
Constraint quantities		
θ	**0.328%**	0.246%
maxτ (kPa)	78.42	88.46
minτ (kPa)	-131.96	**-135.73**
minσ_{yy} (kPa)	-505.98	-592.45
maxσ_{yy} (kPa)	-122.07	-124.53
$\sigma_{b,s,max}$ (kPa)	22570.73	21476.85
$\sigma_{b,t,max}$ (kPa)	4265.33	1183.35
$\sigma_{b,h,max}$ (kPa)	2029.97	2180.68
$\sigma_{b,s,min}$ (kPa)	-23795.84	-23385.15
$\sigma_{b,t,min}$ (kPa)	-7336.37	-8128.82
$\sigma_{b,h,min}$ (kPa)	-2176.36	-2501.06
Algorithm details		
Min. value of obj. fun. (m²)	4.47	6.12
Number of generations	73	64
Number of fun. evaluations	1480	1300

Πίνακας 0.6: Αποτελέσματα βέλτιστου σχεδιασμού για τις δυο περιπτώσεις αντιστηρίξεων.

0.6.5. Βαθμονόμηση των καταστατικών ιδιοτήτων του υλικού ενός ελαστικού μεσω διαδικασίας βελτιστοποίησης

Θεωρούνται οι ιδιομορφές και ιδιοσυχνότητες στην περιοχή των χαμηλών συχνοτήτων προκειμένου να αναπτυχθεί ένα ρεαλιστικό μοντέλο ελαστικού αυτοκινήτου, με βάση αντίστοιχα αριθμητικά δεδομένα από τη βιβλιογραφία. Αυτό επιτυγχάνεται μεσω μιας διαδικασίας βελτιστοποίησης, η οποία προσαρμόζει διάφορες καταστατικές παραμέτρους του υλικού κατασκευής του ελαστικού, ούτως ώστε οι ιδιομορφές του ελαστικού στο βέλτιστο σημείο να βρίσκονται κοντά στις πειραματικές (δεδομένες από τη βιβλιογραφία) το περισσότερο δυνατό. Δεδομένου του μικρού αριθμού των μεταβλητών σχεδιασμού, χρησιμοποιήθηκαν απλές μέθοδοι μαθηματικού προγραμματισμού για την επιλυση του ανωτέρω προβλήματος βελτιστοποίησης. Μέθοδοι που ανήκουν σε αυτή την κατηγορία είναι οι σειριακές μέθοδοι τετραγωνικού προγραμματισμού για μη γραμμικές διαδικασίες βελτιστοποίησης (που χρησιμοποιήθηκαν σε αυτή την

ενότητα), και άλλες. Περισσότερες λεπτομέρειες για αυτές τις μεθόδους παρουσιάζονται από τους Nocedal & Wright (2006).

Η διατομή του ελαστικού, P235/75R17, φαίνεται στην Εικόνα 0.25. Το ελαστικό αποτελείται από τη ζώνη, το πέλμα, και τα πλευρικά τοιχώματα, τα οποία προσομοιώνονται με υπερελαστικό υλικό, αντιπροσωπευτικό του καουτσούκ. Το υπερελαστικό υλικό μοντελοποιείται με δυναμικό ενέργειας παραμόρφωσης αποτελούμενο από έναν πολυωνυμικό όρο (Mooney – Rivlin model). με σειρά Prony μοναδικού όρου, για την προσομοίωση της εν χρόνω ιξωδοελαστικότητας (Bekakos et al., 2016). Η ζώνη περιέχει οπλισμό δυο στρώσεων και τον οπλισμό του σκελετού, όπου ο τελευταίος εκτείνεται πάνω από την περιοχή της ζώνης και καλύπτει τα πλευρικά τοιχώματα. Οι δυο στρώσεις της ζώνης και ο σκελετός διακριτοποιούνται με επιφανειακά στοιχεία με δυνατότητα παραμόρφωσης στρέβλωσης. Το χείλος διακριτοποιείται με γραμμικά στοιχεία συνδέσμου δυο κόμβων για αξονοσυμμετρικές επίπεδες γεωμετρίες, και τα υπόλοιπα τμήματα της διατομής διακριτοποιούνται με τετρακομβικά διγραμμικά στοιχεία μειωμένης ολοκλήρωσης με ελεγχόμενο hourglassing.
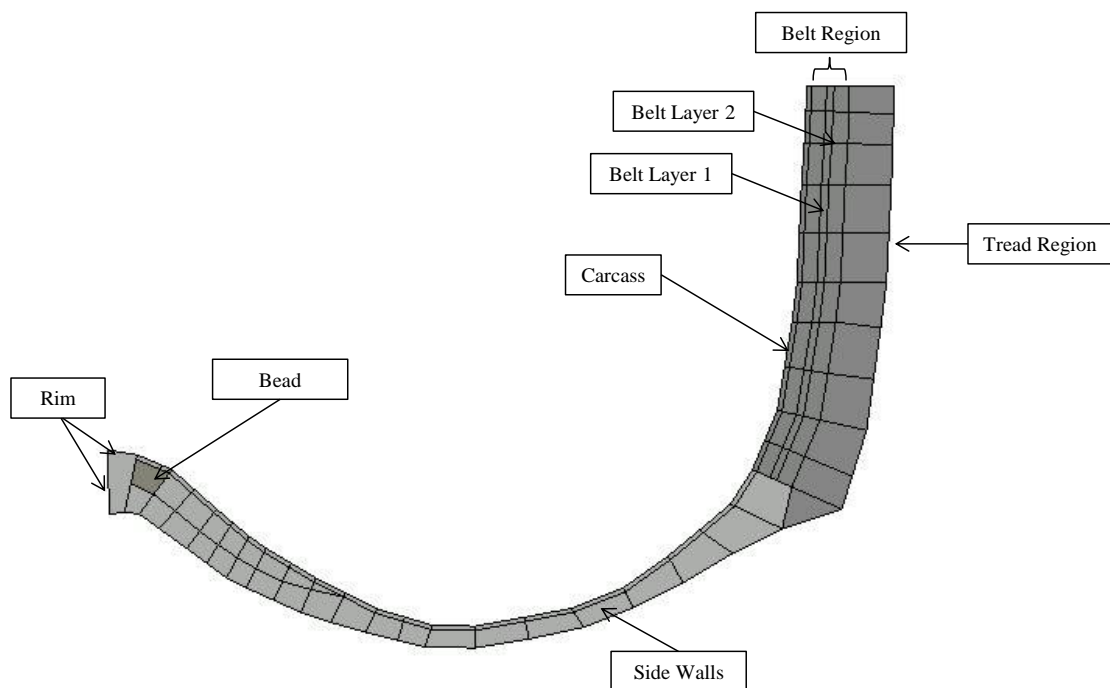
Εικόνα 0.25: Γεωμετρία ημιδιατομής του ελαστικού.

Τα αποτελέσματα της διαδικασίας βελτιστοποίησης φαίνονται στον Πίνακα 0.7.

	Initial model	Optimised model	Wheeler et al. (2005)	Deviation (%)
Design Variables				
A_{belt} (m²)	$2.11868*10^{-7}$	$3.64826*10^{-7}$	N/A	-
$A_{carcass}$ (m²)	$4.20835*10^{-7}$	$8.01133*10^{-8}$	N/A	-
C_{10} (Pa)	10^6	$10^6 +0.01489$	N/A	-
Eigenfrequencies				
f_1 [0,0] (Hz)	36.85	30.86	31.7	2.66
f_2 [0,0] (Hz)	37.17	35.85	35	2.43
f_3 [1,1] (Hz)	43.85	36.92	37.8	2.33
f_4 [1,1] (Hz)	43.85	36.92	37.8	2.33
f_5 [1,0] (Hz)	65.07	58.75	58.5	0.43
f_6 [1,0] (Hz)	65.07	58.75	58.5	0.43
f_7 [2,1] (Hz)	76.33	68.41	66.1	3.49
f_8 [2,1] (Hz)	76.33	68.41	66.1	3.49
f_9 [2,0] (Hz)	86.65	78.67	79.5	1.04
f_{10} [2,0] (Hz)	86.65	78.67	79.5	1.04
f_{11} [3,0] (Hz)	104.36	96.42	97.6	1.21
f_{12} [3,0] (Hz)	104.36	96.42	97.6	1.21
f_{13} [3,1] (Hz)	117.07	107.9	102.7	5.06
f_{14} [3,1] (Hz)	117.07	107.9	102.7	5.06
f_{15} [4,0] (Hz)	122.65	114.9	115.9	0.83
f_{16} [4,0] (Hz)	122.65	114.9	115.9	0.83
Algorithm Details				
Min. value of obj. function	-	8.59	-	-
Number of obj. function evaluations	-	25	-	-

Πίνακας 0.7: Αποτελέσματα της διαδικασίας βελτιστοποίησης των ιδιοτιμών του ελαστικού που εξετάζεται στην παρούσα ενότητα.

Άλλες εφαρμογές του Abaqus2Matlab που παρουσιάζονται στην παρούσα διατριβή είναι οι εξής:

Βέλτιστος σχεδιασμός έναντι λυγισμού δοκών με πολλαπλές διαμήκεις ενισχύσεις στη διατομή, υποκείμενη σε συνδυασμένη κάμψη και διάτμηση
Πρόβλεψη του συντελεστή λυγισμού ενισχυμένων δοκών με χρήση αλγορίθμων βαθιάς μηχανικής μάθησης
Μια υπολογιστική μέθοδος για τη διεξαγωγή μη γραμμικών προσαρμοστικών αναλύσεων pushover κατασκευών μεσω προσομοίωσης Abaqus

o.7.   Συμπεράσματα

Τα κύρια συμπεράσματα της παρούσας διατριβής είναι τα ακόλουθα:

- Η οικογένεια αλγορίθμων μη γραμμικής γενικευμένης εν χρόνω ολοκλήρωσης, απλού βήματος απλής λύσης που αναπτύχθηκαν περιέχει αλγορίθμους πολύ αποτελεσματικοί ακριβείς και ευσταθείς, ακόμα και με αυξημένο μέγεθος χρονικού βήματος ολοκλήρωσης, ενώ οι αλγόριθμοι συνεχούς επιτάχυνσης, που περιλαμβάνουν τη μέθοδο HHT-a ως ειδική περίπτωση, δίνουν τα πιο ακριβή αποτελέσματα για τις περισσότερες περιπτώσεις που μελετήθηκαν.

- Αναπτύχθηκε νέο λογισμικό επεξεγασίας ισχυρών εδαφικών κινήσεων, το οποίο υπερισχύει σε ακρίβεια σε ορισμένες περιπτώσεις, λόγω χρήσης προχωρημένων αλγορίθμων παραμετροποιήσιμων ώστε να έχουν ελεγχόμενες ιδιότητες αριθμητικής απόσβεσης, διασκεδασμού και υπερακοντισμού. Εχει δε το μεγάλο πλεονέκτημα του δωρεάν προσβάσιμου ανοικτού κώδικα, που το κάνει ανάρπαστο για ερευνητικούς και διδακτικούς σκοπούς.

- Αποδεικνύεται ότι υπάρχουν μοναδικές βέλτιστες κατανομές δυσκαμψίας οιονεί γραμμικού σχήματος, για την ομοιόμορφη κατανομή καθ' ύψος της ενέργειας ιξωδοελαστικής και υστερητικής απόσβεσης σε ελαστικά και ελαστοπλαστικά επίπεδα διατμητικά κτίρια. Η ομοιόμορφη κατανομή της ενέργειας απόσβεσης παρέχει αυξημένη ασφάλεια έναντι ολικής κατάρρευσης κατα τη διάρκεια ισχυρών σεισμικών γεγονότων.

- Εισάγεται η καινοτόμα έννοια της γραμμικής κατεύθυνσης ενός μη γραμμικού αλγορίθμου βελτιστοποίησης που οδηγεί σε βασική εξοικονόμηση υπολογιστικού φόρτου για τη λύση μη γραμμικών προβλημάτων βελτιστοποίησης.

- Αποδεικνύεται ότι οι τεχνητές σεισμικές καταγραφές που παράγονται με τη μεθοδολογία του κεφαλαίου 5 της παρούσας διατριβής είναι πιο ρεαλιστικές για τον αντισεισμικό σχεδιασμό των κατασκευών, σε σχέση με τις τεχνητές καταγραφές που παράγονται με τις υπάρχουσες μεθόδους.

- Αποδεικνύεται ότι το Abaqus2Matlab, ένα νέο λογισμικό που αναπτύχθηκε κατα τη διάρκεια εκπόνησης της παρούσας διατριβής, με κύριο σκοπό την επίλυση προβλημάτων βελτιστοποίησης, είναι εύρωστο και αποτελεσματικό, εφαρμόσιμο σε μια μεγάλη ποικιλία προβλημάτων μηχανικού.

Ο κύριος σκοπός της παρούσας διατριβής είναι να θεμελιώσει καινοτόμες μεθοδολογίες για το σχεδιασμό των κατασκευών βασισμένο στις δυναμικές τους

ιδιότητες και τη σεισμική τους απόκριση με τη χρήση προηγμένων υπολογιστικών μεθόδων.

Acknowledgements

I would like to thank a number of people who, in many different ways, contributed to the completion of this dissertation.

First and foremost, I would like to show my greatest appreciation to my advisor, Professor **Manolis Papadrakakis**, for his excellent guidance, his invaluable support and his confidence in me throughout all these years of my PhD research. His comments and suggestions were innumerably valuable throughout my study, and certainly I am inspired by his love and unstinting devotion to science, as well as his enthusiastic encouragement.

I would also like to express my deepest thanks to the other two members of the PhD advisory committee: Associate Professor **Nikolaos Lagaros** of the School of Civil Engineering, National Technical University of Athens (NTUA), for his great support, encouragement and friendship and emeritus professor **Yannis Dafalias** of the School of Applied Mathematical and Physical Sciences, National Technical University of Athens (NTUA) for his kindness, support and availability whenever necessary. I also thank them for their time for the careful reading of the manuscript and their valuable comments and suggestions which contributed to improving the quality of the dissertation.

In addition, I would like to thank Prof. **Vagelis Plevris,** Associate Professor at the Department of Civil Engineering and Energy Technology of OsloMet—Oslo Metropolitan University in Oslo, Norway and, for his continuous support and true friendship since my very early research steps. Our fruitful discussions have always been a great source of inspiration for me. The feedback and valuable suggestions on technical issues from **Manolis Georgioudakis**, postdoctoral researcher at the NTUA were of vital importance in the endeavor of achieving the goals of the dissertation.

Last, but certainly not least, I would like to thank **Andreas-Marios Tsainis**, who has proved to be a devoted friend. The discussions with him about various scientific and social issues have always been (and are) inspiring and his endless humour has made my life happier and easier.

Finally, I am grateful to my beloved father **Christos**, mother **Nerantzia** and sister **Marina**, for their understanding and patience all these years of my PhD research. I would like to express the deepest appreciation to **Vicky**, for her understanding and her whole-heartedly support of my research activity with patience. Her love and encouragement is a powerful source of inspiration and energy for me in the pursuit of my research goals, while her companionship always helps me lead my thoughts to other important aspects of life, as well.

Athens, January 2022
George Papazafeiropoulos

Table of Contents

3   OpenSeismoMatlab: A New Open-source Software for Strong Ground Motion Data Processing 51

4   A new energy-based structural design optimization concept under seismic actions 89

List of Figures

List of Tables

lxxviii

List of Listings

lxxx

Chapter 1

■ ■ ■ ■ ■

1 Introduction

1.1 Motivation

The design of any structural system aims to minimize its construction and operational costs and improve its structural performance. In this regard, engineers use either simple design rules based on experience, or structural optimization procedures in order to improve or optimize the design. It is obvious that this task is often difficult, given the large number of parameters that affect the design and the overall complexity of structural response.

Direct time integration (or time stepping, or step by step) methods are a widely used approach to solve dynamic linear or nonlinear response analysis problems. In these methods the equilibrium equations are satisfied at discrete time points (or steps) of the loading and the response history. The response during each step is calculated from the displacement and velocity at the beginning of the step and from the history of loading during the step. Thus the response for each step is an independent analysis problem. The dynamic direct time integration methods have to satisfy certain criteria in order to be suitable for the integration of the differential equation of motion in the linear or nonlinear regime. In linear dynamic response, emphasis is given in accuracy, whereas in nonlinear dynamic response numerical stability is of primary interest. The large number of criteria that have to be satisfied has led to the development of dynamic time integration algorithms the results of which depend highly on the nature of the problem considered, i.e. while any algorithm may be suitable for dynamic analysis involving a specific time stepping and/or constitutive model, it may be inappropriate for dynamic analysis involving different characteristics of the two aforementioned factors. The classification of the various direct time integration algorithms into categories highlights this fact. From the aforementioned points, it is obvious that there is the need for the development of a direct time integration algorithm that will be able to be universally applied to any dynamic structural analysis problem. Among the goals of this thesis, is to cover this need.

The optimization methodologies can be widely classified in the following two generic formulations:

i.   Deterministic Optimization;
ii. Stochastic Optimization.

Algorithms of the first category proceed towards the optimum solution by following a specific path. Apart from the value of the objective function, they need additional information at each point to determine the direction in which they will proceed. On the contrary, algorithms of the second category use only the objective value and are able to find the global optimum in a stochastic way. In Chapter 4 of this thesis a new algorithm of the first category is developed which involves an innovative way to obtain additional information in each evaluation point of the path towards the optimum, whereas in Chapter 5 a new algorithm of the second category is presented which is able to handle the optimization problem considered.

1.2   Objectives and scope

The goal of the thesis is the development of innovative computational techniques for the optimum design of structures which respond due to dynamic (seismic) loading. Also, the necessary numerical tools for the implementation of the new computational techniques are provided. This is achieved by (a) developing methodologies for the direct step by step integration of the differential equations of dynamic equilibrium in the time domain, (b) the development of a computational tool for processing of raw strong-motion acceleration time series to produce compatible acceleration, velocity and displacement time series, acceleration, velocity and displacement elastic and inelastic response spectra, Fourier amplitude spectra, and standard earthquake-engineering intensity measures, (c) development of a novel iterative optimization algorithm of Newton type with line search capabilities, especially designed for linear elastic and elastoplastic shear buildings, which finds the optimum distribution of storey stiffness and strength for a prescribed fundamental (small strain) eigenperiod of the building, so that the distribution of dissipated energy along the height of the building becomes uniform, (d) development of a mixed real – integer genetic algorithm with appropriately customized genetic operators for the optimization of the ground motion acceleration time histories used for the dynamic analysis of structures in the framework of their seismic design, enabling in this way the more realistic seismic design of structures, and (e) the development of a new software, Abaqus2Matlab, which serves to integrate Abaqus and Matlab for various optimization procedures including, but not limited to, optimum structural design based on the dynamic properties and dynamic response of a structure. This interface between these well-known codes not only benefits from the image processing and the integrated graph-plotting features of Matlab, but opens up

new opportunities in results post-processing, statistical analysis and mathematical optimization, among many other possibilities.

For the efficient solution of the aforementioned problems, efforts have been made for maximizing accuracy while retaining the computational effort as low as possible, through the optimized implementation of the various algorithms and their associated codes.

All the issues described above have been addressed in the thesis, as will be described in detail in the following chapters. Furthermore, via numerical applications of realistic design problems the proposed computational framework is evaluated and tested. The original contribution of the thesis is presented in detail in Section 7.1 of the Conclusions (Chapter 7).

1.3   Organization and outline

The thesis consists of six chapters in total, plus the bibliography and three appendices at the end of it. Its structure is organized as follows:

Chapter 1 is the introduction of the dissertation which provides a general description of the motivation, the goals pursued, as well as a brief description of the contents of each chapter.

Chapter 2 introduces a generalized dynamic time – integration algorithm framework for non-linear structural dynamics. The nonlinear versions of the General Single Step Single Solve time integration algorithms' family are formulated and outlined in an explicit flowchart which describes the nonlinear integration procedure in detail. Afterwards the various algorithms are applied to some benchmark dynamic analysis problems.

Chapter 3 presents the development of OpenSeismoMatlab, which is an innovative open-source software for strong ground motion data processing, written in MATLAB, and is based on the family of the dynamic time integration algorithms presented in Chapter 2. This software is capable of processing of raw strong-motion acceleration time series to produce compatible acceleration, velocity and displacement time series, acceleration, velocity and displacement elastic and inelastic response spectra, Fourier amplitude spectra, and standard earthquake-engineering intensity measures.

In Chapter 4 a new optimization concept is introduced which involves the uniform distribution of the dissipated seismic input energy among the various storeys of an arbitrary planar shear building. This is achieved by appropriately adjusting the stiffnesses and strengths of the various storeys of the building. The optimization technique uses a gradient method based on equivalent linear structures, instead of the traditional practice of calculating the gradients from the nonlinear objective function. The family of the direct time integration algorithms presented in Chapter 2 is used for the dynamic analysis required for the estimation of the dissipated seismic input energy.

Chapter 5 introduces a novel spectra-matching framework, which employs a linear combination of raw ground motion records to generate artificial acceleration time histories perfectly matching a target spectrum, taking into account not only the acceleration but also the seismic input energy equivalent velocity. The optimization procedures employed in Chapter 5 use solvers that involve the use of OpenSeismoMatlab, among others.

Chapter 6 presents a novel software, Abaqus2Matlab, that connects Abaqus, a sophisticated finite element package, with Matlab, the most comprehensive program for mathematical analysis. Abaqus2Matlab is used for various applications, including the optimum design of a retaining wall based on its seismic response, and the inverse analysis for the calibration of the constitutive properties of a tyre based on its dynamic properties.

Chapter 7 contains the conclusions, the original contribution of the thesis, and directions for future research.

Finally, the bibliography is presented followed by Appendix A which contains a listing of publications by the author. Each Chapter is accompanied by its corresponding references and notation.

Chapter **2**

■ ■ ■ ■ ■

2 A Generalized Algorithm Framework for Non-Linear Structural Dynamics

2.1 Introduction

In this chapter a generalized time integration algorithm that incorporates several well-known algorithms as special cases is extended into the nonlinear regime. The behavior of the algorithm during the time integration of the equation of motion of nonlinear dynamic structural problems is studied. After a literature review of the available time integration schemes used for problems of nonlinear structural dynamics and of the family of linear GSSSS algorithms, the nonlinear schemes are formulated and outlined in an explicit flowchart which describes the nonlinear integration procedure in detail. Afterwards, the nonlinear family of algorithms is applied to six elementary benchmark SDOF problems involving the dynamic response of SDOF systems with various stiffness and damping properties, as well as to two advanced benchmark problems, which involve a 3dof structure representing finite element systems containing rigid connections, penalty factors and other such types of constraints and a simple undamped pendulum with large rotations.

The dynamic analysis of engineering structures under dynamic loading (earthquake, impact, etc.) with the finite element method results in a set of ordinary differential equations as follows:

$$M\ddot{u} + p(u, \dot{u}) = f(t) \tag{2.1}$$

where M is the mass matrix, p is the internal force vector, which is in general a nonlinear function of the displacements u and velocities $\dot{u}$ and is equal to the sum of the forces in the structure due to stiffness and damping, and f is the external force vector which is a function of time t. In the case of a linear elastic structure with viscous damping $p(u, \dot{u})$ is equal to:

$$p(u, \dot{u}) = \bar{K}u + \bar{C}\dot{u} \tag{2.2}$$

Where $\overline{K}$ is the stiffness matrix and $\overline{C}$ is the damping matrix, both of them independent of the displacement and velocity. Linear equations of dynamic equilibrium of the form of (2.1) in which $p(u,\dot{u})$ is given by (2.2) can be solved using various superposition methods in the time or frequency domain, which greatly simplify the problem. However, in dynamic analysis of nonlinear response, superposition cannot be used and one has to resort to step-by-step methods.

Direct time integration (or time stepping, or step by step) methods are a widely used approach to solve dynamic linear or nonlinear response analysis problems. In these methods the equilibrium equations are satisfied at discrete time points (or steps) of the loading and the response history. The response during each step is calculated from the displacement and velocity at the beginning of the step and from the history of loading during the step. Thus the response for each step is an independent analysis problem.

The most common characteristics of time integration schemes are the following:

•   Stability. An integration scheme is said to be stable if the numerical solution, under any initial conditions, does not grow without bound. An algorithm is unconditionally stable for linear problems if the convergence of the solution is independent of the size of the time step, that is, the errors are not amplified from one step to the next no matter how long a time step is. In mathematical terms, numerical stability is provided, when the spectral radius ρ of the amplification matrix A defined by:

$$\begin{bmatrix} u \\ \dot{u}\Delta t \\ \ddot{u}\Delta t^2 \\ ... \\ u^{(a)}\Delta t^a \end{bmatrix}_i = A \begin{bmatrix} u \\ \dot{u}\Delta t \\ \ddot{u}\Delta t^2 \\ ... \\ u^{(a)}\Delta t^a \end{bmatrix}_{i-1} \tag{2.3}$$

is not more than unity throughout the dynamic analysis, i.e.

$$\rho = \max\left(|\lambda_1|,|\lambda_1|,...,|\lambda_\alpha|\right) \le 1 \tag{2.4}$$

where a denotes the order of the derivative of u and $\lambda_1,\lambda_2,...,\lambda_a$ are the eigenvalues of the amplification matrix A.

•   Convergence. An integration scheme is convergent if the numerical solution approaches the exact solution as the size of the time step tends to zero. Suppose that the error is defined as follows:

$$E = \left\| u^{\Delta t} - u^{exact} \right\| \tag{2.5}$$

where $u^{\Delta t}$ is an approximation of u depending on the time step Δt, and $\| \ \|$ denotes an arbitrary norm. If the following relation holds:

$$\lim_{\Delta t \to 0} E = 0 \tag{2.6}$$

and there is a convergence trend shown as a straight line with positive integer slope in the $\log(E)$ versus $\log(\Delta t)$ plot, as shown in Figure 2-1, then the integration scheme is said to be properly convergent, whereas the slope q is the order of accuracy of the integration scheme. If there is not any piece of the $\log(E)$ versus $\log(\Delta t)$ curve which is linear with positive integer slope, then the integration scheme is said not to be properly convergent.
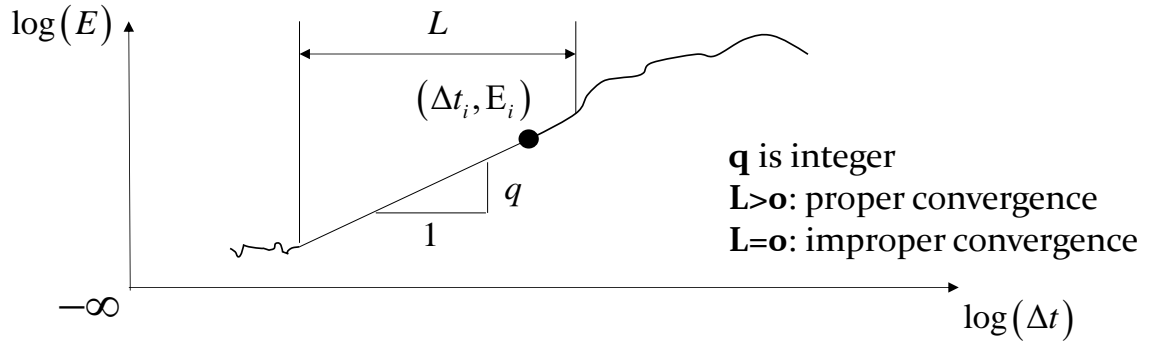
Figure 2-1: Typical variation of error with time step size for converging approximate time integration algorithms.

• Accuracy. The order of accuracy of a time integration scheme is the largest positive integer q for which the following relation holds:

$$\lim_{\Delta t \to 0} \frac{\left\| u_i^{\Delta t} - u_{i-1}^{\Delta t} \right\|}{\left(\dfrac{\Delta t_{i-1}}{\Delta t_i} \right)^q - 1} = 0 \tag{2.7}$$

where Δt_i is the time step size at iteration i of the equation limit (2.7). Two numerical errors are associated with the accuracy of any algorithm: (a) numerical dispersion (often expressed in terms of period elongation) and (b) numerical dissipation (often expressed in terms of either the amplitude decay or the algorithmic damping ratio). Although the order of accuracy is conceptually independent from the numerical stability, there is some relationship between the two, in the form of various restrictions, e.g. Dalquist Barriers (Wood, 1990).

• Algorithmic dissipation. It is a kind of filtering of the higher frequency oscillations, necessary to eliminate the spurious high frequency modes inherent in a finite element mesh. For a SDOF system, the plot of the amplitude of the spectral radius ρ of the amplification matrix A, with respect to $\omega\Delta t$ (or equivalently $\Delta t / T$), represents the capability of the time integration scheme to damp out the higher modes of a finite element model, which are sometimes introducing considerable error in the response. If $\rho < 1$, smaller values of ρ entail larger degree of elimination of the higher frequency oscillations. $\rho = 0$ means complete elimination (see Figure 2-2).

• Self-starting. This type of algorithms requires data from two time steps to proceed the solution. If data from more than two time steps are needed, the algorithm must be implemented with a starting procedure.
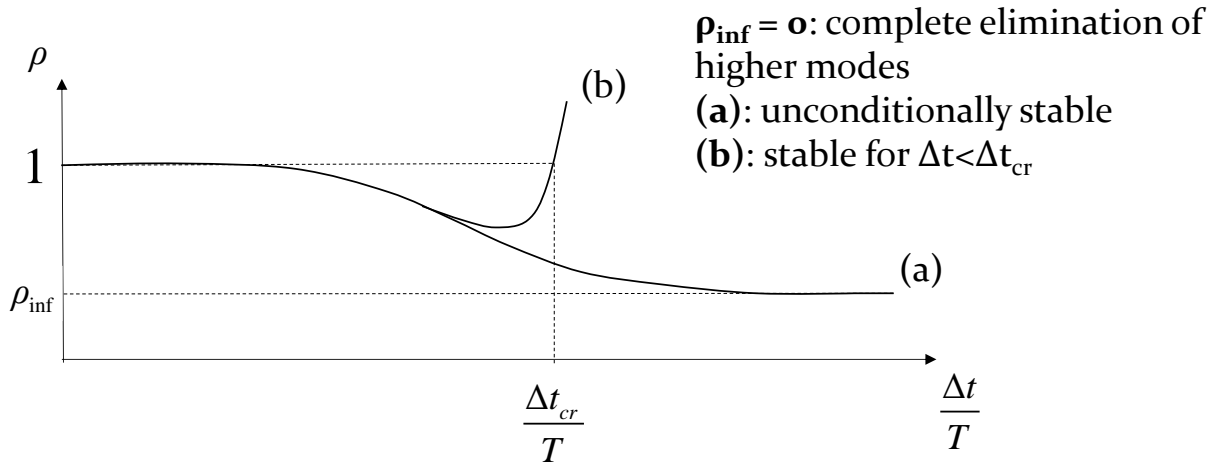
Figure 2-2: Typical variation of spectral radius with $\Delta t / T$ for conditionally and unconditionally stable time integration algorithms.

- Overshooting. It is the tendency of an algorithm to exceed heavily the exact solution in the first few time steps, but eventually to converge to the exact solution.

Taking into account the above characteristics, a time integration scheme should have the following desirable features (Zhou & Tamma, 2004):

- Unconditional stability.
- At least second-order accuracy in time.
- No more than zero-order displacement and velocity overshooting behavior with minimal numerical dissipation and dispersion.
- Self-starting features with no more than one set of single-field system of implicit equations to be solved at each time step to include ease of implementation and computational simplicity.

Regarding linear dynamic response, accuracy is the main concern, since many time integration algorithms are unconditionally stable. However, algorithms which are unconditionally stable for linear dynamics, often lose this stability for nonlinear dynamics, and therefore numerical stability is of primary interest in such cases.

In this chapter, after a concise literature review about the numerical direct time integration algorithms applicable to the dynamic equilibrium equations of structural analysis of the form (2.1), a modification of the group of general single step single solve algorithms is presented, which can account for nonlinear dynamic response. This group of algorithms has already been published in the literature (Zhou & Tamma, 2004), but only for the case of linear dynamic response. The modification consists of introducing a Newton-Raphson iterative procedure inside each increment (or time step). Afterwards, the modified algorithm is applied for the direct time integration of the equation of motion (in the form of eq. (2.1)) of nonlinear dynamic systems the details of which can be found in section 2.2. The effect of all the characteristics of the various integration schemes used, as presented above, on the resulting dynamic response is studied and the relative performances of the various time integration schemes used are

compared. Apart from its robustness in solving nonlinear problems, it is proved that the algorithm can be designed to cope with cases with any degree of nonlinearity.

The simplest direct time integration method for dynamic analysis is the piecewise exact method in which the equation of motion is solved exactly for linear loading during each time step, in which it is assumed that the actual loading history has constant slope (Clough & Penzien, 2003). Although the equation of motion is solved rigorously during each time step, the linear interpolation of the excitation function introduces some error into the calculated response; this can be eliminated either by reducing the length of the time step, or adjusting it so that the introduced loading history fits best the actual one.

The numerical direct time integration schemes can be classified as either explicit or implicit. An explicit scheme is one in which the response values for the next step are calculated only from quantities belonging to the current step. On the other hand, an implicit scheme is one in which the expressions giving the values for the next step include one or more values of the next step, and therefore successive iterations are needed to arrive to the solution for the next step. Implicit methods lead in general to increased computational effort, although it is possible for some of them to be converted into an explicit formulation.

Algorithms that require two or more implicit systems to be solved simultaneously at each time step have improved properties (Argyris et al., 1973), but they require twice or more the computational effort of the simple implicit systems.

Another classification that can be made is according to the formulation used to ensure conservation (or decay) of energy within a time step which is a sufficient condition for algorithmic stability (Kuhl & Crisfield, 1999). This energy criterion is summarized in the following inequality:

$$\left(U_{n+1} + K_{n+1} \right) - \left(U_n + K_n \right) \leq W_{ext} \qquad (2.8)$$

where U_n and U_{n+1} represent the strain energy at the beginning and at the end of the time step respectively, K_n and K_{n+1} are the corresponding kinetic energies and W_{ext} represents the work done by external forces within the time step. This classification results in the following three categories of algorithms which satisfy inequality (2.8):

- Algorithms which employ numerical dissipation. The algorithms considered in the present chapter fall in this category.
- Algorithms extending others by using constraints of energy conservation imposed via Lagrange multipliers (Constraint Energy Method), the first of which was presented in (Hughes & Caughey, 1978).
- Algorithms which enforce energy conservation algorithmically such as the energy-momentum method presented in (Kuhl & Ramm, 1996). One of the first algorithms of this category was presented in (Simo & Tarnow, 1992). In

the absence of external loading these algorithms are designed to obey the following laws:

$$\frac{dL_t}{dt} = 0, \qquad \frac{dJ_t}{dt} = 0, \qquad \frac{dE_{tot}}{dt} \leq 0 \qquad\qquad (2.9)$$

where L_t is the linear momentum, J_t is the angular momentum and E_t^{tot} is the total energy. Combinations of algorithms of different categories from the above have also been made, such as combinations of numerical dissipation algorithms and algorithms ensuring energy conservation algorithmically presented in (Kuhl & Crisfield, 1999; Armero & Petöcz, 1996).

Due to the lack of a general time integration algorithm suitable for various complex nonlinear dynamic structural systems, attempts have been made to combine two or more known algorithms into new more efficient ones. This concept of composition is another alternative for the construction of time integration schemes which possess desired properties, depending on the properties of the parent algorithms being combined. Algorithm composition can be done in two main ways:

- Each time step is divided into two or more substeps, at which different independent integration schemes are applied. Equilibrium is satisfied at each time substep. The final solution depends on the algorithms used as well as on the way of partition of the time steps. The most representative method is presented by Bathe and his collaborators (Bathe, 2007; Bathe & Baig, 2005; Bathe & Noh, 2012) in which the trapezoidal rule is combined with a three-point backward difference method in two equal substeps. Generally, different ways of segmentation of the integration steps into substeps require additional parameters to be introduced (Matias Silva & Mendes Bezerra, 2008).

- Different difference formulae are combined in one whole time step to inherit their advantages. Representative algorithms of this category have been presented in (Liu et al., 2012; Rezaiee-Pajand et al., 2011; Fung, 1998). For example, Liu et al. (2012) have proposed an efficient backward Euler time integration algorithm by composing the two point and three-point backward Euler formulae, which is a self-starting, two-step, second-order accurate algorithm with the same computational effort as the trapezoidal rule. However, with the increase of ratio $\Delta t / T$, this method results in the largest amplitude decay and period elongation, compared to Newmark, Bathe and Alpha methods.

An additional category of methods to solve time dependent problems is the family of temporal discretization techniques, adopted by approximation of a continuous time interval with temporal finite elements. The Whole Element Method (WEM) in which time is incorporated along with the other spatial variables into a direct variational method is outlined in (Rosales & Filipich, 2002). A time-discontinuous Galerkin (TDG) method was presented by Li & Wiberg (1996) whereas a mixed finite element method was developed by Fung et al.

(1998). Similar time integration algorithms have been developed by Argyris & Scharpf, 1969; Gellert, 1978; Riff & Baruch, 1984a, 1984b; Golley, 1996; Chien & Wu, 2000; Bar-Yoseph, 2000.

2.2   Modified nonlinear time integration algorithm

2.2.1   The linear generalized single step single solve algorithm

The equation of motion of a Single Degree of Freedom (SDOF) linear structure is given by the combination of the linear SDOF counterparts of (2.1) and (2.2):

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = f(t) \tag{2.10}$$

with initial conditions:

$$u(0) = u_0, \quad \dot{u}(0) = \dot{u}_0 \tag{2.11}$$

Equation (2.10) can be applied to MDOF structures, given that the latter can be decomposed into a finite number of SDOF structures using various superposition methods. In (Zhou & Tamma, 2004) a family of general single step single solve (GSSSS) algorithms, namely algorithms which do not involve matrix multiplications and involve only one single system solve in a single time step is studied. A single size of the system refers to the number of degrees-of-freedom resulting from its spatial discretization. It is shown that the Dahlquist theorem (Dahlquist, 1963) holds not only for the linear multistep methods (LMS), but also for the general single step single solve (GSSSS) time integration algorithms, which are spectrally identical to the former. This theorem states that a GSSSS algorithm which is unconditionally stable, can be at most second order accurate.

According to the theory presented by Zhou & Tamma, (2004), equation (2.10) can be represented as a time weighted residual as follows:

$$\int_{t_n}^{t_{n+1}} W\left(M\ddot{u} + C\dot{u} + Ku - f\right)dt \tag{2.12}$$

where the weighted time field is assumed to be of the form

$$W = 1 + w_1\Gamma + w_2\Gamma^2 + w_3\Gamma^3 \tag{2.13}$$

and:

$$\Gamma = \tau/\Delta t, \quad \tau = t - t_n, \quad \Delta t = t_{n+1} - t_n \tag{2.14}$$

The dependent field variables (u, $\dot{u}$, $\ddot{u}$) can be approximated by the following asymptotic series expansions:

$$u = u_n + \Lambda_1\dot{u}_n\tau + \Lambda_2\ddot{u}_n\tau^2 + \Lambda_3\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}\tau^3 \tag{2.15}$$

$$\dot{u} = \dot{u}_n + \Lambda_4\ddot{u}_n\tau + \Lambda_5\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}\tau^2 \tag{2.16}$$

$$\ddot{u} = \ddot{u}_n + \Lambda_6\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}\tau \tag{2.17}$$

and the load vector is expanded to first order via a Taylor series:

$$f = f_n + \frac{f_{n+1} - f_n}{\Delta t} \tau \tag{2.18}$$

The updates of displacement and velocity are given by the equations:

$$u_{n+1} = u_n + \lambda_1 \dot{u}_n \Delta t + \lambda_2 \ddot{u}_n \Delta t^2 + \lambda_3 \left(\ddot{u}_{n+1} - \ddot{u}_n \right) \Delta t^2 \tag{2.19}$$

$$\dot{u}_{n+1} = \dot{u}_n + \lambda_4 \ddot{u}_n \Delta t + \lambda_5 \left(\ddot{u}_{n+1} - \ddot{u}_n \right) \Delta t \tag{2.20}$$

The update of acceleration is given by substitution of equations (2.13) to (2.20) into (2.12) as follows:

$$\begin{aligned}
\left(\mu_6 M + \mu_5 C \Delta t + \mu_3 K \Delta t^2 \right) \ddot{u}_{n+1} = \\
-M \left(\ddot{u}_n - \mu_6 \ddot{u}_n \right) \\
-C \left(\dot{u}_n + \mu_4 \ddot{u}_n \Delta t - \mu_5 \ddot{u}_n \Delta t \right) \\
-K \left(u_n + \mu_1 \dot{u}_n \Delta t + \mu_2 \ddot{u}_n \Delta t^2 - \mu_3 \ddot{u}_n \Delta t^2 \right) \\
+ \left(1 - W_1 \right) f_n + W_1 f_{n+1}
\end{aligned} \tag{2.21}$$

or in a simplified form:

$$\tilde{M} \ddot{u}_{n+1} = \tilde{F}_n \tag{2.22}$$

where

$$\begin{aligned}
\tilde{F}_n \left(K, C, f \right) = -M \left(\ddot{u}_n - \mu_6 \ddot{u}_n \right) - C \left(\dot{u}_n + \mu_4 \ddot{u}_n \Delta t - \mu_5 \ddot{u}_n \Delta t \right) \\
-K \left(u_n + \mu_1 \dot{u}_n \Delta t + \mu_2 \ddot{u}_n \Delta t^2 - \mu_3 \ddot{u}_n \Delta t^2 \right) + \left(1 - W_1 \right) f_n + W_1 f_{n+1}
\end{aligned} \tag{2.23}$$

and

$$\tilde{M} \left(K, C \right) = \mu_6 M + \mu_5 C \Delta t + \mu_3 K \Delta t^2 \tag{2.24}$$

The subscript of $\tilde{F}_n$ denotes the time step at which the quantities u, $\dot{u}$, $\ddot{u}$ are evaluated for its calculation. The time step at which K, C and f are evaluated is denoted by a separate subscript placed at these quantities. The constants W_i are given by:

$$W_i = \frac{\dfrac{1}{1+i} + \displaystyle\sum_{j=1}^{3} \dfrac{w_j}{1+i+j}}{1 + \displaystyle\sum_{j=1}^{3} \dfrac{w_j}{1+j}}, \quad i = 1, 2, 3 \tag{2.25}$$

There are 12 independent integration constants that are needed in order to apply equations (2.21), (2.19) and (2.20) to proceed to the next step. These are W_1, μ_1, μ_2, μ_3, μ_4, μ_5, μ_6, λ_1, λ_2, λ_3, λ_4, λ_5. Each combination of these parameters defines a unique algorithm, and can be considered in some way as the algorithm's signature. Many known time integration algorithms, which will be presented later, result from suitable selection of these parameters. In the study by Zhou & Tamma (2004), the integration parameters are calculated by imposing several different constraints to the algorithm, regarding order of accuracy, overshooting behavior (in terms of displacement and velocity orders), spurious roots at the high and low frequency limits, dissipation and dispersion properties, bifurcation of the principal roots, etc, which results in the derivation of 9 different

algorithms belonging to the above family. In this chapter, W_i is calculated directly from (2.25), after specifying the parameters w_1, w_2, w_3, so the number of integration constants needed is 14.

2.2.2   Design of the linear generalized single step single solve algorithm – special cases

An algorithm is termed to have the property of continuous acceleration, if the acceleration $\ddot{u}_{n+1}$ calculated at $t = t_n$ satisfies the equation of motion (strong form) at $t = t_n$. Otherwise, the algorithm is termed to have the property of discontinuous acceleration (Zhou & Tamma, 2004).

The procedure for designing the algorithm presented in the previous section to apply it to time integration problems (i.e. setting its 14 integration constants), is presented by Zhou & Tamma (2004). The algorithms of the generalized single step family are shown in Table 2-1.

The values of the integration constants are shown for various known integration schemes in tables. In Table 2 the parameters of the central difference method, the general family of Newmark methods (Newmark, 1959), the Average Constant Acceleration method (Newmark, 1959), the Linear Acceleration method (Newmark, 1959), the Backward Acceleration Method (Ascher & Petzold, 1998), and the Fox-Goodwin formula (Fox & Goodwin, 1949) are shown. In the case of the general family of Newmark methods, β and γ are the well-known Newmark constants. In Table 2-3 the parameters are given for the zero-order displacement, first-order velocity overshooting algorithms, presented by Zhou & Tamma (2004). It has to be mentioned that the formulas presented in Table 2-3 correspond to three special cases of these zero-order displacement, first-order velocity overshooting algorithms, namely the generalized a-method, the HHT-a method and the WBZ a-method, presented in Chung & Hulbert, 1993; Hilber et al., 1977; Wood et al., 1980) respectively. In order to evaluate the integration constants, the spectral radius ρ_∞, which is the minimum absolute value of the principal roots of the amplification matrix at the high-frequency limit, has to be first assigned a desired value, which must lie in the range given at the first row of Table 2-3, Table 2-4 and Table 2-5. If $\rho_\infty = 1$, the resulting algorithm is non-dissipative. In Table 2-4 the parameters of the zero-order displacement and zero-order velocity overshooting algorithms are presented (Zhou & Tamma, 2004). Table 2-5 shows the parameters of the first-order displacement and zero-order velocity overshooting algorithms. If $\rho_\infty = 1$, the first-order displacement, zero-order velocity, optimal numerical dissipation and dispersion, the first-order displacement, zero-order velocity, continuous acceleration and the first-order displacement, zero-order velocity, discontinuous acceleration algorithms recover, the first the mid-point rule a-form algorithm, and the two last the Newmark average acceleration a-form algorithm. In Table 2-2, Table 2-3, Table 2-4 and

Table 2-5, the number inside the angle brackets after the name of each algorithm is the number with which the algorithm is referred to in this chapter. This is done merely for easy identification of each algorithm throughout this chapter.

2.2.3   Modification of the linear algorithm for nonlinear dynamic response

In this section the family of generalized linear algorithms presented above is modified to account for materially nonlinear dynamic response. In general, to proceed from the current step (u_n, $\dot{u}_n$, $\ddot{u}_n$) to the next time step (u_{n+1}, $\dot{u}_{n+1}$, $\ddot{u}_{n+1}$), the secant stiffness and damping matrices are needed, which usually depend on u_{n+1} and $\dot{u}_{n+1}$. Since the latter are unknown, the tangent stiffness and damping matrices are calculated and iterations are performed to arrive to a converged solution. Convergence is attained via a Newton-Raphson iterative procedure. In some time integration algorithms, this iteration is avoided by using the initial tangent matrices instead of updating them, even though this approximation is not correct in principle.

The outline of the modified nonlinear time integration algorithm used in this chapter is shown in Figure 2-3. The given data are the mass, stiffness and damping properties of the SDOF oscillator and the imposed external force, denoted by M, $\bar{K}$, $\bar{C}$, f respectively. Note that the various quantities correspond to a MDOF system in general; this means that u, $\dot{u}$, $\ddot{u}$, f are column vectors, and M, $\bar{K}$, $\bar{C}$ are square matrices. The symbol ./ denotes right array division, namely the division of two vectors of the same size in an element by element fashion.

Before the application of the algorithm, the necessary integration constants are calculated and the maximum tolerance tol_{max} and the maximum number of iterations until convergence k_{max} are set.

Method name	Displacement overshooting order	Velocity overshooting order	Type
U0-V0-Opt	0	0	optimal numerical dissipation and dispersion
U0-V0-CA	0	0	continuous acceleration
U0-V0-DA	0	0	discontinuous acceleration
U0-V1-Opt	0	1	optimal numerical dissipation and dispersion
U0-V1-CA	0	1	continuous acceleration
U0-V1-DA	0	1	discontinuous acceleration
U1-V0-Opt	1	0	optimal numerical dissipation and dispersion
U1-V0-CA	1	0	continuous acceleration
U1-V0-DA	1	0	discontinuous acceleration

Table 2-1: Algorithms of the generalized single step single solve (GSSSS) family.

	Central Difference Method	Family of Newmark Methods (Newmark, 1959)	Newmark Average Constant Acceleration (Newmark, 1959)	Newmark Linear Acceleration (Newmark, 1959)	Newmark Backward Acceleration (Ascher & Petzold, 1998)	Fox-Goodwin formula (Fox & Goodwin, 1949)
w_1	-15	-15	-15	-15	-15	-15
w_2	45	45	45	45	45	45
w_3	-35	-35	-35	-35	-35	-35
μ_1	1	1	1	1	1	1
μ_2	0	β	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{12}$
μ_3	0	β	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{12}$
μ_4	$\frac{1}{2}$	γ	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
μ_5	$\frac{1}{2}$	γ	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
μ_6	1	1	1	1	1	1
λ_1	1	1	1	1	1	1
λ_2	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
λ_3	0	β	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{12}$
λ_4	1	1	1	1	1	1
λ_5	$\frac{1}{2}$	γ	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$

Table 2-2: Integration parameters for various known time integration schemes.

	U_0-V_1-Opt (Chung & Hulbert, 1993)	U_0-V_1-CA (Hilber et al., 1977)	U_0-V_1-DA (Wood et al., 1980)
ρ_∞	[0 1]	[1/2 1]	[0 1]
w_1	$-15\dfrac{1-2\rho_\infty}{1-4\rho_\infty}$	$-15\dfrac{1-2\rho_\infty}{2-3\rho_\infty}$	-15
w_2	$15\dfrac{3-4\rho_\infty}{1-4\rho_\infty}$	$15\dfrac{2-5\rho_\infty}{2-3\rho_\infty}$	45
w_3	$-35\dfrac{1-\rho_\infty}{1-4\rho_\infty}$	$-35\dfrac{1-3\rho_\infty}{2(2-3\rho_\infty)}$	-35
μ_1	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{2\rho_\infty}{1+\rho_\infty}$	1
μ_2	$\dfrac{1}{2(1+\rho_\infty)}$	$\dfrac{\rho_\infty}{1+\rho_\infty}$	$\dfrac{1}{2}$
μ_3	$\dfrac{1}{(1+\rho_\infty)^3}$	$\dfrac{2\rho_\infty}{(1+\rho_\infty)^3}$	$\dfrac{1}{(1+\rho_\infty)^2}$
μ_4	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{2\rho_\infty}{1+\rho_\infty}$	1
μ_5	$\dfrac{3-\rho_\infty}{2(1+\rho_\infty)^2}$	$\dfrac{\rho_\infty(3-\rho_\infty)}{(1+\rho_\infty)^2}$	$\dfrac{3-\rho_\infty}{2(1+\rho_\infty)}$
μ_6	$\dfrac{2-\rho_\infty}{1+\rho_\infty}$	1	$\dfrac{2}{1+\rho_\infty}$
λ_1	1	1	1
λ_2	$\dfrac{1}{2}$	$\dfrac{1}{2}$	$\dfrac{1}{2}$
λ_3	$\dfrac{1}{(1+\rho_\infty)^2}$	$\dfrac{1}{(1+\rho_\infty)^2}$	$\dfrac{1}{(1+\rho_\infty)^2}$
λ_4	1	1	1
λ_5	$\dfrac{3-\rho_\infty}{2(1+\rho_\infty)}$	$\dfrac{3-\rho_\infty}{2(1+\rho_\infty)}$	$\dfrac{3-\rho_\infty}{2(1+\rho_\infty)}$

Table 2-3: Integration parameters for zero-order displacement, first order velocity overshooting algorithms.

	U_o-V_o-Opt <8>	U_o-V_o-CA <9>	U_o-V_o-DA <10>
ρ_∞	[0 1]	[1/3 1]	[0 1]
w_1	$-15\dfrac{1-2\rho_\infty}{1-4\rho_\infty}$	$-15\dfrac{1-5\rho_\infty}{3-7\rho_\infty}$	-15
w_2	$15\dfrac{3-4\rho_\infty}{1-4\rho_\infty}$	$15\dfrac{1-13\rho_\infty}{3-7\rho_\infty}$	45
w_3	$-35\dfrac{1-\rho_\infty}{1-4\rho_\infty}$	$140\dfrac{\rho_\infty}{3-7\rho_\infty}$	-35
μ_1	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{1+3\rho_\infty}{2(1+\rho_\infty)}$	1
μ_2	$\dfrac{1}{2(1+\rho_\infty)}$	$\dfrac{1+3\rho_\infty}{4(1+\rho_\infty)}$	$\dfrac{1}{2}$
μ_3	$\dfrac{1}{2(1+\rho_\infty)^2}$	$\dfrac{1+3\rho_\infty}{4(1+\rho_\infty)^2}$	$\dfrac{1}{2(1+\rho_\infty)}$
μ_4	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{1+3\rho_\infty}{2(1+\rho_\infty)}$	1
μ_5	$\dfrac{1}{(1+\rho_\infty)^2}$	$\dfrac{1+3\rho_\infty}{2(1+\rho_\infty)^2}$	$\dfrac{1}{1+\rho_\infty}$
μ_6	$\dfrac{3-\rho_\infty}{2(1+\rho_\infty)}$	1	$\dfrac{3+\rho_\infty}{2(1+\rho_\infty)}$
λ_1	1	1	1
λ_2	$\dfrac{1}{2}$	$\dfrac{1}{2}$	$\dfrac{1}{2}$
λ_3	$\dfrac{1}{2(1+\rho_\infty)}$	$\dfrac{1}{2(1+\rho_\infty)}$	$\dfrac{1}{2(1+\rho_\infty)}$
λ_4	1	1	1
λ_5	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{1}{1+\rho_\infty}$

Table 2-4: Integration parameters for zero-order displacement, zero order velocity overshooting algorithms.

	U_1-V_0-Opt <11>	U_1-V_0-CA <12>	U_1-V_0-DA <13>
ρ_∞	$[0 \quad 1]$	$[1/2 \quad 1]$	$[0 \quad 1]$
w_1	$-30\dfrac{3-8\rho_\infty+6\rho_\infty^2}{9-22\rho_\infty+19\rho_\infty^2}$	$-60\dfrac{2-8\rho_\infty+7\rho_\infty^2}{11-48\rho_\infty+41\rho_\infty^2}$	$-30\dfrac{3-4\rho_\infty}{9-11\rho_\infty}$
w_2	$-15\dfrac{25-74\rho_\infty+53\rho_\infty^2}{2\left(9-22\rho_\infty+19\rho_\infty^2\right)}$	$15\dfrac{37-140\rho_\infty+127\rho_\infty^2}{2\left(11-48\rho_\infty+41\rho_\infty^2\right)}$	$15\dfrac{25-37\rho_\infty}{2\left(9-11\rho_\infty\right)}$
w_3	$-35\dfrac{3-10\rho_\infty+7\rho_\infty^2}{9-22\rho_\infty+19\rho_\infty^2}$	$-35\dfrac{5-18\rho_\infty+17\rho_\infty^2}{11-48\rho_\infty+41\rho_\infty^2}$	$-35\dfrac{3-5\rho_\infty}{9-11\rho_\infty}$
μ_1	$\dfrac{3-\rho_\infty}{2\left(1+\rho_\infty\right)}$	$\dfrac{1+3\rho_\infty}{2\left(1+\rho_\infty\right)}$	$\dfrac{3+\rho_\infty}{2\left(1+\rho_\infty\right)}$
μ_2	$\dfrac{1}{\left(1+\rho_\infty\right)^2}$	$\dfrac{2\rho_\infty}{\left(1+\rho_\infty\right)^2}$	$\dfrac{1}{1+\rho_\infty}$
μ_3	$\dfrac{1}{\left(1+\rho_\infty\right)^3}$	$\dfrac{2\rho_\infty}{\left(1+\rho_\infty\right)^3}$	$\dfrac{1}{\left(1+\rho_\infty\right)^2}$
μ_4	$\dfrac{3-\rho_\infty}{2\left(1+\rho_\infty\right)}$	$\dfrac{1+3\rho_\infty}{2\left(1+\rho_\infty\right)}$	$\dfrac{3+\rho_\infty}{2\left(1+\rho_\infty\right)}$
μ_5	$\dfrac{2}{\left(1+\rho_\infty\right)^3}$	$\dfrac{4\rho_\infty}{\left(1+\rho_\infty\right)^3}$	$\dfrac{2}{\left(1+\rho_\infty\right)^2}$
μ_6	$\dfrac{2-\rho_\infty}{1+\rho_\infty}$	1	$\dfrac{2}{1+\rho_\infty}$
λ_1	1	1	1
λ_2	$\dfrac{1}{2}$	$\dfrac{1}{2}$	$\dfrac{1}{2}$
λ_3	$\dfrac{1}{2\left(1+\rho_\infty\right)}$	$\dfrac{1}{2\left(1+\rho_\infty\right)}$	$\dfrac{1}{\left(1+\rho_\infty\right)^2}$
λ_4	1	1	1
λ_5	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{1}{1+\rho_\infty}$	$\dfrac{3-\rho_\infty}{2\left(1+\rho_\infty\right)}$

Table 2-5: Integration parameters for first-order displacement, zero order velocity overshooting algorithms.

2.3   Benchmark SDOF systems for evaluation of algorithms' performance

In this section, 13 different time integration schemes presented in Table 2-2, Table 2-3, Table 2-4 and Table 2-5 are compared through their application for solving a number of elementary (benchmark) problems. The schemes compared are the Newmark Average Constant Acceleration method (Newmark, 1959), denoted by <1>, the Newmark Linear Acceleration method (Newmark, 1959), denoted by <2>, the Newmark Backward Acceleration method (Ascher & Petzold,

1998), denoted by <3>, the Fox-Goodwin formula (Fox & Goodwin, 1949), denoted by <4>, the U0-V1-Opt, denoted by <5>, the U0-V1-CA, denoted by <6>, the U0-V1-DA, denoted by <7>, the U0-V0-Opt, denoted by <8>, the U0-V0-CA, denoted by <9>, the U0-V0-DA, denoted by <10>, the U1-V0-Opt, denoted by <11>, the U1-V0-CA, denoted by <12>, and the U1-V0-DA, denoted by <13>, algorithms. The last 9 integration schemes are presented by Zhou & Tamma (2004) and details about their notation can be seen in Table 2-3, Table 2-4 and Table 2-5.

In this section, 6 benchmark dynamic analyses of nonlinear SDOF systems have been considered which have various constitutive relations, damping types, and the dynamic response of which occurs purely due to nonzero initial conditions (unforced). These applications have been studied in order to assess the performance of the various time integration schemes. Their description will be made in the following sections. In each problem, the ρ_∞ parameter is selected to be equal to zero, or the lowest possible value for all integration algorithms used. Concerning the Newton-Raphson iterative procedure used, the maximum convergence tolerance and the maximum number of iterations are $\text{tol}_{\text{max}} = 0.01$ and $k_{\text{max}} = 200$ respectively. All units involved in the calculations belong to the SI system. For simplification of the calculations, the mass of all the SDOF systems considered is set to unity, without loss of generality. The time step used is $\Delta t = 0.01$ and the duration of the dynamic analysis is equal to 100 time steps (1 sec) for all problems.

2.3.1   Undamped SDOF oscillator with hardening spring

The first benchmark problem studied is a SDOF oscillator with hardening spring, for which the equation of motion is:

$$m\ddot{u} + S_1 u \left(1 + S_2 u^2\right) = 0 \tag{2.26}$$

This type of oscillator represents the well-known unforced and undamped Duffing oscillator (Duffing, 1918). The system is conservative and its total energy is constant and given by analytical integration of (2.26):

$$E_{\text{tot}} = \frac{1}{2} m\dot{u}^2 + \frac{1}{2} S_1 u^2 + \frac{1}{4} S_1 S_2 u^4 \tag{2.27}$$

In this example $S_1 = 1000$, $S_2 = 0.1$ and the initial conditions are $u_0 = 1.5$ and $\dot{u}_0 = 0$.

2.3.2   Undamped SDOF oscillator with softening spring

The second benchmark problem studied is a SDOF oscillator with softening spring, for which the equation of motion is:

$$m\ddot{u} + S \tanh(u) = 0 \tag{2.28}$$

Set $w_1, w_2, w_3, W_1\Lambda_1, W_2\Lambda_2, W_3\Lambda_3, W_1\Lambda_4, W_2\Lambda_5, W_1\Lambda_6, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5,$

Find W_1 from eq. (2.25) for $i = 1$

Initialize $u_n = u_0$, $\dot{u}_n = \dot{u}_0$

Find $K_0 = \bar{K}(u_0, \dot{u}_0)$, $C_0 = \bar{C}(u_0, \dot{u}_0)$, $p_0 = \bar{p}(u_0, \dot{u}_0)$

Find $\ddot{u}_0 = (f_0 - p_0)/M$

Set $K_n = K_0$, $C_n = C_0$, $p_n = p_0$, $u_n = u_0$, $\dot{u}_n = \dot{u}_0$, $\ddot{u}_n = \ddot{u}_0$

for n = 1:length(**f**)-1

    Initialize k = 1

    Initialize **qda** $= \text{tol}_{max}$

    Find $\ddot{u}_{n+1}^1 = \tilde{M}^{-1}(K_n, C_n)\tilde{F}_n(K_n, C_n, p_n)$ from eq. (2.22)

    while $\max(\text{abs}(\mathbf{qda})) \geq \text{tol}_{max}$ **&** $k \leq k_{max}$

        Iteration k of increment n+1:

        Set $\ddot{u}_{n+1}^k = \ddot{u}_n + \ddot{u}_{n+1}^1$

        Find u_{n+1}^k and $\dot{u}_{n+1}^k$ according to (2.19) and (2.20) respectively

        Find $K_{n+1}^k = \bar{K}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $C_{n+1}^k = \bar{C}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $p_{n+1}^k = \bar{p}(u_{n+1}^k, \dot{u}_{n+1}^k)$

        Find the residual $R_{n+1}^k = \tilde{F}_{n+1}^k(K_{n+1}^k, C_{n+1}^k, p_{n+1}^k) - \tilde{M}(K_{n+1}^k, C_{n+1}^k)\ddot{u}_{n+1}^1$

        Set $d\ddot{u}_{n+1}^k = \ddot{u}_n + (\ddot{u}_{n+1}^1 + da)$, where da is a constant infinitesimal variation of acceleration

        Find du_{n+1}^k and $d\dot{u}_{n+1}^k$ from (2.19) and (2.20) respectively

        Find $dK_{n+1}^k = \bar{K}(du_{n+1}^k, d\dot{u}_{n+1}^k)$, $dC_{n+1}^k = \bar{C}(du_{n+1}^k, d\dot{u}_{n+1}^k)$, $dp_{n+1}^k = \bar{p}(du_{n+1}^k, d\dot{u}_{n+1}^k)$

        Find the residual $dR_{n+1}^k = d\tilde{F}_{n+1}^k(dK_{n+1}^k, dC_{n+1}^k, dp_{n+1}^k) - d\tilde{M}(dK_{n+1}^k, dC_{n+1}^k)d\ddot{u}_{n+1}^k$

        Find $qda = [R_{n+1}^k./\ddot{u}_{n+1}^1]./[(dR_{n+1}^k - R_{n+1}^k)./da]$

        Update $\ddot{u}_{n+1}^1 = (1 - qda)\ddot{u}_{n+1}^1$

        Update $k = k + 1$

    end

    Set $\ddot{u}_{n+1}^k = \ddot{u}_n + \ddot{u}_{n+1}^1$

    Find u_{n+1}^k and $\dot{u}_{n+1}^k$ according to (2.19) and (2.20) respectively

    Find $K_{n+1}^k = \bar{K}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $C_{n+1}^k = \bar{C}(u_{n+1}^k, \dot{u}_{n+1}^k)$, $p_{n+1}^k = \bar{p}(u_{n+1}^k, \dot{u}_{n+1}^k)$

    Assign for next increment: $K_{n+1} = K_{n+1}^k$, $C_{n+1} = C_{n+1}^k$, $p_{n+1} = p_{n+1}^k$, $u_{n+1} = u_{n+1}^k$, $\dot{u}_{n+1} = \dot{u}_{n+1}^k$, $\ddot{u}_{n+1} = \ddot{u}_{n+1}^k$

Figure 2-3: Pseudocode of the implementation of the nonlinear time integration algorithms used in this chapter.

The system is conservative and its total energy is constant and given by analytical integration of (2.28):

$$E_{tot} = \frac{1}{2}m\dot{u}^2 + S\ln\left(\cosh\left(u\right)\right) \tag{2.29}$$

In this example $S = 1000$ and the initial conditions are $u_0 = 0.3$ and $\dot{u}_0 = 0$.

2.3.3   Undamped SDOF oscillator with linear elastic spring

The third benchmark problem studied is a SDOF oscillator with linear elastic spring, for which the equation of motion is:

$$m\ddot{u} + ku = 0 \tag{2.30}$$

The system is conservative and its total energy is constant and given by analytical integration of (2.30):

$$E_{tot} = \frac{1}{2}m\dot{u}^2 + \frac{1}{2}ku^2 \tag{2.31}$$

In this example $k = 1000$ and the initial conditions are $u_0 = 4$ and $\dot{u}_0 = 0$.

2.3.4   SDOF oscillator with linear elastic spring and viscous damping

The fourth benchmark problem studied is a SDOF oscillator with linear elastic spring and viscous damping, for which the equation of motion is:

$$m\ddot{u} + c\dot{u} + ku = 0 \tag{2.32}$$

The damping coefficient is given by the equation:

$$c = 2\xi\sqrt{\frac{k}{m}} \tag{2.33}$$

where ξ is the critical viscous damping ratio equal to 2%. The system is not conservative; its total energy gradually decreases due to the damping force according to the equation:

$$E_{tot} = \left(\frac{1}{2}m\dot{u}_0^2 + \frac{1}{2}ku_0^2\right)\exp\left(\frac{-2\xi\sqrt{k/m}}{\sqrt{1-\xi^2}}t\right) \tag{2.34}$$

In this example $k = 1000$ and the initial conditions are $u_0 = 4$ and $\dot{u}_0 = 0$.

2.3.5   SDOF oscillator with linear elastic spring and Coulomb damping

The fifth benchmark problem studied is a SDOF oscillator with linear elastic spring and Coulomb damping, for which the equation of motion is:

$$m\ddot{u} + ku + F\frac{\dot{u}}{|\dot{u}|} = 0 \tag{2.35}$$

The system is not conservative; its total energy gradually decreases due to the Coulomb friction according to the equation:

$$E_{tot} = \frac{1}{2}k\left(u_0 - 2u_F \frac{\sqrt{k/m}}{\pi}t\right)^2 \tag{2.36}$$

and $u_F = F/k$ is a constant which may be interpreted as the static deformation of the system due to friction force F. In this example $k = 1000$, $u_F = 0.05$ and the initial conditions are $u_0 = 4$ and $\dot{u}_0 = 0$.

2.3.6   SDOF oscillator with linear elastic spring and hysteretic damping

The sixth benchmark problem studied is a SDOF oscillator with linear elastic spring and hysteretic damping, for which the equation of motion is:

$$m\ddot{u} + c\dot{u} + ku = 0 \tag{2.37}$$

The damping coefficient is given by the equation:

$$c = \eta\sqrt{\frac{k}{m}} \tag{2.38}$$

where η is the critical hysteretic damping ratio equal to 5%. The system is not conservative; its total energy gradually decreases due to the hysteretic damping force. Analytical equation for the estimation of the energy loss does not exist; the comparison is made with reference to the energy loss associated with the same benchmark problem, calculated for a much smaller time step. In this example $k = 1000$ and the initial conditions are $u_0 = 4$ and $\dot{u}_0 = 0$. The time step used for the reference energy loss is equal to $\Delta t = 0.0001$ and the duration of the dynamic analysis in this case is equal to 10000 time steps.

2.3.7   Results of benchmark problems in terms of total energy

Since no energy considerations were made for the formulation of the linear version of the time integration algorithm family assessed in this chapter, the evaluation of the various algorithms can be made in terms of energy measures, for the benchmark problems of the previous section. An efficient nonlinear time integration scheme should result in an accurate calculation of the energy of these systems. In cases that the analytical calculation of the energy is possible, the numerically calculated energy is compared to the former; in cases that this calculation is not possible (only for problem 6), the comparison is made with the energy calculated for a much smaller time step, a result which is practically the same for all algorithms involved. All the numerical analyses involved in this chapter were performed using MATLAB programming language.

In the diagrams of Figure 2-4 - Figure 2-9, the vertical axis represents the natural logarithm of the total energy of the vibrating system (E_{tot}) and the horizontal axis represents time. To avoid confusion due to the large number of algorithms being studied here, a separate graph has been constructed for each algorithm, and incorporated as a subplot in each Figure. The various subplots of these figures have the same scaling in their axes, to enable numerical

comparisons by simple observation of the deviation of the various curves from the dashed curve which gives the accurate value of the total energy of each system. The first four graphs in each figure from Figure 2-4 - Figure 2-9, distributed in the two first horizontal rows present results of the elementary time integration schemes considered in this chapter (i.e. the Newmark ACA, Newmark LA, Newmark BA and Fox-Goodwin methods), and the last nine graphs in each figure, distributed in the last three rows, present results of the advanced integration schemes (i.e. those belonging to the family of the GSSSS algorithms).

Another point to be highlighted is that, since the size of the time step and the duration of the dynamic analysis is the same for all the benchmark problems, and the small-strain stiffness and mass of the SDOF systems analyzed is the same, these factors which affect the performance of the time integration schemes are isolated from the results, and the differences observed are solely due to the differences in the quality of the algorithms (i.e. the magnitude of the numerical error introduced by them). This fact enables the easier understanding of the numerical performance of the time integration schemes considered in this chapter.

In Figure 2-4 the natural logarithm of the total energy of the undamped SDOF system with hardening spring (benchmark problem described in section 2.3.1) is presented for each algorithm. The exact value of the total energy is calculated at the onset of the vibration and should remain constant throughout the dynamic response, since there is no energy loss due to damping or other sources, and is plotted in all subplots with a dashed line. It is noted that the Continuous acceleration (CA) algorithms, along with the Newmark Average Acceleration Method perform generally better than the others in this problem. The least energy error is observed for the HHT-a method (or U_0-V_1-CA method), i.e. the zero order displacement-first order velocity-continuous acceleration method. Furthermore, the U_0-V_0-Opt and U_0-V_0-DA algorithms seem to perform better than their U_0-V_1 or U_1-V_0 counterparts. It was checked that if the size of the time step is sufficiently reduced, all the algorithms of the GSSSS family produce the exact response. The largest numerical error is observed in the case of U_1-V_0-DA algorithm. Finally, it is observed that numerical errors associated with the time integration schemes may result in increase of the total energy in some cases.

In Figure 2-5 the natural logarithm of the total energy of the undamped SDOF system with softening spring (benchmark problem described in section 2.3.2) is presented for each algorithm. Again the most accurate method is the HHT-a method, in the graph of which the difference between the numerically calculated total energy and the analytical reference energy is hard to discern. A difference is noted in the performance of the Newmark Average Constant Acceleration algorithm between the cases of the undamped SDOF system with hardening spring and the undamped SDOF system with softening spring. As it is easily seen, this algorithm loses its accuracy in the latter case, which is comparable to that of the other elementary integration schemes. The superiority of the Continuous

Acceleration advanced algorithms is remarkable. The largest error is observed for the case of U1-V0-DA algorithm, as in the case of the undamped SDOF system with hardening spring. In general, undulations are exhibited in the energy curves of the less accurate algorithms, either advanced or elementary. The total energy calculated may decrease or increase with respect to the reference energy curve, depending on the numerical error introduced by each algorithm.
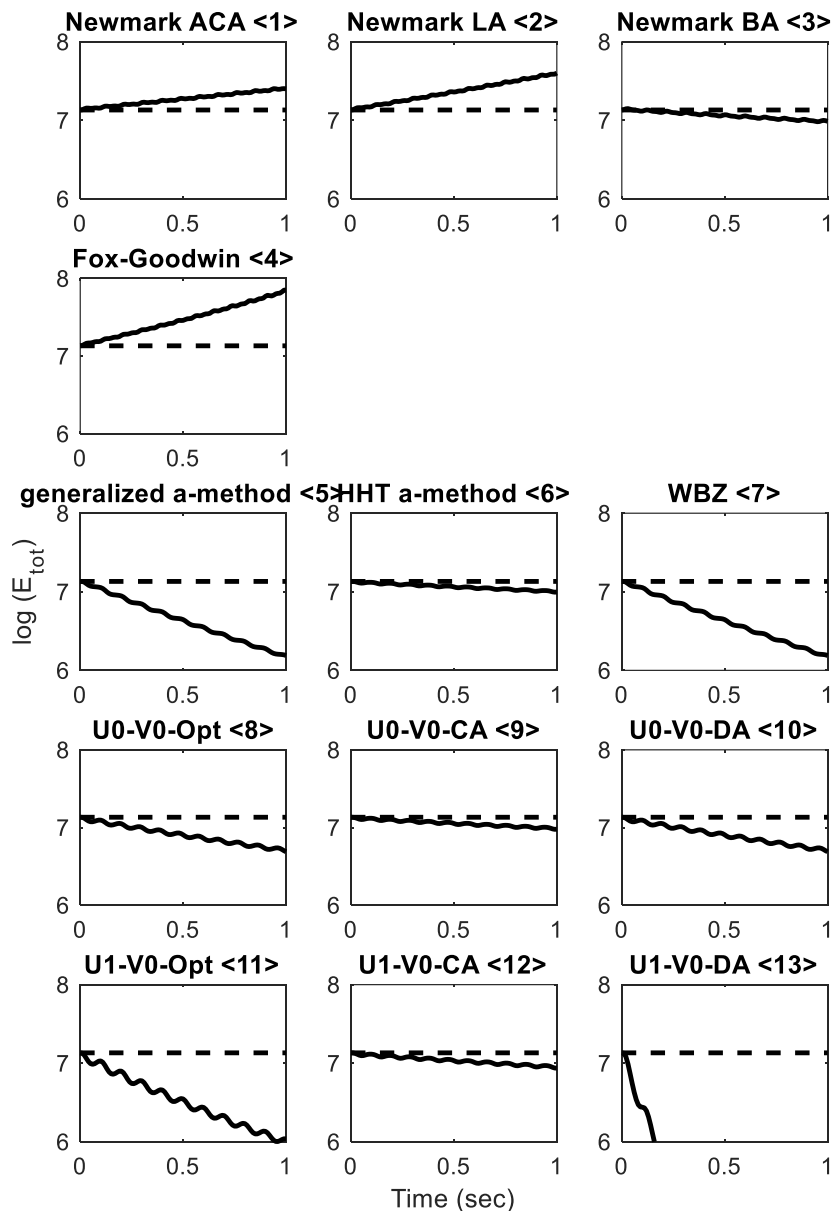
Figure 2-4: Natural logarithm of the total energy for problem 1 (undamped SDOF system with hardening spring).

Newmark ACA <1>   **Newmark LA <2>**   **Newmark BA <3>**

Fox-Goodwin <4>

generalized a-method <5> **HHT a-method <6>**   **WBZ <7>**

U0-V0-Opt <8>   **U0-V0-CA <9>**   **U0-V0-DA <10>**

U1-V0-Opt <11>   **U1-V0-CA <12>**   **U1-V0-DA <13>**

Time (sec)
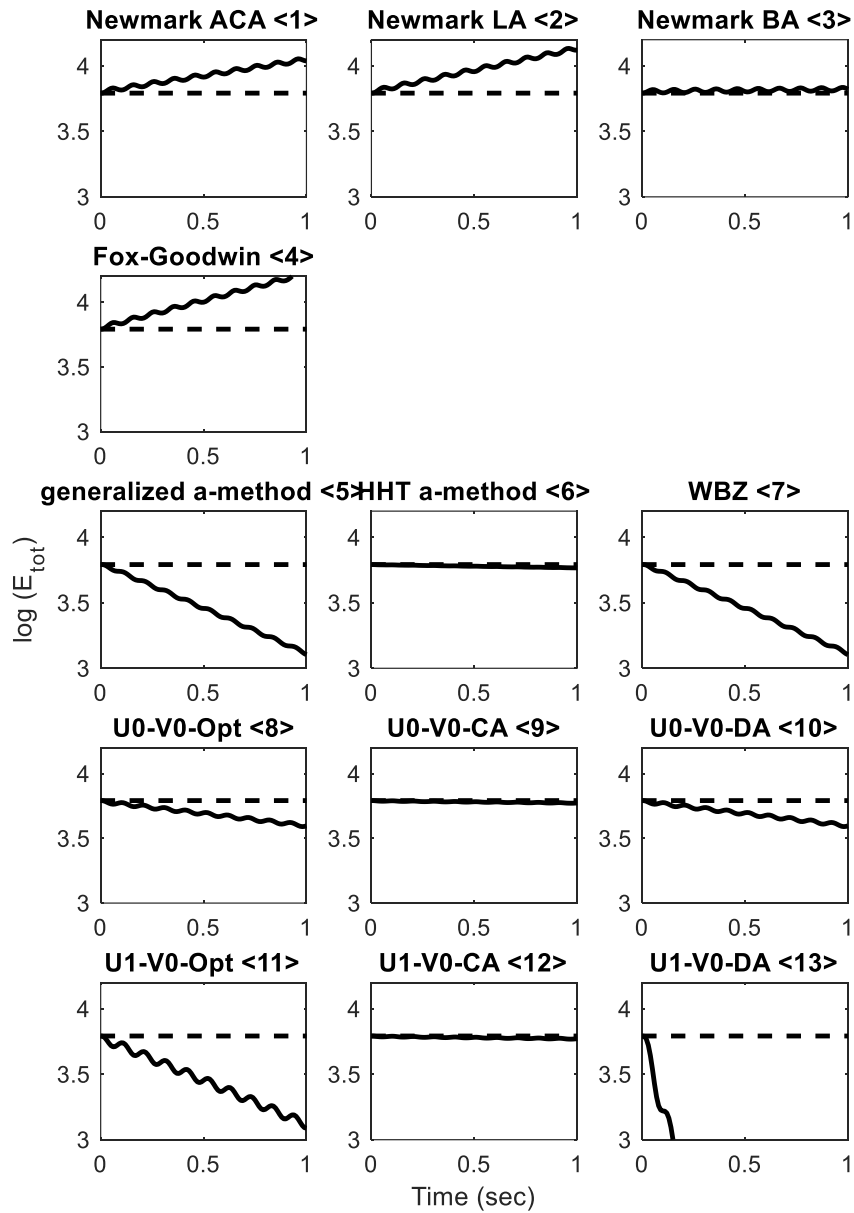
Figure 2-5: Natural logarithm of the total energy for problem 2 (undamped SDOF system with softening spring).
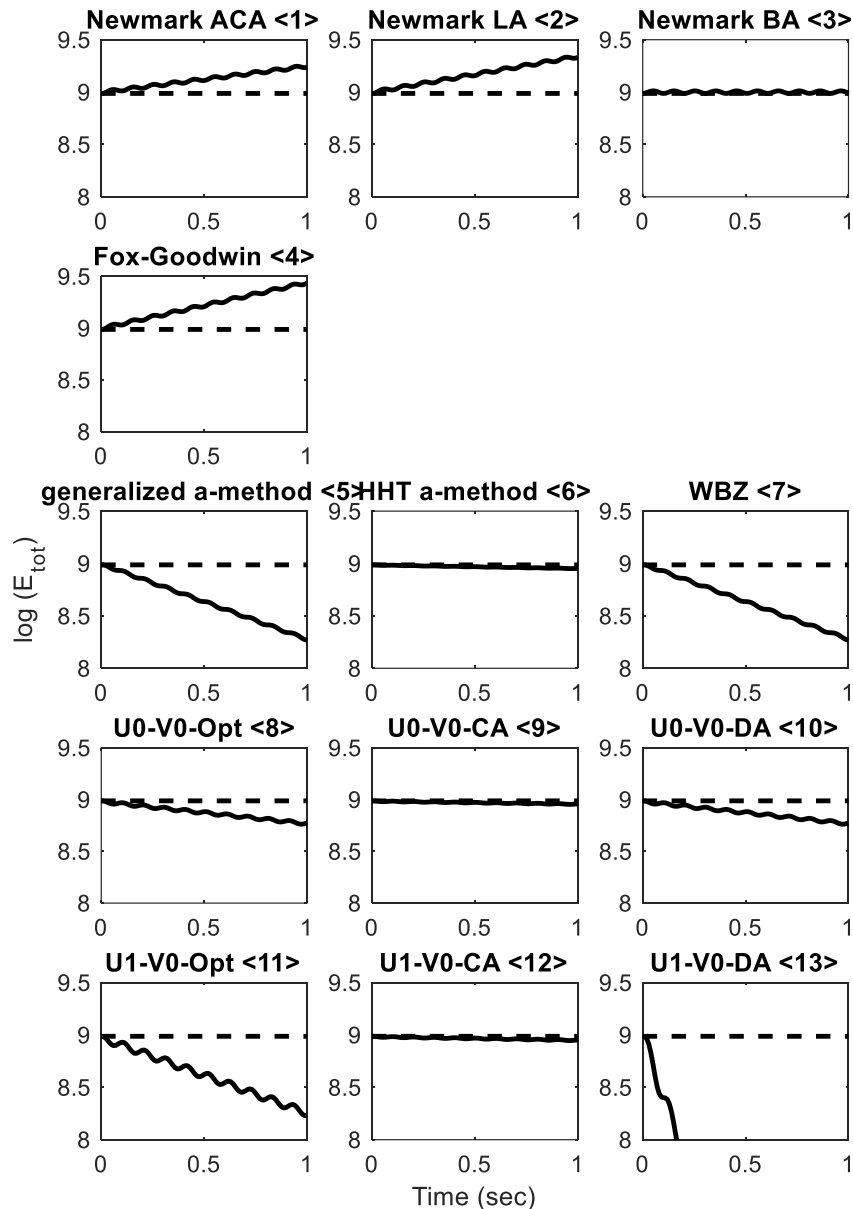
Figure 2-6: Natural logarithm of the total energy for problem 3 (undamped SDOF system with linear elastic spring).

In Figure 2-6, the natural logarithm of the total energy of the undamped SDOF system with linear elastic spring (benchmark problem described in section 2.3.3) is presented. It is seen that the HHT-a method is the most accurate, and among the elementary methods, the Newmark Backward Acceleration method is of comparable accuracy with the HHT-a method. Apart from the Continuous Acceleration and the Newmark BA methods, the remaining algorithms give less accurate results, of various orders of error. The least accurate method is the U1-V0-DA method.

In Figure 2-7, the natural logarithm of the total energy of the linear elastic SDOF system with viscous damping (benchmark problem described in section 2.3.4) is presented. Similarly to the undamped linear elastic SDOF case, the HHT-a and Newmark BA methods are the most accurate among the advanced and the elementary methods respectively. It has to be pointed out that the total energy of the system decreases with time due to the presence of viscous damping; therefore the reference energy curves in the subplots have negative slope and are straight because the energy decrease is exponential and its natural logarithm linear.

In Figure 2-8, the natural logarithm of the total energy of the linear elastic SDOF system with damping due to Coulomb friction (benchmark problem described in section 2.3.5) is presented. The most accurate estimation of energy is provided by the Continuous acceleration algorithms and the Newmark Backward Acceleration method. The performance of the U1-Vo-DA is the poorest as in the previous examples. The reference energy curves are not straight lines, because the rate of energy dissipation is quadratic and not exponential, as can be seen from eq.(2.36) .

In Figure 2-9 the natural logarithm of the total energy of the linear elastic SDOF system with hysteretic damping (benchmark problem described in section 2.3.6) is presented. Similarly to the results of the other problems, the most accurate algorithms prove to be the Continuous Acceleration algorithms. The most accurate scheme appears to be the HHT-a method among the advanced algorithms, and the Newmark Backward Acceleration scheme among the elementary algorithms. Generally, the energy plots show periodic fluctuations and the Uo-Vo-Opt and Uo-Vo-DA algorithms seem to perform better than their Uo-V1 or U1-Vo counterparts, a general trend observed in Figure 2-4 - Figure 2-9. Analytical calculation of the reference total energy is not possible in this benchmark problem and for this reason it is computed by setting a much smaller time step, as outlined in section 2.3.6. The linear configuration of the various curves implies that the energy dissipation during the free vibration of a hysteretically damped SDOF oscillator varies exponentially with time; this type of variation was previously observed also for the viscously damped SDOF oscillator presented in Figure 2-7 and described in section 2.3.4.

It is generally observed in the numerical results in terms of energy presented above that the most accurate scheme in the linear and nonlinear regimes is the HHT-a method. The time integration algorithms considered in this chapter (either advanced or elementary) do not always ensure that the variation of energy is as expected (e.g. conserved for undamped and unforced systems). However, any degree of accuracy can be achieved for all types of oscillators considered in this chapter, if the time step is sufficiently reduced.
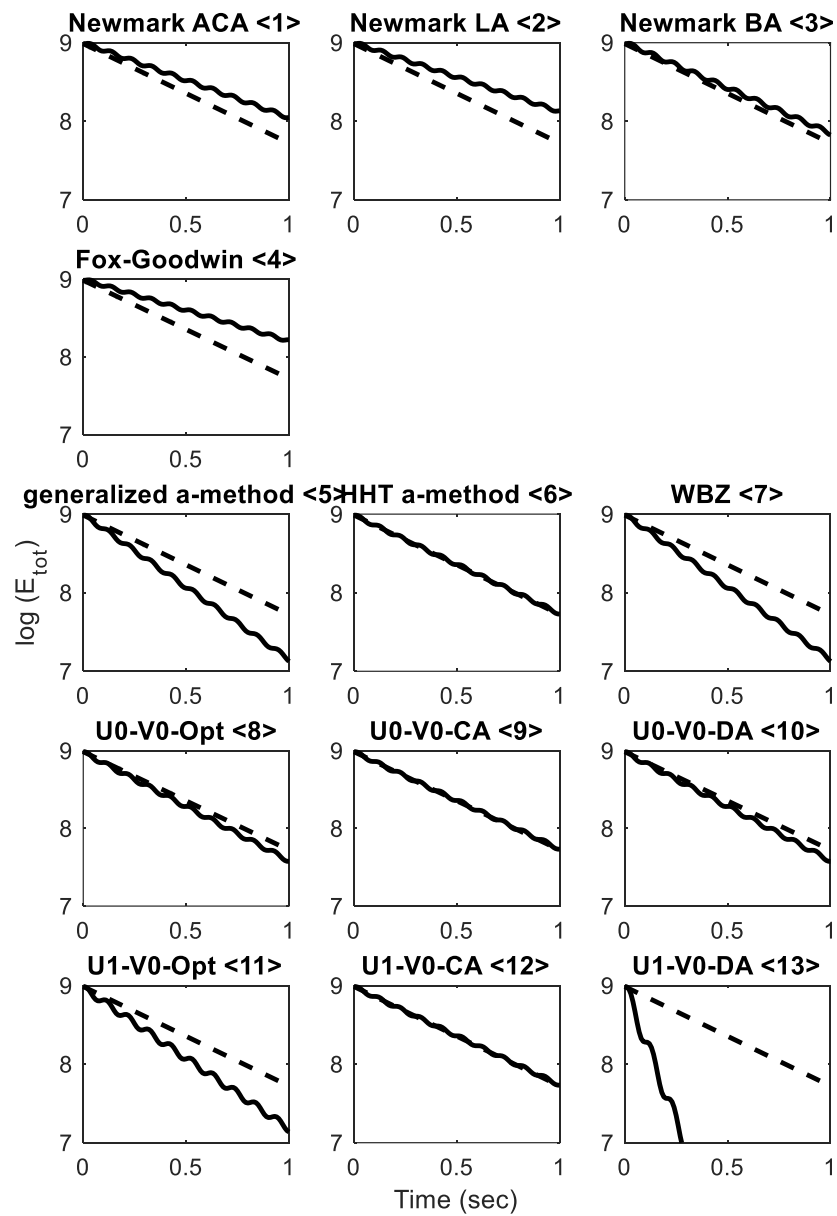
Figure 2-7: Natural logarithm of the total energy for problem 4 (linear elastic SDOF system with viscous damping).
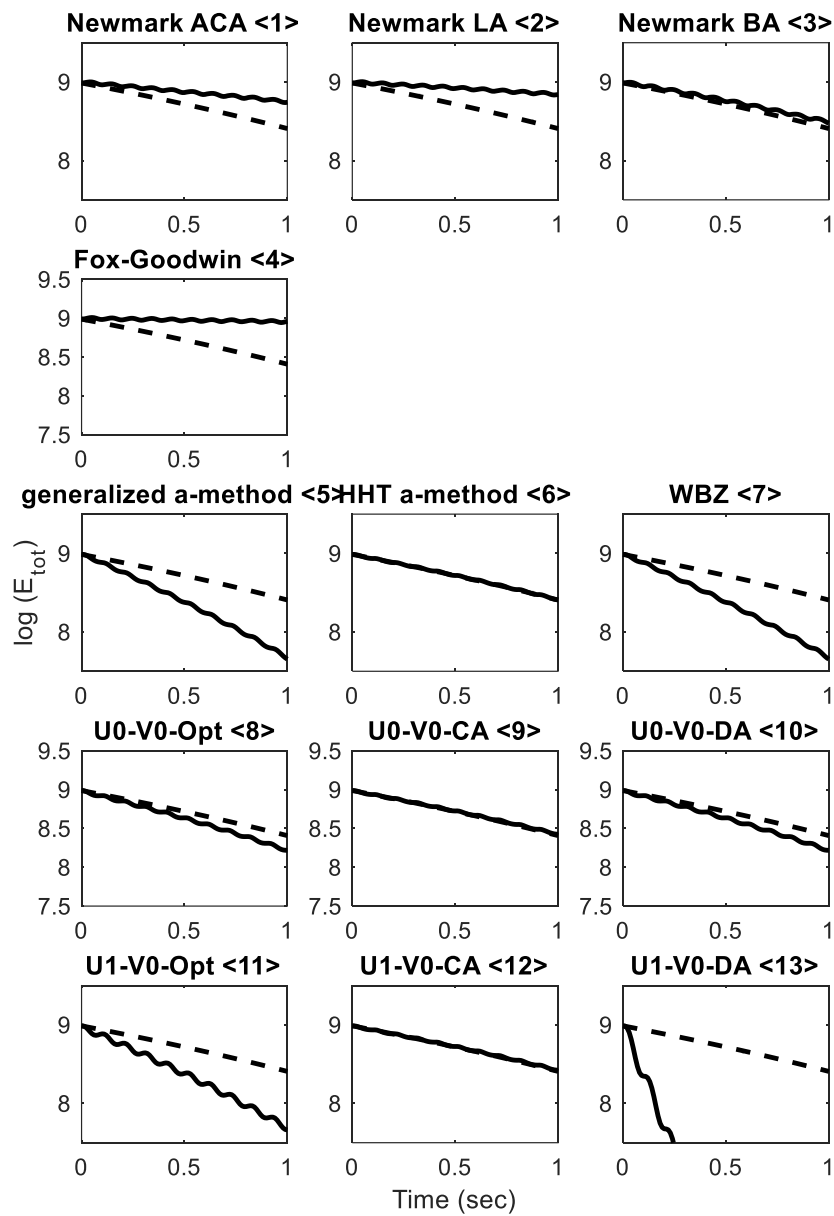
Figure 2-8: Natural logarithm of the total energy for problem 5 (linear elastic SDOF system with damping due to Coulomb friction).
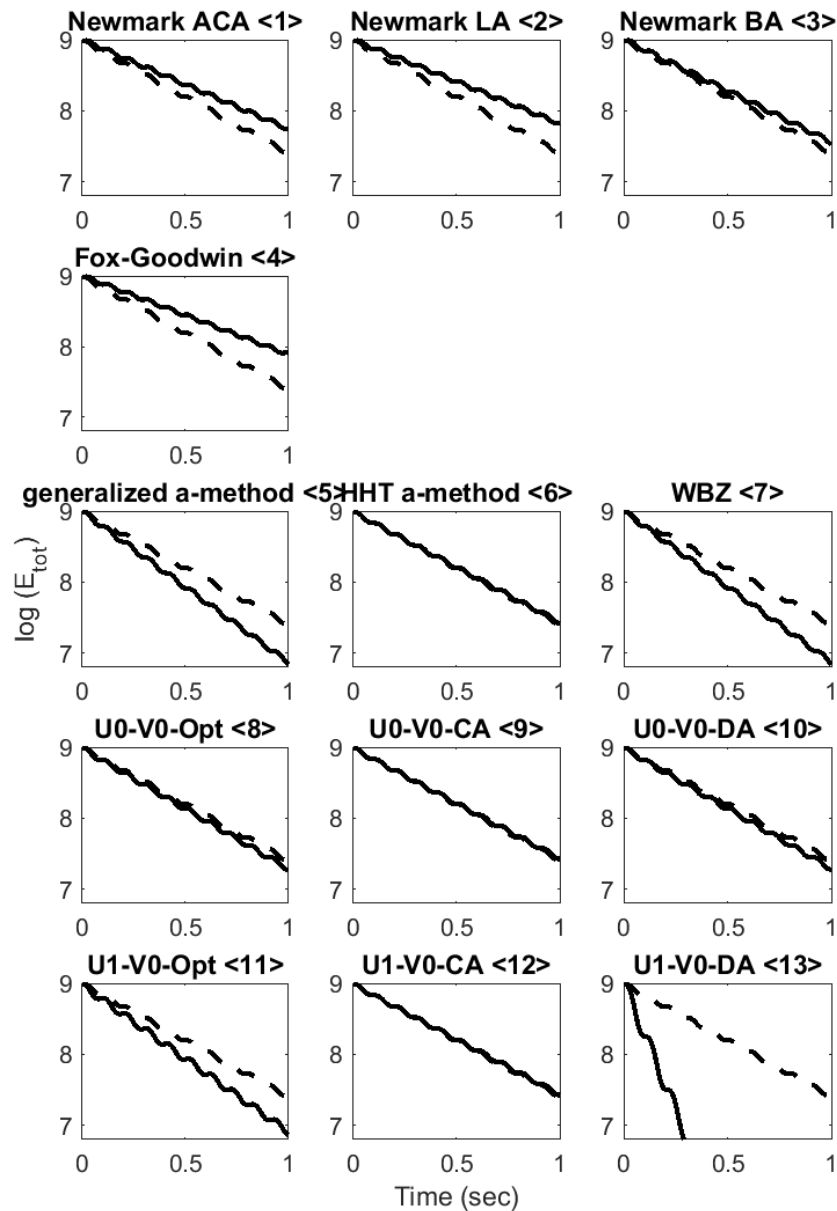
Figure 2-9: Natural logarithm of the total energy for problem 6 (linear elastic SDOF system with hysteretic damping).

It is generally observed in the numerical results in terms of energy presented above that the most accurate scheme in the linear and nonlinear regimes is the HHT-a method. The time integration algorithms considered in this chapter (either advanced or elementary) do not always ensure that the variation of energy is as expected (e.g. conserved for undamped and unforced systems). However, any degree of accuracy can be achieved for all types of oscillators considered in this chapter, if the time step is sufficiently reduced.

2.3.8   Comparison of algorithm performance in linear and nonlinear regime

In this section the performance of the various time integration schemes is evaluated for linear and nonlinear problems. Comparisons are made between these two categories of analysis. Since the time step, the time duration, and the algorithmic constants are the same for all benchmark problems analyzed, and the stiffness of the various oscillators in their small-strain region is the same, the comparisons in this section are made with reference to the quality of the time integration schemes and their associated numerical error.

In Figure 2-10, a dimensionless quantity equal to the relative error of the total energy of the various systems is plotted against integration time. Three schemes are used for the integration of the SDOF systems involved: the Newmark Average Constant Acceleration (ACA), Newmark Linear Acceleration (LA) and Newmark Backward Acceleration (BA) methods. The various curves include linear as well as nonlinear energy error. It is seen that, depending on the algorithm used, the various SDOF systems yield different integration errors. In the case of Newmark ACA algorithm, the largest error is observed for the viscoelastic and elastic system with Coulomb damping, whereas the lowest energy error is observed for softening and linear elastic undamped systems. In the case of Newmark LA algorithm, the hardening system shows the maximum relative error, whereas the softening and linear elastic undamped systems show the minimum relative error. Finally, Newmark BA algorithm yields the most accurate results for linear elastic undamped system, whereas the maximum error is observed for the hardening and viscoelastic systems.

Similar to Figure 2-10, Figure 2-11 compares the relative error of the total energy of the various systems, plotted against integration time, where the continuous acceleration (CA) algorithms are used. It is apparent that the hardening system yields relatively large energy numerical error for all three CA members of the GSSSS family. The maximum error is generally observed for the U1-V0-CA (<12>) algorithm, for all problems considered in this chapter. Compared to the error of the hardening system, the error of the remaining systems is considerably lower. The error is much lower if Continuous Acceleration algorithms are used, compared to the elementary time integration algorithms the results of which are shown in Figure 2-10.

Corresponding results with those of the previous paragraph are observed in Figure 2-12 and Figure 2-13, for the optimal numerical dissipation and dispersion (Opt) algorithms and the discontinuous acceleration (DA) algorithms respectively. The hardening system exhibits the maximum total energy error for all these types of algorithms, as can be seen in both figures. The lowest energy error is observed for the linear viscoelastic system for all optimum numerical dissipation and dispersion algorithms, as well as discontinuous acceleration algorithms. Finally, the error associated with the optimal numerical dissipation and dispersion algorithms and the discontinuous acceleration algorithms is

certainly higher in general than that of the continuous acceleration methods, for the same SDOF benchmark problems studied.

The main conclusion from this figure is that the linear and nonlinear versions of the GSSSS algorithm family yield comparable percentages of total energy error, and in many instances, the error of the total energy of nonlinear systems is lower than the error in the total energy of linear systems. Given that in each subplot the problems considered have the same initial conditions, time step, integration algorithm, nonlinear convergence parameters, etc., the differences between the various curves in each subplot originate mainly from the differences between the constitutive properties of the oscillators. It is a fact that the existence of iterations within a time step is a source of additional error. The linear integration algorithms do not require any iterations within a time step. The opposite is the case with the nonlinear time integration algorithms. Consequently, it is expected that the error accumulated in the nonlinear response will be higher than the error of the linear response. However, from the results presented in this chapter, it is concluded that this does not happen in a regular basis; there are cases in which the error of a nonlinear problem is lower than that of a linear problem, while the same integration scheme is used. From this observation, the conclusion is drawn that the extension of the linear versions of the GSSSS algorithms into the nonlinear regime with incorporation of Newton–Raphson iterations results in nonlinear time integration algorithms which possess a similar (and sometimes higher) level of accuracy with their linear counterparts.

The results presented in Figure 2-10 -Figure 2-13 are summarized in Figure 2-14, where the maximum relative error of the total energy for each SDOF system – integration algorithm pair and for the duration of the oscillation is shown in the form of bar chart. In the horizontal axis each label refers to the type of SDOF system and the time integration algorithm with which its response is calculated, whereas in the vertical axis, the percentage of relative energy error is shown. The lowest error is observed for the Continuous Acceleration algorithms (including HHT algorithm) for the linear elastic with Coulomb damping, undamped softening, undamped linear elastic and viscoelastic oscillators and the Newmark Backward Acceleration algorithm for the undamped linear elastic oscillator. On the other hand, the error associated with the Discontinuous Acceleration algorithms and the undamped hardening oscillator is the largest among all cases and algorithms considered in this chapter.
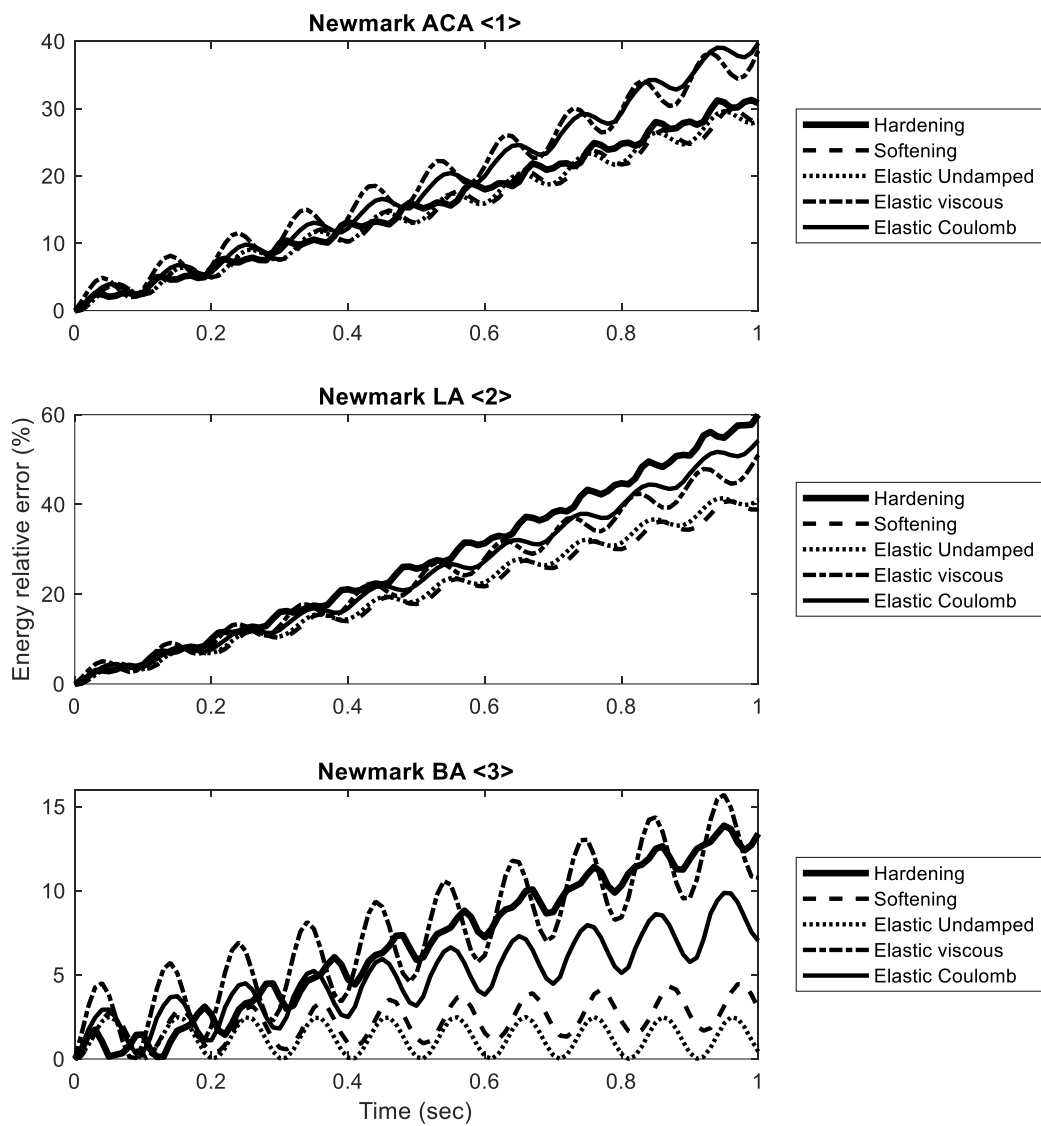
Figure 2-10: Time history of the relative error of the total energy of the SDOF oscillators of problems 1–5 integrated by the Newmark ACA, Newmark LA and Newmark BA algorithms of the GSSSS family.
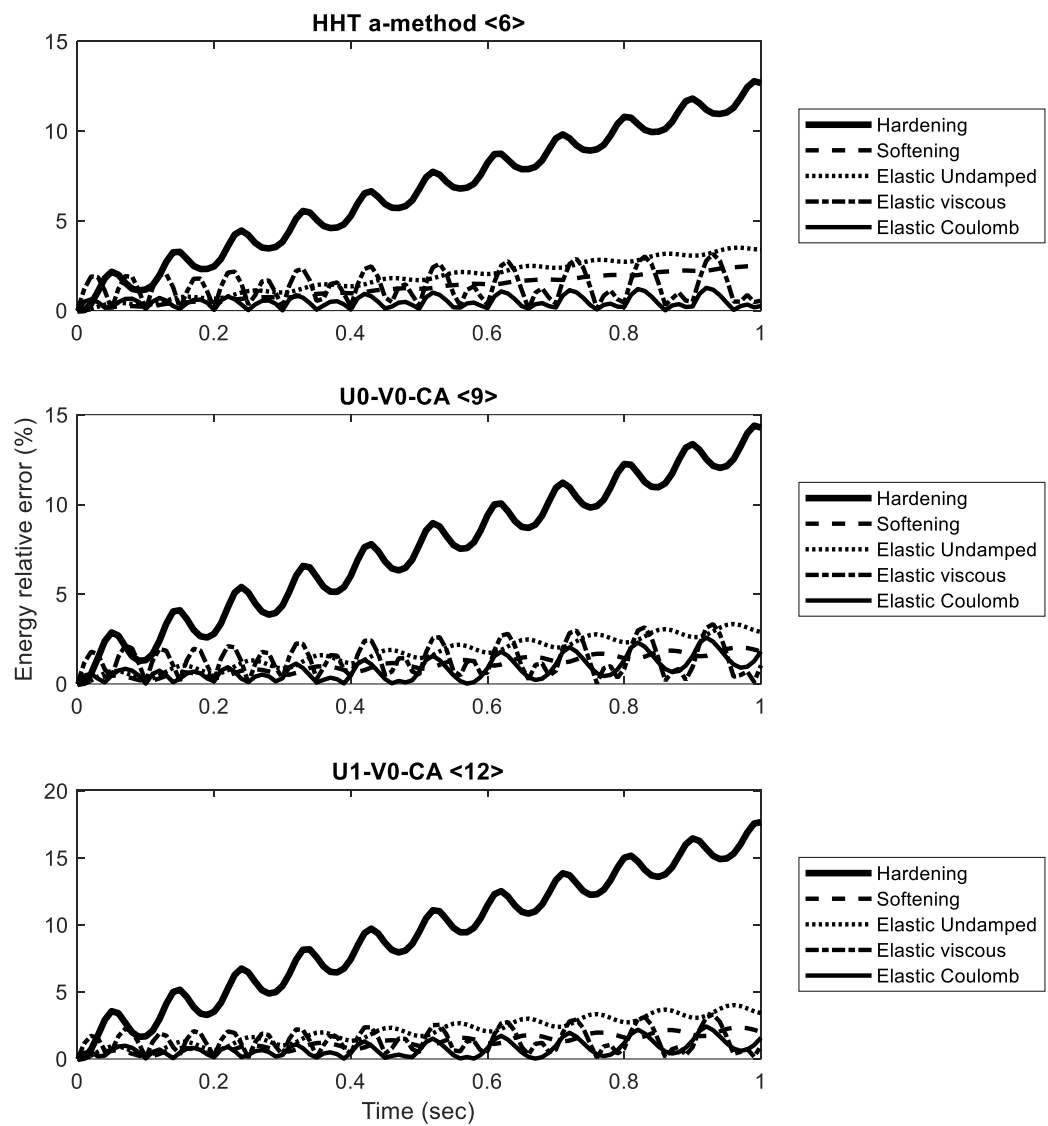
Figure 2-11: Time history of the relative error of the total energy of the SDOF oscillators of problems 1–5 integrated by the Continuous Acceleration (CA) algorithms of the GSSSS family.
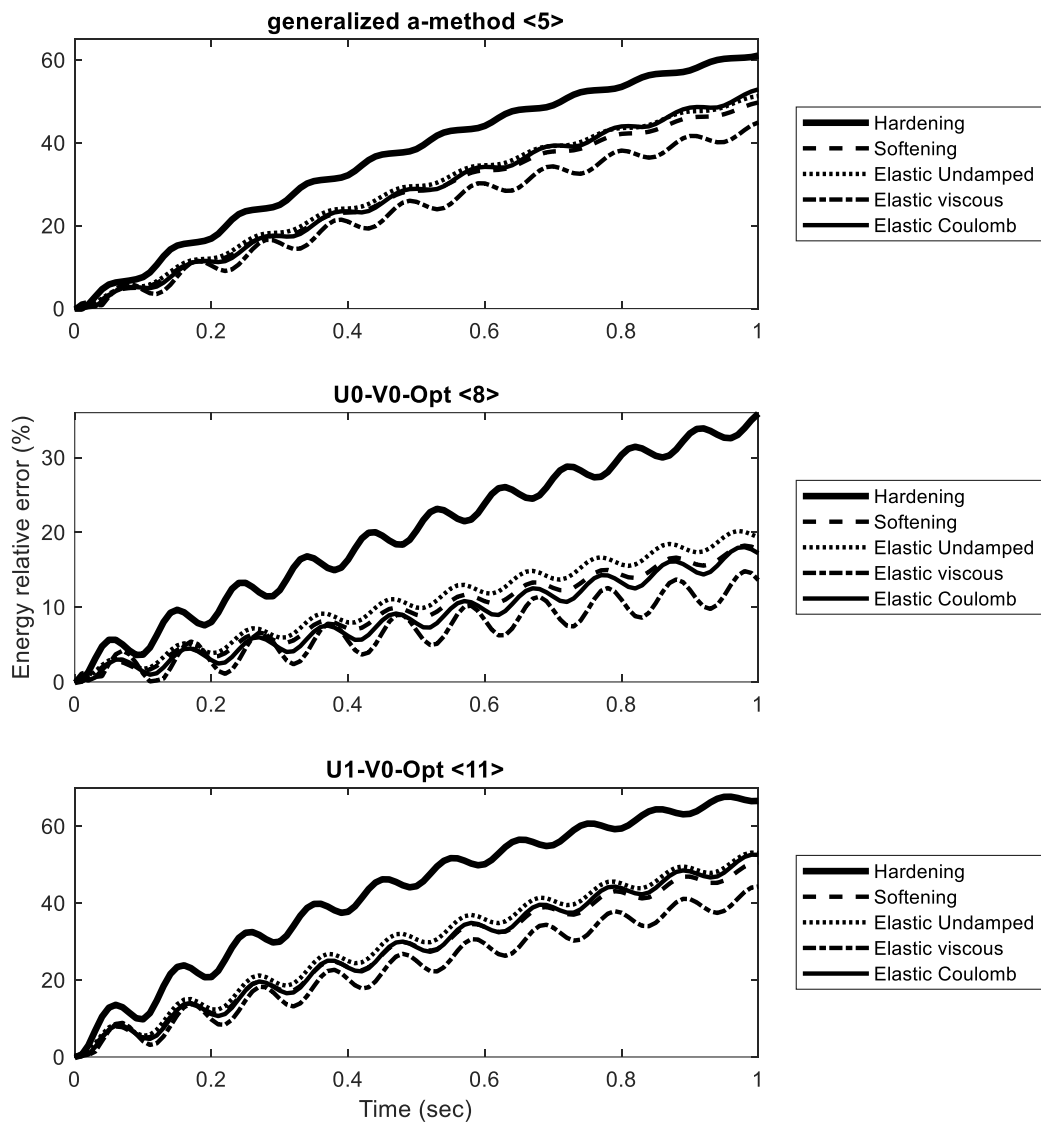
Figure 2-12: Time history of the relative error of the total energy of the SDOF oscillators of problems 1–5 integrated by the optimal numerical dissipation and dispersion (Opt) algorithms of the GSSSS family.
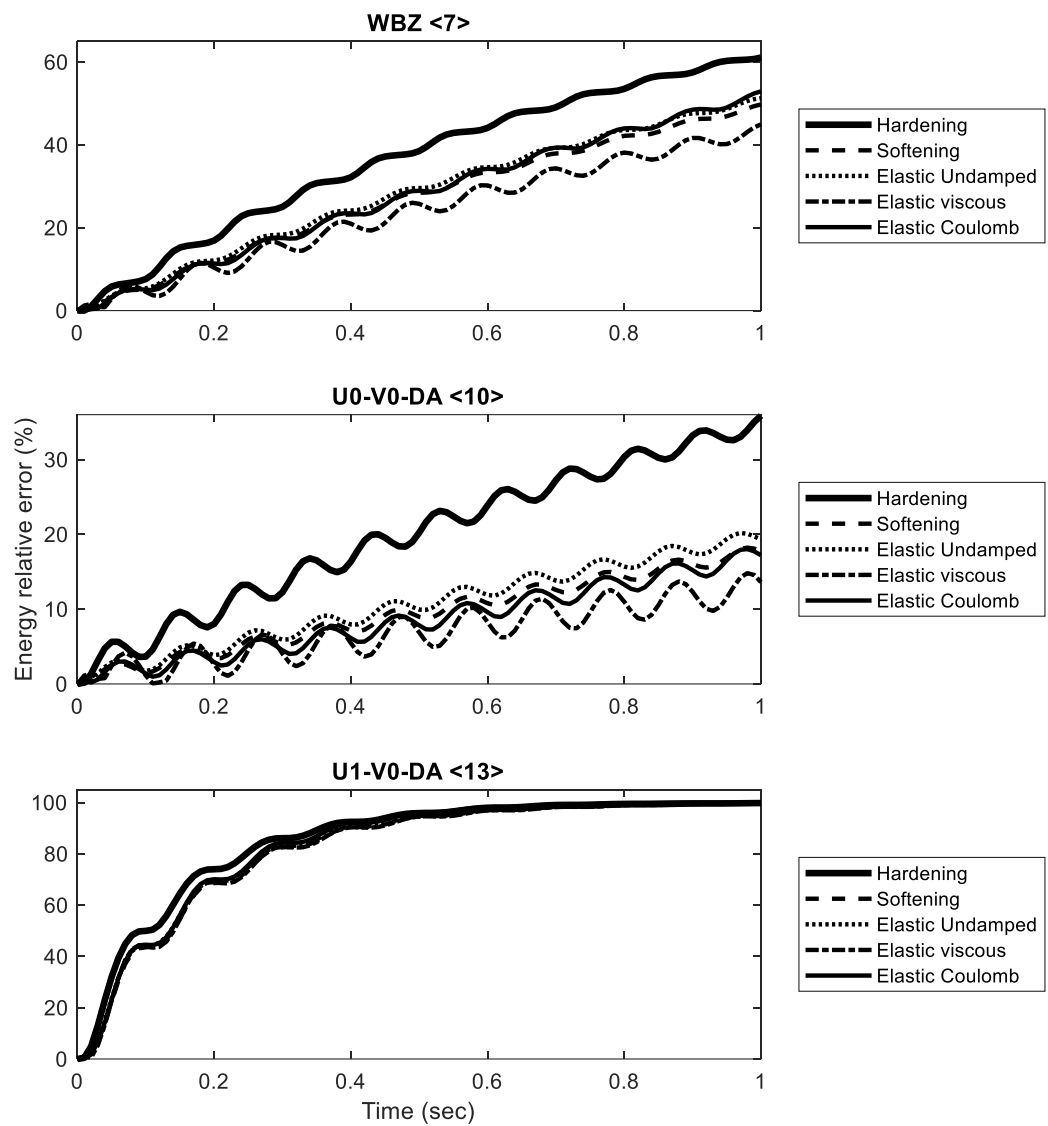
Figure 2-13: Time history of the relative error of the total energy of the SDOF oscillators of problems 1–5 integrated by the Discontinuous Acceleration (DA) algorithms of the GSSSS family.
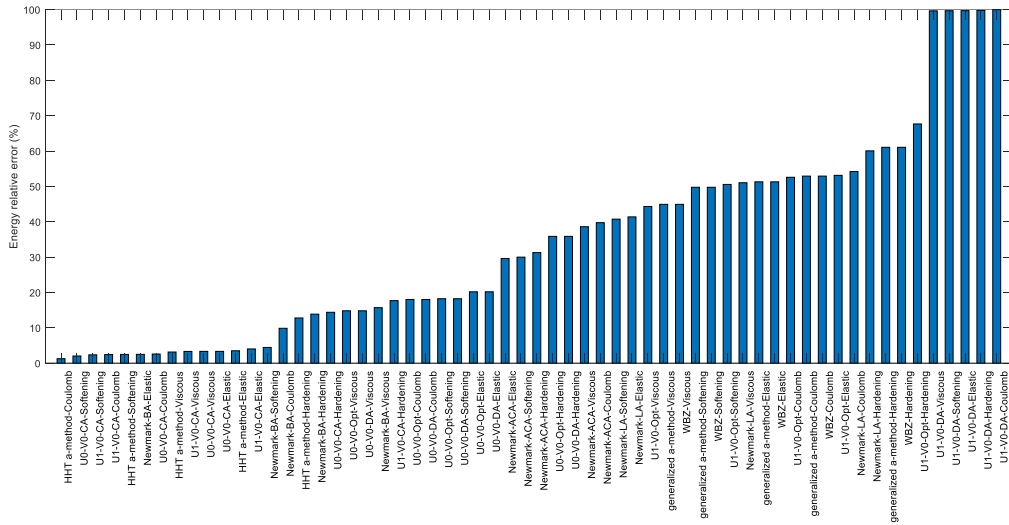
Figure 2-14: Maximum relative error of the total energy of the SDOF oscillators of problems 1–5 integrated by the nonlinear integration algorithms developed in this chapter.

2.4   Benchmark problem of undamped linear elastic system with 3 DOFs for evaluation of algorithms' performance

In this section, the problem studied by Bathe & Noh (2012) is considered, which is a 3-degree-of-freedom spring system shown in Figure 2-15. This problem is used as a benchmark problem mainly due to the fact that it represents finite element models which include high stiffness elements, as well as flexible elements. High stiffness elements can involve, for example, rigid connections or penalty factors or various types of constraints of the model. Such high stiffness values used in finite element models have rarely any physical meaning; they are almost always used to provide constraints. This system is studied here in order to compare the dynamic response produced by the various time integration algorithms with the "exact" presented by Bathe & Noh (2012). The numerical data $k_1 = 1e^7$, $k_2 = 1$, $m_1 = 0$, $m_2 = 1$, $m_3 = 1$ are used for the system properties. Also, node 1 has prescribed displacement equal to:

$$u_1 = \sin \omega_p t \tag{2.39}$$

where $\omega_p = 1.2$. The governing equation of motion is the following:

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \\ \ddot{u}_3 \end{bmatrix} + \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} R_1 \\ 0 \\ 0 \end{bmatrix} \tag{2.40}$$

which can be rewritten for the unknown displacements u_2 and u_3 as follows:

$$\begin{bmatrix} m_2 & 0 \\ 0 & m_3 \end{bmatrix}\begin{bmatrix} \ddot{u}_2 \\ \ddot{u}_3 \end{bmatrix} + \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix}\begin{bmatrix} u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} k_1 u_1 \\ 0 \end{bmatrix} \qquad (2.41)$$

The initial conditions are considered to be zero for both the displacements and the velocities. The response of the system is estimated for a time duration of 10 sec, and the time step used is $\Delta t = 0.2618 \sec$.
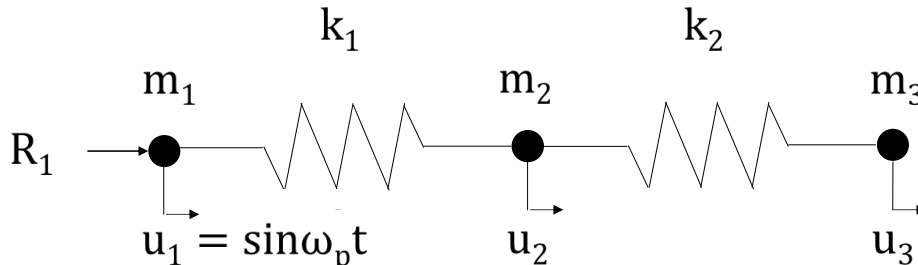
Figure 2-15: Model of three degrees of freedom spring system (Bathe & Noh, 2012).

The displacement, velocity and acceleration at degree of freedom 2 of the 3dof system are calculated and compared to the corresponding results presented by Bathe & Noh (2012). The displacement response of the degree of freedom 2 of the 3DOF oscillator is illustrated in Figure 2-16. The exact solution, as presented by Bathe & Noh (2012) is plotted with a dashed curve. It is evident that for all advanced algorithms, except for the U1-V0-DA algorithm, the numerically calculated displacement time history practically coincides with the exact solution. Among the elementary time integration schemes, only the Newmark Backward Acceleration method manages to trace the response, whereas all others fail. It is concluded that the majority of the advanced algorithms reproduce the exact displacement result of the time integration procedure regarding the structural system studied by Bathe & Noh (2012). Things become different in the case of the velocity at dof 2 of the 3DOF undamped linear elastic system, as the response is computed accurately only by the U0-V1 and U1-V0 algorithms except for the U1-V0-DA algorithm, as seen in Figure 2-17. Spurious oscillations around the exact solution are observed in the response produced by the U0-V0 algorithms. All the elementary integration schemes fail to reproduce the exact velocity response; the erroneous behavior of the Newmark BA algorithm has been already shown by Bathe & Noh (2012). It seems that the generalized-a and the WBZ methods are more accurate than the HHT-a method in the first steps of the response, without this meaning that the latter is generally subordinate with respect to the others.

The acceleration time history at dof 2 of the 3DOF undamped linear elastic system is shown in Figure 2-18. The exact solution, as presented by Bathe & Noh (2012), is most accurately reproduced by the generalized-a, WBZ and U1-V0-Opt methods. All the elementary time integration methods fail to accurately calculate the dynamic response. The spurious undulations which are present in certain subplots have already been pointed out (Bathe & Noh, 2012). In the velocity and acceleration time histories some overshooting behaviour is observed in the

beginning of the response, especially in the cases of U0-V1 and U1-V0 algorithms. This is expected due to the way these algorithms are formulated. However, despite this overshooting behaviour, it is observed that they are generally more accurate (with an exception for the DA version U1-V0 algorithm category) than their optimal numerical dissipation and dispersion (Opt) counterparts.
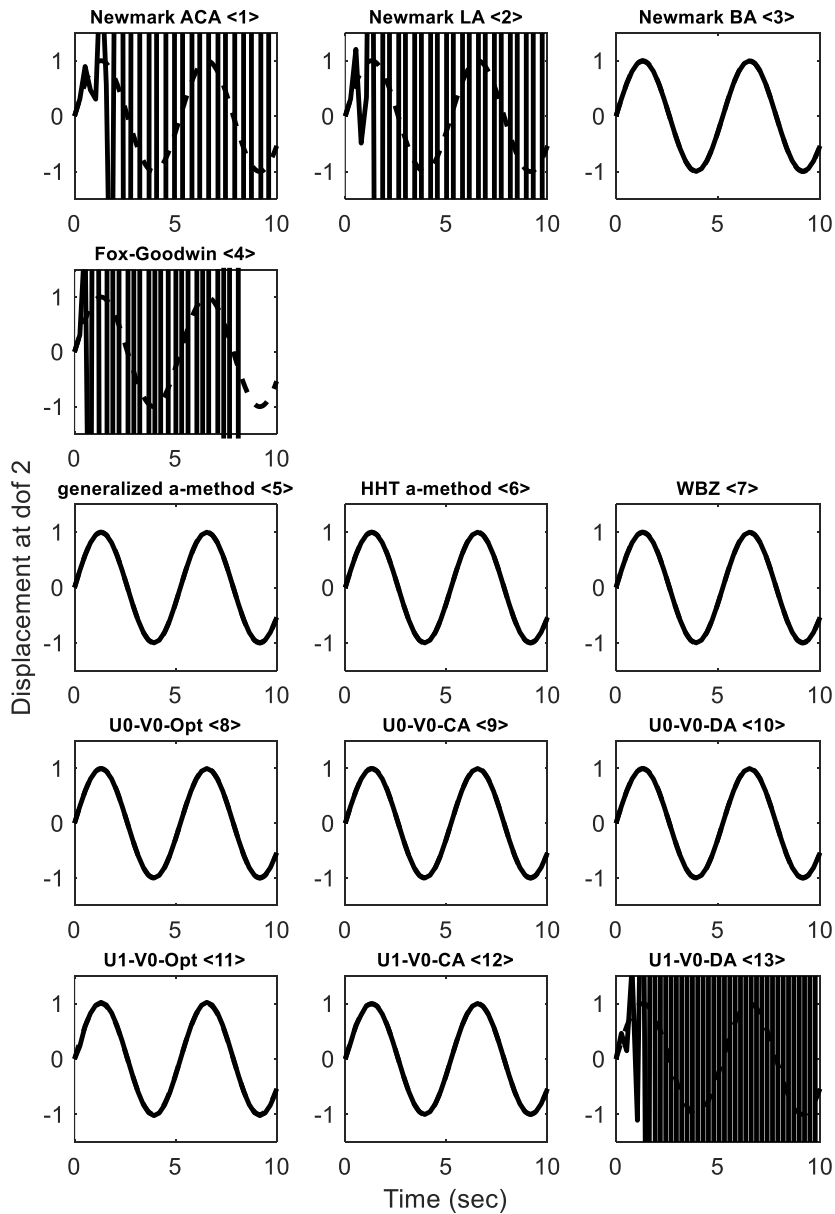
Figure 2-16: Time history of the displacement at the dof 2 of the 3DOF undamped linear elastic system.
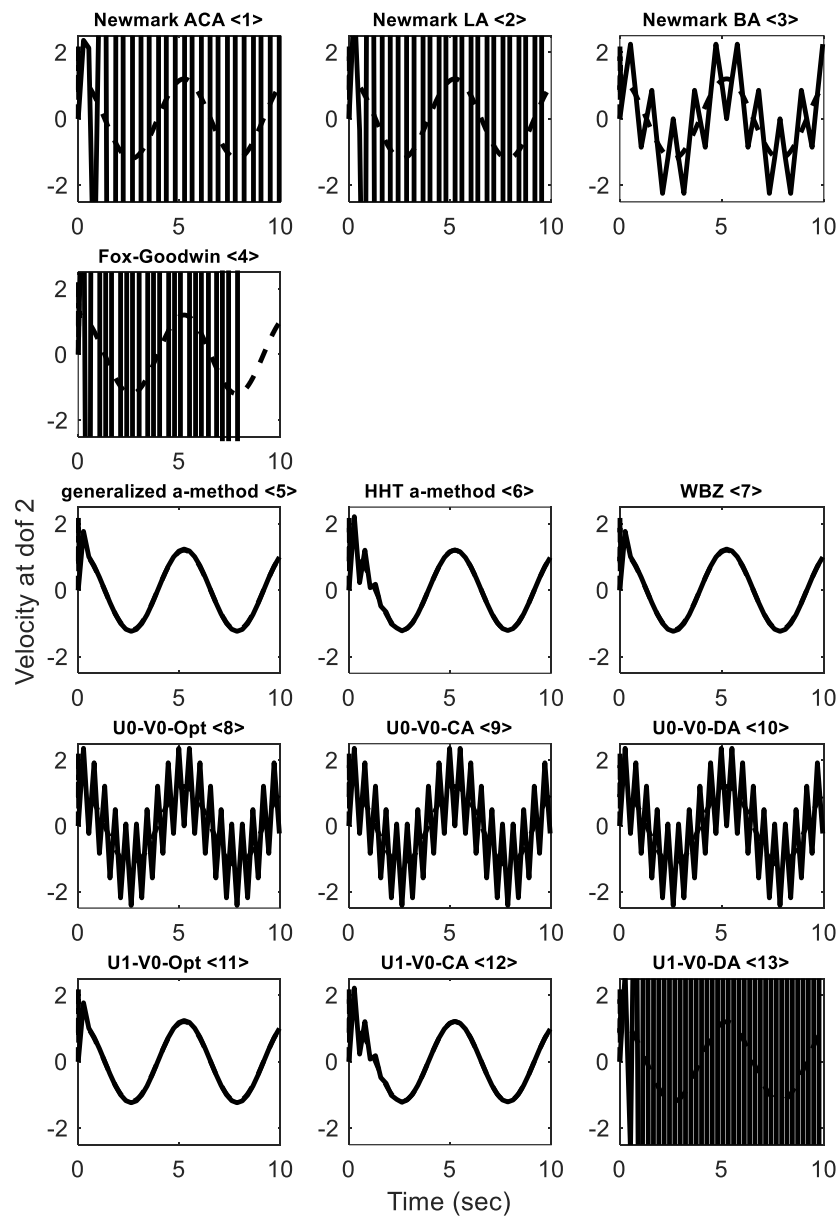
Figure 2-17: Time history of the velocity at the dof 2 of the 3DOF undamped linear elastic system.
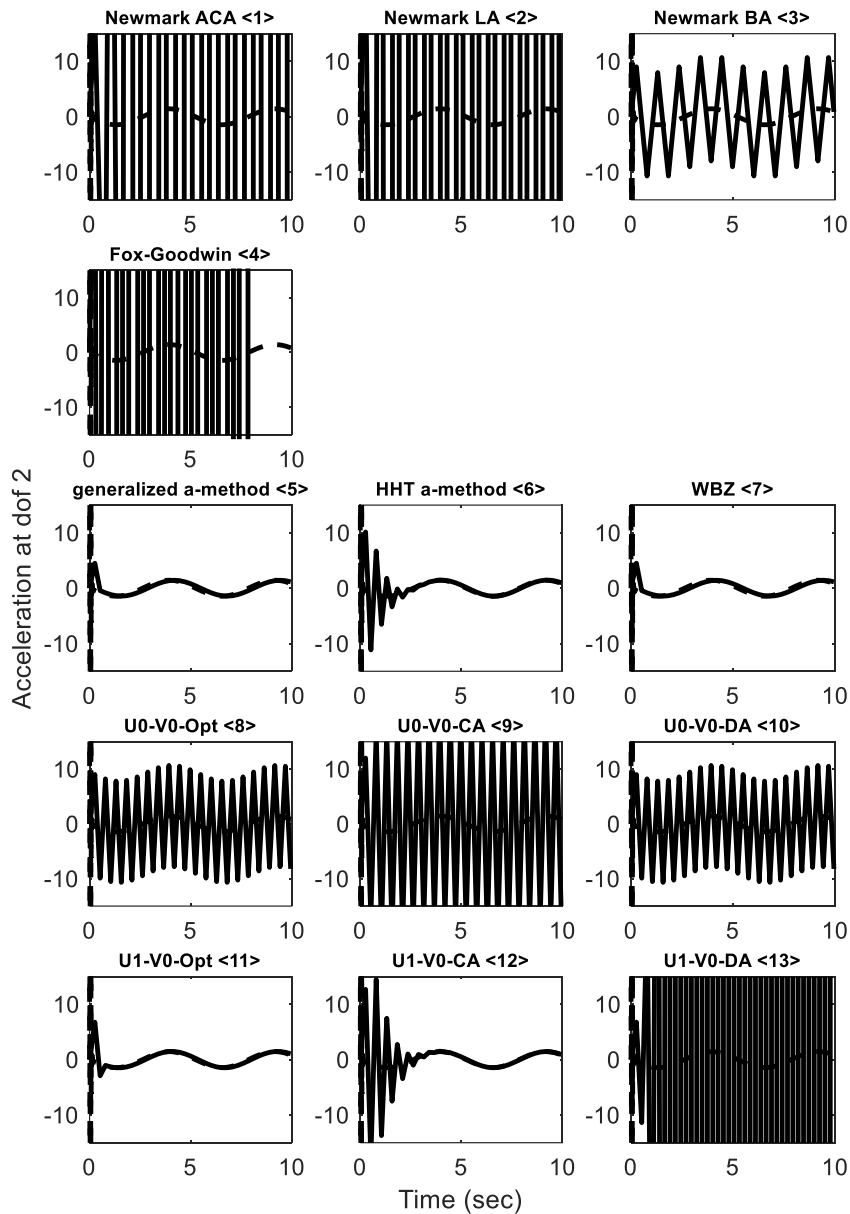
Figure 2-18: Time history of the acceleration at the dof 2 of the 3DOF undamped linear elastic system.

2.5 Benchmark problem of large angular oscillation of a simple undamped pendulum

A fundamental benchmark problem which is widely used for testing the nonlinear performance of various time integration algorithms is the nonlinear oscillation of a pendulum with large rotations. Consider an undamped pendulum in the gravity field which is comprised of a point mass m and a weightless rigid

rod of length L. Here, g is the acceleration of gravity. In this example, as is done in the previous benchmark SDOF problems, in order to verify the accuracy of the family of the GSSSS algorithms the exact solution of the nonlinear free vibration of the pendulum will be employed. The simple pendulum system is shown graphically in Figure 2-19.
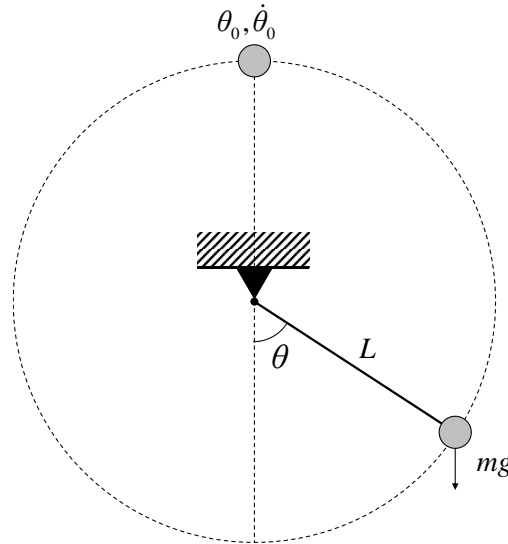
Figure 2-19: Simple undamped pendulum system with large rotations.

Assume the pendulum to be initially at its lower equilibrium point with initial angular velocity $\dot{\theta}_0$. The differential equation of the nonlinear pendulum is (Bornemann et al., 2002):

$$mL\ddot{\theta} + mg\sin(\theta) = 0 \tag{2.42}$$

with the following initial conditions:

$$\theta(0) = 0, \dot{\theta}(0) = \dot{\theta}_0 \tag{2.43}$$

where θ is the angular displacement. After some manipulations, eq. (2.42) can be written in nondimensional form as:

$$\theta_{\bar{\tau}\bar{\tau}} + \sin(\theta) = 0, \theta(0) = 0, \theta_{\bar{\tau}}(0) = \theta_{\tau 0} \tag{2.44}$$

where $\bar{\tau} = \omega_0 t$ and $\omega_0 = \sqrt{g/L}$ and the subscript $\bar{\tau}$ denotes differentiation with respect to $\bar{\tau}$. Among the various special characteristics of the simple undamped pendulum problem, is that the degree of nonlinearity can be adjusted by setting appropriate values to the initial conditions, i.e. if the values of $\theta(0), \theta_{\bar{\tau}}(0)$ increase, nonlinearity increases as well. It is of interest to examine the total energy of this nonlinear system which, given that it is undamped, has to remain constant throughout the time history of its response:

$$E_{tot} = \frac{1}{2}(\theta_{\bar{\tau}})^2 + [1 - \cos(\theta)] = \frac{1}{2}[\theta_{\bar{\tau}}(0)]^2 + [1 - \cos(\theta(0))] \tag{2.45}$$

Assuming that the pendulum oscillates in the plane instead of rotating, the maximum angle θ_{max} can be calculated by setting $\theta_{\bar{\tau}} = 0$ in eq.(2.45):

$$\theta_{max} = \cos^{-1}\left\{\cos\left[\theta(0)\right] - \frac{1}{2}\left[\theta_{\bar{\tau}}(0)\right]^2\right\} \qquad (2.46)$$

It is apparent that $-\pi < \theta_{max} < \pi$ in order for the pendulum to oscillate. In this chapter, $\theta(0) = 0$ and the initial angular velocity is specified so that oscillation instead of rotation is ensured. Therefore, from eq. (2.46) it is obvious that:

$$\theta_{\bar{\tau}}(0)\Big|_{max} = 2 \qquad (2.47)$$

Two different cases are studied here. In the first case, $\theta_{\bar{\tau}}(0) = \sqrt{2}$ which corresponds to $\theta_{max} = \pi/2$ and in the second case , $\theta_{\bar{\tau}}(0) \sim 2$ which corresponds to $\theta_{max} \sim \pi$.

In Figure 2-20 the natural logarithm of the total energy of the simple undamped pendulum with large rotation is presented. The time step is $\Delta t = 0.15 \sec$, the duration of the motion is $20\sec$ and the initial velocity is $\theta_{\bar{\tau}}(0) = \sqrt{2}$ which corresponds to the first case examined. The spectral radius ρ_∞ parameter is selected to be equal to zero, or the lowest possible value for all integration algorithms used. Concerning the Newton-Raphson iterative procedure used, the maximum convergence tolerance and the maximum number of iterations are $tol_{max} = 0.01$ and $k_{max} = 200$ respectively. It is obvious that the most accurate algorithms are the Continuous Acceleration (CA) algorithms, in which the mean value of the total energy of the system is more flat than that of the elementary, Opt and DA algorithms. It seems that the lowest error occurs for the U1-V0-CA algorithm. The total energy of the system that is calculated analytically according to the relation (2.45) is shown with dashed lines and the limits in the various subplots are set to be the same for all algorithms to make visual comparisons of the error easier.

The plots that are shown in Figure 2-21 refer to the second case, in which the initial angular velocity is $\theta_{\bar{\tau}}(0) = 1.99999$. This value is intentionally selected to avoid rotation of the pendulum which happens for $\theta_{\bar{\tau}}(0) \geq 2$, meaning that 2 is a limiting value for $\theta_{\bar{\tau}}(0)$ which can be approached, but not reached. The values of all the remaining parameters are identical to those which were used for in the results presented in Figure 2-20. Regarding the energy error, in Figure 2-21 the trends observed are similar as in Figure 2-20. The main difference here is that there exist "oscillations" in the evolution of the total energy that are more pronounced than those that exist in Figure 2-20. One reason for this is that the degree of nonlinearity is larger in case 2, since the initial velocity is larger.
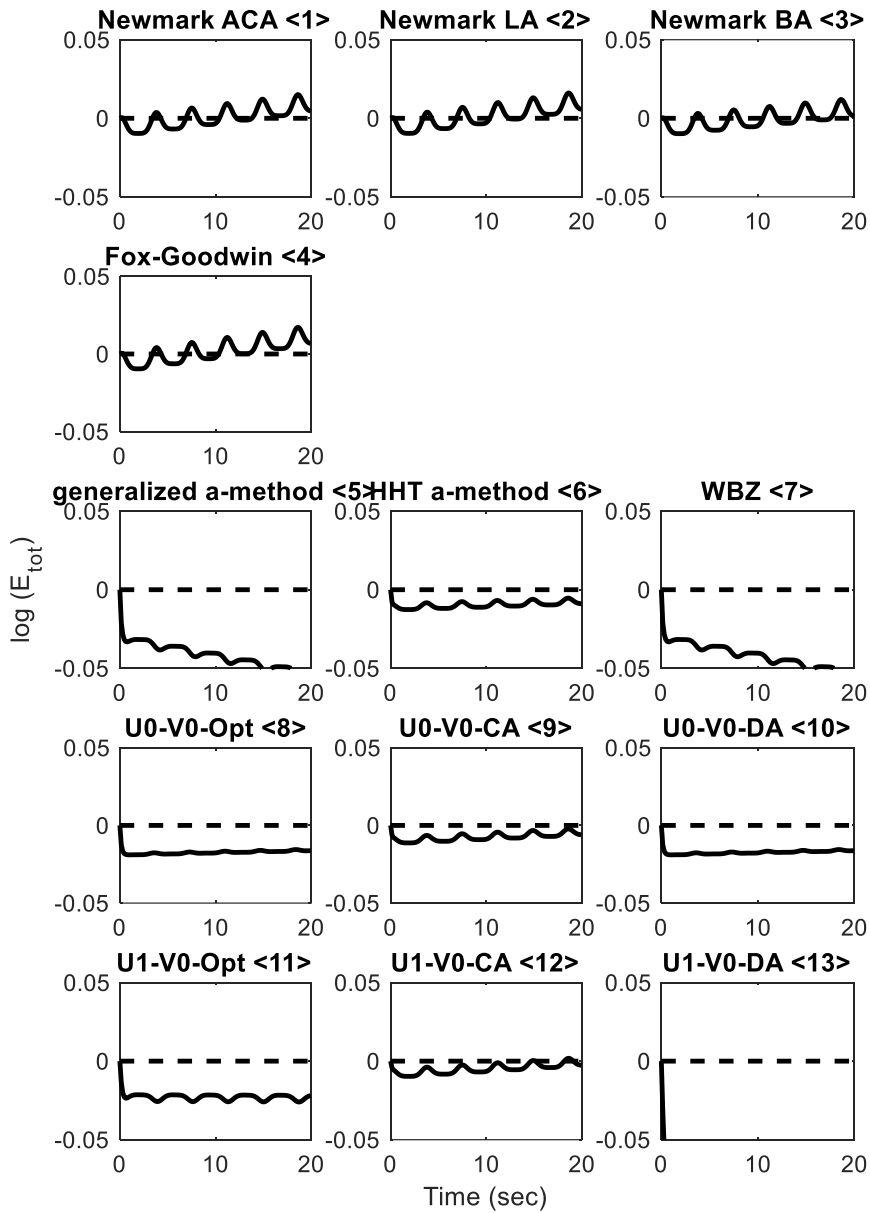
Figure 2-20: Natural logarithm of the total energy for simple undamped pendulum with initial angular velocity $\theta_{\bar{\tau}}(0) = \sqrt{2}$.
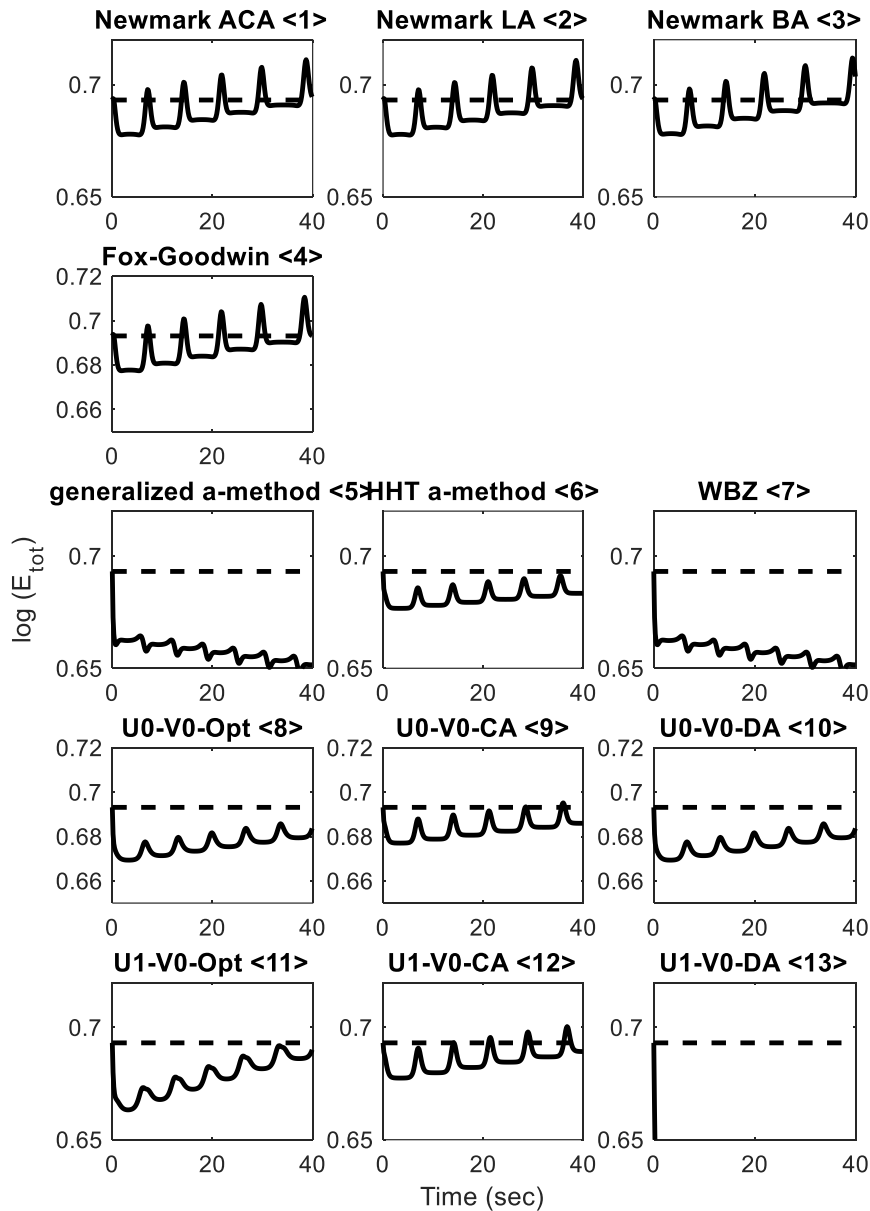
Figure 2-21: Natural logarithm of the total energy for simple undamped pendulum with initial angular velocity $\theta_{\bar{\tau}}(0) = 1.99999$.

2.6   Conclusions

The family of linear generalized single step single solve (GSSSS) algorithms, which includes the most commonly used time integration algorithms as special cases, can be extended to solve materially and geometrically nonlinear dynamic response via a Newton–Raphson iterative procedure. In the nonlinear regime, the extended nonlinear generalized algorithms are very efficient, with acceptable

accuracy and stability, even with increased size of the time step. They perform much better in some cases illustrated in this study, contrary to the elementary algorithms which in practice fail to trace the dynamic response.

It is pointed out that unconditionally stable algorithms for linear problems, may lose their stability in the presence of nonlinearities. For increased time step (and thus lower computational effort), the Continuous Acceleration methods, which include the HHT-a method (<6>) as a special case, exhibit the most accurate response for most of the cases studied. They appear to be the best option, regarding their general performance at the benchmark problems studied compared to the other integration schemes. For sufficiently small time step, all algorithms converge to the exact dynamic response. Further research has to be made to investigate the relation between the stable time increment of the generalized single step single solve algorithms applied in nonlinear problems and various other problem-dependent input data. Apart from this, the numerous integration constants of the algorithms belonging to the GSSSS family, allows for optimization of the values of the integration constants, so that certain difficult dynamic nonlinear problems can be efficiently time-integrated.

2.7   Notation

A : amplification matrix

a: order of the derivative of u

$\overline{C}$: damping matrix

c : damping coefficient of SDOF oscillator

E : error between numerical and exact solutions of dynamic response

E_{tot} : total energy

F : friction force

$\tilde{F}_n$: effective force matrix at time step n

f : external force vector

f_n : external force vector at time step n

g : acceleration of gravity

J_t : angular momentum

K_n : kinetic energy at time step n

$\overline{K}$: stiffness matrix

k : stiffness of SDOF oscillator

k_{max} : maximum number of iterations until convergence

k_i : stiffness at degree of freedom i

L : length of pendulum

L_t : linear momentum

M : mass matrix

$\tilde{M}$: effective mass matrix

m : mass of SDOF oscillator

m_i : lumped mass at degree of freedom i

p : internal force

q : order of accuracy of the integration scheme

S : parameter of SDOF oscillator with hardening spring

S_1 : parameter of SDOF oscillator with softening spring

S_2 : parameter of SDOF oscillator with softening spring

T : period

t : time

t_n : time at step n

tol_{max} : maximum tolerance for convergence of the GSSSS algorithm

U_n : strain energy at time step n

u : displacement

u_F : static deformation of SDOF oscillator due to friction force

u_i : displacement at degree of freedom i

$\dot{u}$: velocity

$\dot{u}_i$: velocity at degree of freedom i

$\ddot{u}$: acceleration

$\ddot{u}_i$: acceleration at degree of freedom i

u_n : displacement at time step n

$\dot{u}_n$: velocity at time step n

$\ddot{u}_n$: acceleration at time step n

$u^{\Delta t}$: approximation of u depending on the time step Δt

u_0 : initial displacement

$\dot{u}_0$: initial velocity

W_{ext} : work done by external forces

W : weighted time field

W_1 : time integration constant given by eq. (1.25) for i=1

w_i : integration constants (i=1...3)

Γ : parameter given by eq. (1.14)

Δt : step of direct time integration scheme

Δt_i : time step size at iteration i of the equation limit (1.7)

η : critical hysteretic damping ratio

θ : angular displacement

θ_{max} : maximum angular displacement

$\dot{\theta}$: angular velocity

$\ddot{\theta}$: angular acceleration

$\dot{\theta}_0$: initial angular velocity

λ_a : eigenvalue of the amplification matrix of order a

$\Lambda_1...\Lambda_6$: time integration constants

$\lambda_1...\lambda_5$: time integration constants

$\mu_1...\mu_6$: time integration constants

ξ : critical viscous damping ratio

ρ : spectral radius of the amplification matrix

ρ_∞ : minimum absolute value of the principal roots of the amplification matrix at the high-frequency limit

τ : parameter given by eq. (1.14)

$\bar{\tau}$: dimensionless time, given by $\bar{\tau} = \omega_0 t$

ω : cyclic frequency

ω_0 : cyclic eigenfrequency of simple undamped pendulum oscillator

ω_p : cyclic frequency of prescribed load

Chapter 3

■ ■ ■ ■ ■

3 OpenSeismoMatlab: A New Open-source Software for Strong Ground Motion Data Processing

3.1   Introduction

In this chapter a new open source software is introduced which can process strong ground motion data. OpenSeismoMatlab is an innovative open-source software for strong ground motion data processing, written in MATLAB. The software implements an elastoplastic bilinear kinematic hardening constitutive model and uses a state-of-the-art single step single solve time integration algorithm featuring exceptional speed, robustness and accuracy. OpenSeismoMatlab can calculate various time histories and corresponding peak values, Arias intensity and its time history, significant duration, various linear elastic response spectra and constant ductility inelastic response spectra, as well as the Fourier amplitude spectrum and the mean period. Due to its open-source nature, the software can be easily extended or modified, having high research and educational value for the professional engineering and research community. In the present chapter, the structure, algorithms and main routines of the program are explained in detail and the results for various types of spectra of 11 earthquake strong ground motions are calculated and compared to corresponding results from other proprietary software.

Earthquake resistant building codes require earthquake engineering studies which, in order to be performed, need strong motion records as original input data. It is therefore important to make realistic selections and processing of the raw input strong motion records in order to calculate the seismic parameters which will help in the estimation of the dynamic response of the structure(s) to be designed. Various software programs have been developed for the selection of the strong ground motions which are used for the dynamic analysis and design of structure(s) (Katsanos & Sextos, 2013; Macedo & Castro, 2017). Among the most

important seismic parameters of a strong ground motion are the various types of spectra (i.e. elastic response spectrum, constant ductility spectrum, constant-damage yield strength spectrum, Fourier spectrum, etc.) which result from the processing of the ground motion and which are used in various seismic design procedures, such as the Dynamic Response Spectrum Analysis (DRSA), the Uncoupled Modal Response History Analysis (UMRHA), the Modal Pushover Analysis (MPA) procedures for dynamic analysis (Chopra, 2012). Furthermore, by adjusting the Fourier spectrum of a strong ground motion, it is possible to control its frequency content. Therefore, the use of a robust and accurate strong motion processing software is critical for the proper seismic design of structures, including strategies for designing earthquake-resistant buildings to ensure the health, safety, and security of building occupants and assets during the structure's lifetime.

The concept of the elastic response spectrum was introduced by G.W. Housner (1959), whereas Newmark & Hall (1969) presented a fundamental work on linear elastic response spectra. Since then a large research effort has been made for the evaluation of the seismic response of linear SDOF systems with particular attention to the effect of input motion and site conditions. Most studies on inelastic response spectra have focused on the selection of the elastic-perfectly plastic material behavior, on taking into account the effects of the duration of the motion and on scaling methods (Newmark et al., 1973; Veletsos & Newmark, 1960; Veletsos et al., 1965; Ziang et al., 2016). In addition, Veletsos & Vann (1971) published among the first studies that systematically investigated the elastic and inelastic structural response to pulse-like excitations (typically not caused by earthquakes).

Many software programs, either free or commercial, have been developed for the processing of strong ground motion data. Some characteristic cases are presented below:

- SMA (Strong Motion Analyst Processing Software) is a commercial Windows-based tool designed to interactively process strong motion accelerograms, featuring instrument correction, data editing, filtering, ground motion integrations, Fourier and Response Spectra calculations, and V1, V2, V3 file format output. It has been developed by the Kinemetrics company.

- EQ-TOOLS (latest version is 3.0) is a free closed source software for earthquake engineering education which allows the user to select, analyze, scale, and modify ground motions. The capabilities of selection and analysis as well as scaling of ground motion records against several types of target spectra, including the ASCE 7 spectrum, spectra from attenuation relationships, and conditional mean spectra are included. Ground motion history analysis, linear response spectrum analysis and Fourier amplitude analysis and a module that enables the modification of ground motions for consideration of site effects are provided. It has been developed by the

George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES).

- PRISM (Processing and Review Interface for Strong Motion data) is a free open-source software used for processing strong-motion records (Jones et al., 2017a; Jones et al., 2017b; Jones et al., 2017c; Kalkan & Stephens, 2017; Jones et al., 2018). It can be installed and run as a stand-alone system on common operating systems such as Linux, Mac and Windows and is flexible and extensible to incorporate new strong motion processing techniques.

- PRISM for Earthquake Engineering (Jeong et al., 2016) includes capabilities for modification, correction, scaling, truncation and baseline correction of earthquake records and it can calculate a variety of strong motion parameters (Arias intensity, elastic and inelastic response spectra, acceleration, velocity, displacement and force-displacement response histories). Various hysteresis models are provided (linear elastic, bi-linear, tri-linear, modified Takeda, Bouc-Wen, and Al-Bermani)

- SEISMOSIGNAL is an interesting, user-friendly and efficient commercial software for processing of strong motion data (Antoniou et al., 2012). Among others, it can calculate the elastic, constant ductility, Fourier amplitude and power spectra and it provides for filtering of high and low frequency record content and estimation of other important seismological parameters, such as the Arias Intensity and the significant and effective durations.

- OPENSIGNAL is a free closed-source software platform for the processing and selection of seismic records, signal processing, response spectra analysis, soil spectra analysis and more (Cimellaro & Marasco, 2014; Cimellaro & Marasco, 2015). It provides filtering uncorrected ground motion records and calculation of the main parameters of a record (Arias Intensity, duration, PGA, PGV, elastic response spectra, etc.).

- USDP (Utility Software for Data Processing) is a computer program that can be used for strong ground-motion data processing by various filtering and baseline adjustment techniques and spectral calculations (Linear spectral analysis, Fourier spectral analysis, Constant strength, ductility and base-shear coefficient nonlinear spectral analysis) for a variety of stiffness and/or strength degrading hysteretic models (Akkar, 2008). It has been developed by the METU-Earthquake Engineering Research Center team and uses public-open Fortran source codes.

- TSPP (Time Series Processing Programs) is a collection of FORTRAN programs that have been developed for processing and manipulating strong-motion accelerograms in terms of displacement, velocity and acceleration time-histories, response and Fourier spectra and filtering (Boore, 2001).

- VIEWWAVE (v2.2.0) is a free closed source software for processing and viewing strong motion records. It can read a large variety of files in many formats and can calculate various waveforms, Fourier and power spectrum, as well as acceleration, velocity and displacement response spectra (Kashima, 2016).

Apart from the above software, some other rather elementary programs have been developed in MATLAB programming language (Kalkan, 2016; Kalkan, 2017a; Kalkan, 2017b; Tazarn, 2011; Carlton, 2015; all submitted to the File Exchange service of the Mathworks official website). However, none of these MATLAB implementations contains advanced time integration algorithms for the extraction of the displacement, velocity and acceleration time histories and the various response spectra. In the present chapter, a new MATLAB open-source software, called OpenSeismoMatlab, is presented which, compared to other similar software, has the following advantages and unique characteristics:

- It uses state-of-the-art time integration algorithms which are more robust and accurate (Papazafeiropoulos et al., 2017a; Papazafeiropoulos et al., 2017b) compared to conventional integration techniques (Newmark, etc.) that are widely used by other software for strong motion data processing. The former algorithms belong to a general single step single solve family and can be adjusted through the specification of 14 independent integration constants to control numerical dissipation and dispersion, continuity of acceleration, and the order of overshooting in displacement and velocity. By adjusting a number of parameters, the user can select from a large family of time integration algorithms, according to Papazafeiropoulos et al. (2017a) and this permits the manual configuration and optimization of the quality of the desired results (time histories, spectra, etc.).

- It is completely free and provided together with its source code (open source). OpenSeismoMatlab is of high educational value, since it contains simply written MATLAB code with comments and is generally easy to be understood by the user. The rationale of the implemented methods is explained in detailed comments within the code. Apart from this, the open-source code format provides the opportunity of extending/upgrading or integrating the software in all possible ways.

- Furthermore, the elastoplastic bilinear kinematic hardening constitutive model which is fundamental for the computation of the nonlinear spectra is accurately formulated and programmed in the software. No simplified versions of the elastoplastic bilinear kinematic hardening constitutive model (Newmark & Hall, 1982; Krawinkler & Nassar, 1992; Miranda & Bertero, 1994) are used, as is usually the case in the literature.

In the following sections, OpenSeismoMatlab is presented, and then it is applied in a number of earthquake records for the calculation of the various response spectra and other quantities.

3.2   Structure and source code of OpenSeismoMatlab

In this section the design of the various algorithms used by OpenSeismoMatlab for the computation of the strong motion data processing results (spectra, time histories, etc.) are given. For each algorithm the structure of the main MATLAB function and its subroutines (if present) are provided and the architecture of the various MATLAB codes within the OpenSeismoMatlab software is presented. Finally, the capabilities and restrictions of OpenSeismoMatlab are discussed. OpenSeismoMatlab can calculate the following strong motion data processing output:

- Time history of velocity
- Time history of displacement
- Peak ground acceleration (PGA)
- Peak ground velocity (PGV)
- Peak ground displacement (PGD)
- Time history of normalized Arias intensity
- Total Arias intensity
- Time interval between 5% and 95% of Arias intensity has occurred (significant duration D_{5-95})
- Linear elastic pseudo-acceleration response spectrum
- Linear elastic pseudo-velocity response spectrum
- Linear elastic displacement response spectrum
- Linear elastic velocity response spectrum
- Linear elastic acceleration response spectrum
- Constant ductility inelastic displacement response spectrum
- Constant ductility inelastic velocity response spectrum
- Constant ductility inelastic acceleration response spectrum
- Fourier amplitude spectrum
- Mean period (T_m)

The source code of OpenSeismoMatlab has been uploaded on two different distribution channels: (i) the File Exchange service of MATLAB central (Papazafeiropoulos, 2018) and (ii) on ResearchGate (Papazafeiropoulos & Plevris, 2018), so that it is publicly available. The source code is organized in folders as follows:

- A folder named "data" contains the acceleration time histories of the earthquakes considered in this chapter, in digitized format
- A folder named "examples" contains three MATLAB scripts which illustrate how the software can be used properly for the generation of Elastic Response Spectra, Fourier Spectra and Constant Ductility Response Spectra
- A folder named "figures" contains the figures that are generated after the execution of the MATLAB scripts contained in the "examples" folder

• A folder named "lib" contains all the subroutine functions of the OpenSeismoMatlab software. These functions are substantial for the application of OpenSeismoMatlab.

In Figure 3-1 the dependency diagram between the various functions included in the OpenSeismoMatlab package is shown. The four main functions are LEReSp for the linear elastic response spectra, CDReSp for the constant ductility response spectra, FASp for the Fourier spectra and baselineCorr for the baseline correction of the input ground motion. The functions DRHA, NLIDABLKIN and HalfStep are called directly by the function CDReSp and are used for Dynamic Response History Analysis, NonLInear Dynamic Analysis with BiLinear KINematic hardening model, and reproduction of an earthquake excitation with the half time step, respectively. The function LIDA is used for Linear Incremental Dynamic Analysis and is called by the functions LEReSp and DRHA, whereas the function BLKIN is called by the function NLIDABLKIN.
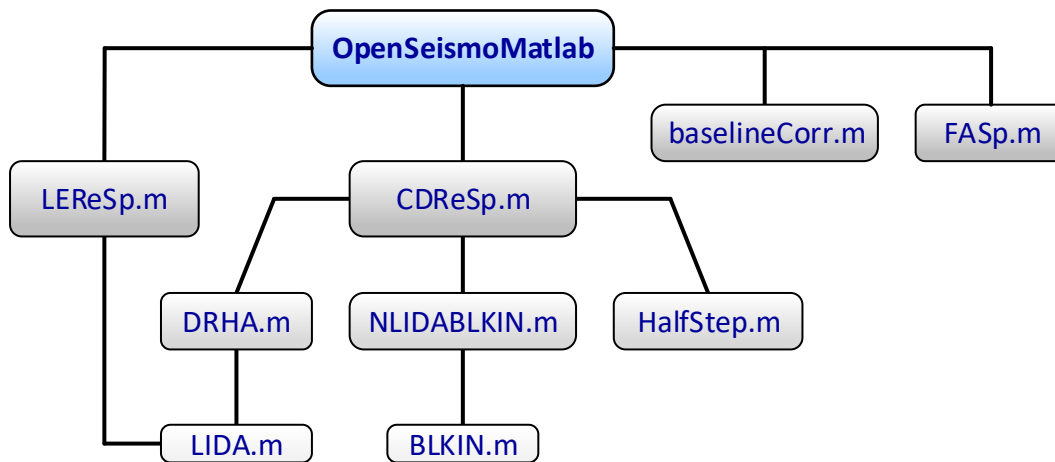
Figure 3-1: Schematic dependency diagram between the various functions included in the OpenSeismoMatlab package.

The beginning section of the main MATLAB function (OpenSeismoMatlab.m) is shown in Listing 1 for purposes of completeness, and to show how the various variables that appear in parts of the main function code presented in the subsequent sections are defined. The two necessary input arguments are the time column vector (denoted as time) and the ground acceleration time history column vector (denoted as xgtt). Apart from these, shows the various default values which are set in the required variable definitions, in case they are not specified by the user. These are the critical damping ratio ξ (denoted in the code as ksi), the lower period limit T1, the upper period limit T2 and the period step dT of the generated response spectra (denoted in the code as T1, T2 and dT respectively), and finally the target ductility μt (denoted in the code as mu).

```matlab
function seismic=OpenSeismoMatlab(time,xgtt,varargin)
%% Initial checks
if nargin<2
    error('Input arguments less than required')
end
if nargin>7
    error('Input arguments more than required')
end
% set defaults for optional inputs
optargs = {0.05,0.04,10,0.05,2};
% skip any new inputs if they are empty
newVals = cellfun(@(x) ~isempty(x), varargin);
% overwrite the default values by those specified in varargin
optargs(newVals) = varargin(newVals);
% place optional args in memorable variable names
[ksi,T1,T2,dT,mu] = optargs{:};
time = time(:);
xgtt = xgtt(:);
dt = time(2)-time(1);
```

Listing 3-1: Source code for the determination of various variables before the main calculations.

3.2.1 Displacement and velocity time histories and peak values

This part of the OpenSeismoMatlab code is quite straightforward. The MATLAB code shown in Listing 3-2 is executed in order to determine the displacement time history of the input motion, the velocity time history, the peak displacement, the peak velocity and the peak acceleration. The time integration is simply performed by the summation of the product of the integrand function by the time step Δt. The velocity and displacement time histories are given by eq. (3.1) and (3.2) respectively:

$$\dot{u}_g^k = \left(\sum_{i=1}^{k} \ddot{u}_g^i \right) \Delta t \tag{3.1}$$

$$u_g^k = \left(\sum_{i=1}^{k} \dot{u}_g^i \right) \Delta t \tag{3.2}$$

where k denotes the kth time step of the earthquake motion. Since the size of the time step is constant throughout the various earthquake motions, eq. (3.1) and (3.2) make use of the cumulative sums (cumsum) of the earthquake acceleration and the earthquake velocity respectively. The peak values are given by the equations (3.3) to (3.5) respectively:

$$PGA = \max\left(\left| \ddot{u}_g \right| \right) \tag{3.3}$$

$$PGV = \max\left(\left| \dot{u}_g \right| \right) \tag{3.4}$$

$$PGD = \max\left(\left|u_g\right|\right) \qquad\qquad (3.5)$$

The user has the option to perform baseline correction to the input acceleration data, if desired, as shown in Listing 3-2. This capability is activated if the boolean variable baselineSw is set equal to true. If so, the function baselineCorr.m is used for this purpose. The source code of baselineCorr.m is shown in Listing 3-3.

```matlab
% TIME SERIES
if baselineSw
    [cor_xg,cor_xgt,cor_xgtt] = baselineCorr(time,xgtt);
    seismic.acc=cor_xgtt;
    seismic.vel=cor_xgt;
    seismic.disp=cor_xg;
else
    % Acceleration time history
    seismic.acc = xgtt;
    % Velocity time history
    seismic.vel = cumtrapz(time,xgtt)*dt;
    % Displacement time history
    seismic.disp = cumtrapz(time,seismic.vel)*dt;
end
% PEAK RESPONSES
% Peak ground acceleration
seismic.PGA = max(abs(xgtt));
% Peak ground velocity
seismic.PGV = max(abs(seismic.vel));
% Peak ground displacement
seismic.PGD = max(abs(seismic.disp));
```

Listing 3-2: Source code for the calculation of the velocity and displacement time histories as well as the peak values.

The function baselineCorr.m performs linear baseline correction for an uncorrected acceleration time history. Initially, first order fitting (straight line) is performed and the fitting line is subtracted from the acceleration time history, giving thus the first correction. Afterwards, this first correction of the acceleration is integrated to obtain the velocity, and then first order fitting (straight line) is reapplied on this velocity time history. The gradient of the straight fitting line is then subtracted from the first correction of the acceleration time history, giving thus the second correction of the acceleration time history. The second correction of the acceleration time history is then simply and doubly integrated to give the corrected velocity and displacement time histories, respectively.

```matlab
function [cor_xg, cor_xgt, cor_xgtt] = baselineCorr(time,xgtt)
dt = time(2)-time(1);
% Least squares fit through acceleration history
p=polyfit(time,xgtt,1);
lsf_cor_xgtt = polyval(p,time);
cor_xgtt1 = xgtt - lsf_cor_xgtt ;
% Integrate for velocity
un_xgt = cumtrapz(time,cor_xgtt1)*dt;
% Least squares fit through velocity history
ca2 = polyfit(time,un_xgt,1);
cor_xgtt = cor_xgtt1 - ca2(1);
% Corrected velocity
cor_xgt = cumtrapz(time,cor_xgtt)*dt;
% Corrected displacement
cor_xg = cumtrapz(time,cor_xgt)*dt;
```

Listing 3-3: Source code for the function baselineCorr.m.

3.2.2 Arias Intensity

The Arias Intensity was proposed as an intensity measure of an earthquake by Arias (1970) and Housner & Jennings (1977), since it was recognized that the peak values alone cannot adequately portray the intensity of a ground motion. It is broadly defined as the sum of the energies per unit mass, dissipated due to the ground motion, by a population of Single Degree of Freedom (SDOF) systems with all natural frequencies. For undamped linear elastic SDOF systems, it can be shown (Arias, 1970; Housner & Jennings, 1977) that the Arias Intensity (AI) is given by:

$$\mathrm{AI} = \frac{\pi}{2}\int_{0}^{t_f} \ddot{u}_g^2(t)\,dt \tag{3.6}$$

where tf is the total duration of the earthquake. For a digitized strong motion data, Arias intensity is given by:

$$\mathrm{AI} = \frac{\pi}{2}\left(\sum_{i=1}^{i_{max}}\left(\ddot{u}_g^i\right)^2\right)\Delta t \tag{3.7}$$

where imax denotes the total number of time increments of the earthquake motion. OpenSeismoMatlab can also output the time history of the normalized Arias Intensity, which expresses how the current $\overline{\mathrm{AI}}_k$ (up to the current time step k, normalized with the total AI) evolves with time during the earthquake motion, as given by eq.(3.8):

$$\overline{\mathrm{AI}}_k = \frac{\mathrm{AI}_k}{\mathrm{AI}} = \frac{\frac{\pi}{2}\left(\sum_{i=1}^{k}\left(\ddot{u}_g^i\right)^2\right)\Delta t}{\mathrm{AI}} \tag{3.8}$$

where AIk is the value of the Arias Intensity at the kth time step of the earthquake motion. The MATLAB code that is used for the above calculations is shown in Listing 3-4:

```matlab
% ARIAS INTENSITY
% time history of Arias Intensity
aint2 = cumsum(xgtt.^2)*pi*dt/2;
% Total Arias Intensity at the end of the ground motion
arias = aint2(end);
seismic.arias = arias;
% time history of the normalized Arias Intensity
seismic.aint2 = aint2/arias;
```

Listing 3-4: Source code for the calculation of the Arias Intensity (total and time history of the normalized AI).

3.2.3   Significant duration

The definition of significant duration is given by Dobry et al. (1978) and Trifunac & Brady (1975). In this definition, the significant duration is defined as the time interval between the time at which 5% of the seismic energy is attained and the time at which 95% of the seismic energy is attained. It is denoted as D5_95. The computational implementation is given in the MATLAB code shown in Listing 3-5. The code outputs both the significant duration (denoted in the code as D_5_95) and the time instants at which 5% and 95% of AI are attained (arranged in a row vector denoted in the code as t_5_95).

```matlab
% SIGNIFICANT DURATION
% elements of the time vector which are within the significant duration
timed = time(aint2>=0.05*arias & aint2<=0.95*arias);
% starting and ending points of the significant duration
seismic.t_5_95 = [timed(1),timed(end)];
% significant duration
seismic.D_5_95 = timed(end)-timed(1);
```

Listing 3-5: Source code for the calculation of the significant duration of an earthquake.

3.2.4   Elastic Response Spectrum

The Linear Elastic Response Spectrum (LEReSp) for a response quantity (acceleration, velocity, displacement, etc.) is a plot of the peak value of the quantity as a function of the natural vibration period (T_n) or frequency (f_n) of a population of linear elastic SDOF systems. Each linear elastic response spectrum is associated with a fixed damping ratio ξ. A flowchart of the calculation of the

linear elastic response spectrum of an earthquake strong ground motion is shown in Listing 3-7.

It is noted that all time integration algorithms require the use of relatively small time-steps in order to deliver sufficiently accurate solutions (Soroushian, 2008; Soroushian, 2017). For this purpose, a maximum value of the ratio between the integration time-step and the period of the oscillator being analyzed is imposed, as shown in Listing 3-7 ($\max\left\{\omega_i\Delta t/(2\pi)\right\}=0.02$). Initially, the program uses the time step of the input acceleration time history as the time step of the dynamic analysis, and then if this is found to violate the aforementioned maximum, the algorithm automatically reproduces the acceleration time history with half the current time step through linear interpolation, so that the threshold value is respected. The maximum limit that is specified above (0.02) leads to sufficiently accurate solutions. However, in OpenSeismoMatlab it can be changed manually by the user, if required, in order to handle special cases. The MATLAB function that reproduces the acceleration time history with half the time step is called HalfStep and its code is shown in Listing 3-6.

```
function uNew = HalfStep(u)
a=[([0;u(1:end-1)]+u)/2,u]';
uNew=a(:);
uNew(1)=[];
end
```

Listing 3-6: MATLAB code for the function HalfStep.m.

The main OpenSeismoMatlab function for the calculation of the linear elastic response spectrum is called "LEReSp" and its source code is shown in Listing 3-8 and Listing 3-9 (as a continuation of Listing 3-8).

It can be seen that the initial conditions for all SDOF systems that are analyzed for the generation of the LEReSp are zero for both velocity and displacement. Furthermore, the time integration algorithm that is used for the dynamic response history analysis of each SDOF system has zero order overshooting behavior for both displacement and velocity, and since the minimum absolute value of the eigenvalues of the amplification matrix (r_{inf}) is equal to unity, this corresponds to the mid-point rule a-form algorithm.

Input: $\ddot{u}_g$, ω, ξ

Initialize SD, SV SA

Set u_0 and $\dot{u}_0$

for each SDOF i with eigenfrequency ω_i

   if $\omega_i \Delta t / (2\pi) > 0.02$

      Reproduce $\ddot{u}_g$ with half time step (from Δt to $\Delta t/2$)

      Set $\Delta t = \Delta t/2$

   end

   Perform dynamic analysis of SDOF with input ($\ddot{u}_g$, ξ, u_0, $\dot{u}_0$)

   Assign $\max(\|u(t)\|)$ to $SD(i)$

   Assign $\max(\|\dot{u}(t)\|)$ to $SV(i)$

   Assign $\max(\|\ddot{u}(t)\|)$ to $SA(i)$

end

Calculate $PSV = \omega SD$ and $PSA = \omega^2 SD$

Output: SD, SV, SA, PSV, PSA

Listing 3-7: Flowchart of the calculation of the linear elastic response spectrum implemented in OpenSeismoMatlab.

It has been proven that, for structural dynamics problems, the so-called linear multi-step methods (LMS) are spectrally identical to a newly developed family of generalized single step single solve (GSSSS) algorithms (Zhou & Tamma, 2004). Three subclasses of computational algorithms can be distinguished based on the overshooting behavior, and additional algorithmic properties such as second-order accuracy, and unconditional stability with numerical dissipative features: (i) zero-order displacement and velocity overshoot algorithms (U0-V0); (ii) zero-order displacement and first-order velocity overshoot algorithms (U0-V1); and (iii) first-order displacement and zero-order velocity overshoot algorithms (U1-V0).

```matlab
function [PSa,PSv,Sd,Sv,Sa]=LEReSp(dt,xgtt,T,varargin)
% set defaults for optional inputs
optargs = {0.05,0.01,'U0-V0-CA',0};
% skip any new inputs if they are empty
newVals = cellfun(@(x) ~isempty(x), varargin);
% overwrite the default values by those specified in varargin
optargs(newVals) = varargin(newVals);
% place optional args in memorable variable names
[ksi,dtTol,AlgID,rinf] = optargs{:};
% initialize
NumSDOF=length(T);
Sd=zeros(NumSDOF,1);
Sv=zeros(NumSDOF,1);
Sa=zeros(NumSDOF,1);
% Set the eigenfrequencies of the SDOF population
omega=2*pi./T;
% Flip eigenfrequency vector in order for the half-stepping algorithm
% (HalfStep function) to work from large to small eigenperiods
omega=omega(end:-1:1);
% set initial conditions
u0=0;
ut0=0;
% zero-order displacement & velocity overshooting behavior and
% optimal numerical dissipation and dispersion
rinf=1; % mid-point rule a-form algorithm
for j=1:length(T)
    omegaj=omega(j);
    % Check if dt/T>dtTol. If yes, then reproduce the time history
    % with the half step
    if dt*omegaj/(2*pi)>dtTol
        xgtt=HalfStep(xgtt);
        dt=dt/2;
    end
    [u,ut,utt] = LIDA(dt,xgtt,omegaj,ksi,u0,ut0,AlgID,rinf);
    % output
    Sd(j)=max(abs(u));
    Sv(j)=max(abs(ut));
    Sa(j)=max(abs(utt));
end
```

Listing 3-8: Source code for the function LEReSp.

```matlab
% Flip output quantities to be compatible with omega
omega=omega(end:-1:1);
Sd=Sd(end:-1:1);
Sv=Sv(end:-1:1);
Sa=Sa(end:-1:1);
% Calculate pseudovelocity and pseudoacceleration
PSv=Sd.*omega;
PSa=Sd.*omega.^2;
end
```

Listing 3-9: Source code for the function LEReSp (continued from Listing 3-8).

The formulation of the general case of a GSSSS algorithm involves the determination of 12 sets of parameters, among which five are related to the accuracy, five are related to the overshoot, dissipation, and dispersion; one parameter set is related to the stability; and one parameter set is related to the second-order approximation for the integration of the load term. Therefore, the user has the freedom to configure the general algorithm by adjusting the time integration constants so that it yields acceptable results for any specific case of application. This capability is also incorporated in OpenSeismoMatlab, since this general time integration framework is an integral part of it. It has been shown that many known time integration algorithms (e.g. the members of the Newmark family) are special cases of this general algorithm framework. A more complete presentation and investigation of the entire time integration algorithm family used in OpenSeismoMatlab has been done by Papazafeiropoulos et al. (2017a).

The function LIDA (Linear Implicit Dynamic Analysis) is utilized for the dynamic analysis of each SDOF system. The internal code of the function LIDA is presented in Listing 3-10. Initially, the time integration constants are calculated so that the time integration scheme corresponds to a zero-order displacement & velocity overshooting behavior and optimal numerical dissipation and dispersion algorithm (see Listing 3-10). The desired properties of this algorithm are second-order accuracy, no overshoot, dissipative with optimal dissipation and dispersion, and unconditional stability. The minimum absolute value of the eigenvalues of the amplification matrix at the high-frequency limit is imposed to be equal to unity to minimize the effect of the spurious root at the low-frequency limit.

The reader is referred to Zhou & Tamma (2004) for more details. The default time integration algorithm of OpenSeismoMatlab (mid-point rule a-form algorithm) can be used for small-scale (non-stiff) undamped problems. Of course, in special cases, e.g. for large scale (stiff) problems with initial displacement or initial velocity, the U1-V0 or the U0-V1 algorithms can be used, respectively. In such cases, the various integration constants can be easily adjusted by the user of OpenSeismoMatlab by appropriate modification of the code shown in Listing 3-10, so that solutions of superior quality can be obtained.

```matlab
function [u,ut,utt] = LIDA(dt,xgtt,omega,varargin)
% set defaults for optional inputs
optargs = {0.05,0,0,1};
% skip any new inputs if they are empty
newVals = cellfun(@(x) ~isempty(x), varargin);
% overwrite the default values by those specified in varargin
optargs(newVals) = varargin(newVals);
% place optional args in memorable variable names
[ksi,u0,ut0,rinf] = optargs{:};
% Integration constants
% zero-order displacement & velocity overshooting behavior and
% optimal numerical dissipation and dispersion
w1=-15*(1-2*rinf)/(1-4*rinf); % suggested
w2=15*(3-4*rinf)/(1-4*rinf); % suggested
w3=-35*(1-rinf)/(1-4*rinf); % suggested
W1=(1/2+w1/3+w2/4+w3/5)/(1+w1/2+w2/3+w3/4); % definition
W1L1=1/(1+rinf);
W2L2=1/2/(1+rinf);
W3L3=1/2/(1+rinf)^2;
W1L4=1/(1+rinf);
W2L5=1/(1+rinf)^2; % suggested
W1L6=(3-rinf)/2/(1+rinf);
l1=1;
l2=1/2;
l3=1/2/(1+rinf);
l4=1;
l5=1/(1+rinf);
```

Listing 3-10: Source code for the first part of the function LIDA.m used for the calculation of the time integration constants.

The dynamic response history analysis is performed through the use of the fast MATLAB function filter. This function proves to be much faster (up to 100x) than the ordinary time integration routines, and filters the data in any input vector (i.e. the time history of acceleration) with a rational transfer function described by two additional input vectors (denominator and nominator) to create the filtered output data (i.e. the time history of the response). The transfer function of the filter function is of the form:

$$Y(z) = \frac{TF_n(1) + TF_n(2)z^{-1} + TF_n(3)z^{-2} + TF_n(4)z^{-3}}{1 + TF_d(1)z^{-1} + TF_d(2)z^{-2} + TF_d(3)z^{-3}} X(z) \tag{3.9}$$

where $X(z)$ is the input signal (i.e. the time history of acceleration), $Y(z)$ is the output signal (i.e. the time history of the SDOF dynamic response), TF_n is a row vector containing the coefficients of the nominator of the transfer function and TF_d is a row vector containing the coefficients of the denominator of the transfer function, as presented in the following.

The calculation of the transfer function denominator and nominator is shown in Listing 3-11. The elements of the amplification matrix are calculated first, and then the invariants of the amplification matrix are found as follows:

$$A = \begin{bmatrix} 1-\dfrac{\lambda_3\Omega^2}{D} & \lambda_1 - \dfrac{\lambda_3(2\xi\Omega+\mu_1\Omega^2)}{D} & \lambda_2 - \lambda_3\dfrac{1+2\mu_4\xi\Omega+\mu_2\Omega^2}{D} \\[4mm] -\dfrac{\lambda_5\Omega^2}{D} & 1-\dfrac{\lambda_5(2\xi\Omega+\mu_1\Omega^2)}{D} & \lambda_4 - \lambda_5\dfrac{1+2\mu_4\xi\Omega+\mu_2\Omega^2}{D} \\[4mm] -\dfrac{\Omega^2}{D} & -\dfrac{2\xi\Omega+\mu_1\Omega^2}{D} & 1-\dfrac{1+2\mu_4\xi\Omega+\mu_2\Omega^2}{D} \end{bmatrix} \qquad (3.10)$$

$$I_A = tr(A) = \sum_{i=1}^{3} A_{ii} \qquad (3.11)$$

$$II_A = \frac{1}{2}\left[tr(A)^2 - tr(A^2) \right] \qquad (3.12)$$

$$III_A = det(A) \qquad (3.13)$$

where $\Omega = \omega\Delta t$ and $D = \mu_6 + 2\mu_5\xi\Omega + \mu_3\Omega^2$, while λ_i and μ_i (i = 1,...,5) are constants of the time integration algorithm. The reader is referred to [27] for the detailed definitions of these integration constants corresponding to the various time integration algorithms. The denominator of the transfer function is given by:

$$TF_d = \left[1-I_A, II_A, -III_A \right] \qquad (3.14)$$

The nominator of the transfer function is calculated as follows:

$$TF_n = \left[B_1, B_2, B_3, B_4 \right] \qquad (3.15)$$

where:

$$B_1 = \frac{\Delta t^2 \lambda_3 W_1}{D}$$

$$B_2 = \frac{\Delta t^2 \left[\lambda_3(1-W_1) - (A_{22}+A_{33})\lambda_3 W_1 + A_{12}\lambda_5 W_1 + A_{13}W_1 \right]}{D}$$

$$B_3 = \frac{\Delta t^2}{D} \left\{ \begin{array}{l} -(A_{22}+A_{33})\lambda_3(1-W_1) + A_{12}\lambda_5(1-W_1) + A_{13}(1-W_1) + (A_{22}A_{33}-A_{23}A_{32})\lambda_3 W_1 - \\ (A_{12}A_{33}-A_{13}A_{32})\lambda_5 W_1 + (A_{12}A_{23}-A_{13}A_{22})W_1 \end{array} \right\}$$

$$B_4 = \frac{\Delta t^2}{D} \left\{ \begin{array}{l} (A_{22}A_{33}-A_{23}A_{32})\lambda_3(1-W_1) - (A_{12}A_{33}-A_{13}A_{32})\lambda_5(1-W_1) + \\ (A_{12}A_{23}-A_{13}A_{22})(1-W_1) \end{array} \right\}$$

$$(3.16)$$

The denominator vector is constructed by the amplification matrix invariants. The nominator vector is constructed as a function of the elements of the amplification matrix and various integration constants. Finally, the displacement of the SDOF system is found at the first and second time instants by using the initial conditions and the amplification matrix. The time history of the displacement is found using the Matlab function filter. Then the velocity is calculated from the system of equations (3.17) and(3.18), in which the only unknowns are the time histories of the velocity and acceleration ($\dot{u}^t$ and $\ddot{u}^t$, respectively):

$$u^{t+\Delta t} = \begin{bmatrix} 1-\dfrac{\lambda_3\Omega^2}{D} \\[1.2em] \lambda_1-\dfrac{\lambda_3(2\xi\Omega+\mu_1\Omega^2)}{D} \\[1.2em] \lambda_2-\lambda_3\dfrac{1+2\mu_4\xi\Omega+\mu_2\Omega^2}{D} \end{bmatrix}^T \begin{bmatrix} u^t \\[0.6em] \Delta t\dot{u}^t \\[0.6em] \Delta t\ddot{u}^t \end{bmatrix} + \dfrac{\lambda_3\Delta t^2}{D}\left[(1-W_1)\left(-\ddot{u}_g^t\right)+W_1\left(-\ddot{u}_g^{t+\Delta t}\right)\right] \quad (3.17)$$

$$m\ddot{u}^t + c\dot{u}^t + ku^t = -m\ddot{u}_g^t \qquad (3.18)$$

and the acceleration time history, with the time histories of the displacement and velocity known, is derived merely from eq. (3.18). See Listing 3-12 for more details about the source code segment in which this computation is performed.

3.2.5 Constant Ductility inelastic Response Spectrum

The Constant Ductility Response Spectrum (CDReSp) is the nonlinear counterpart of the linear elastic response spectrum that is described in the previous section. It is a plot of the peak value of any response quantity as a function of the small strain natural vibration period (T_n) or frequency (f_n) of a population of inelastic (bilinear elastoplastic) SDOF systems. Each CDReSp curve is associated with a fixed critical damping ratio ξ and target ductility μ_t. A flowchart of the calculation of the CDReSp of an earthquake strong ground motion is shown in Listing 3-13. A number of iterations are performed for each SDOF system (i.e. for each eigenfrequency) of the CDReSp, as the way to determine the yield limit of a SDOF structure based on its dynamic response (ductility) is not straightforward. During the iterations the yield limit is continuously adjusted so that the ductility that is calculated is as close as possible to the target ductility. This fact renders the calculation of the CDReSp more computationally expensive than the calculation of the simple LEReSp. Given its large extent, the related MATLAB code for this function as well as the children (called) functions are not presented here but can be easily found online in the OpenSeismoMatlab package of source files at the File Exchange service of MATLAB Central, or at other repositories. Inside the function CDReSp, a series of dynamic analyses of the linear elastic and the bilinear elastoplastic SDOF system are performed, as can be seen in Listing 3-13. For these dynamic analyses, an

appropriate external function is called. The flowchart of this function can be found in Papazafeiropoulos et al. (2017a). It is noted that the constant-ductility inelastic response spectrum is computed through nonlinear dynamic analyses of elastoplastic hysteretic systems, rather than through the simplified approaches that have been proposed in the literature (Newmark & Hall, 1982; Krawinkler & Nassar, 1992; Miranda & Bertero, 1994).

```matlab
% Transfer function denominator
Omega=omega*dt;
D=W1L6+2.*W2L5.*ksi.*Omega+W3L3.*Omega.^2;
A31=-Omega.^2./D;
A32=-1./D.*(2.*ksi.*Omega+W1L1.*Omega.^2);
A33=1-1./D.*(1+2.*W1L4.*ksi.*Omega+W2L2.*Omega.^2);
A11=1+l3.*A31;
A12=l1+l3.*A32;
A13=l2-l3.*(1-A33);
A21=l5.*A31;
A22=1+l5.*A32;
A23=l4-l5.*(1-A33);
% Amplification matrix
A=[A11 A12 A13;A21 A22 A23;A31 A32 A33];
% Amplification matrix invariants
A1=A(1,1)+A(2,2)+A(3,3);
A2=A(1,1)*A(2,2)-A(1,2)*A(2,1)+A(1,1)*A(3,3)-
A(1,3)*A(3,1)+A(2,2)*A(3,3)-...
    A(2,3)*A(3,2);
A3=A(1,1)*A(2,2)*A(3,3)-A(1,1)*A(2,3)*A(3,2)-
A(1,2)*A(2,1)*A(3,3)+A(1,2)*...
    A(2,3)*A(3,1)+A(1,3)*A(2,1)*A(3,2)-A(1,3)*A(2,2)*A(3,1);
% Transfer function denominator
a=[1 -A1 A2 -A3];
% Transfer function nominator
B1=1./D.*dt^2.*l3.*W1;
B2=1./D.*dt^2.*(l3.*(1-W1)-(A22+A33).*l3.*W1+A12.*l5.*W1+A13.*W1);
B3=1./D.*dt^2.*(-(A22+A33).*l3.*(1-W1)+A12.*l5.*(1-W1)+A13.*(1-W1)+...
    (A22.*A33-A23.*A32).*l3.*W1-(A12.*A33-A13.*A32).*l5.*W1+(A12.*A23-
...
    A13.*A22).*W1);
B4=1./D.*dt^2.*((A22.*A33-A23.*A32).*l3.*(1-W1)-(A12.*A33-
A13.*A32).*l5.*(1-...
    W1)+(A12.*A23-A13.*A22).*(1-W1));
b=[B1,B2,B3,B4];
```

Listing 3-11: Source code for the calculation of the denominator and nominator of the transfer function used in filter.m (source code continued from Listing 3-10).

```matlab
% form initial conditions for filter function
% equivalent external force
f=-xgtt;
% stiffness
k=omega.^2;
% damping constants
c=2.*omega.*ksi;
% initial acceleration
utt0=-f(1)-(k*u0+c*ut0);
U_1=A\[u0;dt*ut0;dt^2*utt0];
u_1=U_1(1);
U_2=A\U_1;
u_2=U_2(1);
ypast=[u0,u_1,u_2];
vinit=zeros(1,3);
vinit(3:-1:1) = filter(-a(4:-1:2),1,ypast);
% main dynamic analysis
u=filter(b,a,f,vinit);
% calculate velocity from the following system of equations:
% 1st: the first scalar equation of the matrix equation (60) in
X.Zhou &
% K.K.Tamma (2004)
% 2nd: equation of motion
C_u=omega^2*A(1,3)*dt^2-A(1,1);
C_f=-A(1,3)*dt^2;
C_ut=A(1,2)*dt-A(1,3)*dt^2*2*ksi*omega;
L=1/D*l3*dt^2*((1-W1)*[0;f(1:end-1)]+W1*f);
ut=(u+C_u*[u0;u(1:end-1)]+C_f*[0;f(1:end-1)]-L)/C_ut;
% calculate acceleration from equation of motion
utt=-omega^2*u-2*ksi*omega*ut;
```

Listing 3-12: Source code for the final part of the function LIDA.m used for the dynamic analysis of a linear elastic SDOF system (source code continued from Listing 3-11).

Input: $\ddot{u}_g$, ω, ξ, μ_t

Initialize SD, SV, SA, PSV, PSA

Set m, n_{NR}, u_0 and $\dot{u}_0$

for each SDOF i with eigenfrequency ω_i

   Find the low strain stiffness $k_{hi} = m\omega_i^2$

   if $\omega_i \Delta t/(2\pi) > 0.02$

      Reproduce $\ddot{u}_g$ with half time step (from Δt to $\Delta t/2$)

      Set $\Delta t = \Delta t/2$

   end

   Perform dynamic analysis of linear elastic SDOF ($\ddot{u}_g, k_{hi}, m, \xi, u_0, \dot{u}_0$)

   $u_{peak} = \max(|u(t)|)$

   $f_{peak} = k_{hi}\, u_{peak}$

   $pos = u_{peak}$

   $neg = \dfrac{u_{peak}}{1.5\mu_t}$

   for k from 1 to n_{NR}

      Perform dynamic analysis of bilinear elastoplastic SDOF ($\ddot{u}_g, k_{hi}, m, \xi, u_0, \dot{u}_0, u_{y,k}$)

      $u_{peak}^{NL} = \max(|u^{NL}(t)|)$

      $\mu = u_{peak}^{NL}/u_{y,k}$

      $res_k = \mu_t - \mu$

      if $|res_k|/\mu_t < tol_1$

         break

      else if $k \geq 2$

         Adjust tol_1 depending on the number of iterations k

         Find pos, neg, $u_{y,k+1}$ depending on the sign of res_k, res_{k-1} and $res_k - res_{k-1}$

      else if $k = 1$

         $u_{y,k+1} = neg$

      end

   end

   Calculate $SD(i) = \max(|u^{NL}(t)|) = u_{peak}^{NL}$, $SV(i) = \max(|\dot{u}^{NL}(t)|)$ and $SA(i) = \max(|\ddot{u}^{NL}(t)|)$

   Calculate $PSV(i) = SD(i)\omega_i$ and $PSA(i) = SD(i)\omega_i^2$

end

Output: SD, SV, SA, PSV, PSA

Listing 3-13: Flowchart of the calculation of the constant ductility response spectrum implemented in OpenSeismoMatlab.

3.2.6   Fourier Amplitude Spectrum

The Fourier Amplitude Spectrum (FASp) shows how the amplitude of the strong ground motion varies with frequency. It expresses generally the frequency content of a ground motion and useful information can be extracted from it. The MATLAB code that is used for the calculation of the FASp is shown in Listing 3-14 and Listing 3-15.

```
% FOURIER AMPLITUDE SPECTRUM
[f,U]=FASp(dt,xgtt);
seismic.FAS = U;
seismic.freq = f;
```

Listing 3-14: Source code for the calculation of the FASp.

As can be seen in Listing 3-14, the function FASp.m performs the majority of the calculations needed for the creation of the Fourier Amplitude Spectrum. The MATLAB source code for the function FASp.m is shown in Listing 3-15. The highest frequency that is considered for the calculation of the FASp is the Nyquist frequency of the input acceleration (the last is denoted in the code as xgtt). The Fourier spectrum that is calculated is single-sided and based on the MATLAB function fft which applies the Fast Fourier Transform [based on a library called FFTW (http://www.fftw.org); Frigo & Johnson, 1998]. It is apparent from Listing 3-15 that the transform length n_{FFT} has been set equal to the minimum power of 2 that gives result larger than the length of the input acceleration. This selection can increase the performance of FFT. The Fast Fourier Transform is implemented as follows:

$$U\left(k\right) = \sum_{j=1}^{n_{FFT}} \ddot{u}_g\left(j\right) W_{n_{FFT}}^{(j-1)(k-1)} \tag{3.19}$$

where

$$W_{n_{FFT}} = e^{(-2\pi i)/n_{FFT}} \tag{3.20}$$

is a $n_{FFT}{}^{th}$ root of unity and i in eq. (3.20) is the square root of -1.

3.2.7   Mean Period

The mean period (T_m) is computed from the Fourier amplitude spectrum of an acceleration time history by eq. (3.21) (Rathje et al., 2004):

$$T_m = \frac{\sum_i U_i^2\left(1/f_i\right)}{\sum_i U_i^2} \tag{3.21}$$

for $0.25\text{Hz} \le f_i \le 20\text{Hz}$ with $\Delta f \le 0.05\text{Hz}$. U_i is the Fourier amplitude, f_i is the frequency corresponding to U_i and Δf is the frequency interval. Eq. (3.21)

defines that the mean period is the weighted average of the periods of the spectrum with weighting based on squared Fourier amplitudes. The MATLAB source code that is used for the calculation of T_m is presented in Listing 3-16.

```matlab
function [f,U] = FASp(dt,xgtt)
% Nyquist frequency (highest frequency)
Ny = (1/dt)/2;
% number of points in xgtt
L  = length(xgtt);
% Next power of 2 from length of xgtt
NFFT = 2^nextpow2(L);
% frequency spacing
df = 1/(NFFT*dt);
% Fourier amplitudes
U = abs(fft(xgtt,NFFT))*dt;
% Single sided Fourier amplitude spectrum
U = U(2:Ny/df+1);
% frequency range
f = linspace(df,Ny,Ny/df)';
end
```

Listing 3-15: Source code of the function FASp.

```matlab
% MEAN PERIOD
fi = f(f>0.25 & f<20);
Ci = U(f>0.25 & f<20);
Tm = ((Ci(:)'.^2)*(1./fi(:)))/(Ci(:)'*Ci(:));
seismic.Tm = Tm;
```

Listing 3-16: Source code for the calculation of the mean period of an acceleration time-history.

3.3 Numerical results

In order to verify the results of OpenSeismoMatlab, they are compared with the corresponding results of a commercial strong ground motion data processing software, SeismoSignal. A detailed description of this software is given in Section 3.1. The reason for the selection of this software is that it is easy to use, it has a relatively detailed documentation of high scientific quality and it is accepted as a trustworthy and reliable tool worldwide, since it has been used and tested flawlessly for a number of years by researchers and professionals. However, SeismoSignal uses conventional time integration algorithms, and in certain cases can be susceptible to errors, especially when time integration algorithms with

dissipative and overshooting properties superior to those of the Newmark family of algorithms need to be used. OpenSeismoMatlab comes to improve this inadequacy. For the purposes of comparison, a suite of 11 strong ground motion acceleration time histories have been selected, which are presented in Figure 3-2. Data about the earthquakes that generated these acceleration time histories are shown in Table 3-1.

Earthquake	Year	Station
Imperial Valley	1979	El Centro Array Sta 8, CA, 95 E Cruickshank Rd
Izmit-Kocaeli	1999	Yarimca Petkim
Loma Prieta	1989	Gilroy Array Sta 3, CA, Sewage Plant
Northridge	1994	090 CDMG Station 24278
San Fernando	1971	Castaic, CA, Old Ridge Route
Spitak	1988	Gukasyan
Cape Mendocino	1992	Cape Mendocino, CA, Petrolia
Chi-Chi	1999	Nantou - Hsinjie School, WNT
El Centro	1940	El Centro Terminal Substation Building
Hollister	1961	USGS Station 1028
Kobe	1995	Takarazuka

Table 3-1: Earthquakes the strong motion records of which have been considered in the present chapter.

3.3.1   Peak values of displacement, velocity and acceleration time histories

In Table 3-2 the peak ground displacement (PGD), peak ground velocity (PGV) and peak ground acceleration (PGA) are calculated for 11 strong ground motions with various characteristics. Compared to the corresponding results of SeismoSignal (not shown here), it has been observed that the peak values are almost identical with the largest relative difference within 0.5% of the original value.

3.3.2   Arias Intensity values and time histories

In Table 3-3 the Arias intensity is shown for the same 11 strong ground motion records of Table 3-2. The results of OpenSeismoMatlab coincide with those of SeismoSignal.

In Figure 3-3 the time histories of the normalized Arias Intensity (AI) are presented for the 11 strong ground motions considered in this chapter. The normalization is made with respect to the AI values shown in Table 3-3. Two time

histories are shown in each subplot of this figure. The curves in red color correspond to the results of OpenSeismoMatlab whereas the curves in black color

Earthquake	PGD (m)	PGV (m/s)	PGA (m/s²)
Imperial Valley	1.233	0.553	5.997
Kocaeli	1.538	0.885	3.085
Loma Prieta	0.106	0.364	5.317
Northridge	0.402	0.787	9.707
San Fernando	1.722	0.337	2.654
Spitak	5.801	0.667	1.879
Cape Mendocino	0.349	0.445	10.194
ChiChi	0.422	0.644	9.373
El Centro	0.212	0.363	3.128
Hollister	0.003	0.042	1.347
Kobe	0.267	0.685	6.803

Table 3-2: PGD, PDV and PGA values of the strong motion data considered in this chapter.

Earthquake	Arias Intensity (m/s)
Imperial Valley	1.582
Kocaeli	1.669
Loma Prieta	2.075
Northridge	16.634
San Fernando	0.973
Spitak	0.311
Cape Mendocino	2.386
ChiChi	7.569
El Centro	1.802
Hollister	0.044
Kobe	3.067

Table 3-3: Arias Intensity (AI) of the strong motion data considered in this chapter.
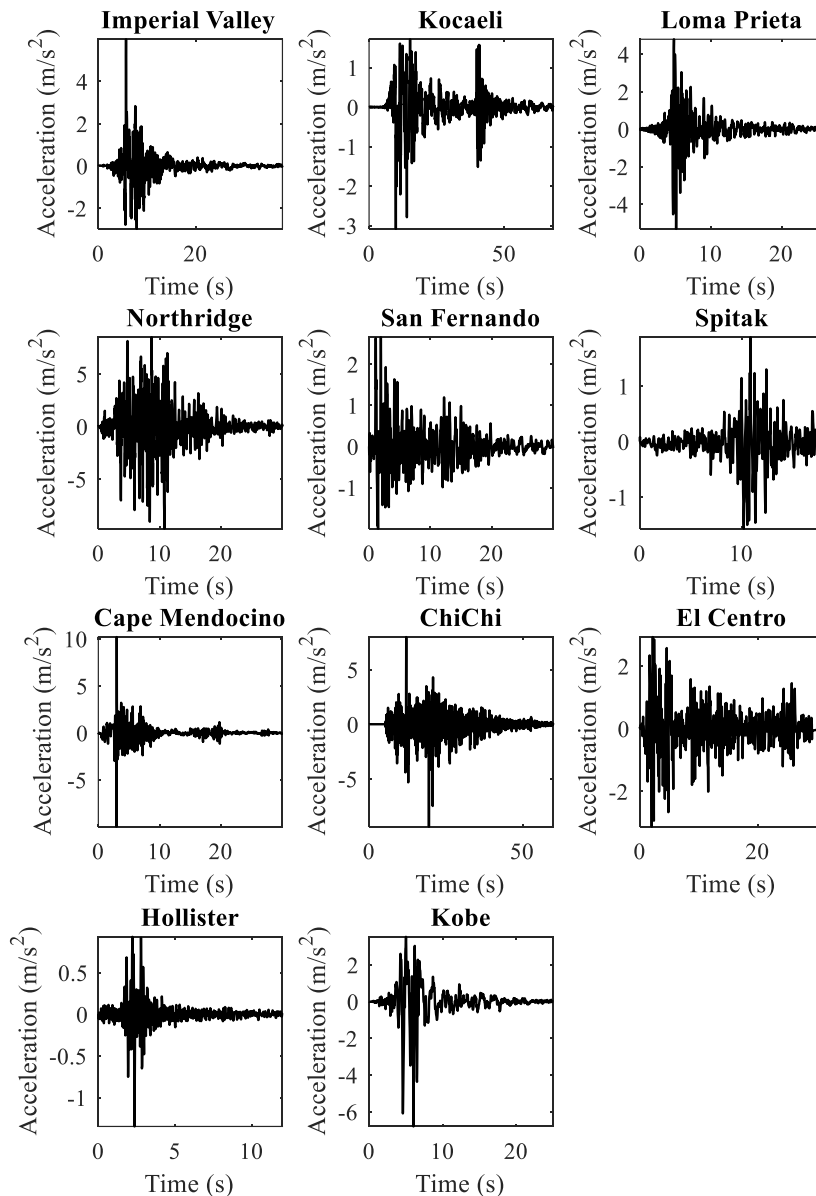
Figure 3-2: Acceleration time histories of the earthquake records considered.

(almost invisible since they are almost fully covered by the red curves) correspond to the results of SeismoSignal. It can be seen that the curves are nearly identical; this shows that the agreement between the two software is very good.

3.3.3   Significant durations 5-95

The significant duration of each strong ground motion record are shown in Table 3-4. The significant duration 5-95 is defined as the time interval between the point at which 5% of the Arias intensity is attained, and the point at which 95% of the Arias intensity is attained. The results of the two programs coincide.
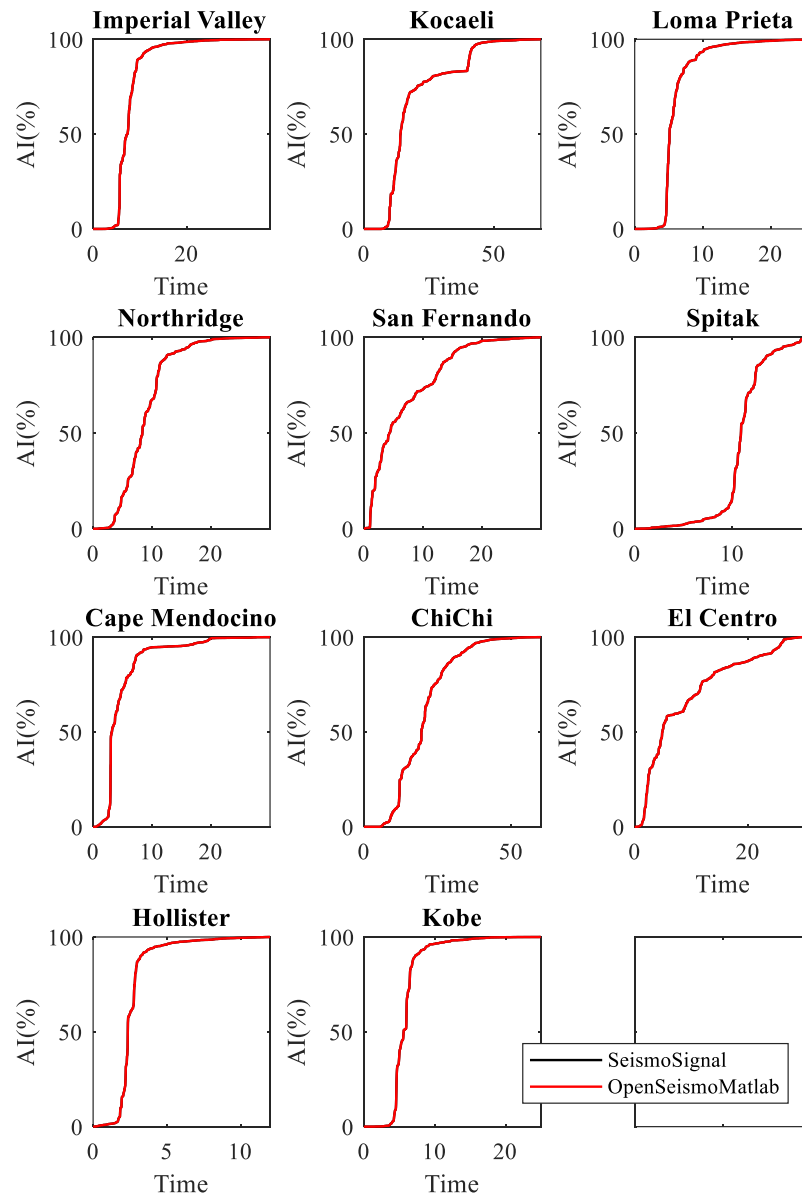
Figure 3-3: Time histories of the normalized Arias Intensity for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.

Earthquake	Significant duration 5-95 (s)
Imperial Valley	6.84
Kocaeli	31.66
Loma Prieta	6.00
Northridge	12.58
San Fernando	15.82
Spitak	8.08
Cape Mendocino	10.04
ChiChi	27.34
El Centro	23.84
Hollister	2.48
Kobe	4.60

Table 3-4: Significant duration of the strong motion data considered in this chapter.

3.3.4 Elastic response spectra

In Figure 3-4, Figure 3-5 and Figure 3-6 the displacement, pseudo-velocity and pseudo-acceleration response spectra are presented for the various strong ground motions considered in this chapter. As done also in other figures, the curves with red and the black color correspond to the results of OpenSeismoMatlab and SeismoSignal, respectively. The comparison is excellent between the two programs.

3.3.5 Constant-ductility inelastic response spectra

In Figure 3-7 and Figure 3-8 the spectral displacement and spectral velocity respectively are shown for the constant-ductility inelastic response spectra of the 11 strong ground motions considered for target ductility equal to 2. Obviously, the difference between the curves of the results of the two software is larger than that between the linear elastic counterparts. The differences between the results can be attributed to the different methods used by the two software, the superiority of the time integration algorithms used by OpenSeismoMatlab and other factors related to the efficiency of the implementation of the various procedures in the code of the two software. Despite these, generally the corresponding results of the two software are reasonably close to each other also in the nonlinear regime.

3.3.6 Fourier amplitude spectra

In Figure 3-9 the Fourier amplitude spectra (FAS) are shown for the strong ground motions considered. Two curves for each record are shown which correspond to the two software being compared. It seems that the various Fourier spectra are nearly identical.
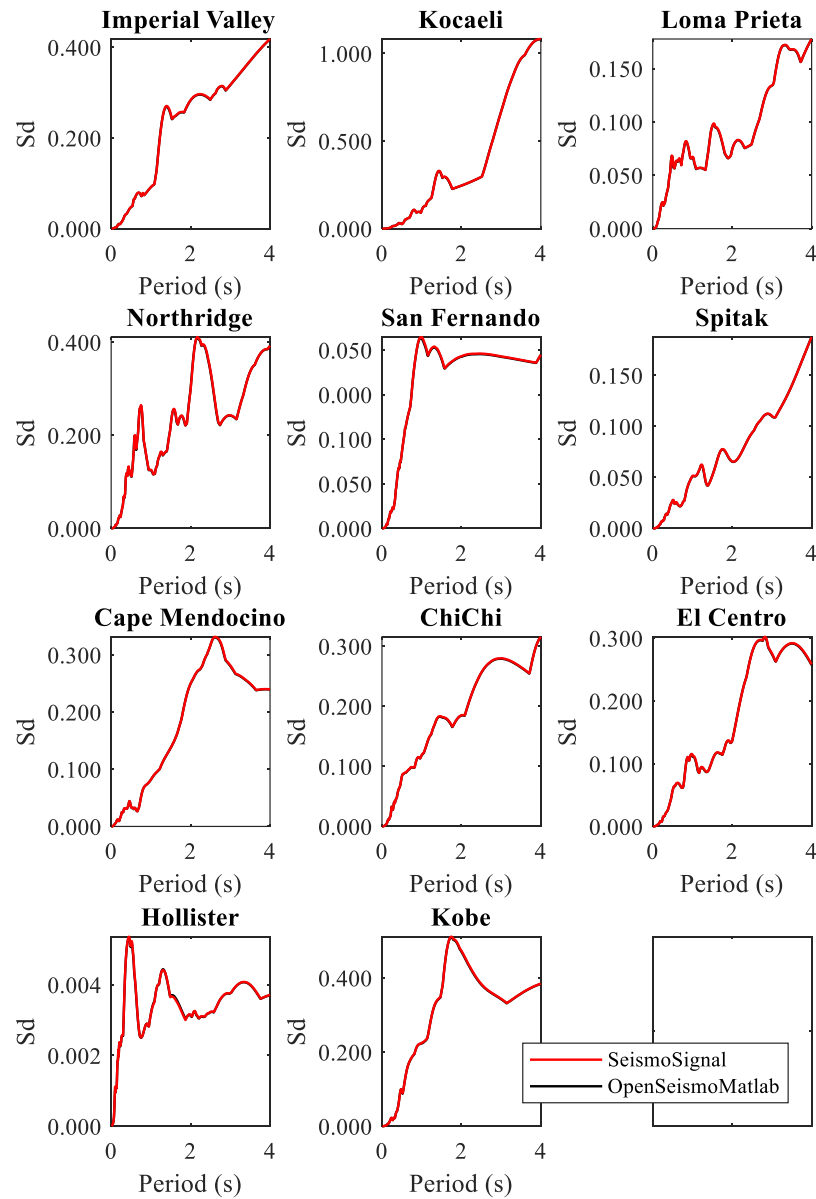
Figure 3-4: Linear elastic displacement response spectra for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.
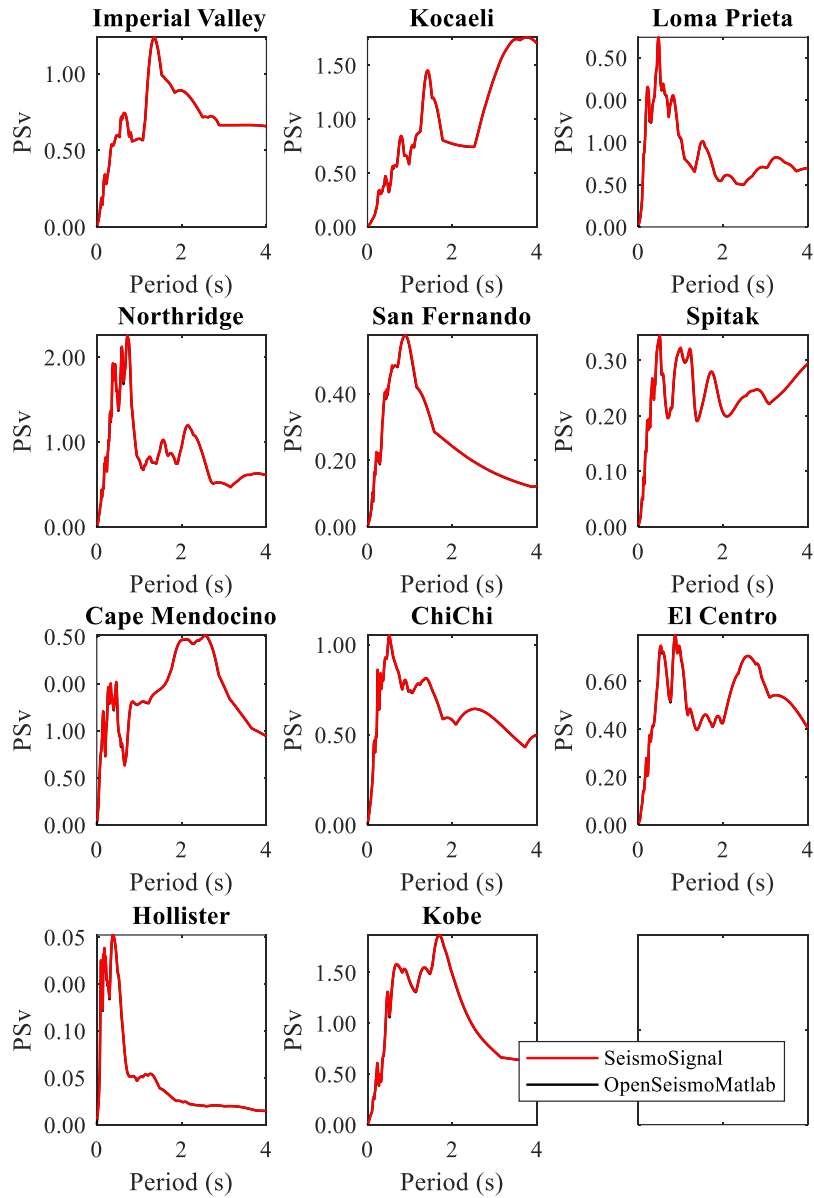
Figure 3-5: Linear elastic pseudo-velocity response spectra for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.
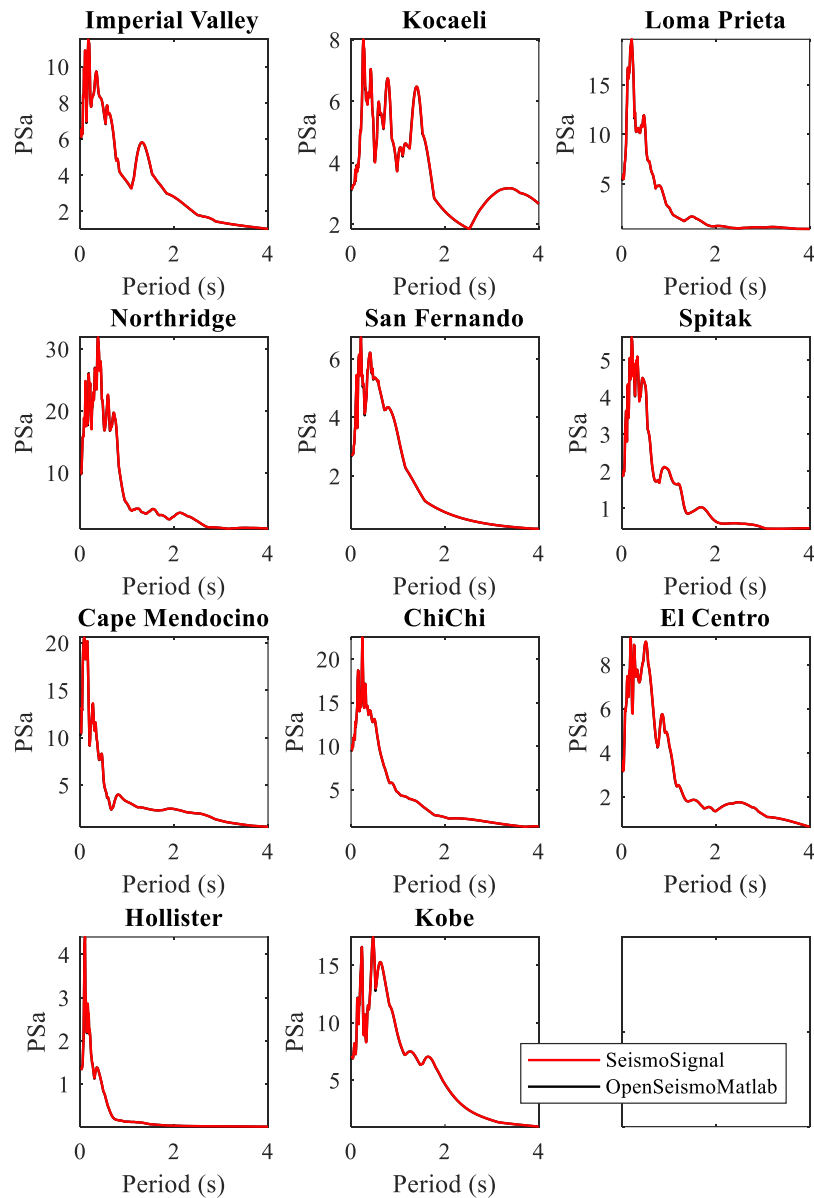
Figure 3-6: Linear elastic pseudo-acceleration response spectra for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.
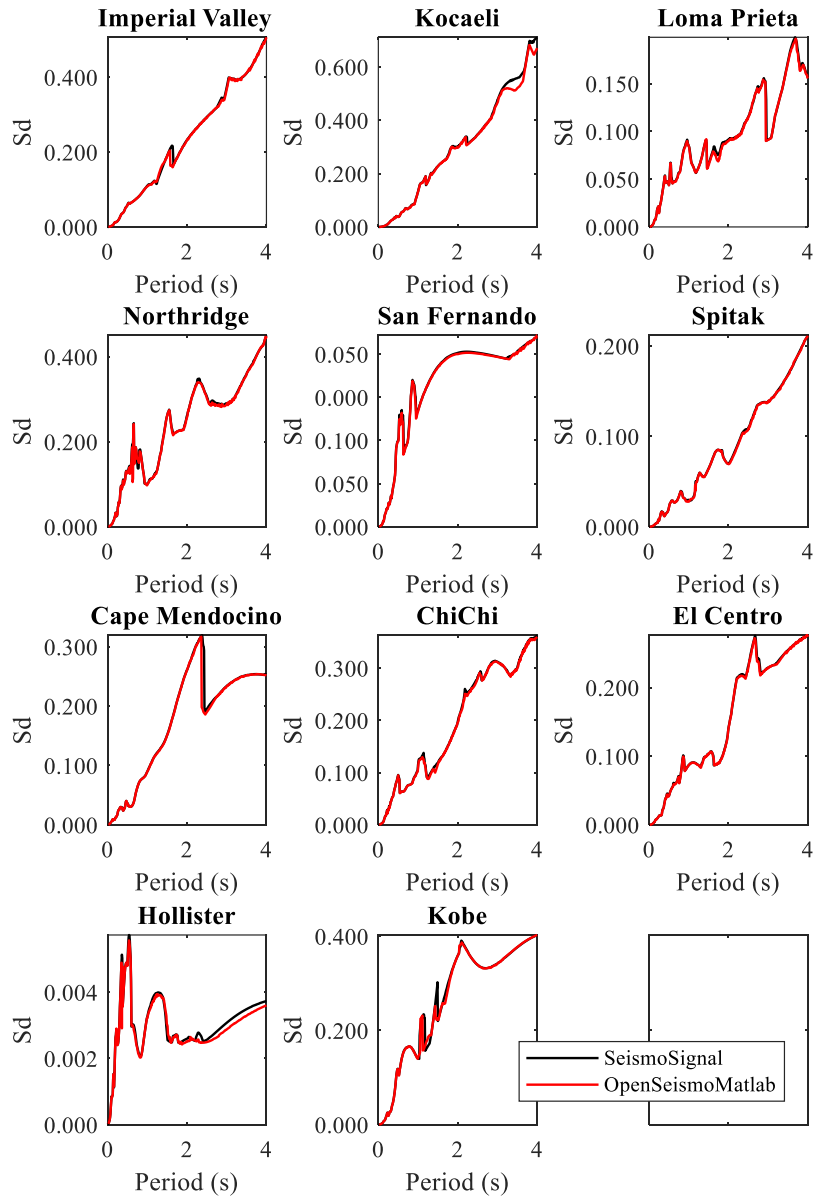
Figure 3-7: Constant ductility inelastic displacement response spectra for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.
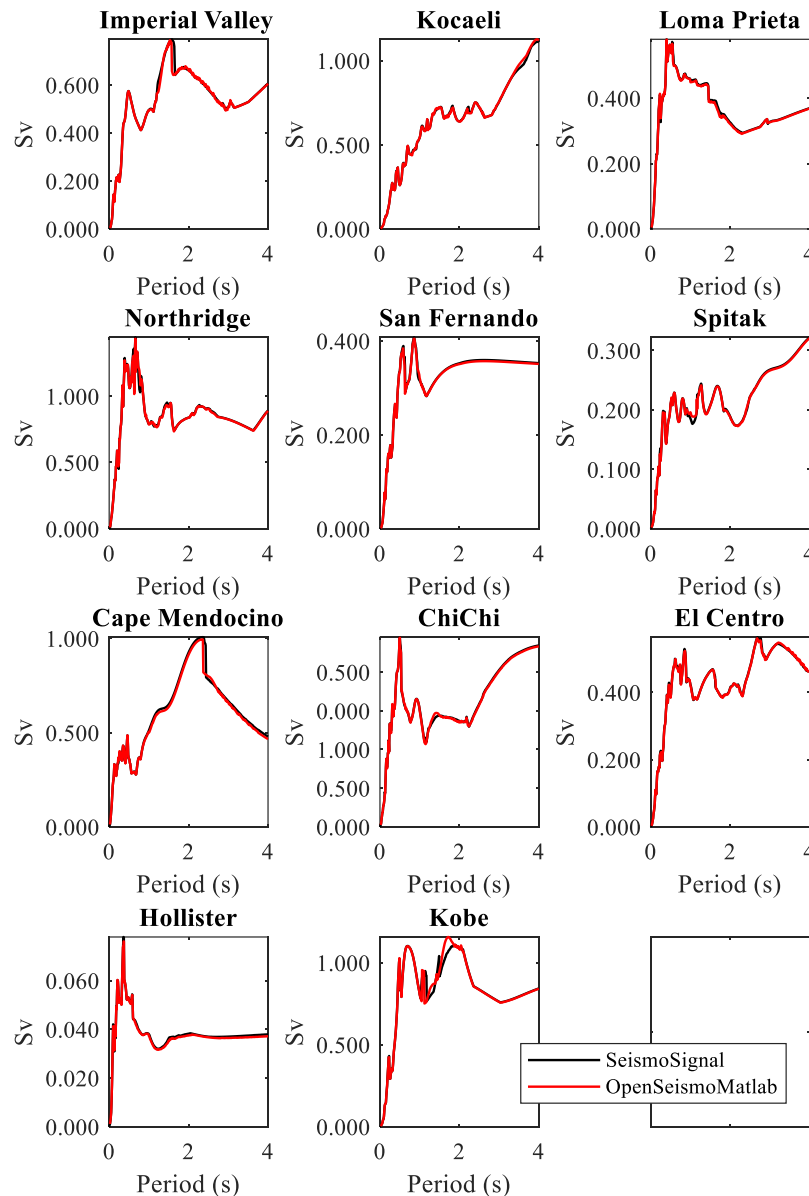
Figure 3-8: Constant ductility inelastic velocity response spectra for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.

3.4   Effect of the time step on the accuracy of the output

In this section, the effect of the time step size on the accuracy of the solutions provided by OpenSeismoMatlab is investigated. The pseudoacceleration (PSa) response spectrum of a sinusoidal acceleration time history (corresponding to the function $\ddot{u}_g = \sin\left(20\pi t\right)$) with critical damping ratio ξ equal to 5% is considered. The excitation is a harmonic (sinusoidal) motion with circular frequency equal to 20π (i.e. frequency 10 Hz and period 0.1 s) and total duration 2 s, whereas it is

digitized in sufficiently small time steps ($\Delta t = 0.0001s$). The PSa spectrum is calculated for OpenSeismoMatlab and SeismoSignal separately and initially a comparison is made between the two solutions. This comparison is shown in Figure 3-10, in which the decimal logarithm of the PSa spectrum is plotted versus the range of eigenperiods considered.

It is obvious that the two curves nearly coincide with each other and from this it can be concluded that practically they both coincide with the real solution, since the time step is relatively small enabling thus high accuracy computations. The differences between the two solutions are very small, found only at the 5th decimal digit. We define as PSa_0 this reference solution, i.e PSa_0 is considered the correct solution for each program. As the time step size increases, a certain degree of error is introduced in the PSa spectrum. A measure of this error can be the root-mean-square deviation between the PSa spectrum for an arbitrary value of Δt and the accurate PSa_0 presented in Figure 3-10, which is estimated by eq. (3.22) as follows:

$$RMSD = \sqrt{\frac{\sum_{i=1}^{n}\left(PSa_{\Delta t}^{i} - PSa_{0}^{i}\right)^2}{n}} \tag{3.22}$$

where $PSa_{\Delta t}$ is the PSa spectrum obtained for time step equal to Δt and n is the number of eigenperiods contained in the PSa spectrum (n=400 in this investigation). The different values of Δt that are considered are 3×10-4 s, 1×10-3 s and 3×10-3 s. For each value of these time steps, two $PSa_{\Delta t}$ spectra are calculated, one by OpenSeismoMatlab and one by SeismoSignal. Then, Eq. (3.22) is applied for the two programs separately, where for each one the corresponding PSa_0 is considered; two separate RMSD curves are extracted and plotted in Figure 3-11 for comparison. It is obvious that the solutions provided by OpenSeismoMatlab have less error than the corresponding solutions provided by SeismoSignal, for the various time step sizes. As a result, it is shown that the quality of the results of OpenSeismoMatlab is superior to that of SeismoSignal, at least under certain circumstances. This is attributed to the fact that advanced time integration algorithms are used by the former.

3.5   Conclusions

A new open-source software for strong ground motion data processing called OpenSeismoMatlab is presented, which uses advanced time integration algorithms, contains open and free source code written in MATLAB, and uses an accurate formulation and implementation of the elastoplastic bilinear kinematic hardening constitutive model. Parts of the code have been presented and
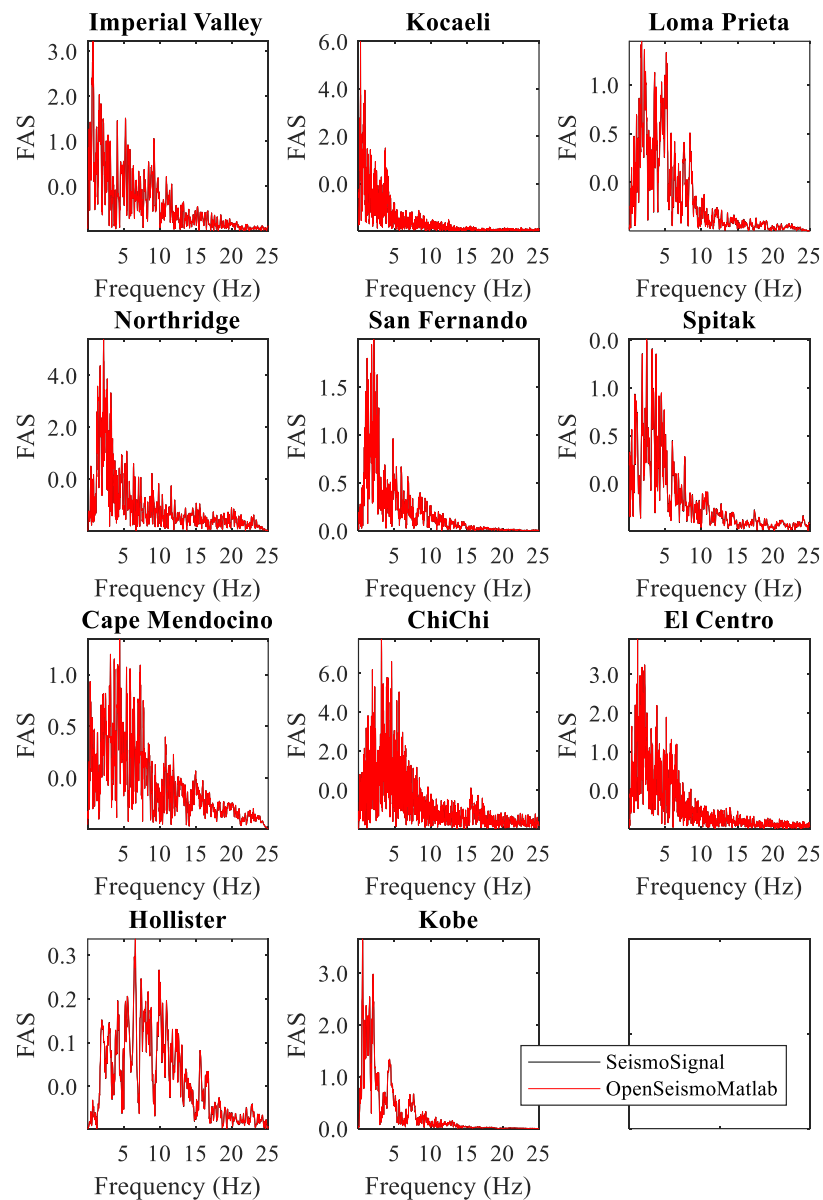
Figure 3-9: Fourier amplitude spectra for the strong motion data considered in this chapter calculated by OpenSeismoMatlab and SeismoSignal.
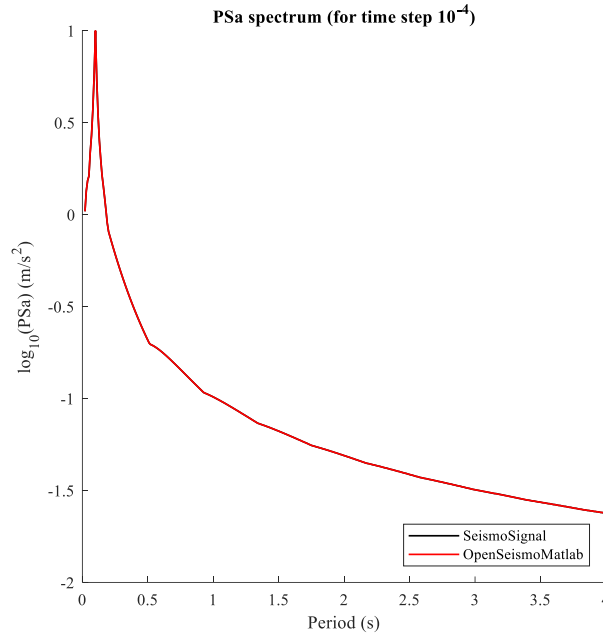
Figure 3-10: Comparison of the pseudoacceleration response spectrum PSa of a sinusoidal acceleration time history for a very small step size (Δt=10-4 s) between OpenSeismoMatlab and SeismoSignal.
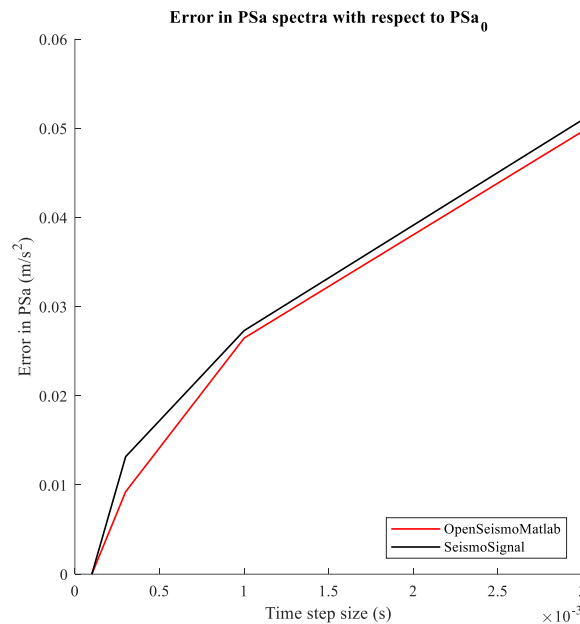
Figure 3-11: Comparison of the error of the pseudoacceleration spectrum (PSa) of a sinusoidal acceleration time history with respect to PSa_0, between OpenSeismoMatlab and SeismoSignal.

explained in detail in this chapter, so that the reader can easily understand the structure and implementation of the software and make various case-dependent adjustments in order to obtain results of the highest quality. The various types of spectra of 11 earthquake strong ground motions have been extracted with

OpenSeismoMatlab and it has been shown that they are nearly identical to corresponding results of SeismoSignal, a reliable commercial proprietary software. In some cases, the quality of the results of the new software is superior to that of SeismoSignal due to the fact that it uses advanced time integration algorithms that allow for controlled dissipation, dispersion and overshooting properties. A numerical investigation was made which showed that OpenSeismoMatlab provides more accurate results than SeismoSignal when the same integration step size is used for both. OpenSeismoMatlab is a unique software that combines innovative numerical algorithms, high quality and robustness and is provided as an open-source tool to the research and professional engineering communities for the seismic design of structures as well as the processing of strong ground motions. The new software can be used for free by students and/or programmers for the seismic design of structures as well as general processing of strong ground motions. Thanks to its open source nature, it can be of high educational value for related university courses and can be easily extended or modified in order to be incorporated in higher level software.

3.6   Notation

Δt : step of direct time integration scheme

$\lambda_1...\lambda_5$: time integration constants

$\mu_1...\mu_6$: time integration constants

μ: ductility ratio achieved for dynamic nonlinear analysis of a SDOF system

μ_t: target ductility ratio

ξ : ratio of critical viscous damping

Ω: Normalized circular eigenfrequency

ω: circular eigenfrequency

AI: Arias Intensity

$\overline{AI}$: Normalized Arias Intensity

c: damping coefficient

D_{5_95}: Significant duration of an earthquake

dT: period step of the generated response spectra

errΔt: error associated with the time step Δt

f_i : frequency corresponding to Fourier amplitude U_i

f_n: eigenfreuency of the nth SDOF system

f_{peak}: maximum linear elastic force

i_{max}: Total number of time increments of the earthquake motion

k: stiffness

m: mass

n: number of eigenperiods contained in the PSa spectrum

n_{FFT} : the transform length of the fft function

n_{NR}: Newton Raphson number of iterations for convergence to the target ductility

PGA: Peak Ground Acceleration

PGV: Peak Ground Velocity

PGD: Peak Ground Displacement

PSA: Pseudo Spectral Acceleration

$PSa_{\Delta t}$: PSa spectrum for time step equal to Δt

PSV: Pseudo Spectral Velocity

RMSD: Root-mean-square deviation between the PSa spectrum for an arbitrary value of Δt and PSa_0

SA: Spectral acceleration

SD: Spectral displacement

SV: Spectral velocity

T_1: lower period limit of the generated response spectra

T_2: upper period limit of the generated response spectra

TF_d: vector containing the coefficients of the denominator of the transfer function

TF_n: vector containing the coefficients of the nominator of the transfer function

t_f: total duration of the earthquake

T_m: mean period of an acceleration time-history

T_n: eigenperiod of the nth SDOF system

u_0: initial displacement

u_g: earthquake ground displacement

$\dot{u}_0$: initial velocity

$\dot{u}_g$: earthquake ground velocity

$\ddot{u}_g$: earthquake ground acceleration

u_{peak}: maximum value of the absolute displacement time history of a linear SDOF system

u: displacement time history of the linear elastic SDOF system

u^t: displacement at time t

$\dot{u}^t$: velocity at time t

$\ddot{u}^t$: acceleration at time t

$u_{y,k}$: yield limit of a SDOF system at k^{th} iteration

u^{NL}: displacement time history of the bilinear elastoplastic SDOF system

u_{peak}^{NL}: maximum value of the absolute displacement time history of a nonlinear SDOF system

U: Fourier amplitude

W_1: time integration constant

Chapter 4

■ ■ ■ ■ ■

4 A new energy-based structural design optimization concept under seismic actions

4.1 Introduction

In this chapter a new optimization concept is introduced which involves the optimization of nonlinear planar shear buildings by using a gradient method based on equivalent linear structures, instead of the traditional practice of calculating the gradients from the nonlinear objective function. The optimization problem is formulated as an equivalent linear system of equations in which a target fundamental eigenfrequency and an equally dissipated energy distribution among the storeys of the building are the components of the objective function. The concept is applied in a modified Newton-Raphson algorithm in order to find the optimum stiffness distribution of two representative linear or nonlinear MDOF shear buildings, so that the distribution of viscously damped and hysteretically dissipated energy respectively over the structural height is uniform. A number of optimization results are presented in which the effect of the earthquake excitation, the critical modal damping ratio and the normalized yield interstorey drift limit on the optimum stiffness distributions is studied. Structural design based on the proposed approach is more rational and technically feasible compared to other optimization strategies (e.g. uniform ductility concept), whereas it is expected to provide increased protection against global collapse and loss of life during strong earthquake events. Finally, it is proven that the new optimization concept not only reduces running times by as much as 91% compared to the classical Newton-Raphson optimization algorithms, but also it can be applied in other optimization algorithms which use gradient information to proceed to the optimum point.

Optimization techniques play an important role in various occasions in structural design, where they can be used by engineers, decision makers, etc. to find the best possible solution. Optimization methods used for structural design can be

classified into various categories, i.e. deterministic or stochastic (based on whether the model involves a fully specified or probabilistic formulation), constrained or unconstrained, local or global, etc. The objective of any structural optimization algorithm is to select among various possible design cases the optimum case which will minimize cost, maximize safety, and at the same time comply with the various design and construction constraints, if present.

In modern structural design for static and/or dynamic loading it is intended to design structures that will partially respond in the inelastic range, since this design proves to be more economical. Especially in seismic design, inelastic behavior is acceptable within certain limits, determined by the tradeoff between structural safety and economy. Besides, many structures have resisted earthquakes during which much higher inertia forces were induced to them than their strength calculated through linear elastic force based design. The concept of ductility was introduced to justify the latter and as a design tool for the former. These facts are realized by most current seismic design codes, mainly based on the traditional force based design procedures, which take these effects into account by introducing modification factors to reduce seismic force and overstrength demands depending on the structural system and the ductility desired. However, both force-based and displacement based design concepts are based only on the peak responses of a structure subject to an earthquake; the loading history or the time history of its response are not taken into account. The peak response does not provide enough information on how the structure has performed nonlinearly during an earthquake ground motion; there are various quantities which accumulate within the structure, such as the plastic energy absorbed by the structural components. The latter is a good indication of the damage suffered by the structure, especially in reinforced concrete structures. Therefore, it should be understood that seismic design should be time-history dependent and not based only on peak response at specific time instances. Based on the above a new design method has appeared, based on the energy input and dissipation in structures, named Energy Based Design (EBD). According to this method, an energy-dissipating mechanism has to be designed, which must have the ability to absorb greater amounts of energy than the input energy to a structure during strong ground motion, in order to ensure that the structure will efficiently resist earthquake motions.

Apart from the ductility of the construction material, the seismic performance of a structure is affected by its structural configuration and the distribution of strength and stiffness. Most collapses during or after past earthquakes have occurred to some extent due to incorrect structural configuration. The creation of soft storeys are a characteristic example of deficient structural behavior, where excessive ductility and drift are observed at a single floor of a building, leading to local collapse. Most buildings are designed according to the concept of equivalent static forces prescribed by seismic codes. The heightwise distribution of these forces results from the inherent assumption that the vibration modes of the

structure are linear elastic. On the other hand, according to the EBD concept, it is assumed that the structure responds non-linearly; consequently the assumption of linear elastic modes does not lead to realistic calculation of equivalent force distributions of the structure, and thus does not necessarily ensure optimum seismic performance, or even safety.

In this chapter, a new iterative optimization algorithm of Newton type with line search capabilities especially designed for structural optimization is presented and implemented for the optimum structural design in terms of the energy absorbed during an ensemble of seismic excitations. More specifically, the objective of the optimization process is to minimize the variation of the dissipated energy distribution along the height of a MDOF planar shear building, by finding the optimum distribution of storey stiffness and strength, for a prescribed fundamental (small strain) eigenperiod of the building. The optimization procedure is applied both for linear elastic and elastoplastic buildings. Based on the optimization results, the effects of different earthquakes, fundamental eigenperiods, number of storeys and amount of viscous damping along the height of the building on the optimum strength distributions are investigated and discussed.

4.2   Literature review

In most seismic design codes for buildings worldwide, the seismic effects on structures are taken into account in simplistic ways which refer to linear elastic structural models, or lateral force methods of analysis, e.g. CEN (1998); KBC (2009); IBC (2006); UBC (1997); NZS1170 (2004); AIJ (1996). For example, in Part 1, section 4.3.3.2 of CEN (1998), the horizontal seismic force distributions to be applied for design are determined based on the elastic properties of the structure, or even on a triangular distribution of horizontal displacements. It is apparent that these force distributions usually do not lead to evenly distributed dynamic distress of building structures, and therefore attempts have been made in the past to calculate these distributions in an optimum way by enforcing that distress and damage are equidistributed among the storeys of a building.

A first approach is to apply the theory of uniform deformation to determine the optimum seismic forces (Moghaddam & Hajirasouliha, 2004). According to this concept, while in most conventional design cases the ductility demand will vary among the floors of a building, leading either to material partially working or to material less than required, it is enforced that the maximum interstorey drift is uniformly distributed heightwise, and equal to the maximum allowable limit. Thus the condition of uniform deformation results in optimum use of material. The uniform deformation theory has been successfully applied in various studies for optimum seismic design of fixed-base shear buildings (Moghaddam & Hajirasouliha, 2006; Park & Medina, 2007; Hajirasouliha & Moghaddam, 2009; Hajirasouliha et al., 2012; Hajirasouliha & Pilakoutas, 2012). It

is concluded that there is generally a unique optimum distribution of structural properties, regardless of the load distribution used for the initial design (Moghaddam & Hajirasouliha, 2006). Furthermore, if the fundamental period increases, the load pattern usually increases at the top storeys due to higher mode effects. On the contrary, if the ductility demand increases the load pattern increases at the lower floors. Furthermore, a methodology to avoid concentrated deformation of a building and to distribute damage uniformly along the height without utilizing any optimization algorithm is proposed by Park & Medina (2007). The distributions implemented in current seismic codes are modified by suitably adjusting specific parameters using regression techniques. The resulting relationships remain highly empirical, and applicable only for structures with properties similar to those examined. In the study of Hajirasouliha & Moghaddam (2009) it is shown that structures designed according to the average of optimum load patterns, for the same ductility demand, have relatively less structural material in comparison with those designed conventionally. A practical method to redistribute material in reinforced concrete frames in order to achieve uniform deformation or damage is proposed, and is extended to take into account multiple performance objectives (Hajirasouliha et al., 2012). Moreover, the average strength distribution resulting from optimum strength distributions, calculated for various synthetic earthquakes representing a typical building code design spectrum and corresponding to either minimum ductility or minimum cumulative damage, is found and it is shown that structures accordingly designed can have up to 37% less structural weight compared to conventionally designed structures (Hajirasouliha & Pilakoutas, 2012). Also, structures with nonuniform mass distributions are considered and optimized in that study.

Soil-structure interaction effects are taken into account for the application of the uniform deformation theory for the optimum seismic design of shear buildings (Ganjavi & Hao, 2012a; Ganjavi & Hao, 2012b; Ganjavi & Hao, 2013). The effect of Soil-Structure interaction (SSI) on the optimum strength distribution at a MDOF structure, initially linear elastic and later nonlinear, based on the uniform deformation theory is examined (Ganjavi & Hao, 2012a; Ganjavi & Hao, 2012b). It is found that among the various seismic codes, UBC (1997). leads to the best performance, and that when SSI is taken into account, things change substantially regarding the optimum strength distribution, while optimum strength distributions established for fixed-based structures lose their efficiency when applied in their flexible foundation counterparts.

Besides these, hysteretic energy dissipation in a structure during an earthquake is the key factor related to the amount of damage in it. A structure is considered to resist an earthquake ground motion provided that the energy input to the structure from the earthquake is lower than its energy absorption capacity. Following this, the EBD concept as well as the determination of elastic and/or hysteretic energy distributions, have been examined for MDOF systems (Berg & Thomaides, 1960; Penzien, 1960; Zahrah & Hall, 1982; Akiyama, 1985; Nakamura &

Yamane, 1986; Léger & Dussault, 1992; Rodriguez, 1994; Nakashima et al., 1996; Connor et al., 1997). The energy dissipation in inelastic single degree of freedom structures is studied by Berg & Thomaides (1960), where it is found that usually when the yield level decreases, the amount of seismic energy transferred to the structure decreases. It is highlighted that design of multistorey buildings involving nonlinear response is not straightforward, because the number of points of occurrence and the distribution of inelastic deformations cannot be readily predicted. The elastoplastic shear-beam type models with several degrees of freedom are investigated and it is found that higher modes may have a large contribution in the overall structural response when the plastic deformations are relatively low, and therefore the lateral force distribution considered for design should be carefully selected (Penzien, 1960). The energy absorption in, and the inelastic behavior of, SDOF and 2-DOF structures during strong earthquake excitation are investigated (Zahrah & Hall, 1982). A shear-beam type 2-DOF structure is considered, with member behavior modeled by an elastoplastic bending moment-end rotation relationship. It is found that the amount of input energy in the 2-DOF system is the same as in an equivalent SDOF system with the same fundamental eigenfrequency, and the amount of energy dissipated by yielding is roughly equal to that of an equivalent SDOF system with the same damping and displacement ductility as the original structure. Moreover, an attempt for balanced energy dissipation among the two floors is made, whereas for structures with a balanced energy dissipation accurate predictions of their dynamic response can be made using modal analysis with a modified response spectrum. The effect of different mathematical models of viscous damping, computed from either the initial elastic or the tangent inelastic system properties, on the seismic hysteretic energy dissipation in MDOF structures is studied by Léger & Dussault (1992). A new parameter for measuring seismic damage capacity is proposed by Rodriguez (1994). It uses the energy dissipated by a structure in inelastic deformations and a structural overall drift, and it yields consistent results with building damage observed in practice. The energy behavior of buildings with hysteretic dampers modeled as bilinear elastic systems is examined by Nakashima et al. (1996), where the yield strength distribution over the storeys is such that all the DOFs of the building would yield simultaneously under the static design earthquake force profile specified in the Japanese Seismic Design Code (1981). It is found that the hysteretic energy profile becomes more uniform for increasing values of the post-yield to pre-yield stiffness ratio. A procedure for the distribution of seismic energy demand over the floors of a MDOF system solely by modal superposition of energy shapes which are established from a static pushover analysis is presented (Chou & Uang, 2003). Similar equivalent SDOF system concepts have also been used in the context of modal pushover analysis to estimate the hysteretic energy demand without the need for nonlinear time history analysis (Ghosh & Collins, 2006; Prasanth et al., 2008). Equations for the distribution of hysteretic energy for

MDOF systems, which are suitable for hard soil sites and for buildings the dynamic response of which is mainly controlled by the first mode shape are proposed (Wang & Yi, 2012), whereas a simplified method of distribution of hysteretic energy over the height of moment resisting frames, based on the work done by external forces applied at the storeys during the displacements of the latter is proposed by Mezgebo, (2015). A special case of this distribution scheme occurs if the story displacements are assumed to be proportional to the story mode shape values, which is similar to the relation proposed by Wang & Yi (2012).

Apart from the above, it has been shown that the addition to the structure of dampers of various types leads to modification of the hysteretic energy or maximum interstorey drift patterns. Optimum stiffness distribution along the building height has been proposed by Uetani et al. (2003). Optimum placement of oil, hysteretic and inertial mass dampers in order to minimize the maximum interstorey drift of the structure has been examined (Murakami et al., 2013). Detailed methods for addition of dampers in structures to optimize performance-based design for earthquakes can be found in Takewaki (2011). Despite the large amount of the literature being concerned with hysteretic energy distributions in shear buildings, to the best of the authors' knowledge the investigation of the conditions for uniform distribution of hysteretic energy along the height of a shear building has not been yet addressed in the literature. It was shown that it is possible to find an optimal stiffness distribution over the height of a linear elastic MDOF building to minimize the total seismic input energy, a ratio of which is the hysteretic energy responsible for structural damage (Shargh & Hosseini, 2010; Shargh & Hosseini, 2011). This optimum stiffness distribution results in minimum value of total dissipated hysteretic energy (Shargh et al., 2012).

The issue of optimum seismic design of nonlinear MDOF structures by modification of the stiffness and strength properties in order to achieve a uniform hysteretic energy dissipation pattern over the structure's height requires the formulation of a theory of uniform hysteretic energy dissipation, similar to the theory of uniform deformation already used for optimum seismic design and presented in the previous section. It has been shown that according to the latter with decreasing lateral yield strength the ductility demand decreases and if the former becomes lower than a certain point this trend is reversed (Penzien, 1960). However, the variation of hysteretic energy demand with yield strength is not as obvious as the variation of ductility demand with yield strength; it also depends on additional factors such as the duration of the seismic event. This entails that more robust techniques than those used for the uniform deformation theory have to be used to find optimum structural properties for uniform hysteretic energy distributions. An attempt to develop a new powerful optimization technique is made in this chapter to solve the uniform hysteretic energy problem in planar

MDOF systems, a problem that has not been solved yet, to the best of the author's knowledge. Therefore, two main novelties are considered in this chapter:

•   Formulation of the theory of uniform energy dissipation and optimum design of shear buildings according to the former, and

•   Development of a new fast and robust energy-based optimization technique.

4.3   Numerical modeling

4.3.1   Structural model

The most common model used for the dynamic response history analysis of building structures is the shear beam model. This system is represented by a viscously damped spring-mass system, where the mass is considered to be concentrated on each floor level and the storey shear force versus storey deflection relationship is presumed to be bilinear with a very low nonzero positive post-yield stiffness, so that the model responds effectively as elastic - perfectly plastic. The building deforms only in shear, since it is assumed that the floors are axially and flexurally rigid. Regularity with regards to the mass distribution along the height of a building is assumed and also it is presumed that changes in the stiffness distribution lead to negligible changes in the mass distribution (resulting from changes e.g. in the cross section of the columns, etc). Moreover, it is assumed that the floor masses move horizontally only within a vertical plane. Two MDOF systems are analysed: one 5-storey building and one 10-storey building. For each of the two MDOF systems the height of the storeys is assumed to be equal to 3m and the mass per floor is assumed to be equal to 25000kg. For each building a realistic value of fundamental eigenfrequency f_0 has been assumed; for the 5-storey building it is set equal to 2Hz (corresponding to fundamental eigenperiod 0.5sec) and for the 10-storey building it is set equal to 1Hz (corresponding to fundamental eigenperiod 1sec). The well-known rule of thumb that the fundamental eigenperiod of a building is equal to 0.1 sec multiplied by the number of storeys was used. Both buildings are considered to be fixed at their base, whereas their behavior is assumed to be either linear or nonlinear. Both linear and nonlinear buildings have been examined in this chapter. Also, for the nonlinear MDOF systems uniform non-dimensional yield displacement is assumed for all the floors, i.e. lateral stiffness is assumed as proportional to shear strength at each story. Damping is included through a classical damping matrix resulting from the superposition of the damping matrices of all linear elastic modes of the structures which have the same modal damping ratio. A number of horizontal seismic excitations are imposed at the base of the MDOF systems, resulting in their dynamic response. The two MDOF shear buildings analysed are shown in Figure 4-1.

Several types of hysteresis models are employed in research and engineering practice to predict the response of steel and reinforced concrete members subjected to cyclic loading. Six of them have been presented by Decanini & Mollaioli (2001) where a methodology for the assessment of the seismic energy demands imposed in structures is applied. To model the nonlinear force-deformation behavior of shear buildings, the elastic-perfectly plastic constitutive model is chosen in this chapter.
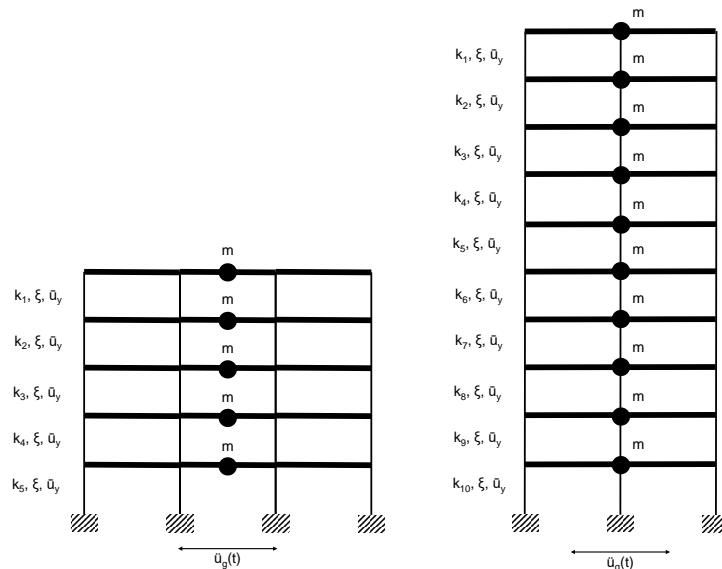
Figure 4-1: Planar shear building models analysed in this chapter with 5 and 10 storeys.

The elastoplastic model is chosen as a reference hysteretic model, since the introduction of a more sophisticated model of nonlinear response would complicate the range of validity of the optimization results, subtracting thus from generality. The elastoplastic model is considered as the fundamental model of hysteretic behavior and furthermore it is easier to be compared with other models. The methodology introduced in the present chapter is virtually a general framework for the optimum design of shear buildings, opening thus the way for more specialized treatments of the problem using advanced structural constitutive models. Bilinear hysteretic behavior is simulated using two linear models, corresponding to the two branches of the hysteresis loop.

The basic idea is that each branch of the hysteresis may be described by an equation of the form $f_K = k(u-d)$, where f_K is the restoring force, u is the displacement and d, k are the equilibrium displacement and pre- or post-yield stiffness at the last application of the elastoplastic model. The restoring force is zero when $u=d$. By suitable loops over the floors of the MDOF structure (counting from top to bottom of the shear building) and identification of transitions between the elastic loading, plastic loading and elastic unloading states, the interstorey forces and stiffnesses are calculated and passed to the time

integration algorithm. An indicative one-cycle force-displacement diagram of the bilinear elastic model is shown in Figure 4-2. The elastic-perfectly plastic constitutive model used in this chapter is implemented as follows:

- (*) Form the square pre-yield stiffness matrix K from k_{hi}
- Find the eigenfrequencies ω_i and eigenvectors φ_i of the linear elastic (pre-yield) structure with stiffness K and mass M for which the following relations hold:

$$\left|K - \omega_i^2 M\right| = 0 \tag{4.1}$$

$$\left(K - \omega_i^2 M\right)\varphi_i = 0 \tag{4.2}$$

and calculate the elastic pre-yield tangent damping matrix:

$$C = \sum_{i=1}^{n_{dofs}} 2\xi\omega_i M\varphi_i\varphi_i^T M \tag{4.3}$$

(**)Read the values of u, $\dot{u}$ and add a zero element to u, $\dot{u}$ to account for the fixed base:

$$u = \begin{bmatrix} u \\ 0 \end{bmatrix}, \dot{u} = \begin{bmatrix} \dot{u} \\ 0 \end{bmatrix} \tag{4.4}$$

for i from 1 to n_{dofs}, where n_{dofs} is the number of degrees of freedom (or storeys) of the structure, do the following:

- Compute the stiffness force of the current storey from k and d stored from previous application of the elastoplastic model (see step beginning with three asterisks (***) below):

$$p_{K,i} = k_i\left(u_i - u_{i+1} - d_i\right) \tag{4.5}$$

Compute the yielding force level:

$$p_{y,i} = k_{lo,i}\left(u_i - u_{i+1}\right) + \left(k_{hi,i} - k_{lo,i}\right)u_{y,i} \cdot \text{sign}\left(\dot{u}_i - \dot{u}_{i+1}\right) \tag{4.6}$$

Check for yielding or load reversal and update k_i and d_i accordingly

- If $k_i = k_{hi,i}$ and $\dot{u}_i - \dot{u}_{i+1} > 0$ and $p_{K,i} > p_{y,i}$, the system has exceeded its positive yield force level. Update as follows:

$$k_i = k_{lo,i} \tag{4.7}$$

$$d_i = \left(1 - k_{hi,i}/k_{lo,i}\right)u_{y,i} \tag{4.8}$$

If $k_i = k_{hi,i}$ and $\dot{u}_i - \dot{u}_{i+1} < 0$ and $p_{K,i} < p_{y,i}$, the system has exceeded its negative yield force level. Update as follows:

$$k_i = k_{lo,i} \tag{4.9}$$

$$d_i = \left(k_{hi,i}/k_{lo,i} - 1\right)u_{y,i} \tag{4.10}$$

If $k_i = k_{lo,i}$ and $p_{K,i}\left(\dot{u}_i - \dot{u}_{i+1}\right) < 0$, the system reloads from negative ultimate displacement or unloads from positive ultimate displacement. Update as follows:

$$k_i = k_{hi,i} \tag{4.11}$$

$$d_i = u_i - u_{i+1} - k_{lo,i} / k_{hi,i} \left(u_i - u_{i+1} - d_i \right) \tag{4.12}$$

Update the global force vector:

$$p_{K,int,i+1} = k_i \left(u_i - u_{i+1} - d_i \right) \tag{4.13}$$

for i from 1 to n_{dofs} find the elastoplastic forces of all storeys of the structure (with respect to its base):

$$p_i = p_{K,int,i+1} - p_{K,int,i} \tag{4.14}$$

Add the contribution of linear elastic damping to the internal force:

$$p = p + C\dot{u} \tag{4.15}$$

(***) Store k and d, and go back to the step beginning with an asterisk (*). Alternatively, store C, k and d, and go back to the step beginning with two asterisks (**).

The dimensionless interstorey drift yield limit is considered to be uniform along the height of the MDOF shear building, and is given by the equation:

$$\bar{u}_y = \frac{ku_y}{\max\left\{ |\ddot{x}_g| \right\} m} = u_y \frac{\left(2\pi f_0\right)^2}{\max\left\{ |\ddot{x}_g| \right\}} \tag{4.16}$$

where k is the pre-yield stiffness of a hypothetical SDOF system, u_y is its yield limit, m is its mass, f_0 is its fundamental eigenfrequency (which is considered equal to that of the building analysed) and $\ddot{x}_g$ is the time history of the earthquake acceleration.
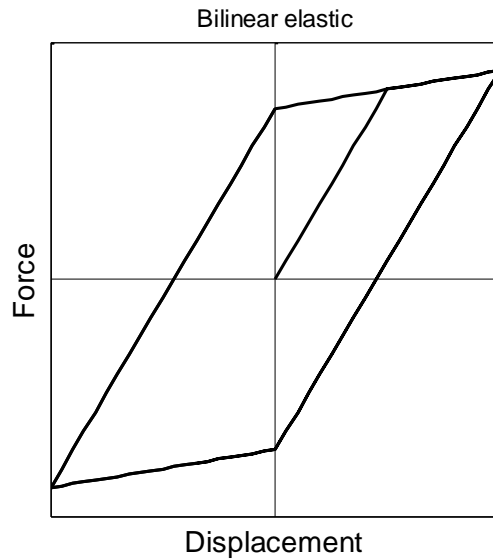
Figure 4-2. Force-displacement diagram showing one cycle of the bilinear elastic model used in this chapter.

4.3.2   Time integration algorithm for evaluation of structural response

The hysteretic energy demand can be accurately computed through a nonlinear dynamic time-history analysis of the structure subjected to a given earthquake ground acceleration. For the dynamic response history analyses

performed in this chapter, the family of nonlinear direct time integration algorithms presented by Papazafeiropoulos et al. (2017) is used. This family of algorithms is described by the following basic relationships:

- The updates of displacement and velocity:

$$u_{n+1} = u_n + \lambda_1 \dot{u}_n \Delta t + \lambda_2 \ddot{u}_n \Delta t^2 + \lambda_3 \left(\ddot{u}_{n+1} - \ddot{u}_n \right) \Delta t^2 \qquad (4.17)$$

$$\dot{u}_{n+1} = \dot{u}_n + \lambda_4 \ddot{u}_n \Delta t + \lambda_5 \left(\ddot{u}_{n+1} - \ddot{u}_n \right) \Delta t \qquad (4.18)$$

- The update of acceleration:

$$\tilde{M}_n^k \ddot{u}_{n+1} = \tilde{f}_n^k \qquad (4.19)$$

where

$$\tilde{f}_n^k \left(K_n^k, C_n^k, f_n^k \right) = -M \left(\ddot{u}_n^k - \mu_6 \ddot{u}_n^k \right) - C_n^k \left(\dot{u}_n^k + \mu_4 \ddot{u}_n^k \Delta t - \mu_5 \ddot{u}_n^k \Delta t \right)$$
$$- K_n^k \left(u_n^k + \mu_1 \dot{u}_n^k \Delta t + \mu_2 \ddot{u}_n^k \Delta t^2 - \mu_3 \ddot{u}_n^k \Delta t^2 \right) + \left(1 - W_1 \right) f_n^k + W_1 f_{n+1}^k \qquad (4.20)$$

and

$$\tilde{M}_n^k \left(K_n^k, C_n^k \right) = \mu_6 M + \mu_5 C_n^k \Delta t + \mu_3 K_n^k \Delta t^2 \qquad (4.21)$$

The residual equivalent force, which becomes zero if an iteration within an increment reaches equilibrium:

$$g_n^k = \tilde{f}_n^k - \tilde{M}_n^k \ddot{u}_n^k \qquad (4.22)$$

Any scheme of the aforementioned algorithm family needs 15 integration constants (of which 14 are independent) to be uniquely defined. See Papazafeiropoulos et al. (2017) for a complete list of the known time integration schemes which are special cases of this family. The time integration algorithm used here has optimal numerical dissipation and dispersion and zero order overshooting in displacement and velocity (Uo-Vo-Opt). In addition, equilibrium iterations are made within each increment by the use of a Newton-Raphson (N-R) procedure. The last updates the stiffness matrix at each iteration, until an equilibrium state is reached and the time integration algorithm proceeds to the next increment. It is possible, however, that during the iterations within an increment the algorithm does not converge, usually due to the fact that the stiffness of the structure changes abruptly between pre- and post-yielding state. In this case, the iterations are terminated and the last meaningful solution is accepted for equilibrium.

4.3.3   Energy-based design optimization problem

The minimization of the deviation of the energy distribution along the height of a building is treated in this chapter as an unconstrained optimization problem, the components of which are described in detail in the following paragraphs.

4.3.3.1   Design variables

The design variables of the optimization procedure are simply the stiffness of each storey of the two buildings under consideration, namely for the 5-storey and 10-storey buildings there are 5 and 10 design variables respectively. The various

stiffness distributions encountered during the optimization process have to respect some upper and lower limits, to ensure that computations remain meaningful and that no premature termination of the process occurs. For the stiffness of each storey, an upper and a lower limit is imposed which remains constant during the optimization procedure. Moreover, if at an iteration the new value of x_k violates any of the upper and lower bounds, the step length is appropriately decreased by applying a line-search algorithm, so that the new value of x_k lies within the upper and lower limits, whereas the Newton direction remains unchanged. This line search algorithm is described in section 4.3.4 below. The upper and lower bounds are equal to 1E9 N/m and 1E6 N/m respectively.

4.3.3.2   Objective function

The objective of the optimization procedure employed in this chapter is to find the stiffness distribution that corresponds to uniform energy dissipation over the structural height, either in terms of energy dissipation due to viscous damping for linear elastic structures, or in terms of energy dissipation due to hysteresis for elastic-perfectly plastic structures. However, the enforcement of uniform energy dissipation alone does not lead to a unique stiffness/strength distribution of the structure; the magnitude of the energy dissipated has to be additionally determined. The latter is done by imposing that the structure will have a specific fundamental eigenfrequency f_0 which controls the energy input in the structure. From the above it is concluded that the objective function has to be defined in a way that not only the distribution of the energy dissipation, but also the fundamental eigenfrequency of the structure have to be calculated as functions of the design variables (stiffness distribution along the height).

In this chapter the gradient of the objective function for elastic-perfectly plastic structures is defined as:

$$\nabla f_{obj}\left(x_k\right) = \frac{y_k}{\bar{y}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0}\right)^q \tag{4.23}$$

where the exponent q serves as a weighting factor between the energy distribution and the desired fundamental eigenfrequency of the building and is selected in a manner that maximizes the convergence rate of the optimization process. In this chapter q is set equal to 10 for all optimization analyses. Here, y_k is the vector of the hysteretically dissipated energy distribution along the height of the structure, $\bar{y}_k$ is its mean value and $\omega_{0,k}$ is the fundamental cyclic eigenfrequency of the structure having stiffness distribution x_k. Analogous equation holds for the linear elastic structures:

$$\nabla f_{obj}\left(x_k\right) = \frac{d_k}{\bar{d}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0}\right)^q \tag{4.24}$$

where d_k is the vector of the damped energy distribution along the structural height and $\overline{d}_k$ its average. It must be noted here that the explicit definition of the objective function is not of interest here, since it does not have any physical meaning; only its gradient is considered, which becomes zero at the point of optimum design.

4.3.3.3   Earthquakes considered

Nine earthquake records have been studied, which are the following: Imperial Valley (1979), Kobe (1995), Izmit-Kocaeli (1999), Cape Mendocino (1992), Loma Prieta (1989), Chi-Chi (1999), Imperial Valley (1940), Spitak (1988), San Fernando (1971). More details about these earthquake records can be seen in Table 4-1.

Earthquake	Station	Instrument	Component
Imperial Valley 1979	El Centro Array Sta 8, CA, 95 E Cruickshank Rd	Ground level	140
Kobe 1995	Takarazuka	Ground level	0
Izmit-Kocaeli 1999	Yarimca Petkim	Basement	0
Cape Mendocino 1992	Cape Mendocino, CA, Petrolia	Ground level	90
Loma Prieta 1989	Gilroy Array Sta 3, CA, Sewage Plant	Ground level	0
Chi-Chi 1999	Nantou - Hsinjie School, WNT	Free-field	90
Imperial Valley 1940	El Centro Terminal Substation Building	Ground level	N-S
Spitak 1988	Gukasyan	Free-field	0
San Fernando 1971	Castaic, CA, Old Ridge Route	Ground level	291

Table 4-1: Earthquake excitations considered in this chapter and their characteristics.

4.3.4   Optimization algorithm

In this chapter a gradient optimization strategy is employed to find optimum stiffness (and strength) distributions at MDOF shear buildings. Gradient based optimization methods search for a minimum of a scalar function $f_{obj}(x_k)$ of a vector including the floor stiffnesses as design variables x_k iteratively, by approximating the objective function by a Taylor series expansion around x_k:

$$f_{obj}(x_k + x) \approx f_{obj}(x_k) + \left(\nabla f_{obj}(x_k)\right)^T x + \frac{1}{2}x^T \nabla^2 f_{obj}(x_k)x \qquad (4.25)$$

At each optimization step, a direction e_k and a step length a_k are calculated based on the current value of the stiffness distribution x_k, and the latter as well as the objective function are updated based on the following equations:

$$x_{k+1} = x_k + a_k e_k \tag{4.26}$$

$$f_{obj}(x_{k+1}) \tag{4.27}$$

The algorithm begins with a random initial stiffness distribution x_0. The above process is repeated until the convergence criterion is satisfied, at which point the optimization algorithm terminates. The formal version of the Newton direction method involves a quadratic approximation of the objective function realized through the calculation of the Hessian matrix as follows:

$$e_k = -\left[\nabla^2 f_{obj}(x_k)\right]^{-1} \nabla f_{obj}(x_k) \tag{4.28}$$

where $\nabla^2 f_{obj}(x_k)$ is the jacobian. In this chapter, however, the above Newton direction is modified by adding a constant multiplied by the unity matrix, which is proved to stabilize the whole behavior of the optimization algorithm:

$$e_k = -\left[\nabla^2 f_{obj}(x_k) + NR_{stab}I\right]^{-1} \nabla f_{obj}(x_k) \tag{4.29}$$

where $\nabla^2 f_{obj}(x_k) + NR_{stab}I$ is the modified jacobian. Equation (4.29) can be rewritten due to equations (4.23) and (4.24) as follows:

$$e_k = -\left[\nabla\left\{\frac{d_k}{\bar{d}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0}\right)^q\right\} + NR_{stab}I\right]^{-1}\left[\frac{y_k}{\bar{y}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0}\right)^q\right] \tag{4.30}$$

and

$$e_k = -\left[\nabla\left\{\frac{d_k}{\bar{d}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0}\right)^q\right\} + NR_{stab}I\right]^{-1}\left[\frac{d_k}{\bar{d}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0}\right)^q\right] \tag{4.31}$$

for elastic-perfectly plastic and linear elastic MDOF structures respectively.

In this chapter equations (4.30) and (4.31) are used for the computation of the modified Newton direction, without explicit consideration of the objective function $f_{obj}(x_k)$. Note that the modified jacobian of equation (4.30) is not a function of the hysteretically dissipated energy y_k, but the damping energy of the equivalent linear elastic MDOF system d_k. The equivalent linear elastic MDOF system of a given elastic-perfectly plastic MDOF system is defined as the latter with its yield limit set equal to infinity (i.e. the former is defined by the behavior of the latter for small strains). This new way of calculation of the jacobian accelerates by far the optimization process of the nonlinear MDOF system, despite the minor loss in accuracy that is associated with this option.

Given that the calculation of the derivative of the energy distribution requires the largest part of the total computational effort required for the optimization process, it is concluded that this new logic of gradient optimization of nonlinear structures is vital for the reduction of the computational load. In addition, this

rationale introduces the concept of optimization points found from nonlinear structural response and directions found from equivalent linear structural response. The last concept can be applied not only in the case of Newton direction methods, but in many other types of optimization methods, utilizing either line searches or trust regions, such as steepest descent, conjugate gradient, subspace minimization, Broyden class algorithms, etc. In a future study, the authors will deal with how the aforementioned concept can be applied for improving the performance of such algorithms.

Two optimization procedures are implemented in this chapter. The first concerns the optimization of the linear elastic structure with respect to damping energy, using equations (4.31) and (4.26), whereas the second concerns the optimization of the elastic-perfectly plastic structure using equations (4.30) and (4.26). After having estimated the optimum stiffness distribution of the linear elastic structure, this distribution is used as the initial point for the optimization of the nonlinear structure. The optimization procedure implemented in this chapter is as follows:

- Initialize:

$$k = 1 \tag{4.32}$$

$$x_k = x_0 \tag{4.33}$$

$$r_k = -r \tag{4.34}$$

While the vector $|r_k|/|r|$ contains at least one value higher than tol_r:

- Check if the hessian has to be updated. If yes, calculate it from the relation(4.35), else omit this step and proceed to the following steps:

$$J_k = \nabla \left\{ \frac{d_k}{\overline{d}_k} - \left(\frac{\omega_{0,k}}{2\pi f_0} \right)^q \right\} + NR_{stab}I \tag{4.35}$$

Solve for the quasi-Newton direction e_k according to equations (4.30) or(4.31).

- Find a trial value for x_k by assuming a unit step along the direction e_k, using equation (4.36):

$$x_{k+1} = x_k + e_k \tag{4.36}$$

If any value of the new vector x_{k+1} is not within the upper and lower limits u_b and l_b respectively:

- Perform line search for the step in the direction e_k as follows:

$$a_{1k} = \min\{u_b - x_k\} / \left(\max\{e_k\} - \min\{e_k\} \right) \tag{4.37}$$

$$a_{2k} = \min\{x_k - l_b\} / \left(\max\{e_k\} - \min\{e_k\} \right) \tag{4.38}$$

$$a_k = \min\{a_{1k}, a_{2k}\} \tag{4.39}$$

Adjust x_{k+1} for the next iteration according to equation (4.26).

- Calculate the new residual for the next iteration:

$$r_k = \nabla f_{obj}\left(x_{k+1}\right) \tag{4.40}$$

Update the design variables and the iteration counter for the next iteration of the while loop:

$$x_k = x_{k+1} \tag{4.41}$$

$$k = k+1 \tag{4.42}$$

Regarding the aforementioned optimization parameters, the values $NR_{stab} = -3E-6$ and $tol_r = 0.01$ are specified in this chapter. The optimization algorithm implemented in this chapter can be easily applied in the case of irregular structures and give proper optimum stiffness distributions, not only for linear, but also for nonlinear shear buildings.

4.4   Typical hysteretic energy distributions for shear building

Typical distributions of the energy dissipated due to hysteresis during elastoplastic response of the 5-storey and 10-storey buildings considered in this chapter are shown in Figure 4-3. It has been assumed that the buildings have uniform stiffness distributions along their height, which are scaled so that they correspond to fundamental eigenfrequencies equal to 2 Hz and 1 Hz respectively. As it has been often observed in practice, the largest amounts of energy are concentrated at the bottom floors of the buildings for all the earthquake records considered. At the top floors the energy is much lower, and sometimes becomes zero (i.e. the upper floors do not participate as an energy absorption mechanism during seismic response). It is seen that generally the energy distributions vary nonlinearly with height. The largest energy demand on the building is generally imposed by the Kobe (1995), Kocaeli (1999) and Chi-Chi (1999) earthquakes. Figure 4-3 shows clearly the reason for which the damages caused by an earthquake accumulate at the lower floors, and why soft storey mechanisms develop more often at these levels. This phenomenon is undesirable; there is the need to equidistribute the seismic energy absorbed by the building among all storeys, in order to exploit the construction material as much as possible, and maximize structural safety. This chapter tries to cover this need by proposing a family of new fast optimization algorithms which has already been presented in the previous sections.

4.5   Optimum stiffness distributions for linear structures

In this section the optimum stiffness distributions are shown for the cases of the linear elastic versions of the 5-storey and 10-storey planar shear buildings considered in this chapter. The optimum stiffness distributions refer to the specific fundamental eigenfrequencies prescribed for both buildings (2 Hz and 1 Hz respectively) and various values of the critical damping ratio; uniformity of the dissipated energy due to viscous damping is enforced as has been already

discussed. Apart from the optimum stiffness distributions, the effects of various factors are discussed in the next.

4.5.1 Effect of earthquake excitation on optimum stiffness and energy distributions

Two families of optimum stiffness distributions along with their corresponding damping energy distributions are shown in Figure 4-4 and Figure 4-5 for the 5-storey and 10-storey MDOF systems analyzed in this chapter, respectively. It seems that the optimum stiffness generally has a regular distribution, where the largest value is at the first storey and the lowest at the top storey.
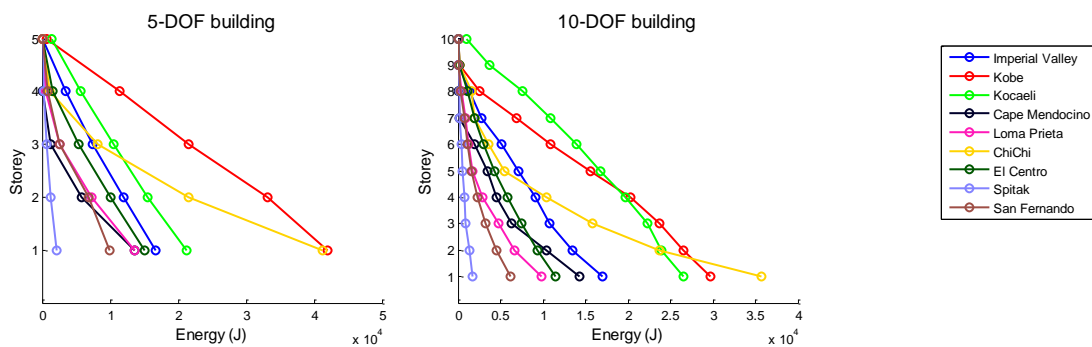
Figure 4-3: Distributions of energy dissipated due to hysteresis for the 5- and 10-storey shear buildings with uniform stiffness along their height, ξ=5%, ūy=0.01 and fundamental eigenfrequencies 2Hz and 1Hz respectively, for various earthquake records.

Similar results with Figure 4-4 are presented in Figure 4-5, where the optimum stiffness and optimum damping energy distributions for the 10-storey shear building are shown. It is obvious that the stiffness distributions of the 10-storey MDOF systems are regular and have generally their largest value at the bottom of the structure and their lowest value at the top. The general observation is that the stiffness distribution which corresponds to uniform damping energy over the height of a shear building is generally independent of the earthquake motion with which the building is excited. By comparing Figure 4-5 with Figure 4-4, it can be stated that, the stiffnesses of the 10-storey building are generally close to those of the 5-storey shear buildings. On the other hand, the energy distributions of the 10-storey building seem to be generally lower than those of the 5-storey building.

4.5.2 Effect of modal damping on optimum stiffness and energy distributions

In Figure 4-6 and Figure 4-7 the effect of critical modal viscous damping ratio on the optimum distributions of stiffness and damping energy for both 5-storey and 10-storey systems considered in this chapter is illustrated. In Figure 4-6 and Figure 4-7 results regarding the El Centro earthquake record are presented. It is

apparent that the two shear buildings have nearly identical stiffness distributions for the various critical damping ratios in the case of the El Centro earthquake record. Regarding the damping energy distributions, it can be seen generally that as the damping ratio increases, the amount of the dissipated energy also increases. Apart from this, with increasing damping ratio, the difference between successive dissipated energy distributions of the 5-storey system and the 10-storey system becomes lower. Finally, another thing to be noted is that the energy distributions of the 5-storey building are generally larger than those of the 10-
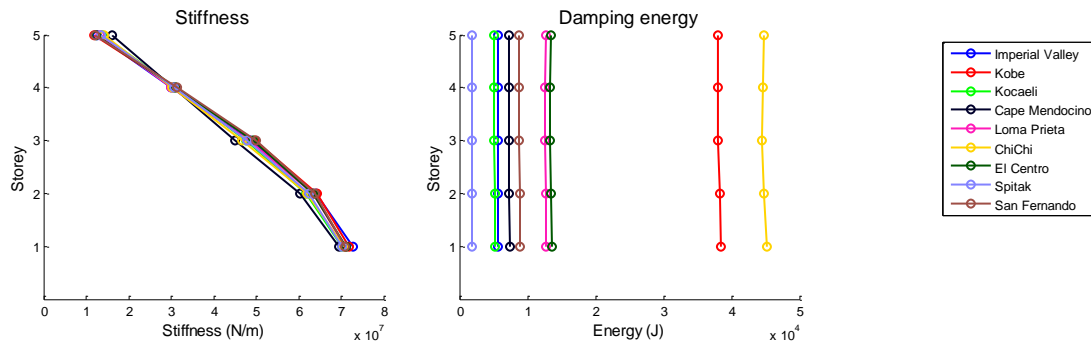
Figure 4-4: Optimum distributions of elastic stiffness and energy dissipated due to damping for the 5-storey shear building with ξ=5% for various earthquake records.
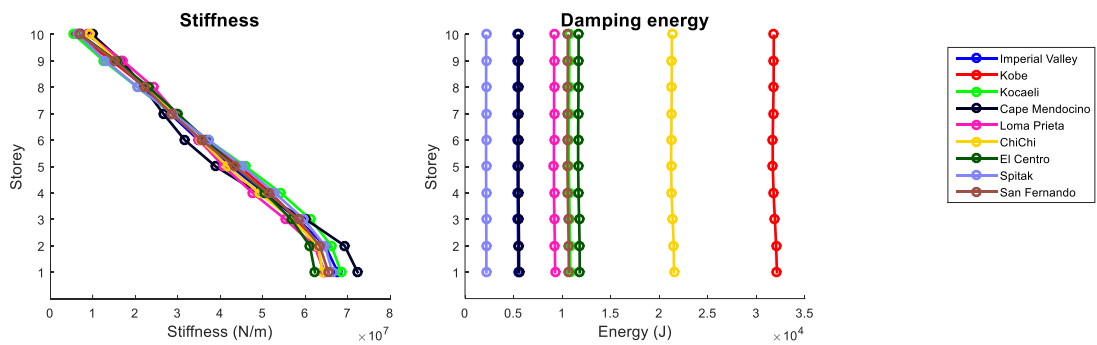
Figure 4-5: Optimum distributions of elastic stiffness and energy dissipated due to damping for the 10-storey shear building with ξ=5% for various earthquake records.
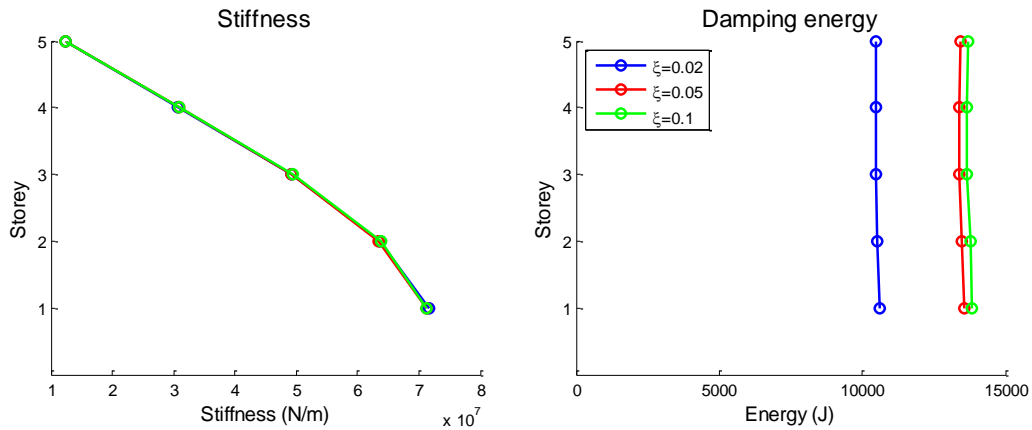
Figure 4-6: Optimum distributions of elastic stiffness and energy dissipated due to damping for the 5-storey shear building subject to the El Centro earthquake record for various critical modal damping ratios.
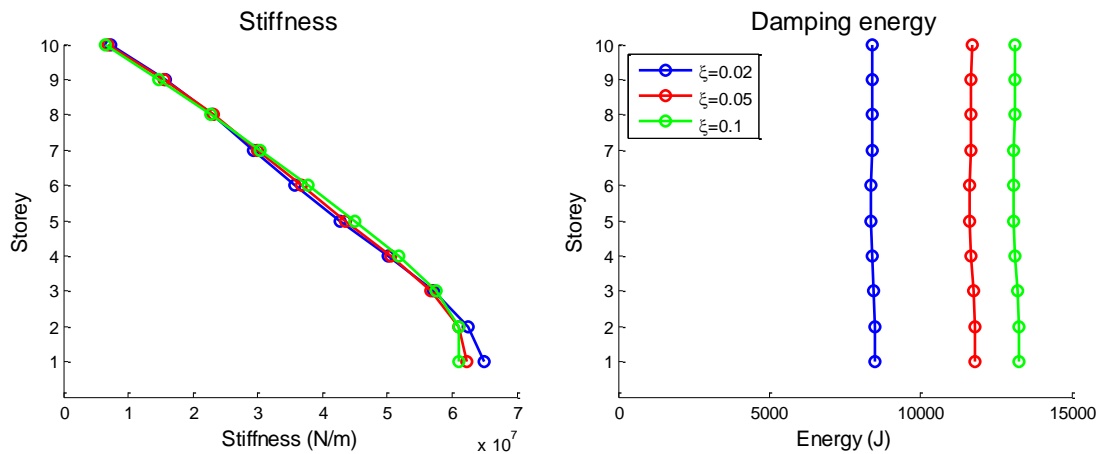
Figure 4-7: Optimum distributions of elastic stiffness and energy dissipated due to damping for the 10-storey shear building subject to the El Centro earthquake record for various critical modal damping ratios.

storey building.

4.6 Optimum stiffness distributions for elastic – perfectly plastic structures

In this section, optimization results are presented for the elastic-perfectly plastic counterparts of the planar shear buildings considered in the previous section. Along with the critical modal damping ratio, an additional parameter is taken into account here, which is the normalized interstorey drift yield limit, defined in equation (4.16). The fundamental eigenfrequencies of the two buildings remain the same as those in the linear elastic case: 2 Hz and 1 Hz for the 5- and 10-storey building respectively. The uniform normalized interstorey drift yield limit is assumed to be $\bar{u}_y = 0.1$ and $\bar{u}_y = 0.04$ for the 5- and 10-storey building respectively. It has to be noted here that the effective seismic force for

which the structure will be designed can be easily calculated from the optimum stiffness distribution multiplied by the uniform yield interstorey drift limit.

4.6.1 Effect of earthquake excitation on optimum stiffness and energy distributions

The optimum pre-yield stiffness distributions and its corresponding hysteretic energy distributions are shown in Figure 4-8, in the left and right subplots, respectively, for the 5-storey shear building considered in this chapter. It is observed that the stiffness distributions generally decrease from bottom to top, as was seen in the linear elastic case in Figure 4-4. It is noted that a general (quasi-linear) optimum stiffness distribution trend exists which is followed by the stiffness distributions for the various earthquake records considered, for both 5-storey and 10-storey buildings, perhaps with the slight exception of the Cape Mendocino (1992) and Loma Prieta (1989) earthquakes in the cases of 5- and 10-storey shear buildings respectively. The general stiffness distribution trend can be used in each case for structural design, at least in the preliminary stage. Concerning the hysteretic energy distributions for optimum stiffness at the right subplot of the figure, it is seen that the hysteretic energy that is suffered by the two buildings in the case of Kobe (1995) earthquake appears to be the largest of all earthquakes. The Kobe (1995) earthquake yields also the largest damping energy distribution in the case of the linear elastic 10-DOF system with optimum stiffness distribution (Figure 4-5). The hysteretic energy distribution of the Spitak (1988) earthquake remains to be the lowest of all earthquakes for both buildings. The above results lead to the conclusion that there is some close relation between the linear viscous damping energy and elastoplastic hysteretic energy that is dissipated at the storeys of a shear building.

Figure 4-9 shows the optimum pre-yield stiffness distributions and their associated hysteretic energy distributions for the elasto-plastically responding 10-storey shear building subject to various seismic excitations. It is seen that a general trend is again followed by the majority of the earthquakes considered. Regarding the energy distributions, it is seen again that the maximum hysteretic energy distribution occurs for the Kobe (1995) earthquake and the minimum for the Spitak (1988) earthquake. Additionally, the former is larger for the optimal 5-storey building than that for the optimal 10-storey building.
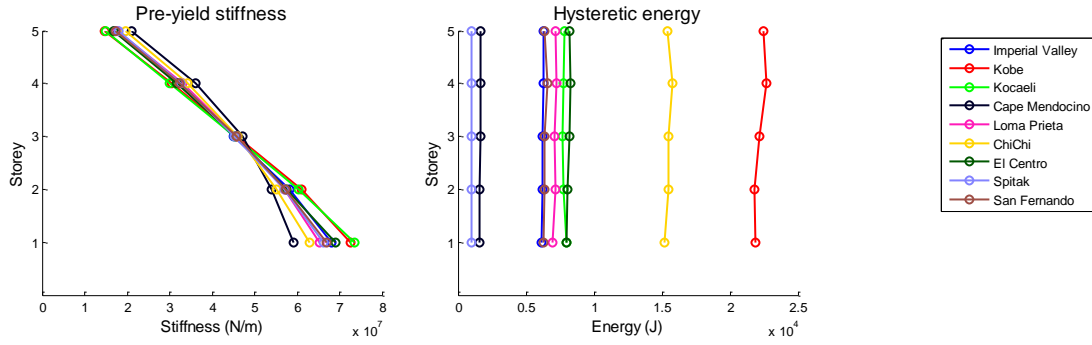
Figure 4-8: Optimum distributions of elastic pre-yield stiffness and energy dissipated due to hysteresis for the 5-storey shear building with ξ=5%, ūy=0.1 for various earthquake records.
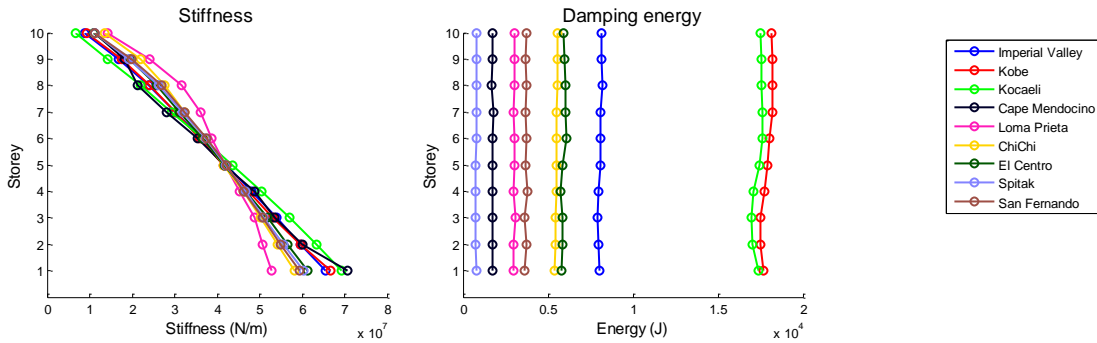
Figure 4-9: Optimum distributions of elastic pre-yield stiffness and energy dissipated due to hysteresis for the 10-storey shear building with ξ=5%, ūy=0.04 for various earthquake records.

4.6.2 Effect of critical modal damping ratio on optimum stiffness and energy distributions

The effect of critical modal damping ratio on the optimum stiffness and energy distributions of the two shear buildings is shown in Figure 4-10 and Figure 4-11 for the El Centro earthquake. It is observed that the various optimum stiffness distributions are nearly identical for the various cases of damping ratio, whereas it seems that as the damping ratio increases, the hysteretic energy distribution decreases. This can be explained by considering that the earthquake energy that is input to a shear building can be dissipated through either damping or hysteretic elastoplastic response. As the damping ratio increases, the energy dissipated through damping also increases. As a consequence of this, the portion of the input energy that is dissipated through hysteresis decreases. Apart from this, it is also observed that in all cases examined the 5-storey building has larger hysteretic energy distributions than the 10-storey building. Finally, the stiffness distributions for the 10-storey building are slightly lower than the corresponding distributions of the 5-storey building.
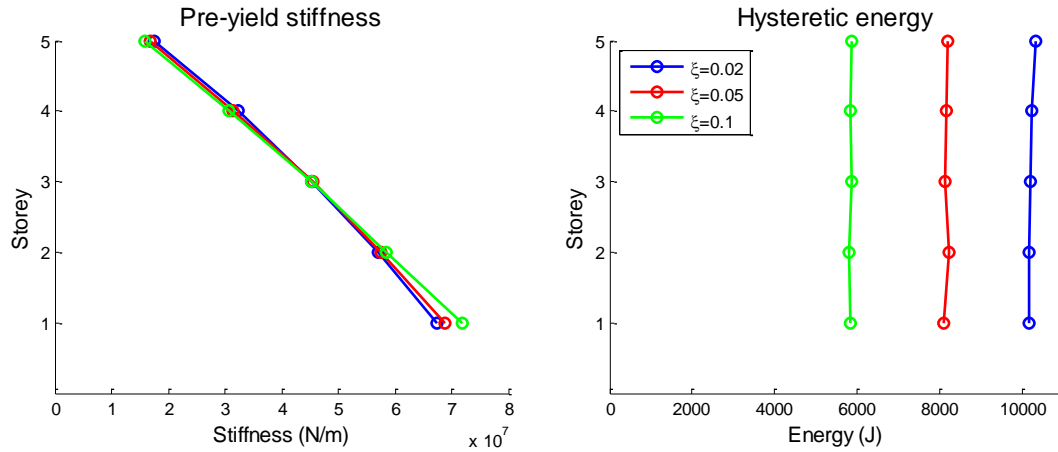
Figure 4-10: Optimum distributions of elastic pre-yield stiffness and energy dissipated due to hysteresis for the 5-storey shear building with ūy=0.1 subject to the El Centro earthquake record for various critical modal damping ratios.
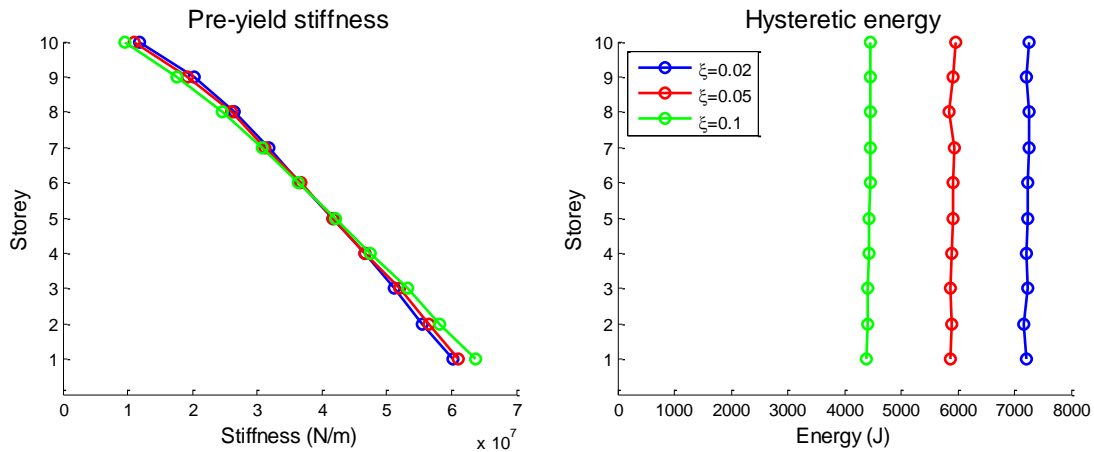
Figure 4-11: Optimum distributions of elastic pre-yield stiffness and energy dissipated due to hysteresis for the 10-storey shear building with ūy=0.04 subject to the El Centro earthquake record for various critical modal damping ratios.

4.6.3 Effect of normalized yield interstorey drift on optimum stiffness and energy distributions

The distribution of the normalized interstorey drift yield limit at a shear building is another factor that affects profoundly its structural response in the elastoplastic regime. This parameter is taken to be uniform for all storeys, and is calculated based on equation (4.16). In Figure 4-12 and Figure 4-13 the effect of this parameter is illustrated for the 5-storey and 10-storey shear building respectively with $\xi = 0.05$, subject to the El Centro earthquake excitation. It is observed that the stiffness distributions are relatively close to each other both for 5-storey and for 10-storey buildings, for the two values of $\overline{u}_y$. This can be attributed to the fact that $\overline{u}_y$ is uniform over the building height for both cases.

Besides, the energy distributions show clearly that as $\bar{u}_y$ decreases, the dissipated energy due to hysteresis increases, as is expected. This can be explained by the fact that as $\bar{u}_y$ decreases, the structure responds at the perfectly plastic branch of the elastoplastic response for larger time intervals, and therefore the effect of plasticity becomes greater, leading to larger hysteresis loops and thus increased hysteretic energies. Finally, it can be noted that the 5-storey building has slightly lower stiffness distributions and dissipates larger amounts of hysteretic energy than the 10-storey building.
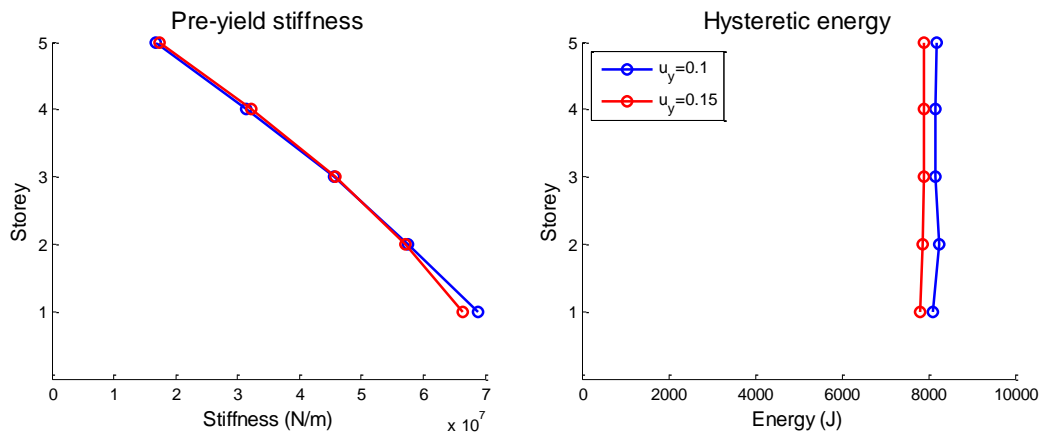
Figure 4-12: Optimum distributions of elastic pre-yield stiffness and energy dissipated due to hysteresis for the 5-storey shear building with ξ=5% subject to the El Centro earthquake record for various normalized yield interstorey drifts.

4.7   Effectiveness of the new optimization concept

For every new optimization algorithm, the question arises, how it increases the effectiveness, speed, etc. of the optimization process to which it is applied. The new optimization concept presented in this chapter can be applied for any energy-based optimization problem, and we need to see how the algorithm behaves for typical examples already presented in previous sections. In Figure 4-14 the evolution of the standard deviation of the hysteretic energy distribution is shown as a function of the normalized running time for the 5- and 10-storey shear buildings considered in this chapter with $f_0 = 2\text{Hz}$, $\xi = 0.05$, $\bar{u}_y = 0.1$ and $f_0 = 1\text{Hz}$,
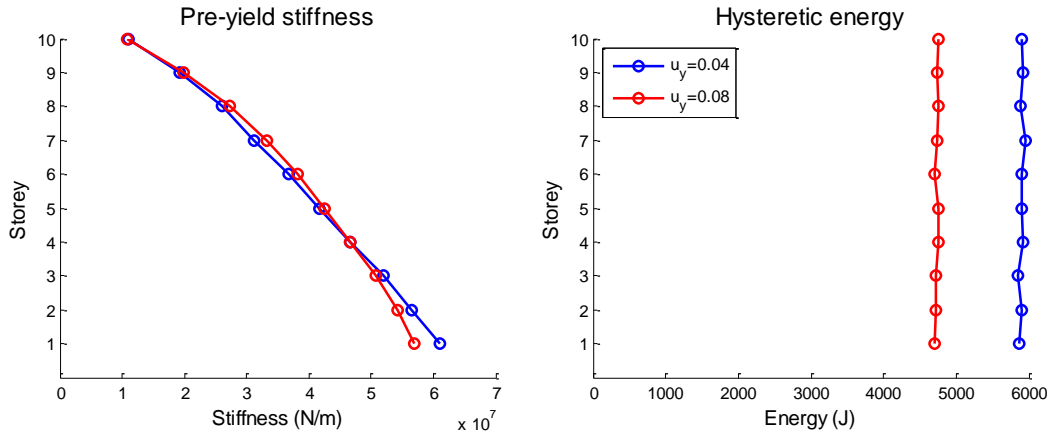
Figure 4-13: Optimum distributions of elastic pre-yield stiffness and energy dissipated due to hysteresis for the 10-storey shear building with ξ=5% subject to the El Centro earthquake record for various normalized yield interstorey drifts.

$\xi = 0.05$, $\overline{u}_y = 0.04$ respectively subject to the El Centro earthquake record. The optimum stiffness and energy distributions for the two cases are shown in Figure 4-12 and Figure 4-13. It is seen that the running times of the NR algorithm using linear derivatives are much lower than those with nonlinear derivatives. The running time of each optimization problem is normalized with respect to the running time of the optimization algorithm using nonlinear derivatives, hence the running time of the two nonlinear derivative algorithms is set to unity (100%).

It is clearly seen that the novel optimization algorithm proposed in this chapter can be roughly as much as 11 times faster than the traditional NR for the 10-storey system and roughly 7 times faster for the 5-storey system. For further increasing number of storeys, the novel algorithm is expected to be even over 11 times faster than the ordinary NR, saving thus a great amount of computational effort. It has to be noted here that, for comparison purposes, the initial stiffness distributions with which the algorithms began were set to be identical for both sets of cases, and equal to the linear elastic optimum stiffness distributions shown in Figure 4-6 and Figure 4-7 for $\xi = 0.05$. Since the algorithms begin from the same initial distribution to solve essentially the same problem (in terms of earthquake record and various structural properties), the differences in the running times and the general behavior are affected merely by the nature of the algorithm and its properties. The results of the optimization studies shown in Figure 4-14, are shown in Table 4-2.

It is seen that, the proposed NR algorithm in this chapter, while it retains the number of iterations approximately at the same levels with the classic NR, it can reduce the execution time by as much as 85% in the case of the 5-storey building and by 91% in the case of the 10-storey building. The reduction in the running

time is expected to increase for buildings with more storeys, or generally structures with more degrees of freedom. As a result, as the problem becomes more complicated, the proposed algorithm is expected to perform better compared to the classic NR method. Finally, it has to be pointed out that in both sets of cases, the final optimum stiffness distribution result was identical for both the classical NR method and the proposed optimization algorithm.
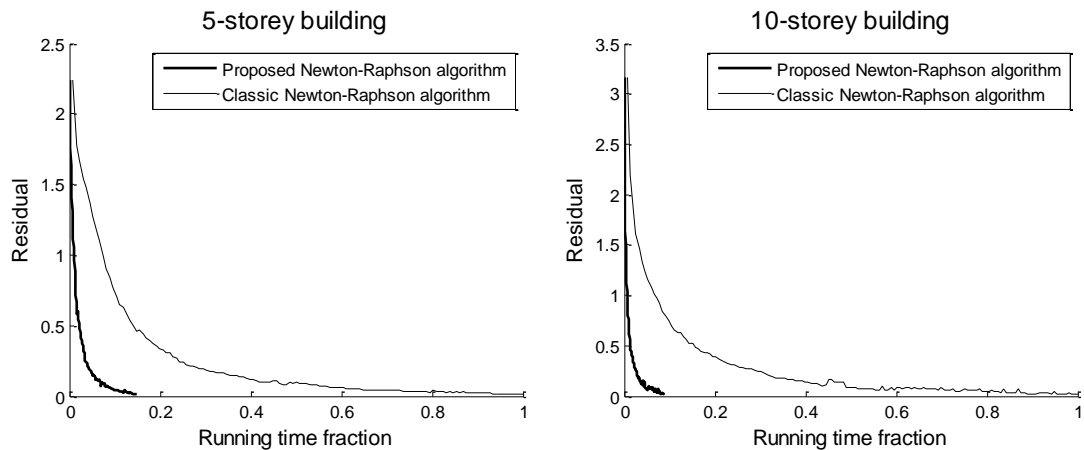
Figure 4-14: Evolution of the standard deviation of the hysteretic energy distributions during various optimization processes for the classic N-R optimization procedure and the proposed N-R optimization procedure, for 5- and 10-storey shear buildings with f0=2 Hz, ξ=5%, ūy=0.1 and f0=1 Hz, ξ=5%, ūy=0.04 respectively subject to the El Centro earthquake record.

Case	Normalized running time	Time reduction	Iterations
5-storey, Newton-Raphson	100%	-	136
5-storey, proposed algorithm	14.9%	85.1%	135
10-storey, Newton-Raphson	100%	-	153
10-storey, proposed algorithm	8.8%	91.2%	143

Table 4-2: Numerical results of the optimization processes the evolution of which is presented in Figure 4-14

4.8 Conclusions

The main conclusions drawn from this chapter are the following:
- It is shown that there exist unique optimum stiffness distributions which correspond to equidistributed viscous damping and hysteretic energy dissipation for linear elastic and elastoplastic planar shear building structures, respectively.
- In addition, the optimum stiffness distribution for both elastic and elastoplastic shear buildings appears generally to have a quasi-linear shape

(slightly curved), with the maximum value at the bottom floor and the minimum value at the top floor of the structure. This shape is generally independent of the earthquake excitation and offers the possibility for the development of simple methods for the calculation of the optimum stiffness distribution in shear buildings.

• Structural design based on the proposed approach is more rational and technically feasible compared to the uniform ductility concept, whereas it is expected to provide increased protection against global collapse and loss of life during strong earthquake events.

It is finally proved that the novel concept of linear directions equipped with a stabilizer for optimization of nonlinear problems, as applied for the modification of a simple full N-R method, leads to substantial computational savings, since, although the number of iterations required for convergence remains roughly the same, the running times can be reduced by a factor equal to 11. It is obvious that the new modified N-R algorithm is robust and efficient. The new concept presented in this chapter can be applied to other commonly used algorithms, which is the aim of future research to be conducted by the authors.

4.9   Notation

a_k : step length of update of x_k at iteration k

C : damping matrix

d : equilibrium displacement

d_i : equilibrium displacement at degree of freedom i at the last application of the elastoplastic model

d_k : damping energy distribution vector at iteration k of the optimization procedure

$\bar{d}_k$: average of damping energy distribution at iteration k of the optimization procedure

e_k : direction of update of x_k at iteration k

E_d : damping energy of a SDOF system

f : equivalent external loading vector due to seismic excitation imposed on the structure

$\tilde{f}$: effective force vector

y_k : hysteretic energy distribution vector at iteration k of the optimization procedure

$\bar{y}_k$: average of hysteretic energy distribution at iteration k of the optimization procedure

f_0 : fundamental eigenfrequency of SDOF or MDOF structure for small deformations

f_k : restoring force

f_{obj} : objective function

g : residual equivalent force vector

I : unity matrix

J_k : jacobian matrix (first derivative of energy distribution) at iteration k

k : iteration number or stiffness associated with a degree of freedom

k_{hi} : pre-yield stiffness vector

$k_{hi,i}$: pre-yield stiffness at degree of freedom i

k_i : stiffness at degree of freedom i at the last application of the elastoplastic model

$k_{lo,i}$: post-yield stiffness at degree of freedom i

K : stiffness matrix

l_b : lower bound of stiffness distribution x_k

M : mass matrix

$\tilde{M}$: effective mass matrix

m : lumped mass per storey of SDOF or MDOF systems

n_{dofs} : number of degrees of freedom of the structure

NR_{stab} : Newton-Raphson stabilizer constant for optimization procedure

p_i : internal force due to stiffness at degree of freedom i

$p_{K,i}$: internal force at degree of freedom i due to stiffness

$p_{K,int,i}$: interstorey force between degrees of freedom i and i+1 due to stiffness

$p_{y,i}$: yield force at degree of freedom i

q : exponent of eigenfrequency ratio

r : initial value of residual for the optimization procedure

r_k : residual at iteration k of optimization procedure

T_0 : fundamental eigenperiod of SDOF or MDOF structure for small deformations

tol_r : tolerance of $|r_k|/|r|$

u : displacement

u_b : upper bound of stiffness distribution x_k

u_i : displacement at degree of freedom i

$\dot{u}_i$: velocity at degree of freedom i

$\ddot{u}_i$: acceleration at degree of freedom i

u_y : yield displacement

$u_{y,i}$: yield displacement at degree of freedom i. Yielding occurs if the interstorey drift between degrees of freedom i and i+1 exceeds $u_{y,i}$.

$\bar{u}_y$: dimensionless yield interstorey drift

W_l : time integration constant

$\ddot{x}_g$: earthquake ground acceleration

x_k : stiffness distribution at iteration k of optimization procedure

x_0 : initial value of stiffness distribution to start optimization procedure

Δt : step of direct time integration scheme

$\lambda_1 ... \lambda_5$: time integration constants

$\mu_1 ... \mu_6$: time integration constants

ξ : ratio of critical viscous damping of the system, assumed to be unique for all storeys of the structure

φ_i : i^{th} fundamental eigenmode of structure

$\omega_{0,k}$: fundamental cyclic eigenfrequency of structure with stiffness distribution x_k for small deformations

ω_i : i^{th} fundamental cyclic eigenfrequency of structure

Chapter 5

■ ■ ■ ■ ■

5 Selecting and Scaling of Energy-Compatible Ground Motion Records

5.1 Introduction

A novel spectra-matching framework is proposed, which employs a linear combination of raw ground motion records to generate artificial acceleration time histories perfectly matching a target spectrum, taking into account not only the acceleration but also the seismic input energy equivalent velocity. This consideration is leading to optimum acceleration time histories which represent actual ground motions in a much more realistic way. The procedure of selection and scaling of the suite of ground motion records to fit a given target spectrum is formulated by means of an optimization problem. Characteristic ground motion records of different inherent nature are selected as target spectra, to verify the effectiveness of the algorithm. In order to assess the robustness and accuracy of the proposed methodology the seismic performance of single- and multi- degree of freedom structural systems has been also considered. The portion of the seismic input energy that is dissipated due to viscous damping action in the structure is quantified. It is shown that there exists a good agreement between the target and optimized spectra for the different matching scenarios examined, regardless of the nature of target spectra, demonstrating the reliability of the proposed methodology.

The response history analysis for the seismic design and the evaluation of the performance of structures has evolved along with the rapid increase in the computational power of the various engineering software. This has enabled not only the application of a faster and more accurate linear elastic time history analysis of structures having some thousands degrees of freedom, but also of the nonlinear time history response analysis which is becoming more and more common nowadays. Traditionally, the seismic design of structures is based on a force-based and/or displacement based approach, in which the effect of the earthquake loading is quantified using the peak ground and response spectra

acceleration of the corresponding ground motion record. However, the current status of the various norms regarding the selection of suitable ground motion records that meet specific requirements is rather simplified, which, despite the robustness of the various finite element models available for seismic design, may account for significant source of error in structural design. Therefore, the selection of appropriate sets of ground motion records for linear/nonlinear dynamic analysis of structures remains a challenge.

5.2   Literature review

Although, large ground motion databases are today widely available, in engineering practice, the problem of record selection is tackled either through scaling a real ground motion, or generating them artificially. A state-of-the-art review on the available methods for selection and scaling of ground motion records is presented by Katsanos et al. (2010), whereas some critical issues in record selection and manipulation are presented by Iervolino et al. (2008). In case of limited availability of appropriate real acceleration time-histories, simulated strong motion records can be used (Boore, 2009; Graves and Pitarka, 2010). The generation of artificial/simulated spectrum-compatible ground motion records has some disadvantages against real ground motions. Artificial records have generally a large number of cycles of strong motion, which leads to increased energy content compared to real ground motions. Adjusting the Fourier spectrum of a real ground motion in the frequency domain with a view to matching a target spectrum at specific frequencies affects amplitude, frequency content and phasing, which generally tends to increase the total input energy. The same deficiencies are observed also in the simulated records, which may not produce similar nonlinear response in structures as real records due to unrealistic phasing as well as peaks and troughs effects (Atkinson and Goda, 2010).

An alternative formulation of the loading effect of earthquakes on structures can be based on the earthquake input energy, which is the internal product of force and displacement. Energy considerations for the seismic design of structures constitutes the basis of the energy-based seismic design (EBSD) approach and is gaining extensive attention (Uang and Bertero, 1988; Chou and Uang, 2003; Surahman, 2007; Leelataviwat et al., 2009; Jiao et al., 2011; López Almansa et al., 2013). Since in the EBSD methods the energy-absorption capacity of the structure and the input energy that comes from the ground motion are compared for seismic design, it is imperative to develop and use design energy input spectra (DEIS).

EBSD has many benefits and compensates the deficiencies related to the use of conventional acceleration or pseudo-acceleration response spectra as follows: (a) It accounts for the effects of duration of the cyclic loading of the earthquake ground motion. Therefore, it can adequately capture the different type of time histories (impulsive, non-impulsive, periodic with long-duration pulses, etc.)

regarding their destructive potential. (b) It enables the quantitative evaluation of the cumulative structural damage in terms of hysteretic energy without the need to use equivalent viscous damping and/or ductility reduction. (c) There is no interdependence between the earthquake input energy and the structural resistance in terms of energy dissipation capacity, (d) The input energy that a structure experiences during an earthquake is governed primarily by its eigenperiod and mass and less by its strength or damping, except for the short-period range (Zahrah and Hall, 1984; Akiyama, 1985; Kuwamura and Galambos, 1989). This has been verified experimentally by Tselentis et al. (2010). Therefore the input energy is a stable quantity that does not depend on many factors and thus is simpler to handle and interpret.

Given the advantages of the EBSD over the traditional approaches, the incorporation of not only acceleration spectra but also energy-based spectra for the generation of artificial ground motion records is an interesting alternative that could lead to more realistic spectrum-compatible design records (Chapman, 1999; Tselentis et al., 2010). Actually, it has been demonstrated that if the hazard is assessed on the basis of the earthquake input energy, the hazard posed by larger magnitude earthquakes contributes more to the total seismic hazard at a specific site, than that based on spectral acceleration (Tselentis et al., 2010). It is noted that the input energy spectrum that is obtained elastically is valid also for inelastic systems since the strength and plastification of the structure do not practically affect the total energy input (López Almansa et al., 2013; Dindar et al., 2015).

In this chapter a novel spectra-matching framework is developed, to generate artificial acceleration time histories perfectly matched a target spectrum. Apart from the well-known design acceleration spectrum that is prescribed by the various norms and guidelines, the seismic input energy equivalent velocity spectrum is also taken into account. This consideration is leading therefore to optimum acceleration time histories which represent actual motions in a much more realistic way. In order to produce elastic spectra that match as closely as possible to a given target spectrum, the procedure of selection and scaling of the suite of ground motion records to fit a given target spectrum is formulated as an optimization problem. Three characteristic ground motion records of different inherent nature are selected as target spectra, to verify the effectiveness of the proposed algorithm, ensuring that its performance is target spectrum independent assuming different matching scenarios. The optimization results have shown that there exists a good agreement between the target and optimum spectra for each case examined, regardless of the nature of target spectrum, demonstrating the reliability and performance of the proposed methodology.

5.3   Numerical modeling

The main goal of this chapter is to obtain artificial ground motion records by performing as minimum as possible number of operations on the raw ground motion data. These ground motion records are linearly combined together forming a suite of records. The procedure of selection and scaling of the suite of ground motion records to fit a given target spectrum is formulated as an optimization problem. In this section, the processing of the raw ground motion data as well as the ingredients for the formulation of the optimization problem are presented.

5.3.1   Processing raw ground motion data

A linear combination of real accelerograms requires only selection and scaling of the latter and does not alter their inherent characteristics (e.g. non-stationarity, coda, phase content, etc.), which have to be preserved in order to obtain realistic artificial records as a result of the linear combination. Since the real records have various durations, linear combination cannot be applied directly to the acceleration time histories. However, it can be applied to their Fourier spectra in the frequency domain which have the same length for all motions; the resulting time history can be obtained by the inverse Fourier transform of the Fourier spectra of a suite of m ground motion records as follows:

$$\ddot{u}_{g,c} = \text{IFFT}\left(\sum_{i=1}^{m} x_i \text{FFT}\left(\ddot{u}_{g,i} \right) \right) \tag{5.1}$$

where $\ddot{u}_{g,i}$ is the acceleration time history, $\text{FFT}\left(\ddot{u}_{g,i}\right)$ is its Fast Fourier Transform, x_i is the combination coefficient respectively of the ith ground motion, $\text{IFFT}(\ )$ is the inverse Fourier transform and $\ddot{u}_{g,c}$ is the linear combination of the accelerations of ground motions records in the suite. Given that the Fourier transform of any real ground motion record is a linear transformation, it can be established that Eq. (5.1) effectively combines linearly the various records involved. In this way, the artificial time history that is generated depends only on selection and scaling of the participating ground motion records and also on the values of the combination coefficients x_i, i.e. scale factors.

It is apparent that an inverse Fourier transform of a signal in the frequency domain which is a linear combination of Fourier-transformed signals, requires a time step which has to be identical to that used for the Fourier transform of the original records, in order to obtain in this way realistic linear combinations of real ground motions. For this purpose, each record is resampled so that the fixed sampling rate of all records in the data base is unique. This fixed sampling rate

(or fixed time step) is used for the inverse Fourier transform of the linear combination of the Fourier transforms of the resampled motions.

5.3.1.1 Resampling

The resampling technique is based on least-squares linear-phase finite-duration impulse-response (FIR) filter for the rate conversion. The order N_{FIR} of the FIR filter is given by:

$$N_{FIR} = 20 \cdot \max\left(\Delta t_{old}, \Delta t_{new}\right) \tag{5.2}$$

where Δt_{old}, Δt_{new} are the time steps of the ground motion before and after conversion, respectively. The frequency-amplitude characteristics of the FIR filter approximately match those given by the relation:

$$A(f) = \begin{cases} 1 & 0 \le f \le f_0 \\ 0 & f_0 < f \le 1 \end{cases} \tag{5.3}$$

where A is the amplitude that corresponds to frequency f, 1 is the Nyquist frequency and f_0 is given by:

$$f_0 = 1 / \max\left\{\Delta t_{old}, \Delta t_{new}\right\} \tag{5.4}$$

The coefficients of the FIR filter are multiplied by the coefficients of a Kaiser window of length equal to $N_{FIR} + 1$, given by:

$$w(n) = \frac{I_0\left(\beta\sqrt{1 - \left(\dfrac{n - N_{FIR}/2}{N_{FIR}/2}\right)^2}\right)}{I_0(\beta)}, \quad 0 \le n \le N_{FIR} \tag{5.5}$$

where I_0 is the zero-th order modified Bessel function of the first kind. In this chapter, β parameter is selected to be equal to 5. To compensate for the delay of the linear phase filter a number of entries at the beginning of the output sequence are removed. After obtaining the FIR filter designed via a Kaiser window, the raw ground motion record is resampled based on this filter thus obtaining the modified ground motion history.

5.3.1.2 Fast Fourier Transform

The FFT of a raw motion data of Eq. (5.1) is calculated by means of DFT (Discrete Fourier Transform). The DFT of raw motion data $\ddot{u}_g(t)$ is calculated as:

$$\bar{\bar{u}}_g(k\omega) = \sum_{j=1}^{n} \ddot{u}_g(j\Delta t) W_n^{(j-1)(k-1)} \tag{5.6}$$

where $W_n = e^{-2\pi i/n}$ is one of the n roots of unity and $\omega = 1/(2n\Delta t)$. The inverse DFT of $\bar{\bar{u}}_g(k\omega)$ is given by:

$$\ddot{u}_g(j\Delta t) = \frac{1}{n} \sum_{k=1}^{n} \bar{\bar{u}}_g(k\omega) W_n^{-(j-1)(k-1)} \tag{5.7}$$

The execution time of DFT depends on the number of multiplications involved. A direct DFT evaluation takes n^2 multiplications whereas FFT takes $n \log_2 n$ multiplications. It has been proven that the n-point DFT can be obtained from two $n/2$-point transforms, one on even input data and one on odd input data (Frigo and Johnson, 1998; FFTW). Therefore, if n is a power of 2, then it is possible to recursively apply this decomposition until only discrete Fourier transforms of single points are left.

5.3.2   Problem formulation

In mathematical terms the procedure of selection, scaling and linearly combining of ground motion records to fit a given target spectrum is formulated as follows:

$$\text{minimize:} \qquad f(x)$$
$$\text{subject to:} \qquad \begin{aligned} x_{i,min} \leq x_i \leq x_{i,max} \\ i = \{1, 2, ..., D\} \end{aligned} \tag{5.8}$$

where f is the objective function to be minimized, x is the vector of design variables of dimension D, and $x_{i,min}$, $x_{i,max}$ are the lower and upper bounds of its i-th component.

5.3.2.1   Objective function

In this chapter, two types of objective functions are proposed:

(a) Objective function f_{Sa} which consists a measure of the area under the curve of the deviation between the suite and the target spectral accelerations and is defined as follows:

$$f_{Sa} = \int_{T_1}^{T_2} \left| \frac{Sa_c(T) - Sa_t(T)}{Sa_t(T)} \right| p(T) dT \tag{5.9}$$

where Sa_c is the spectral acceleration of the linear combination of the ground motions as obtained from Eq. (5.1) and Sa_t is the target spectral acceleration.

(b) Objective function $f_{Sa-Siev}$ which consists a measure of the sum of the following:

  • The area between the spectral acceleration curves.
  • The area between the equivalent seismic absolute input energy velocity spectra curves.
  • The area between the equivalent seismic relative input energy velocity spectra curves.

$f_{Sa-Siev}$ is given by:

$$f_{Sa-Siev} = \int_{T_1}^{T_2} \left(\left| \frac{Sa_c(T) - Sa_t(T)}{Sa_t(T)} \right| + \left| \frac{Siev_c^{ABS}(T) - Siev_t^{ABS}(T)}{Siev_t^{ABS}(T)} \right| + \left| \frac{Siev_c^{REL}(T) - Siev_t^{REL}(T)}{Siev_t^{REL}(T)} \right| \right) p(T) dT$$
$$\tag{5.10}$$

where Siev_c^{ABS}, Siev_c^{REL} are the spectral equivalent absolute and relative input energy velocities respectively of the suite of the ground motions and Siev_t^{ABS}, Siev_t^{REL} are the target spectral equivalent absolute and relative input energy velocities, respectively. Detailed calculation of Siev_t^{ABS} and Siev_t^{REL} quantities can be found in Uang and Bertero, 1990.

In Eqs. (5.9) and (5.10) $\left|\;\right|$ denotes the absolute value and $p(T)$ is a linear penalty function which is biased towards the lower period range and is given by:

$$p(T) = \frac{(T - T_1) + k_p(T_2 - T)}{T_2 - T_1} \tag{5.11}$$

where T_1, T_2 are the lower and upper period integration limits, T is the period and k_p is a penalty constant. Although baseline correction is performed before the various spectral computations, the penalty function ensures that the displacement and velocity of the acceleration is equal to zero at the start and the end of the time history considered.

5.3.2.2   Design variables

The design variables of the optimization problem are arranged into the vector x which contains $2m$ components, where m is the number of ground motion records in the suite. The first m components are the scale factors (continuous variables) used for the selected ground motions in the suite of Eq.(5.1), and the remaining components, are the IDs (integer variables) of the corresponding selected ground motion. The lower and upper bounds, $x_{i,min}$ and $x_{i,max}$ respectively of the continuous variables, $i = \{1, 2, ..., m\}$, have a significant impact on the performance of the optimization algorithm and the quality of the optimum solution. As the range of values of a design variable gets broader, the optimization algorithm shows a relaxed behavior, which can become unstable for very large upper and/or very small lower limits. Therefore, suitable values for these limits should be selected. The values selected in this chapter are as follows:

$$x_{i,min} = \begin{cases} -2.0 & 1 \le i \le m \\ 1 & m+1 \le i \le 2m \end{cases} \tag{5.12}$$

$$x_{i,max} = \begin{cases} 2.0 & 1 \le i \le m \\ M & m+1 \le i \le 2m \end{cases} \tag{5.13}$$

where M is the total number of the raw ground motions records contained in the database.

As obtained from Eqs. (5.12) and (5.13) the problem considered in this chapter is virtually a mixed-integer optimization problem and for this purpose the optimization algorithm has to be able to handle such a situation.

5.3.3   Mixed integer genetic algorithm

Choosing the proper search algorithm for solving such problem is not a straightforward procedure. Metaheuristic search optimization algorithms achieve efficient performance for a wide range of structural optimization problems. In this chapter, among the plethora of metaheuristic algorithms, a genetic algorithm has been chosen to solve the underlying optimization problem, capable to handle mixed-integer nature of the design variables. This should not be considered as an implication related to the efficiency of other algorithms, since any algorithm available can be used for solving a particular optimization problem based on researcher's experience.

The Genetic Algorithm (GA) is a stochastic global search optimization method introduced by Holland (1992) which emulates the natural biological evolution. GA applies on a population of potential solutions the principle of survival of the fittest to produce better approximations to a solution. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics (selection, crossover and mutation). This process leads to the evolution of individuals that are better suited to their environment than the individuals that they were created from, like in natural evolution process. The algorithm stops when a suitable criterion is met (e.g. current generation GEN equals to maximum number of generations, MAXGEN). A pseudocode of GA is described in Listing 5-1.

Pseudocode of the GA
1   Set parameters
2   Generate the initial population
3   **while** GEN < MAXGEN **do**
4   Fitness calculation
5   Selection
6   Crossover
7   Mutation
8   **end while**
9   Obtain the individual with maximum fitness
10   **return** the best solution

Listing 5-1: The pseudocode of a GA

For the purposes of this chapter, a real-valued representation is adopted as encoding strategy. The use of real-valued genes in GAs offers over binary encodings the following advantages: (i) efficiency of the GA is increased as there is no need to convert chromosomes to phenotypes before each function evaluation, (ii) less memory is required as efficient floating point internal computer representations can be used directly, (iii) no loss in precision by

discretization to binary or other values, (iv) greater freedom to use a variety of genetic operators.

5.3.3.1 Initialization of population

The GA starts with the generation of a random initial population of individuals with uniform distribution in the initial generation. If the initial population is denoted by P_0 and its size (number of individuals) by n_P, then any element of P_0 is given by:

$$x_{i,j} = x_{j,min} + \left(x_{j,max} - x_{j,min}\right) a_{RU} \qquad (5.14)$$

where a_{RU} is a random variable with uniform distribution for which $0 \leq a_{RU} \leq 1$. It is ensured that x_i, $i = \{m+1, m+2, ..., 2m\}$ is a positive integer. In case of a duplicate integer found this is replaced by a random integer value (respecting the upper and lower bounds) different from the calculated ones in P_0.

5.3.3.2 Selection and crossover

The stochastic universal sampling (SUS) is used as a selection function, which provides zero bias and minimum spread. SUS offers an offspring selection procedure that may lead to faster convergence to the solution of a problem than other selection methods, such as e.g. roulette wheel selection.

In addition, to avoid duplicate entries in the ground motion record identities a new crossover scheme is proposed which ensures that the linear combination of the ground motion records examined each time is comprised by unique members. This procedure is described by detail in the following:

If the crossover is performed between two random individuals at generation k, $P_{k,1} = \left\{x_{i1,j}\right\}$ and $P_{k,2} = \left\{x_{i2,j}\right\}$, the individual $P_{k+1,12}$ is produced as a result of the crossover. Initially, three set operations are performed between the two individuals:

a) Intersection between $x_{i1,j}$ and $x_{i2,j}$:

$$x_{1\cap2} = \left\{x_{i1,j}\right\} \cap \left\{x_{i2,j}\right\} \qquad (5.15)$$

b) Subtraction of $x_{i2,j}$ from $x_{i1,j}$:

$$x_{1-2} = \left\{x_{i1,j}\right\} - \left\{x_{i2,j}\right\} \qquad (5.16)$$

c) Subtraction of $x_{i1,j}$ from $x_{i2,j}$:

$$x_{2-1} = \left\{x_{i2,j}\right\} - \left\{x_{i1,j}\right\} \qquad (5.17)$$

The offspring $P_{k+1,12}$ will contain the intersection $x_{1\cap2}$ which contains $n_{1\cap2}$ elements and the vector $\left\{x_{1-2}, x_{2-1}\right\}_1$ which contains $l = m - n_{1\cap2}$ randomly selected elements from the vector formed by concatenating the two differences $\left\{x_{1-2}, x_{2-1}\right\}$:

$$P_{k+1,12} = \left\{ x_{1 \cap 2}, \left\{ x_{1-2}, x_{2-1} \right\}_1 \right\} \tag{5.18}$$

In the case where $x_{1 \cap 2} = \varnothing$ then $\left\{ x_{1-2}, x_{2-1} \right\}_1 = \left\{ x_{1-2}, x_{2-1} \right\}$. Eqs. (5.15) - (5.18) apply both for continuous and integer design variables of the problem.

5.3.3.3 Mutation

In GA, the mutation function uses various distributions from which random numbers (perturbations) are generated and added to the components of the individual that is mutated. In this chapter, the perturbation of the continuous/integer design variables, is performed using a Gaussian/random uniform distribution respectively and are described in detail below.

Continuous variables: The mutation function of continuous design variables follows a Gaussian distribution of zero-mean with standard deviation given by the relation:

$$m_{SC,k} = m_{SC,0} \left(1 - m_{SH} \frac{k}{k_{max}} \right) \tag{5.19}$$

where the standard deviation $m_{SC,k}$ is the fraction of the maximum range of possible perturbations of the design variables (i.e. scale factors) that can be added to an individual in generation k during mutation process. $m_{SC,0}$ is the scale parameter and is equal to the fraction of the maximum range of possible perturbations of the continuous variables at the initial generation (0), whereas m_{SH} is the shrink parameter which controls how fast $m_{SC,k}$ is reduced as generations evolved. Both of the parameters $m_{SC,0}$ and m_{SH} can be arbitrarily selected and their values must be between 0 and 1. $m_{SH} < 0$ or $m_{SH} > 1$ is also possible, but not recommended. For a random individual at generation k, $P_{k,1} = \left\{ x_{i1,j} \right\}$ this operation can be written as follows:

$$P_{k+1,1} = \left\{ x_{i1,j} \right\} + \overline{a}_{GU} m_{SC,k}, \ 1 \le j \le m \tag{5.20}$$

where $m_{SC,k}$ is given by Eq. (5.19) and $\overline{a}_{GU}$ is a vector with entries following a uniform Gaussian distribution.

Integer variables: The mutation function of integer design variables follows a random uniform distribution. Since the random perturbations are not integers in general, the result is rounded towards the nearest integer and then the remainder of its Euclidean division with M is extracted, to ensure that the result does not exceed M value. For a random individual at generation k, $P_{k,1} = \left\{ x_{i1,j} \right\}$ this operation can be written as follows:

$$P_{k+1,1} = \text{mod} \left(\left\langle \left\{ x_{i1,j} \right\} + \left(2\overline{a}_{RU} - 1 \right) m_{SC,k} \right\rangle, M \right), \ m+1 \le j \le 2m \tag{5.21}$$

where the symbol $\langle \ \rangle$ is used to denote the nearest integer of the quantity contained in the brackets, $\overline{a}_{RU}$ is a vector with entries following a uniform random distribution with $0 \leq \overline{a}_{RU,j} \leq 1$, $m_{SC,k}$ is the scale parameter of mutation function (standard deviation of Gaussian distribution at the kth generation), and $\mathrm{mod}(\ )$ denotes the modulo operation, i.e. the remainder of the Euclidean division of between the two arguments. After application of Eq. (5.21)the result is checked for duplicate values of integer components. If so, the duplicates are replaced by a random integer value (respecting the upper and lower bounds) different from the calculated ones.

5.4   Numerical results

The effectiveness of the proposed algorithm is verified by generation of artificial accelerograms which are compliant to target spectra of different inherent nature, ensuring also the independence of the algorithm's performance from the target spectrum. More specific, the acceleration and equivalent input energy velocity response spectra of three ground motion records: a) El Centro Terminal Substation Building record of the 1940 Imperial Valley earthquake, b) Rinaldi record of the 1994 Northridge earthquake and c) Sakarya – SKR record of the 1999 Kocaeli earthquake are defined as target spectra. The target spectra are associated with a far-field ground motion, a near-field ground motion which contains forward directivity effects and a near-field ground motion which contains fling-step effects, respectively (Kalkan and Kunnath, 2006). Typical characteristic of the near-field motions is the presence of high-velocity pulses, which do not exist in typical far-field ground motions. The difference between these two types of motions originates mainly from two factors: (a) the distance between the site where the earthquake is recorded and the seismic fault, (b) the orientation of the last. It is noted that the three target spectra have essentially different general configurations, a fact that results from the different inherent nature of the time histories of the three ground motions.

Two matching scenarios are considered: i) Matching Scenario 1 (Sa matching): Matching only the spectral acceleration and ii) Matching Scenario 2 (Sa–Vei matching): Matching both the spectral acceleration and the equivalent input energy velocity spectra (absolute and relative). In each scenario, the database is comprised of the ground motion records obtained from the European Strong Motion (ESD) database (Iervolino et al., 2010; Ambraseys et al., 2004). After a preliminary screening of the ESD database, a subset database is constructed that consists of 6026 ground motion records corresponding to horizontal earthquake components, i.e. $M = 6026$. The number m of ground motion records in the suite is set to be equal to 20 and the matching range of periods is between $T_1 = 0.1s$ and $T_2 = 4.0s$. The penalty constant k_p is set to be equal to 50.

Furthermore, the tuning parameters of the GA are selected as follows: the population size n_P (number of individuals in each generation) is equal to 80. For reproduction, the number of individuals that are guaranteed to survive to the next generation (elite children) is equal to 5% of the population size, namely $n_E = 0.05 n_P = 4$, and the fraction of the next generation, other than elite children, that is produced by crossover (crossover fraction) is equal to 0.8, i.e. $n_C = 0.8(n_P - n_E) \approx 61$ individuals are produced in each generation. The number of individuals in each generation that are produced by mutation is $n_M = n_P - n_E - n_C = 15$. In the GA used in this chapter no migration occurs, as there are no subpopulations. As stopping criteria for the GA algorithm the maximum number of generations (MAXGEN) is used, i.e. equal to 100. A sensitivity analysis of 30 independent optimization runs is also performed followed by a statistical process on the optimized results. The sensitivity analysis represents a necessary step since the GA optimization procedure does not yield the same results when restarted due to its stochastic nature.

In all cases examined, the objective function is evaluated using OpenSeismoMatlab, an open source tool for earthquake ground motion processing (Papazafeiropoulos and Plevris, 2018). OpenSeismoMatlab performs baseline correction and generates the elastic acceleration and equivalent input energy velocity response spectra which are then used for the calculation of the objective function.

5.4.1   Matching scenario 1

The optimization results for Matching Scenario 1 are depicted in Figure 5-1. For each target record, the black curve represents the target acceleration spectrum, while the red and blue curves represent the spectral acceleration that corresponds to the optimization run (out of the 30 runs) that fits best and worst to the target spectrum, respectively. The coefficient of variation (CoV) of the 30 runs for each period is also depicted by the green curve.

(a)   (b)

(c)
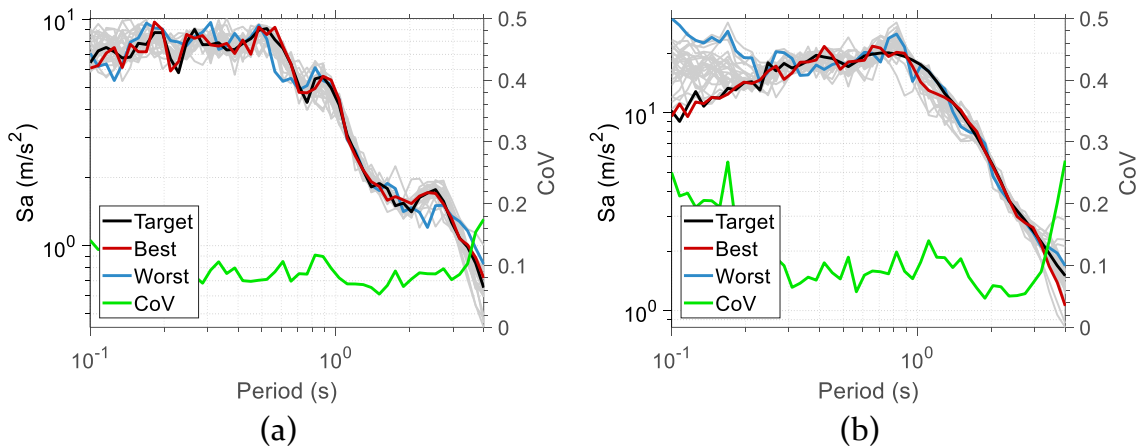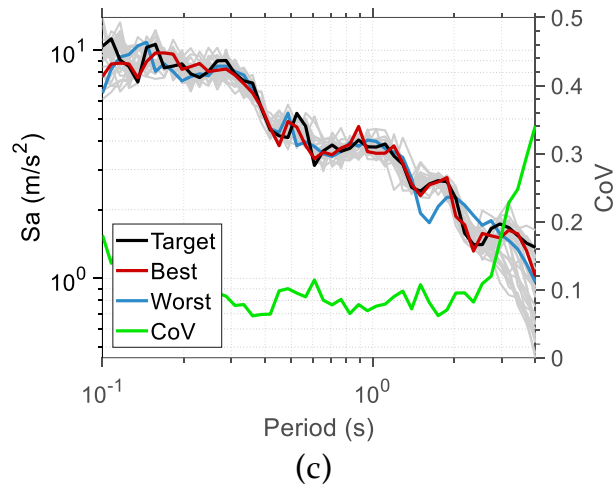
Figure 5-1: Optimization results of Matching Scenario 1 for each target: (a) El Centro, (b) Northridge and (c) Sakarya.

A good agreement is observed between the "best" and "target" spectra in all cases examined while the CoV value increases near the bounds of the matching period range. This is mostly attributed to the range of the periods involved in the calculation of the objective value (see Eqs. (9) and (10)) which is defined in a way that it covers the eigenperiods of a structure. This means that the period range used in the matching procedure and consequently the optimized acceleration time history are period-dependent. In this chapter, an extended period range is selected to highlight the applicability of the proposed methodology for a variety of structures. However, most of civil structures have eigenperiods that are concentrated near the middle of the range considered, where the CoV values are minimum and high accuracy can be achieved. Furthermore, the finite number of ground motions in the suite of the linear combination contributes to large CoV values in general. As the number of the ground motions in the suite decreases, the methodology becomes more cumbersome, since the time history given by the suite has less flexibility. Hence, as the number of the ground motions increases, the matching becomes generally better. Finally, the shape of the penalty function in Eq. (11) has an important effect on the optimized response spectrum of each optimization run, since the weighting of the deviation from the target spectrum for the matching period range considered is not uniform, as has been already mentioned in the previous section.
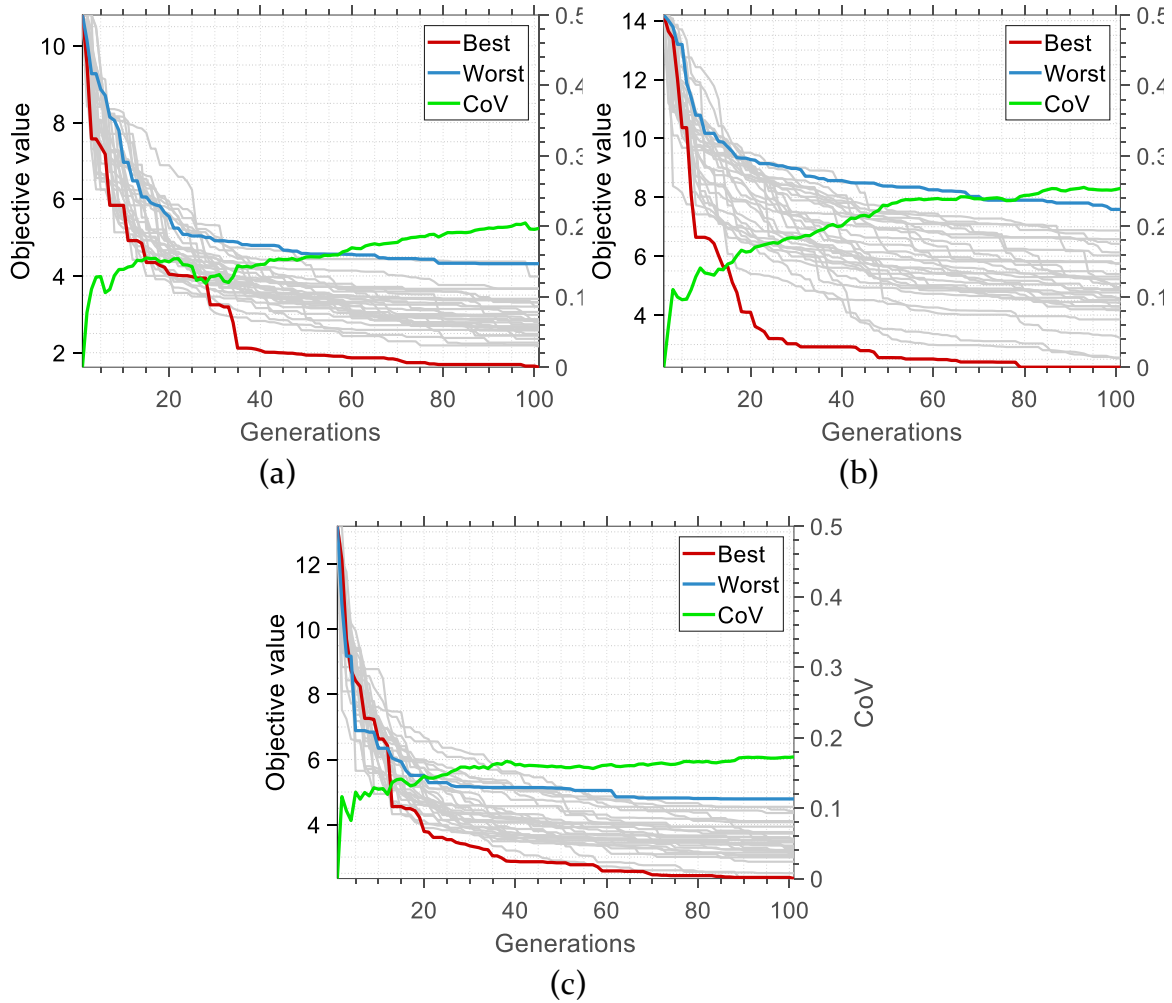
Figure 5-2: Optimization history of the 30 independent runs of the Matching Scenario 1 for each target: (a) El Centro, (b) Northridge and (c) Sakarya.

Figure 5-2 shows the convergence history of the 30 independent optimization runs of Matching Scenario 1. Each curve represents the objective value of the best individual at each generation of a given optimization run. The red (blue) curve represents the evolution of the objective value that corresponds to the optimization run (out of the 30 runs) that fits best (worst) to the target spectrum. It can be noted that in the case of El Centro earthquake the best individual of the final generation for the best independent run corresponds to roughly 14% of the objective value of the best individual of the initial generation. The best individual of the final generation for the worst independent run corresponds to roughly 40.3% of the objective value of the best individual of the initial generation. In the case of Northridge earthquake these percentages are roughly equal to 16.1% and 53.6% respectively, and in the case of Sakarya earthquake they are 17.7% and 36.9% respectively.

The trend of all convergence histories shows that the approach to the optimum value is quick and relatively smooth, which is achieved by proper adjustment of the crossover and mutation rates, in order to ensure sufficient

population diversity in each generation. It seems that, while the coefficient of variation among the optimization histories increases at the early stages of the optimization process, there is a point after which it stabilizes until termination. The magnitude of the final stabilized value of the CoV value is a measure of the complexity of the optimization space. As it is expected, larger CoV values corresponds to increased diversity between the various optimization runs, in terms of the path followed by the best individual of each optimization run. The largest CoV value of the objective value of the best individual among the various optimization runs at the final generation occurs in the case of Northridge earthquake, an observation that correlates well with the large dispersion of the optimum spectra, especially in the low period range, in Figure 5-1b.

5.4.2   Matching scenario 2

The optimization results for Matching Scenario 2 are depicted in Figure 5-3. Nearly the same traits that are mentioned for Figure 5-1 are observed; the proposed algorithm gives higher CoV values in the lower and higher limits of the matching period range considered.
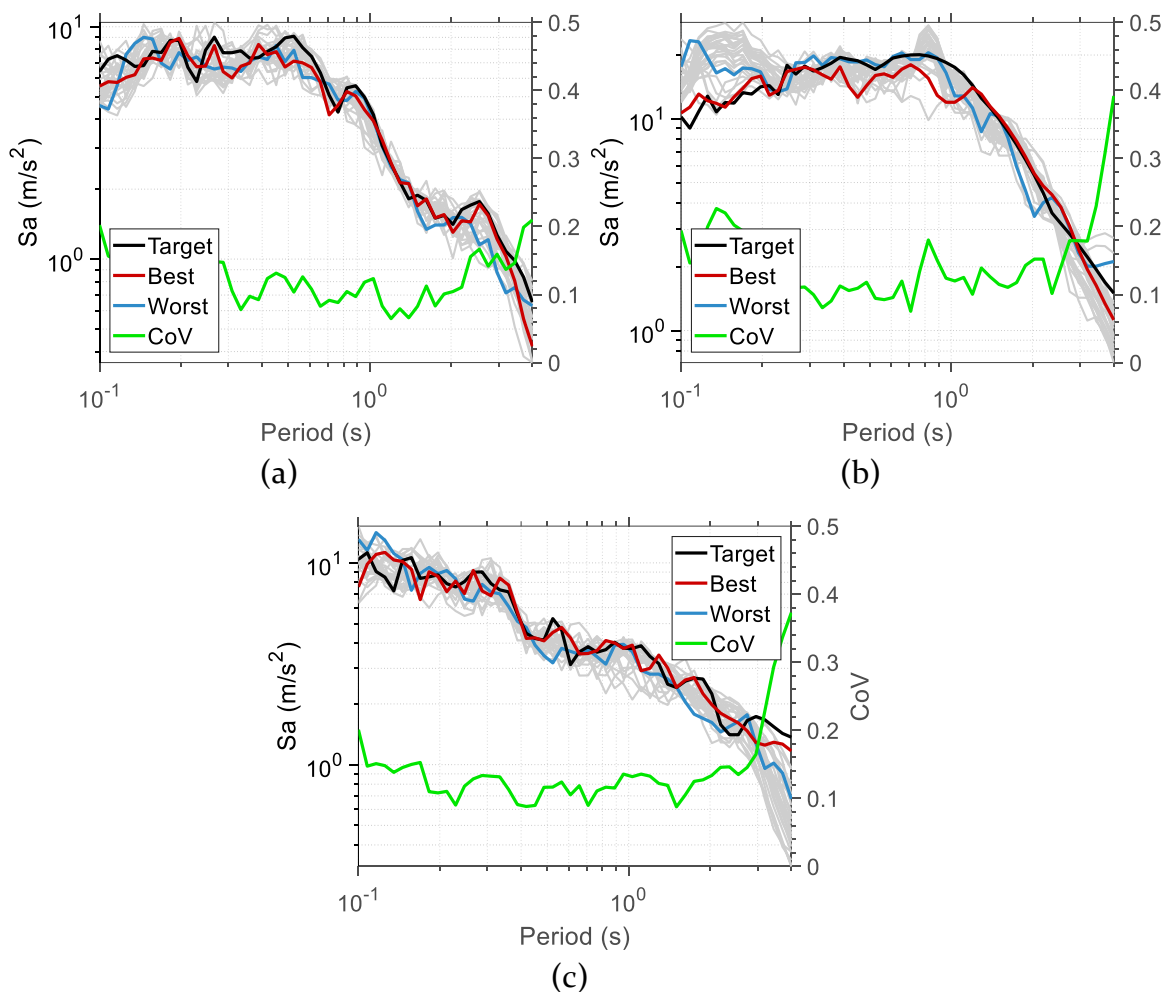
Figure 5-3: Results of Matching Scenario 2 regarding spectral acceleration for each target: (a) El Centro, (b) Northridge and (c) Sakarya.

In Figure 5-4 and Figure 5-5, the absolute seismic input energy equivalent velocity (SievABS) and the relative seismic input energy equivalent velocity (SievREL) spectra for each target spectrum are presented respectively. A very close agreement between the target and corresponding optimized spectra is also observed in this case. Although the CoV plots exhibit local peaks and troughs, all of them fluctuate around the value of 10%, regardless of the target spectrum.

In a similar rationale, Figure 5-6 depicts the convergence history of the 30 independent optimization runs of Matching Scenario 2. It is apparent that in the case of El Centro earthquake the best optimization run gives result equal to 32.8% of the best objective value of the initial population, whereas the worst result is roughly equal to 48.3% of the initial best objective value. In the case of Northridge earthquake the best and worst results are roughly equal to 32% and 55.7% respectively of the initial best objective value. Similarly, the corresponding percentages for the Sakarya earthquake are 26% and 41.1%. Interestingly, the lowest (best) percentage appears in the case of Sakarya earthquake whereas the highest (worst) percentage appears in the case of Northridge earthquake. The smooth convergence in optimization histories demonstrates the reliability of the proposed algorithm not only for matching the target spectral acceleration, but also for matching both target acceleration and target seismic input energy equivalent velocity spectra.

(a)   (b)

(c)
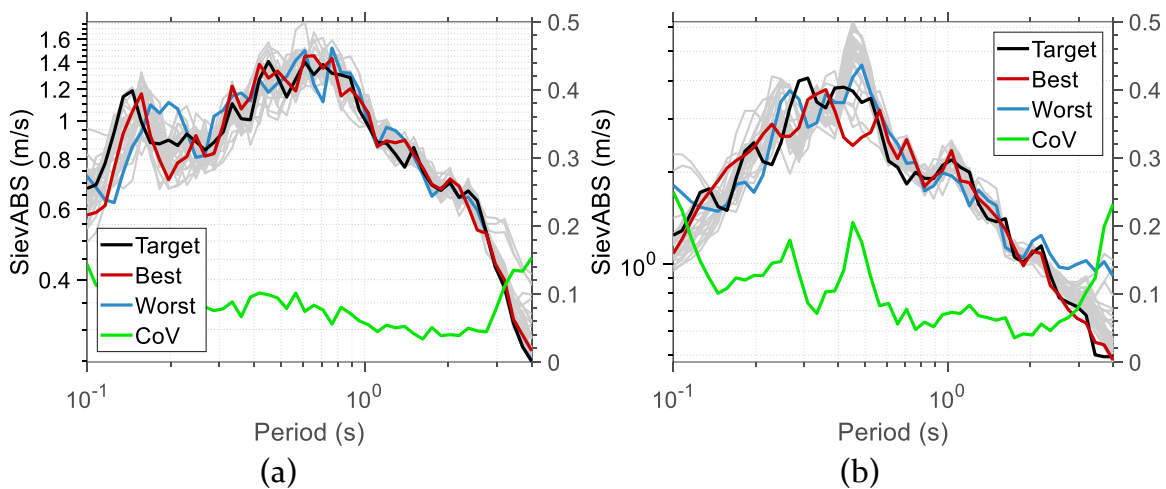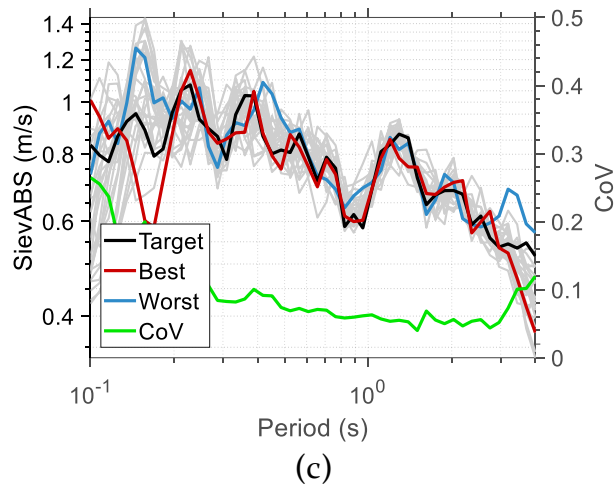
Figure 5-4: Results of Matching Scenario 2 regarding equivalent absolute seismic input energy velocity spectra (SievABS) for each target: (a) El Centro, (b) Northridge and (c) Sakarya.

(a)

(b)

(c)
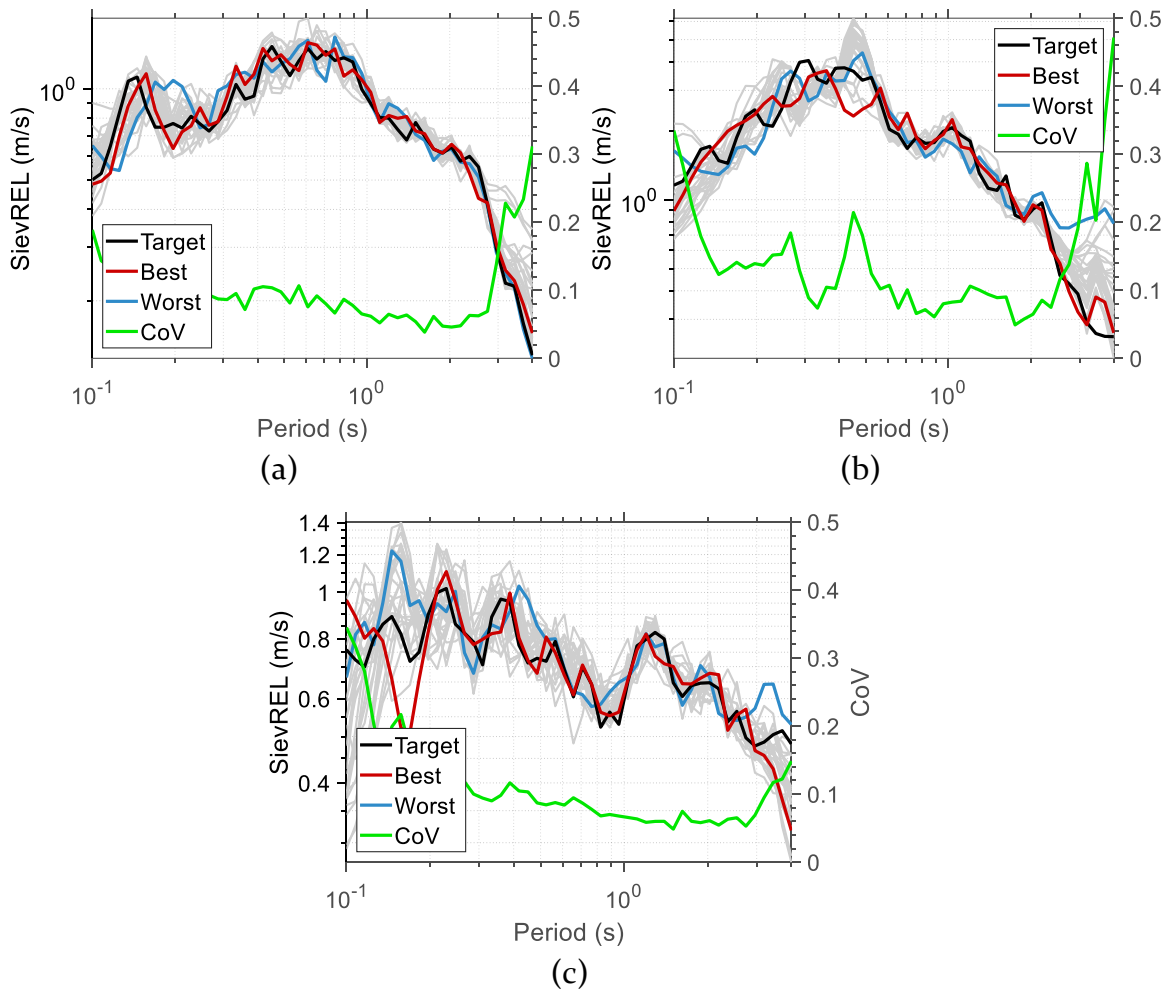
Figure 5-5: Results of Matching Scenario 2 regarding equivalent relative seismic input energy velocity spectra (SievREL) for each target: (a) El Centro, (b) Northridge and (c) Sakarya.
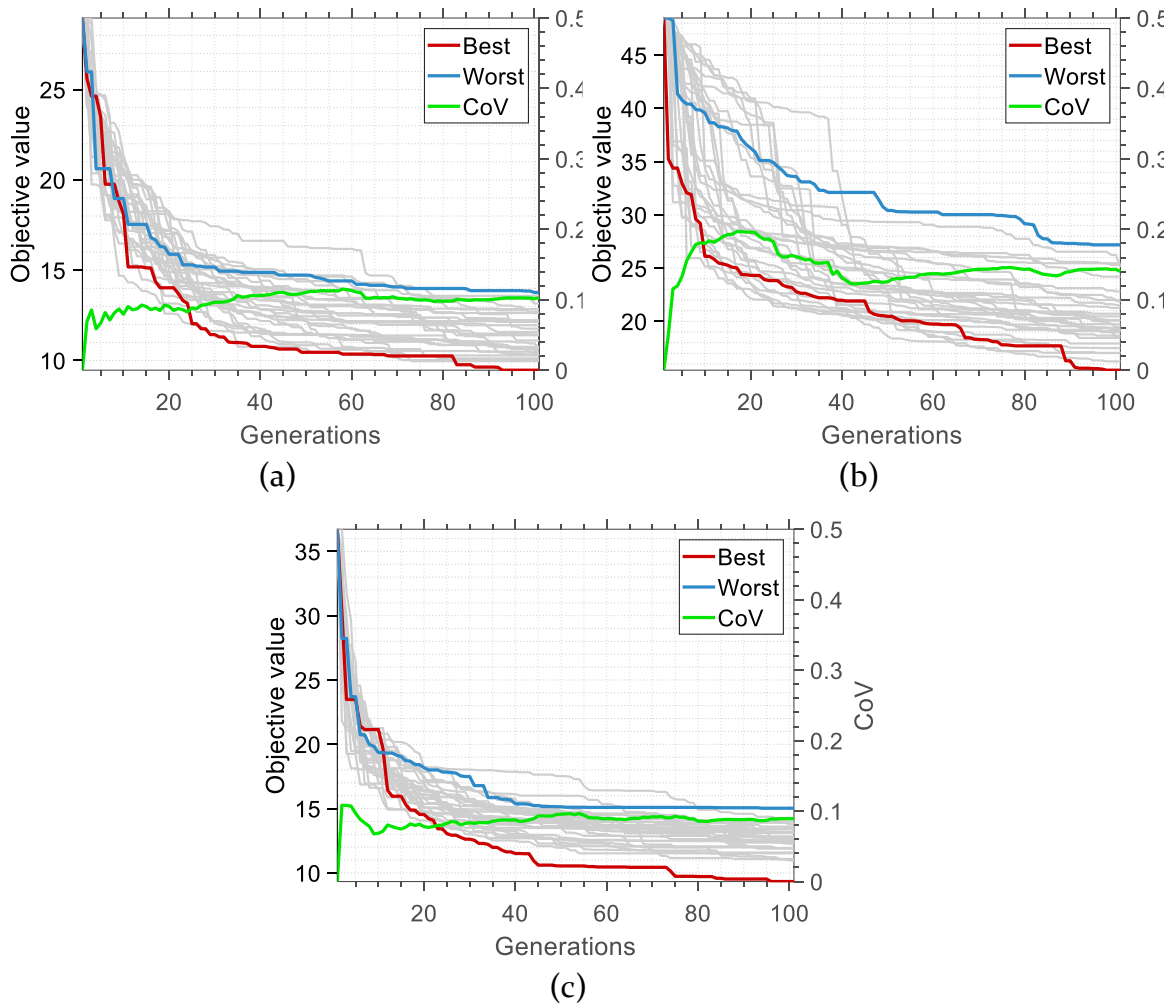
Figure 5-6: Optimization history of the 30 independent runs of the Matching Scenario 2 for each target: (a) El Centro, (b) Northridge and (c) Sakarya.

5.4.3 Comparison of the two scenarios

A one-to-one comparison between the performance of the two scenarios shows that the CoV is generally higher in Scenario 2. This occurs because the optimization problem of Scenario 1 is more "relaxed" than the Scenario 2. In Scenario 1, the objective function is related only with a single target spectrum (acceleration), while in Scenario 2 the objective function is related with three target spectra (acceleration, absolute velocity, relative velocity), at the same time. This relation establishes an indirect "constraint" which implies that, with respect to the target acceleration spectrum only, the optimized solution of Scenario 2 will have higher deviation than that of Scenario 1, which interprets the higher CoV values in Figure 5-3 when compared to Figure 5-1. Consequently, in the case of Scenario 2 the possible "paths" of the population evolution towards the optimum are far fewer and therefore the population diversity is lower compared to Scenario 1, which explains the reduced CoV in the last generation in Figure 5-6 (Scenario 2), compared to that in Figure 5-2 (Scenario 1). Finally, it is noted that

as the generations increase, the CoV fluctuation is smoother in the case of Scenario 2, related to the increased robustness of the algorithm in this case.

5.5   Verification of the proposed methodology

In order to assess the robustness and accuracy of the proposed methodology the seismic performance of single- and multi- degree of freedom structural systems has been considered. To this end, nonlinear response history analyses were conducted for the optimized accelerograms of the two Matching Scenarios as resulted for the three target ground motion records in Section 3. The response results are compared in terms of the goodness-of-fit with the respective response result of the target ground motion. The seismic input energy that is dissipated due to viscous damping action in the structure (damping energy) is also quantified.

5.5.1   Energy definitions

The seismic input energy that is absorbed by an inelastic single degree of freedom (SDOF) structural system during an earthquake can be defined by integrating the equation of motion of the system as follows:

$$\int_0^u \ddot{u}\bar{m}du + \int_0^u \dot{u}\bar{c}du + \int_0^u f_s du = -\int_0^u \bar{m}\{I\}\ddot{u}_{g,c}du \tag{5.22}$$

where $\bar{m}$ is the mass matrix, $\bar{c}$ is the viscous damping coefficient matrix, f_s is the resistance force due to stiffness, I is the unit influence vector of the structure and $\ddot{u}_{g,c}$ is the linear combination of the accelerations of ground motions records in the suite as defined in Section 2.1. Eq. (5.22) stands as a statement of energy balance of the system and can be rewritten as:

$$E_k(t) + E_d(t) + E_s(t) + E_y(t) = E_I(t) \tag{5.23}$$

With regard to Eq. (5.22) the first integral gives the kinetic energy E_k, the integral on the right-hand side gives the input energy E_I imparted from the ground motion to the structure and the last integral on the left-hand side is equal to the sum of the linear elastic recoverable strain energy E_s and the plastic irrecoverable strain energy E_y. The damping energy term E_d is defined as follows:

$$E_d(t) = \int_0^u \dot{u}\bar{c}du \tag{5.24}$$

The definitions of the aforementioned energy quantities are given for a structure whose mass is acted upon by a force equal to $p_{eff}(t) = -\bar{m}\{I\}\ddot{u}_{g,c}$, i.e. they are based on the consideration of the structural motion relative to the base, rather

than the total motion of the structure. The two types of energy formulations (relative and absolute) are equivalent but the former is more intuitive and simplifies the calculations when it comes to multi degree of freedom (MDOF) structural systems. Eqs (5.22) to (5.24) correspond to a SDOF system in mathematical terms and their extension to MDOF systems can be done in a straightforward manner.

5.5.2   SDOF system results

Three SDOF systems involving a bilinear elastoplastic constitutive model with kinematic hardening are analyzed for each target ground motion. The eigenperiod, the critical damping ratio, the post-yield stiffness ratio (i.e. the ratio of the post-yield stiffness to the initial small strain stiffness of the structure), and the ductility demand are same for all the SDOF systems and equal to 0.5 sec, 5%, 1% and 1.1 respectively. The three systems have different yield displacements, equal to 0.052 m, 0.1 m and 0.025 m for the El Centro, the Northridge and the Sakarya target ground motion, respectively. The reader is referred to Papazafeiropoulos et al. (2017) for more details about the implementation of the bilinear elastoplastic constitutive model with kinematic hardening and the time integration algorithm that were used in this chapter.

The small ductility value specified for all target ground motions denotes that structures only with slightly nonlinear behavior are considered in this chapter; for cases of severely nonlinear response the scenarios presented in this chapter for calculation of the design artificial ground motion is an open research issue. For such cases it would be better to consider the inelastic response spectra, rather than elastic response spectra in matching scenarios. In addition, the physical properties of each SDOF system remain the same for the estimation of its dynamic response for each target ground motion as well as the optimized ground motions obtained from the two matching scenarios. Based on an arbitrarily selected value of ductility demand (equal to 1.1, to ensure a slightly nonlinear response) for each target ground motion the yield displacement that is calculated was used also for the corresponding optimized ground motions obtained from the two matching scenarios in all nonlinear time history response analyses.

In Figure 5-7, the time variation of the damping energy per unit mass for each target motion and the optimized ground motion records produced from the two matching scenarios is depicted. A good agreement is observed in all cases since the damping energy of the optimized ground motion (red line) is very close to that of the respective target ground motion (black line). To quantify this agreement, the normalized error e_{ij} for the ith story (in the case of SDOF systems i is always equal to 1) and jth matching scenario, which is proportional to

the area between a matching scenario and the target ground motion curves was used as a metric of this goodness-of-fit, defined as follows:

$$e_{ij} = \frac{\int_0^{T_d} \left| E_{d,i}^j(t) - E_{d,i}^T(t) \right| dt}{\int_0^{T_d} E_{d,i}^T(t) dt} \tag{5.25}$$

where $E_{d,i}^j$ and $E_{d,i}^T$ is the damping energy for the jth scenario and the target ground motion respectively.
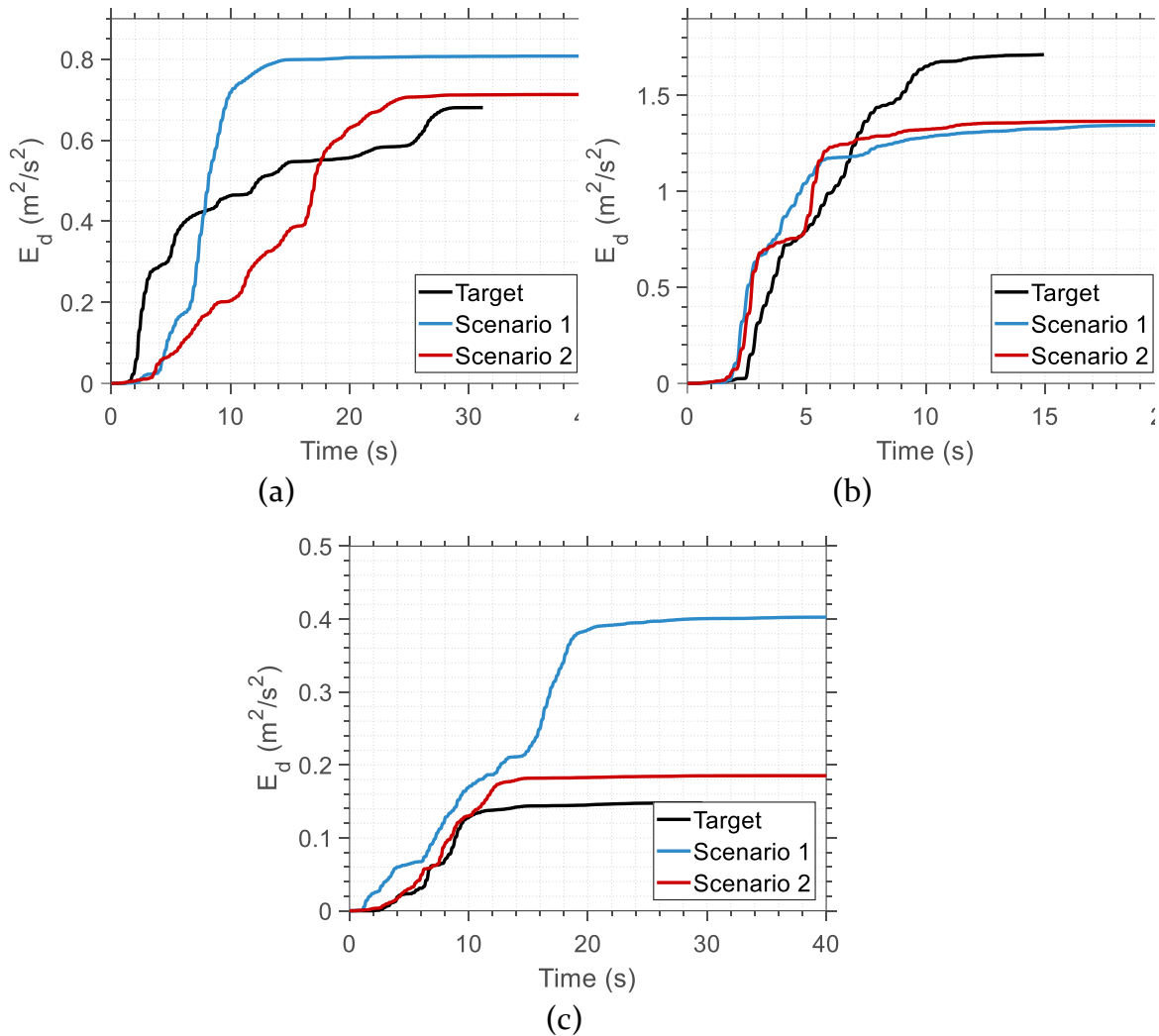
Figure 5-7: Energy dissipated by viscous damping per unit mass over time for the optimized artificial ground motions of the two matching scenarios and for each target ground motion: (a) El Centro, (b) Northridge and (c) Sakarya.

Even in the case of Northridge target ground motion, it is indicative that the damping energy corresponding to Scenario 2 is slightly closer to the respective curve of the target motion, although there is not much difference between the two scenarios (17% as seen in Table 2). This fact, in combination with the large

value of the dissipated energy per unit mass may be a consequence of the special characteristics of Northridge earthquake, which contains a high velocity pulse (forward directivity effect) as a near-field ground motion.

Target ground motion	Matching Scenario 1	Matching Scenario 2	Difference (%)
El Centro	0.4065	0.2794	31.3
Northridge	0.2289	0.1901	17.0
Sakarya	1.2104	0.2310	80.9

Table 5-1: Normalized error of the damping energy between the optimized and the target ground motion records.

5.5.3 MDOF system results

Two model buildings were analyzed as a 3-DOF and 9-DOF structural systems. More specific, the model buildings are a 3-story (LA3) and a 9-story building (LA9) designed as standard office buildings and situated on a stiff soil (soil type S2), following the local code requirements for the Los Angeles city (UBC 1994), and according to the provisions of the FEMA/SAC project, presented in FEMA-355C (2000). The plan and elevation of their effective structural models, along with the various cross sections of its members are shown in Figure 5-8. The perimeter moment-resisting frames act as the structural system of the building. The column bases of the moment resisting frames are considered as fixed. Furthermore, the design of the buildings for the two orthogonal directions is quite similar, and therefore only half of the structure is considered in the analysis in each case.
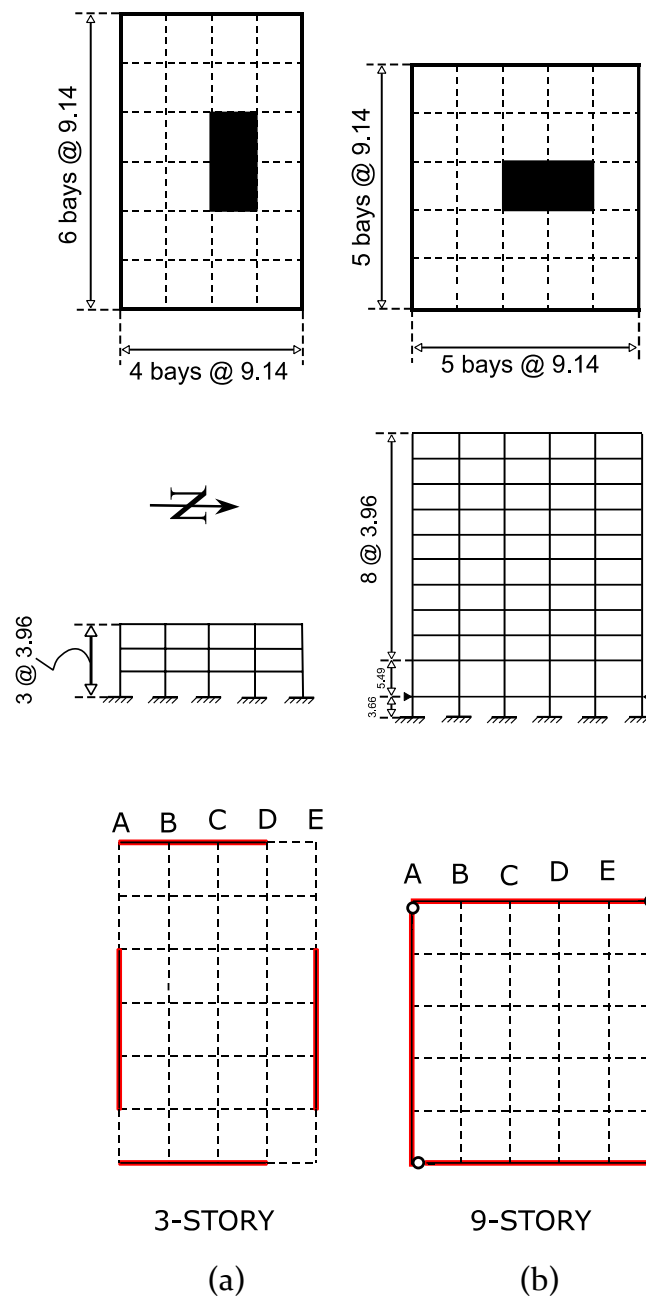
Figure 5-8: Plan view, typical floor and elevation view of the effective model of the (a) LA3 and (b) LA9 model buildings.

The benchmark buildings are simulated as a 3-DOF and a 9-DOF structural system involving the same bilinear elastoplastic constitutive model with kinematic hardening, as in the SDOF system analyzed previously. Their fundamental eigenperiods are equal to 1.01 sec and 2.85 sec, respectively. The post-yield stiffness ratio and critical damping ratio were set equal to 1% and 5%, respectively. The yield displacement and ductility demand of each story for the 3-DOF and 9-DOF structural systems are shown in Table 3. The maximum ductility

at any story does not exceed the value of 2. Usually, an interpolative iterative procedure is necessary to obtain the yield displacement for a target ductility demand (Chopra, 2017). However, for each target ground motion in each building the yield displacement is assumed as uniform distributed across all storeys and is calculated so that the maximum ductility demand is equal to 2 at least in one story of the building. For both of the LA3 and LA9 buildings the maximum ductility demand is observed at the first story. The ductility of the remaining storeys is much lower or even lower than 1 (i.e. story remains linear elastic).

Target Ground Motion			El Centro	Northridge	Sakarya
LA3	Yield displacement [m]		0.0283	0.1681	0.0356
	Ductility demand	Story 2	1.19	0.99	0.99
		Story 3	0.62	0.48	0.49
LA9	Yield displacement [m]		0.0685	0.166	0.1193
	Ductility demand	Story 2	0.41	0.38	0.38
		Story 3	0.44	0.41	0.38
		Story 4	0.39	0.39	0.30
		Story 5	0.51	0.58	0.30
		Story 6	0.47	0.61	0.25
		Story 7	0.52	0.74	0.28
		Story 8	0.36	0.53	0.20
		Story 9	0.18	0.26	0.10

Table 5-2: Yield displacement and ductility demand of each story for the 3-DOF and 9-DOF structural systems

For each target ground motion, three nonlinear response history analyses were conducted using as excitation the target ground motion and the two optimized ground motions resulting from the two matching scenarios. Figure 5-9-Figure 5-11 show the time history of the damping energy at the three storeys of the building for each target ground motion and the optimized ground motion records. Again, a good agreement is observed in all cases since the damping energy of the optimized ground motion (red line) is very close to that of the respective target ground motion (black line).
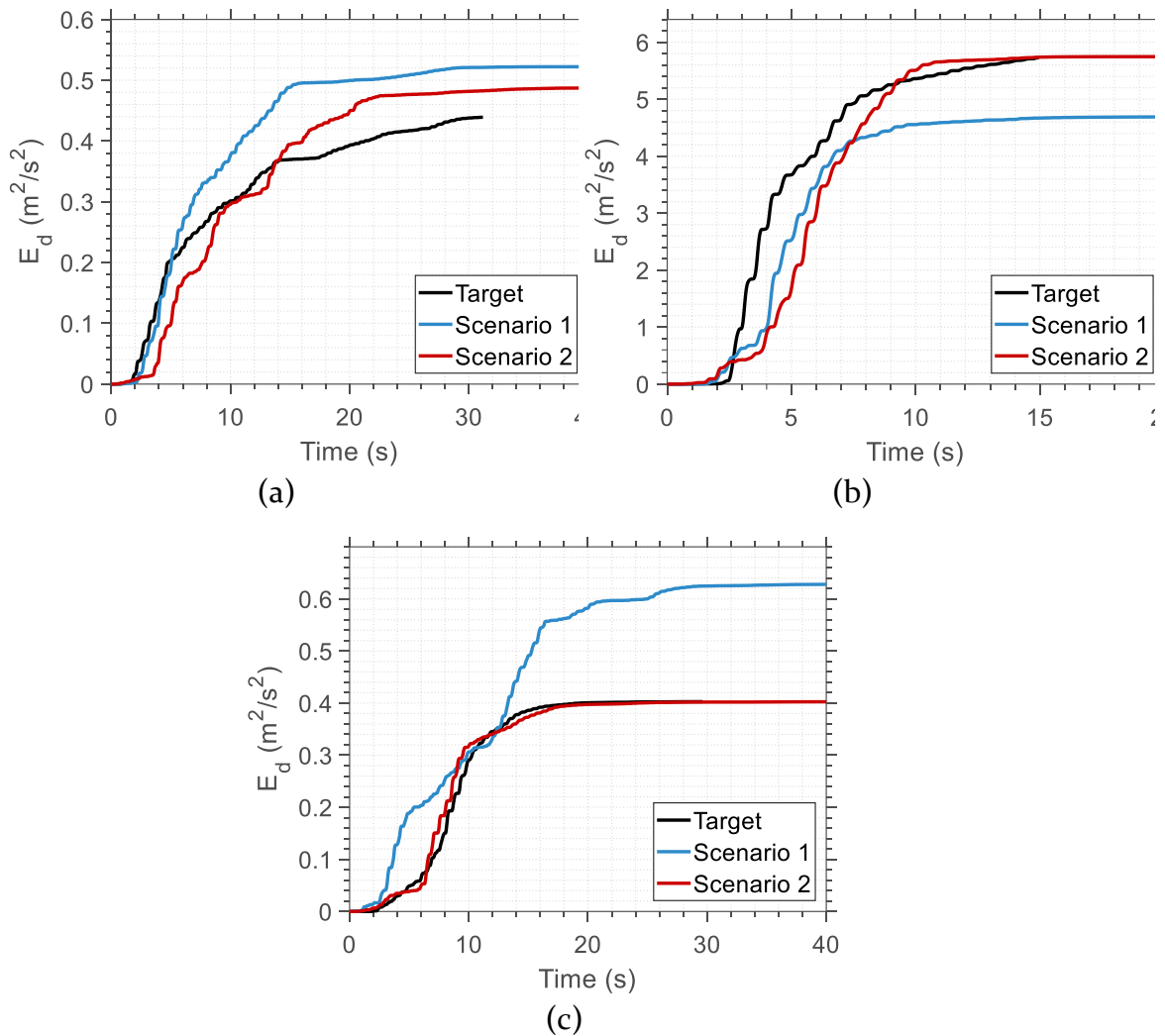
Figure 5-9: Time variation of energy dissipated at the 1st story of the 3-DOF system for the optimized and the target ground motion records: (a) El Centro, (b) Northridge and (c) Sakarya.

(a)

(b)

(c)
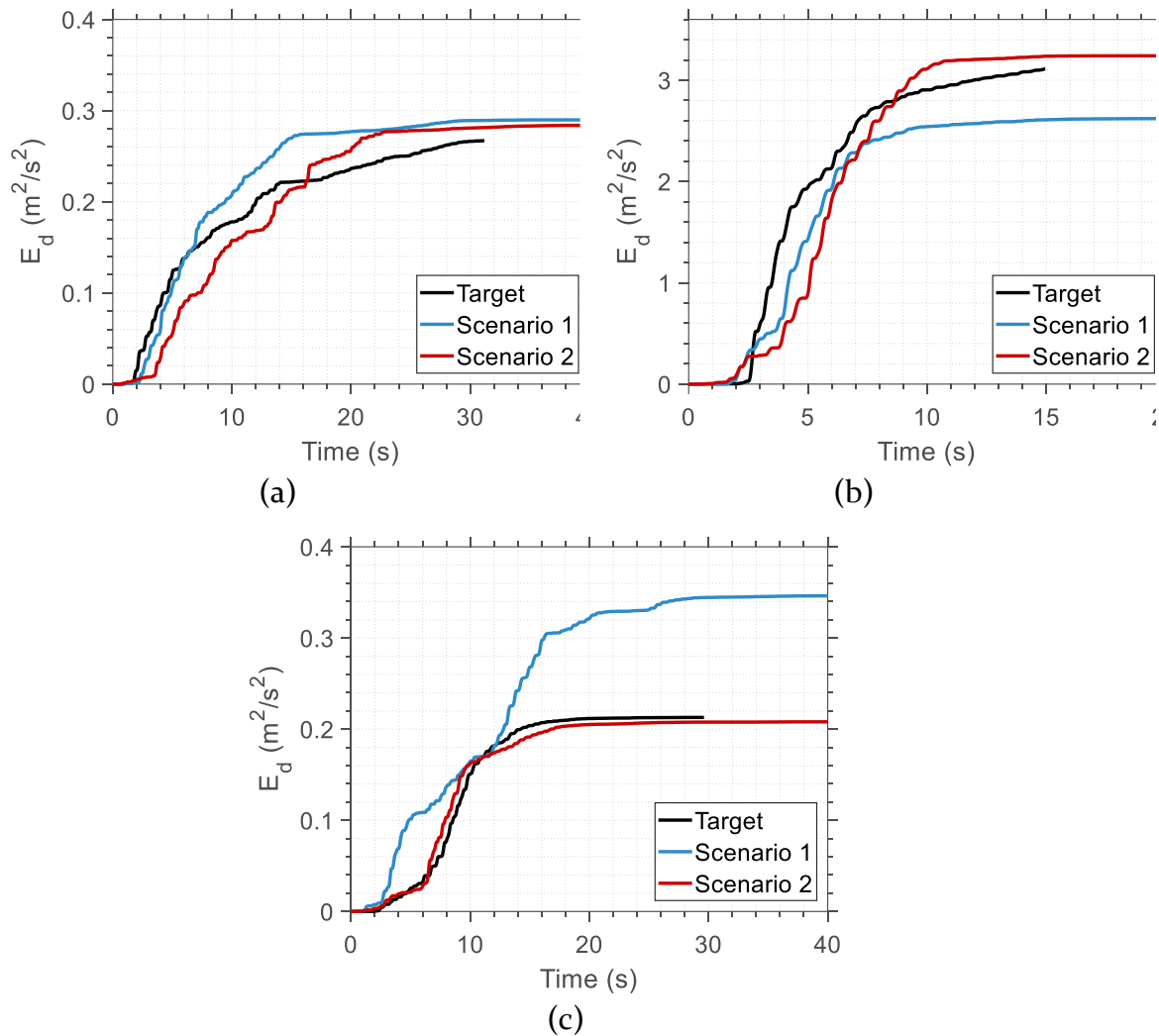
Figure 5-10: Time variation of energy dissipated at the 2nd story of the 3-DOF system for the optimized and the target ground motion records: (a) El Centro, (b) Northridge and (c) Sakarya.
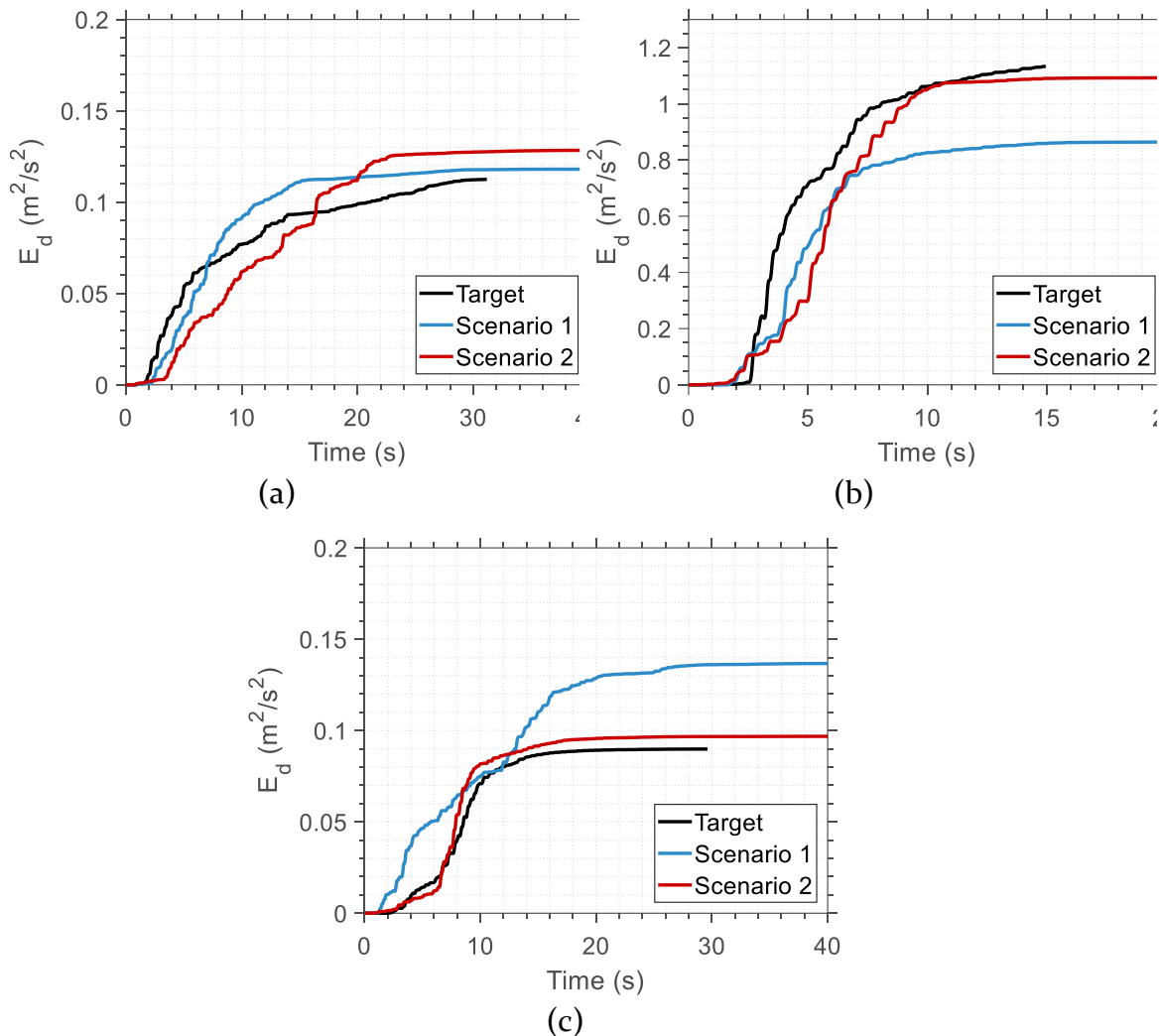
Figure 5-11: Time variation of energy dissipated at the 3rd story of the 3-DOF system for the optimized and the target ground motion records: (a) El Centro, (b) Northridge and (c) Sakarya.

To quantify this agreement the normalized error as defined in Eq. (5.25) was used as a metric of this goodness-of-fit. Figure 5-12 shows the normalized error of the damping energy between the optimized and the target ground motion records for each floor of the 3-DOF structural system. The min/max errors for the two scenarios are 15%/48% and 3%/20%, respectively. It is observed that the proposed algorithm (Scenario 2) yields far lower error compared to Scenario 1. Although the error of Scenario 2 remains lower, only in the case of the dynamic response of the third floor of the 3-DOF system for the El Centro target motion Scenario 2 gives greater error compared to Scenario 1 (28% higher). It is worth noting that in the case of the Sakarya target ground motion the error of the Scenario 2 is 78.3% lower compared to Scenario 1. This is directly related with the low CoV values observed in Section 3 for this specific case, a fact that also proves the robustness and accuracy of the proposed methodology.
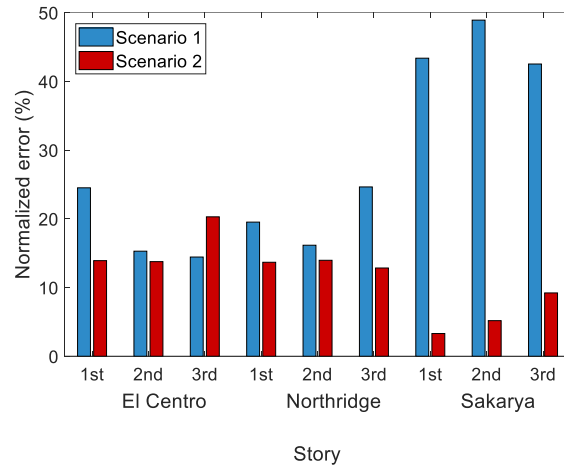
Figure 5-12: Normalized error of the damping energy between the optimized and the target ground motion records for each floor of the 3-DOF structural system.

Figure 5-13 shows the time history of the damping energy at a typical story (i.e. first story) of the LA9 building for each target ground motion and the optimized ground motion records. Again, a good agreement is observed in all cases since the damping energy of the optimized ground motion is very close to that of the respective target ground motion. To quantify this agreement, Figure 5-14 shows the normalized error of the damping energy between the optimized and the target ground motion records for each story of the 9-DOF structural system. The min/max errors for the two scenarios are 8.2%/88.7% and 9.8%/38.5%, respectively. It is observed that the proposed algorithm (Scenario 2) yields far lower error compared to Scenario 1. The error of Scenario 2 remains higher, only in the case of the dynamic response of the upper storeys of the 9-DOF system for the Northridge target motion. This deviation is attributed to the dynamic characteristics of the structural system mainly affected by the near field effects of the specific ground motion. It is worth noting that the maximum error of the Scenario 1 is 130.4% higher compared to the corresponding maximum error of the Scenario 2.
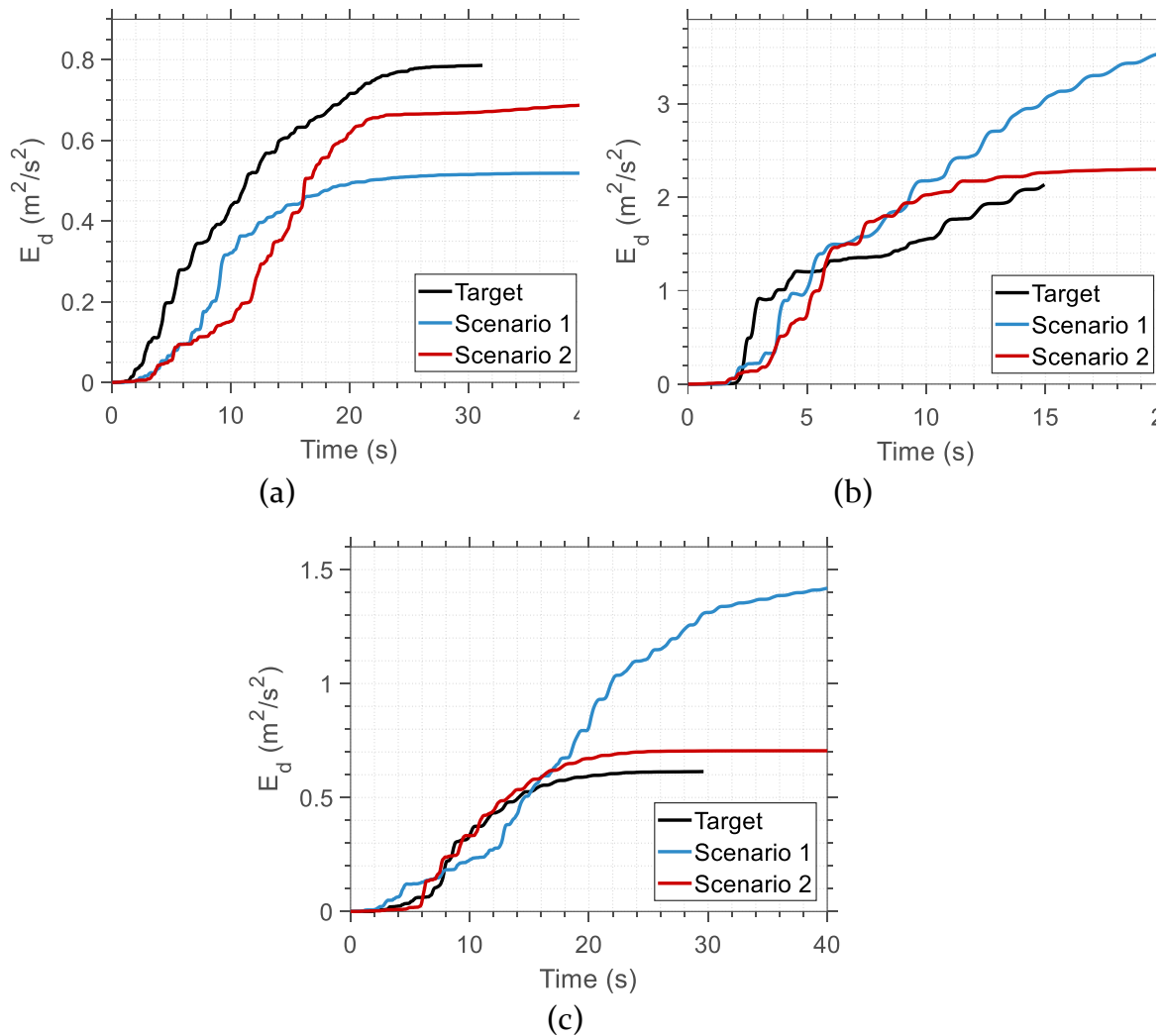
Figure 5-13: Time variation of energy dissipated at the 1st story of the 9-DOF system for the optimized and the target ground motion records: (a) El Centro, (b) Northridge and (c) Sakarya.

(a)   (b)

(c)
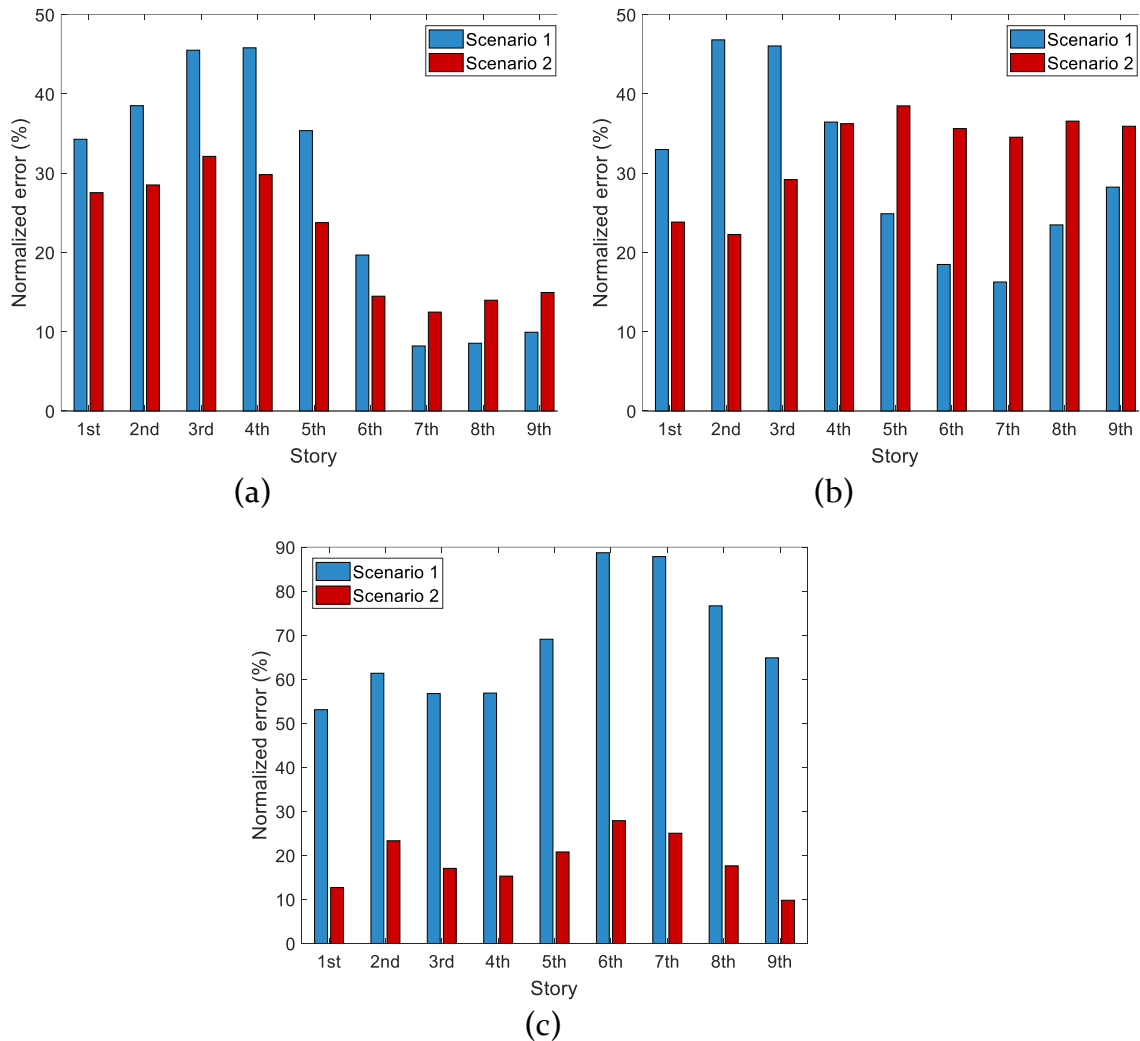
Figure 5-14: Normalized error of the damping energy between the optimized and the (a) El Centro, (b) Northridge and (c) Sakarya target ground motion records for each floor of the 9-DOF structural system

5.6   Conclusions

In this chapter a novel spectra-matching framework is developed, which employs a linear combination of raw ground motion records to generate artificial accelerograms. To this end, apart from the well-known design acceleration spectrum that is prescribed by the various norms and guidelines, the seismic input energy equivalent velocity spectrum is also taken into account.

This consideration is leading therefore to optimized acceleration time histories, which represent actual motions in a much more realistic way. In order to produce elastic spectra that match as closely as possible to a given target spectrum, the procedure of selection and scaling of a suite of ground motion records to fit a given target spectrum is formulated as an optimization problem. Three characteristic ground motion records of different inherent nature are selected as

target spectra, to verify the effectiveness of the proposed algorithm, ensuring that its performance is not ground motion record-dependent assuming different matching scenarios. The optimization results have shown that there exists a good agreement between the target and optimized spectra for each case examined, regardless of the nature of target spectrum. Finally, it is proved that the artificially generated records are much more realistic and suitable for the seismic design of structures, since they reproduce better the real nonlinear structural inelastic response in terms of the damping energy, demonstrating also the reliability and robustness of the proposed methodology.

5.7   Notation

A : amplitude

a_{RU} : scalar variable with uniform random distribution with $0 \leq a_{RU} \leq 1$

$\bar{a}_{GU}$: vector with entries following a uniform Gaussian distribution

$\bar{a}_{RU}$: vector with entries following a uniform random distribution with $0 \leq \bar{a}_{RU} \leq 1$

$\bar{c}$: viscous damping coefficient matrix

$DFT()$: Discrete Fourier Transform

E_d : Energy dissipated due to damping

E_k : Kinetic energy

E_I : Input energy due to earthquake

E_s : Elastic recoverable strain energy

E_y : Energy dissipated due to yielding

f : frequency

f_i : fitness value of the i -th individual

f_s : force due to stiffness

$FFT()$: Fast Fourier Transform

I : unit column vector (influence vector)

I_0 : zeroth – order modified Bessel function of the first kind

$IFFT()$: Inverse Fast Fourier Transform

k : number of generation

k_{max} : maximum number of generations

k_p : constant for penalty of lower period bound

k_S : positive integer for selection function with $0 \leq k_S \leq n_S - 1$

$\bar{k}$: small strain (initial) stiffness matrix

M : number of raw accelerograms that are contained in the earthquake data base

m : number of raw accelerograms combined to produce the artificial time history

$m_{SC,0}$: scale parameter of mutation function (standard deviation of Gaussian distribution at the first generation) at the initial generation (0)

$m_{SC,k}$: scale parameter of mutation function (standard deviation of Gaussian distribution at the first generation) at generation k

m_{SH} : shrink parameter of mutation function (rate of decrease of standard deviation w.r.t. generation number)

$\bar{m}$: mass matrix

N_{FIR} : order of FIR filter

n : length of the Fourier transform

$n_{1\cap2}$: number of elements of the intersection $X_{1\cap2}$

n_C : number of individuals in each generation produced by crossover

n_E : number of elite individuals in each generation

n_M : number of individuals in each generation produced by mutation

n_P : population size

n_S : number of individuals which are selected for breeding in each generation

P_0 : population at zeroth generation (initial)

P_k : population at generation k

$P_{k,1}$: first random individual belonging to population at generation k

$P_{k,2}$: second random individual belonging to population at generation k

$P_{k+1,12}$: offspring from crossover between $P_{k,1}$ and $P_{k,2}$

$p(T)$: penalty function

p_i : probability of selection of the ith individual

r : rank of an individual

Sa_c : spectral acceleration of the linear combination of the selected ground motions

Sa_t : target spectral acceleration

$Siev_c^{ABS}$: spectral equivalent absolute input energy velocity of the linear combination of the selected ground motions

$Siev_c^{REL}$: spectral equivalent relative input energy velocity of the linear combination of the selected ground motions

$Siev_t^{ABS}$: target spectral equivalent absolute input energy velocity

$Siev_t^{REL}$: target spectral equivalent relative input energy velocity

T : eigenperiod

T_1 : lower period limit of the various spectra

T_2 : upper period limit of the various spectra

T_d : duration of the earthquake time history

t : time

u : displacement vector of the structure

$\dot{u}$: velocity vector of the structure

$\ddot{u}$: acceleration vector of the structure

$\ddot{u}_{g,c}$: acceleration time history of the linear combination of the selected ground motions

$\ddot{u}_{g,i}$: acceleration time history of the ith ground motion

W_n : one of the n roots of unity

W : coefficients of Kaiser window

$X_{1 \cap 2}$: intersection between two random individuals $X_{i1,j}$ and $X_{i2,j}$

X_{1-2} : relative complement of individual $X_{i2,j}$ in individual $X_{i1,j}$

X_{2-1} : relative complement of individual $X_{i1,j}$ in individual $X_{i2,j}$

X_i : combination coefficient respectively of the ith ground motion

$X_{i,j}$: j^{th} element of the ith individual

β : constant equal to 5

Δt_{old} : time step of ground motion before resampling

Δt_{new} : time step of ground motion after resampling

ω : cyclic frequency step of the Fourier spectrum

Chapter **6**

■ ■ ■ ■ ■

6 Abaqus2Matlab: An Integrated Optimization Framework for Engineering Applications

6.1 Introduction

Practical FEA applications are often not standalone, but incorporated into a general optimization framework, so that a predefined objective is attained. The reason for this is that usually the various objectives of the optimum structural design are too complex to be implemented within the framework of a finite element analysis. Sometimes an optimization process (e.g. an inverse analysis) is necessary in order to compensate for the lack of modeling information (e.g. constitutive material properties, etc.) and can yield results that are otherwise infeasible to obtain. To implement the optimization procedures that are presented in this chapter, the Abaqus2Matlab software (Papazafeiropoulos et al., 2017) has been used.
Abaqus2Matlab is an effective tool with the following features:

- It provides linking between Abaqus and Matlab. Abaqus analysis can be conducted through Matlab, without interacting with Abaqus/CAE interface, or even Abaqus/Command.

- It transfers efficiently results from Abaqus to Matlab, in an error-proof way, since every contained external function is verified by its application in reading the results of a corresponding Abaqus analysis. The results of the verification of each function are presented in this toolbox in the form of html files.

- It provides the requested results in a form that enables the user to easily manipulate the data for further postprocessing.

- It can read many different kinds of nodal results (results at nodes), elemental results (results at the element integration points or results regarding whole elements) and analysis results (e.g. node definitions, element connectivity, eigenfrequencies and eigenvalues, etc.)

- A complete documentation package is provided along with the source code.

- A sufficient number of functions is included in the toolbox to capture the most usually requested Abaqus results.

- The software, as well as all the associated material and documentation can be accessed and downloaded for free at www.abaqus2matlab.com.

6.2   Abaqus2Matlab – software development

6.2.1   Introduction

Partial Differential Equations (PDEs) govern the physics of most engineering systems. As analytical solutions are limited and generally restricted to idealized cases, the development of efficient and robust numerical methods marks a milestone in the solution of boundary value problems in structural mechanics, electromagnetism, heat transfer, mass diffusion and fluid dynamics, among many other disciplines. The Finite Element Method (FEM) has become the leading numerical technique for solving PDEs in the mechanical, civil, aeronautical and bioengineering industries. Among the wide range of packages available, ABAQUS is undoubtedly one of the most popular finite element tools for academics and practitioners. However, practical applications often require consideration of non-linear conditions, where uncertainties hinder high fidelity numerical predictions. In such circumstances, the use of advanced analysis methodologies –such as optimization procedures, inverse approaches, statistical tools or hybrid experimental–numerical techniques –has proven to compensate for the lack of information, yielding results that are otherwise unobtainable. More specifically, in many cases the objective function is a function of the error between a numerical analysis and an experiment, which needs to be minimized by adjusting various design variables of the numerical model. These design variables almost always involve some or all of the material constitutive parameters, since the constitutive properties of a model have the largest effect on its response. The results of the aforementioned optimization analysis are some "optimum" values of the constitutive properties. Assuming the the constitutive model has been selected in a reasonable way, these "optimum" values will be identical to the "real" constitutive values. The latter values are actually modeling information that is missing for an engineer who wants to perform a direct FEA and also are infeasible to obtain by direct (straight) FE analysis. Therefore optimization is necessary to find this lacking constitutive information of the material (i.e. modeling information), and since Abaqus cannot do inverse analysis by itself, various third-party softwares are used (in this thesis Matlab, which is integrated with Abaqus using Abaqus2Matlab). MATLAB, a multi-paradigm computing environment, is generally considered to be the most powerful software in this regard due to its advanced capabilities in statistics, machine learning, neural networks, curve fitting, model-based calibration and optimization. Yet, a

connection between the two most used packages in, respectively, finite element modeling and mathematical analysis, is still lacking. To fill this gap, a novel software tool is here proposed: Abaqus2Matlab, which allows to run ABAQUS directly from Matlab and to post-process the results, providing a link between the two well-known packages in a non-intrusive and versatile manner. The present proposal enjoys the benefits of MATLAB's user friendly and centralized environment, as opposed to other powerful tools like Python, which require add-on libraries. Abaqus2Matlab is distributed as source code with the aim of facilitating research. Section 6.2 is organized as follows. The structure of ABAQUS results (*.fil) files is described in section 6.2.2 and the way in which Abaqus2Matlab reads the file is presented in section 6.2.3. The software framework and architecture are explained in section 6.2.4 . For more details about the software, the reader is referred to Papazafeiropoulos et al. (2017).

6.2.2   Structure of ABAQUS results (*.fil) files

A medium in which ABAQUS analysis results can be transferred to other software for postprocessing or pre- and postprocessing is the results file. The ABAQUS results file can be written in binary (default) or ASCII format. Generally, the manipulation of results files in ASCII format is easier than in binary format, since they can be transferred between different computer systems and read from many different postprocessing software without special settings. On the other hand, for large problems the results files in ASCII format are significantly larger than the same files in binary format. ABAQUS provides the ascfil facility to convert a results file from binary to ASCII format, as shown in section 6.2.2.4. The discussion from now on will concern only ABAQUS results files in ASCII format.

6.2.2.1   Data item format

Any data item contained in a results file can be either integer, floating point number or character string. Integers begin with the character I, followed by a two digit integer which shows the number of the digits of the integer, followed by the integer itself. If the number of digits of the integer has one digit, the first character after character I is a blank space. For example, integer number "8" would be written as "I 18" and integer number "9999999999" would be written as "I109999999999".

Floating point numbers begin with the character D, followed by the number in the format E22.15 or D22.15, depending on the precision (single or double respectively). For example, number "0.5" in double precision would be written as "D 5.000000000000000D-01".

Character strings begin with the character A, followed by eight characters. If the length of a character string is less than 8, then the trailing positions are filled with blank spaces. If the length of a character string is larger than 8, then the character string is written in consecutive character strings, eight characters at a time. For example, "HOMOGENEOUS TENSION FOR ELEMENT 1" would be

written as "AHOMOGENEAOUS TENSAION FOR AELEMENT A1   ". Note the seven trailing blank spaces after the last character ("1") in the last character string.

6.2.2.2   Record format

The results file is a sequential file, meaning that it contains and stores data records in a specific order. It must be read from the beginning, up to the location of the desired data. All data items are converted into equivalent character strings and written in series which are called (logical) records. Each single line of a results file contains a series of 80 string characters, which may contain whole or part of a record. In the latter case, after completely filling the first line in which a record begins, the record string continues at the subsequent lines till the end of the record. If a record string ends before the end of a line, then the next record starts immediately after the current record in the same line, with continuation in the subsequent lines as explained above. The beginning of each record is indicated by an asterisk (*). Within each record, the data items are arranged immediately behind each other, and therefore it is possible that the end of a line splits a data item, with its first characters belonging to a line and the remaining characters belonging to the next line. The last line of the results file, if partially completed, is filled with blank spaces until the end of the line. Then, a logical record consisting of 80 blanks is inserted as the next line, in order for the end-of-file to be handled correctly. Each record has the format shown in Table 6-1.

Location	Length	Description
1	1	Record length (L)
2	1	Record type key
3	(L-2)	Attributes

Table 6-1: Format of a record written in an ABAQUS results file.

The location number denotes the position in the record, where a series of consecutive data items is written. The number of data items in each series is denoted by the length number. The first series of data items (consisting of a single data item) is an integer showing the record length, i.e. the number of data items which the record contains. The second series of data items (also consisting of a single data item) is an integer showing the record type key. The record type keys are standard indicators set in ABAQUS by convention, and denote the type of data which the record includes. The data items which actually provide useful information for the user (or attributes) are contained in a series of L-2 data items, at the 3rd (and last) position of a record. For example, record key 1900 (Record type: Element definition) for a CPE4R element with element number 2 and nodes 5, 6, 7, and 8 would be written as follows:

*I 18I 41900I 12ACPE4R   I 15I 16I 17I 18

and record key 101 (Output variable identifier: U, i.e. displacements) for node 145 and displacements for the 6 degrees of freedom equal to (0.200000029802322,

0.00, -0.07500000298023224, 1.732049942016602, 1.732049942016602 and 1.732049942016602) would be written as:

*I   19I   3101I   3145D   2.000000029802322D-01D   0.000000000000000D+00D-7.50000029802322   4D-02D   1.732049942016602D+00D   1.732049942016602D+00D 1.732049942016602D+00

In a data record which contains complex values (e.g. in a steady-state analysis), all the real components of the data record are written first and all the imaginary components follow immediately. For example, record key 101 (Output variable identifier: U, i.e. displacements) for node 1 and complex displacements for the 6 degrees of freedom equal to (-1.621881950939540e-16+0.50939i, 0.004367975320916413+0.67975i, -1.558539209401511e-15+0.055i) would be written as:

*I   19I   3101I   11D-1.621881950939540D-16D   4.367975320916413D-03D-1.558539209401511   D-15D   0.509390000000000D+00D   0.679750000000000D+00D 0.055000000000000D+00

6.2.2.3   Output

The types of output that can be written to the results file are the following:

•   element output, nodal output, energy output, modal output, contact surface output, and section output

•   element matrix output

•   substructure matrix output

•   cavity radiation viewfactor matrices

It is possible that a model is defined as an assembly of part instances, the nodes and/or the elements of which have repeated numbering definitions. In this case the local node and element numbers are converted internally into global node and element numbers, which are unique for the model being analyzed. The output in the results file is given in terms of these global identities. A map between user-defined numbers and internal numbers is printed to the data file (*.dat) if any results file output that includes node and element numbers is requested.

Set and surface names that appear in the results file are given along with their corresponding assembly and part instance names, separated by underscores. For example, if Set1 is the name of a set or surface of part Part1, which is instanced in the assembly Assembly1, then this set appears with the name Assembly1_Part1_Set1 in the results file.

6.2.2.4   Generation of ABAQUS Results (*.fil) files

ABAQUS results files can be produced in a variety of ways. The overall implementation which includes the generation of the results file(s) depends on the information flow between ABAQUS and other pre- and postprocessing

software. In order to retrieve the results of an analysis in an easy to handle form, results files in ASCII format must be generated. This can be achieved by determining specific execution procedures, which can involve input (*.inp), restart (*.res), and other types of files which can be found in the ABAQUS Documentation. In each of the input files involved, specific options with specific parameters have to be defined. In addition, the results file generation procedures differ between ABAQUS/Standard and ABAQUS/Explicit. The execution procedures for ABAQUS/Standard and ABAQUS/Explicit, the required files as well as the options in the input files of the single or restart analysis are shown in Table 2. Four procedures are presented, which combine ABAQUS/Standard and ABAQUS/Explicit finite element programs with either single or restart analysis, resulting thus in four different cases. The ABAQUS ascfil utility serves to convert results files from binary to ASCII format. This is particularly useful when ABAQUS/Explicit is used for the analyses, in which the results files generated can be only in binary format. In the case of a restart analysis, the *FILE FORMAT, ASCII and *FILE OUTPUT options for ABAQUS/Standard and ABAQUS/Explicit respectively have to be specified either in the initial or in the restart input files.

Finite element program	Execution command	Files required	Input file options		Result files generated
ABAQUS/Standard - single analysis	ABAQUS job=1	1.inp	Required	Optional	1.fil (ascii)
			*FILE FORMAT, ASCII	*CONTACT FILE	
				*EL FILE	
				*ENERGY FILE	
				*MODAL FILE	
				*NODE FILE	
				*SECTION FILE	
ABAQUS/Standard - restart analysis	ABAQUS job=2 oldjob=1	2.inp	Required	Optional	2.fil (ascii)
		1.mdl	*POST OUTPUT	*CONTACT FILE	
		1.odb	*FILE FORMAT, ASCII	*EL FILE	
		1.stt		*ENERGY FILE	
		1.prt		*MODAL FILE	
		1.res		*NODE FILE	
				*SECTION FILE	
ABAQUS/Explicit - single analysis	ABAQUS job=1	1.inp	Required	Optional	1.fil (binary)
			*FILE OUTPUT	*CONTACT FILE	
				*EL FILE	
				*ENERGY FILE	
				*MODAL FILE	
				*NODE FILE	
				*SECTION FILE	
ABAQUS/Explicit - restart analysis	ABAQUS job=2 oldjob=1	2.inp	Required	Optional	2.fil (binary)
		1.abq	*RESTART,READ	*CONTACT FILE	
		1.mdl	*FILE OUTPUT	*EL FILE	
		1.odb		*ENERGY FILE	
		1.stt		*MODAL FILE	

		1.pac		*NODE FILE	
		1.prt		*SECTION FILE	
		1.res			
		1.sel			
-	ABAQUS ascfil job=1	1.fil (binary)	-		1.fin (ascii)

<p align="center">Table 6-2: Procedures used in ABAQUS for the generation of results (*.fil) files.</p>

6.2.3 Reading of ABAQUS results files with Abaqus2Matlab

This section describes the way an ABAQUS result file is read, in order to obtain the numerical data in an easy to use form. A segment of a results file is shown in Figure 1. As mentioned in a previous section, each single line of a results file contains a series of 80 string characters, which may contain whole or part of a record. The segment shown in Figure 1 contains three records. Before the first record, the last 74 characters of the last record appear. After the third record, the first 98 characters of the next record appear.

```
...
4D-02D 1.732049942016602D+00D 1.732049942016602D+00D 1.732049942016602D+00*I 19I
41901I 3262D 1.147152855992317D-01D-1.638304144144058D-01D 7.500000298023224D-0
2D 1.732049942016602D+00D 1.732049942016602D+00D 1.732049942016602D+00*I 19I 419
01I 3263D 1.285575181245804D-01D-1.532088816165924D-01D 7.500000298023224D-02D 1
.732049942016602D+00D 1.732049942016602D+00D 1.732049942016602D+00*I 19I 41901I
3264D 1.414213627576828D-01D-1.414213627576828D-01D 7.500000298023224D-02D 1.732
049942016602D+00D 1.732049942016602D+00D 1.732049942016602D+00*I 19I 41901I 3265
D 1.532088816165924D-01D-1.285575181245804D-01D 7.500000298023224D-02D 1.7320499
...
```

<p align="center">Listing 6-1: Segment of the contents of an ABAQUS results file.</p>

The way Abaqus2matlab reads the segment of the results file presented in Listing 6-1 will be illustrated. For this purpose, the code used for reading the segment will be shown and explained line by line.

The function Fil2str (fil file to string conversion), the code of which is shown in Listing 6-2, opens the ABAQUS results file for reading only, reads the data in this file by considering it as a string and concatenating lines horizontally, so that the cell array C contains a single line string. It is reminded that all lines starting with "%" are not executed and are treated as comments. Special characters as delimiters, whitespaces or end of line characters are not specified. The concatenation in a single line during execution of textscan does not happen in previous versions of Matlab, and therefore the newline and carriage return characters of the string A contained in the 1 x 1 cell array C have to be deleted (replaced with nothing) by applying two strrep (string replacement) commands consecutively, as shown in lines 19 & 21 of the code shown in Listing 6-2, in order to yield a single line string containing all information of the results file. A single line output string is necessary, since this is the only way to manipulate whole records easily, avoiding interruptions due to continuation to subsequent lines.

After the application of Fil2str function, all lines of the segment in Listing 6-1 will be arranged in a single line as shown in Listing 6-3.

Line	Code
1	`function Rec = Fil2str(ResultsFileName)`
2	`% Open the results file for reading`
3	`fileID = fopen(ResultsFileName,'r');`
4	`% Read data from results file as a string and assign them to a cell array`
5	`% Concatenate each line without specifying delimiter, whitespace or end of`
6	`% line characters`
7	`try`
8	`    C  =  textscan  (fileID,  '%s',  'CollectOutput',  '1', 'delimiter', ...`
9	`        '','whitespace','','endofline','');`
10	`catch`
11	`    C = textscan (fileID, '%s', 'CollectOutput', 1, 'delimiter', ...`
12	`        '','whitespace','','endofline','');`
13	`end`
14	`% Close the results file`
15	`fclose(fileID);`
16	`% Assign A`
17	`A = C{1}{1};`
18	`% Remove newline characters`
19	`A1 = strrep(A,sprintf('\n'),'');`
20	`% Remove carriage return characters`
21	`Rec = strrep(A1,sprintf('\r'),'');`

Listing 6-2: Matlab code of the function Fil2str.m.

```
. . .4D-02D  1.732049942016602D+00D  1.732049942016602D+00D  1.732049942016602D+00*I  1  9I
41901I  32   62D   1.147152855992317D-01D-1.638304144144058D-01D   7.500000298023224D-02D
1.73204994201660  2D+0  0D  1.732049942016602D+00D  1.732049942016602D+00*I  1  9I  41  901I
3263D   1.285575181245804D-01D-1.  532088816165924D-01D   7.500000298023224D-02D    1
.732049942016602D+00D  1.732049942016602D  +00D  1.  732049942016602D+00*I  19I  41901I3264D
1.414213627576828D-01D-1.414213627576828D-01D          7.50000029         8023224D-02D
1.732049942016602D+00D  1.732049942016602D+00D  1.732049942016602D+00*I  19I  41901I  3  265D

1.532088816165924D-01D-1.285575181245804D-01D  7.500000298023224D-02D  1.732049 9...
```

Listing 6-3: Single line string extracted from the data in Listing 6-1.

The single line string, after being produced by Fil2str function, enters another suitable function specified by the user, depending on the type of the results to be extracted from this string. The string shown in Listing 6-3 contains node definition data (which are identified by the record key 1901 in ABAQUS) and a function which can read node definitions from the string must be used. Of course, the string may contain more than one types of data (such as nodal displacements, for example), but there is not a unique function which can extract all types of data from a string. For each type of data to be extracted, the corresponding function has to be used. Abaqus2matlab contains different functions which can read different types of results from a single line string that has been produced from a results file. To avoid confusion, there is a standard naming convention of these functions. For example, in order to extract node

definition results (record key 1901 as mentioned above) the function Rec1901.m has to be used, namely, the name of the function is comprised of "Rec" followed by the record key of the results to be read. In the string shown in Listing 6-3, the results correspond to record key 1901. Therefore, the function Rec1901 has to be used to read these results. The code of all such functions follows a similar logic, which does not differ significantly from the logic described for the code of the function Rec1901.m. The application of this function is explained in the following.

Line	Code
1	`function out = Rec1901(Rec)`
2	`ind = strfind(Rec,'I 41901'); % record key for node output (1901)`
3	`if isempty(ind)`
4	`    out=[];`
5	`    return;`
6	`end`
7	`nextpos=numel('I 41901')+1;`
8	`% Initialize`
9	`NodeNum=zeros(numel(ind),1);`
10	`% Initialize record length matrix`
11	`NW=zeros(numel(ind),1);`
12	`for i=1:numel(ind)`
13	`    % find the record length (NW)`
14	`    Rec2=Rec(ind(i)-7:ind(i));`
15	`    indNW=strfind(Rec2,'*'); % record starts with *`
16	`    % ensure that the record exists and that the record type key is at`
17	`    % location 2`
18	`    if isempty(indNW) || indNW>3`
19	`        ind(i)=NaN;`
20	`        continue;`
21	`    end`
22	`    % number of digits of record length`
23	`    ind1=indNW+2;`
24	`    ind2=indNW+3;`
25	`    a1=str2num(Rec2(ind1:ind2));`
26	`    % Record length (NW)`
27	`    ind1=ind1+2;`
28	`    ind2=ind2+a1;`
29	`    NW(i)=str2num(Rec2(ind1:ind2));`
30	`end`
31	`NodeCoords=zeros(numel(ind),max(NW)-4);`
32	`for i=1:numel(ind)`
33	`    % number of digits of node number`
34	`    ind1=ind(i)+nextpos;`
35	`    ind2=ind(i)+nextpos+1;`
36	`    a1=str2num(Rec(ind1:ind2));`
37	`    % Node number`
38	`    ind1=ind1+2;`
39	`    ind2=ind2+a1;`
40	`    NodeNum(i)=str2num(Rec(ind1:ind2));`
41	`    % Node coordinates`
42	`    for j=1:NW(i)-4`
43	`        % node coordinate`
44	`        ind1=ind2+2;`
45	`        ind2=ind2+23;`
46	`        NodeCoords(i,j)=str2num(Rec(ind1:ind2));`
47	`    end`

```
48   end
49   % Assembly of matrices for output
50   out=[NodeNum NodeCoords];
51   end
```

Listing 6-4: Matlab code of the function Rec1901.m.

The Rec1901.m function works as follows. In order to accelerate matrix storage in Matlab, preallocation of the results matrix has to be made, especially for large output. In order to preallocate the results matrix, the record length has to be known. To find the record length, the positions of the record key in ASCII form ("I 41901") are found first using the strfind function (line 2). The position of the record key is meant to be the position of its first character (i.e. the character I). These positions for the example string in Listing 6-3 are [235   391   546]. After this, a typical check is made if the array ind is empty (i.e. if no string "I 41901" is found). In positive case, the function is exited giving as output an empty matrix (lines 3 – 6). This case can be encountered if in the results file no nodal definition data are written for some reason.

Thereafter, the record length matrix is initialized, having number of rows equal to the number of elements in ind array. It is known that the record length is written one position before the record key number and therefore the pointer goes back from the position of the record key by a default number of 7 characters and stores these characters in string Rec2. After this, the string Rec2 is searched for "*", to determine the positions where the records start. If there is not an asterisk, then this means that the record does not start within these seven characters, and consequently the string "I 41901" does not signify a record key (it could be the number of a node in an element definition for example). Another point to be noted is that indNW (which shows the location of the asterisk (*) within the seven characters preceding the string "I 41901") cannot be larger than 3; this would mean that the first data item of the record includes less than 7-3=4 characters, which is not possible, since if this occurs, only the number of digits of the record length will be known, and not the record length itself (three characters include the character I followed by at most two numerical characters). In any of the two cases, ind is set equal to NaN, so that results in the corresponding positions are not read. After having ensured that indNW shows the position of the beginning of a record, the number of digits of the record length is read using the function str2num, which converts a string into a number. In a similar way, the record length is read (with indexing based on the number of its digits given previously) and assigned to array NW.

Having formed the array NW, its maximum value is taken to set the number of columns of the output matrix at preallocation, denoted as out. The number of rows of this matrix is set to be equal to the number of elements of ind. After this, the elements of ind (i.e. position of the second data item of records giving node definition results) are scanned and for each element the number of digits of node number is determined first, then the node number, and finally the nodal

coordinates, by the insertion of a for loop within each record definition, intended to scan the three coordinates (x,y,z) of each node. Finally, the node numbers and the node coordinates are concatenated horizontally to form the output array out.

6.2.4   Use of Abaqus2Matlab

Before using the Abaqus2Matlab toolbox, the user has to be aware of the various source codes contained and how they are organized. Knowledge of the source codes will enable the user to use the toolbox more effectively to perform the desired postprocessing of the ABAQUS results. In this section, after a description of how the various files are organized in the toolbox, detailed instructions are given for the use of Abaqus2Matlab.

6.2.4.1   Organization of the source code

The source code files and folders used in the toolbox are the following:

- A function named Fil2str.m that converts the contents of the results file into a one-row string from which the desired output is retrieved, as already mentioned in previous sections.

- A folder named OutputAnalysis which contains the functions available for the processing of the results of analysis type (e.g. node definitions, element connectivity, eigenfrequencies and eigenvalues, etc). A table of variables available for analysis output requests is shown in Table 6-3. The first column (with title "record type") describes the variable which is written in the ABAQUS results file for the corresponding record key shown in the second column. In the third column the output variable identifier is written. The output variable identifier is the identifying key for the variables to be written to the results (.fil) or selected results (.sel) file. The keys are defined in the sections "ABAQUS/Standard output variable identifiers" and "ABAQUS/Explicit output variable identifiers" of the ABAQUS Analysis User's Guide. In the fourth column, the Matlab function suitable for the extraction of the corresponding variable from the ABAQUS results file is shown.

- A folder named OutputNodes which contains the functions available for the processing of the results of nodal type (e.g. node displacements, concentrated forces, nodal temperatures, etc). A table of variables available for nodal output requests is shown in Table 6-4 The variables are ordered according to the output variable identifiers' names, alphabetically.

- A folder named OutputElements which contains the functions for the processing of the element results (results at the element integration points or results regarding whole elements, e.g. total strains, section forces and moments, etc). A table of variables available for element output requests is shown in Table 6-5. The variables are ordered according to the output variable identifiers' names, alphabetically.

• A folder named Verification, which contain Matlab scripts for the verification of the Fil2str.m and the various RecX.m functions (where X is the record key). All the functions provided with the Abaqus2Matlab toolbox and associated with obtaining analysis, element or nodal results are verified to ensure that they work correctly and they are not error-prone. In the verification process, an appropriate ABAQUS input file (in which the option for the extraction of the desired results in an ASCII results file (.fil) is specified), is run by ABAQUS. After the ABAQUS analysis terminates and the results file is created in the ABAQUS working directory, it is processed appropriately by Abaqus2Matlab to obtain the requested results. Finally, the results are presented and checked with regard to their class and size. The verification of Abaqus2Matlab toolbox was made using ABAQUS 6.13.

• A folder named ABAQUSInputFiles which contains the input files which are run by ABAQUS for the verification procedure. Each ABAQUS input file is named with a number (let it be Y) which is the record key of the corresponding output variable identifier, followed by the extension ".inp". The ABAQUS input file Y.inp is run by ABAQUS and produces results which are retrieved (after ABAQUS completes the analysis) by the function RecY.m.

• A folder named help which contains all Matlab source codes which are intended to print the contents of the ABAQUS input files contained in the folder ABAQUSInputFiles.

• A folder named html which contains all the html files of the documentation of Abaqus2Matlab, including the html files produced by publishing the verification examples. All the verification examples contained in the folder Verification and the ABAQUS input files contained in the folder help are published by Matlab in this folder and are accessible through the documentation.

ANALYSIS RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Element definitions	1900	-	Rec1900.m
Node definitions	1901	-	Rec1901.m
Modal	1980	-	Rec1980.m

Table 6-3: List of variables available in Abaqus2Matlab for analysis output requests

NODAL RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Nodal Acceleration	103	A	Rec103.m
Concentrated Electrical Nodal Charge	120	CECHG	Rec120.m
Concentrated Electrical Nodal Current	139	CECUR	Rec139.m
Nodal Point Load	106	CF	Rec106.m
Concentrated Flux	206	CFL	Rec206.m

Nodal Coordinate	107	COORD	Rec107.m
Fluid Cavity Volume	137	CVOL	Rec137.m
Electrical Potential	105	EPOT	Rec105.m
Motions (in Cavity Radiation Analysis)	237	MOT	Rec237.m
Normalized Concentration (Mass Diffusion Analysis)	221	NNC	Rec221.m
Temperature	201	NT	Rec201.m
Fluid Cavity Pressure	136	PCAV	Rec136.m
Pore or Acoustic Pressure	108	POR	Rec108.m
Electrical Reaction Charge	119	RCHG	Rec119.m
Electrical Reaction Current	138	RECUR	Rec138.m
Nodal Reaction Force	104	RF	Rec104.m
Residual Flux	204	RFL	Rec204.m
Internal Flux	214	RFLE	Rec214.m
Reactive Fluid Volume Flux	109	RVF	Rec109.m
Reactive Fluid Total Volume	110	RVT	Rec110.m
Total Force	146	TF	Rec146.m
Nodal Displacement	101	U	Rec101.m
Nodal Velocity	102	V	Rec102.m
Viscous Forces Due to Static Stabilization	145	VF	Rec145.m

Table 6-4: List of variables available in Abaqus2Matlab for nodal output requests.

ELEMENT RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Creep Strain (Including Swelling)	23	CE	Rec23.m
Mass Concentration (Mass Diffusion Analysis)	38	CONC	Rec38.m
Concrete Failure	31	CONF	Rec31.m
Coordinates	8	COORD	Rec8.m
Unit Normal to Crack in Concrete	26	CRACK	Rec26.m
Total Strain	21	E	Rec21.m
Total Elastic Strain	25	EE	Rec25.m
Energy (Summed over Element)	19	ELEN	Rec19.m
Energy Density	14	ENER	Rec14.m
Mechanical Strain Rate	91	ER	Rec91.m

Whole element volume	78	EVOL	Rec78.m
Film	33	FILM	Rec33.m
Total Fluid Volume Ratio	43	FLUVR	Rec43.m
Pore Fluid Effective Velocity Vector	97	FLVEL	Rec97.m
Gel (Pore Pressure Analysis)	40	GELVR	Rec40.m
Heat Flux Vector	28	HFL	Rec28.m
Total Inelastic Strain	24	IE	Rec24.m
Logarithmic Strain	89	LE	Rec89.m
Nominal Strain	90	NE	Rec90.m
Nodal Flux Caused by Heat	10	NFLUX	Rec10.m
Plastic Strain	22	PE	Rec22.m
Pore or Acoustic Pressure	18	POR	Rec18.m
Radiation	34	RAD	Rec34.m
Stress	11	S	Rec11.m
Saturation (Pore Pressure Analysis)	35	SAT	Rec35.m
Section Strain and Curvature	29	SE	Rec29.m
Section Force and Moment	13	SF	Rec13.m
Stress Invariant	12	SINV	Rec12.m
Strain Jump at Nodes	32	SJP	Rec32.m
Principal stresses	401	SP	Rec401.m
Average Shell Section Stress	83	SSAVG	Rec83.m
Element Status	61	STATUS	Rec61.m
Section Thickness	27	STH	Rec27.m
Thermal Strain	88	THE	Rec88.m

Table 6-5: List of variables available in Abaqus2Matlab for element output requests

6.2.4.2   Instructions for use of Abaqus2Matlab

To use Abaqus2Matlab, the instructions below have to be followed:

- Ensure that ABAQUS license server is running.
- Open the file named Documentation.m in Matlab and run it (press F5). This action virtually sets up all files and folders contained in the Abaqus2matlab toolbox, including the documentation. It is noted that the files generated during ABAQUS analyses will be placed one level up (outside) of the toolbox folder. The command "S=pwd" finds the directory containing the file Documentation.m, wherever may it be. The command "addpath(genpath(S))" does the setup.
- To extract an arbitrary ABAQUS analysis result from an ABAQUS results file, initially the record key and the output variable identifier have to be specified.

These can be obtained from Table 6-3 for an analysis-type output, Table 6-4 for a node-type output, and Table 6-5 for an element-type output.

- The syntax of each RecX.m function has to be known (especially regarding its output). To view the syntax of an arbitrary RecX.m function type "doc RecX" or "help RecX" (where X is the record key) in the Matlab command window. The first option shows the function manual in a Matlab browser, whereas the second option shows the function manual in the Matlab command window. In the manual of each function the necessary options to be included in the corresponding ABAQUS input file are shown.

- Create the ABAQUS input file and place it in the folder of the Abaqus2Matlab toolbox (at least at the same level as the Documentation.m script and anyway not outside the toolbox folder).

- Run the ABAQUS input file by typing in the Matlab command window "!ABAQUS job=X", then enter. After the analysis terminates, the results file X.fil is generated in the same directory as the X.inp file. The results file is then read by Abaqus2Matlab to extract the requested results.

- Type in the Matlab command window "Rec=Fil2str('X.fil')". The variable Rec is a one-row string containing the information included in the X.fil file.

- Type in the Matlab command window "out=RecX(Rec)". The variable out contains the requested results, extracted from the X.fil results file. It will be generally a double or cell array. For more information about the identity and/or physical meaning of each element contained in this array, one can refer to the manual of the function RecX.m, mentioned in section "Results file output format" of the ABAQUS Analysis User's Guide.

6.3   Abaqus2Matlab – optimization

6.3.1   Aspects of coupling different optimizer and solver software

Structural optimization is a research field dealing with optimal design of load-carrying mechanical structures. The standard form of a structural optimization problem is as follows:

$$\text{Minimize:}$$
$$f(x) \tag{6.1}$$
$$\text{Subject to:}$$
$$g_i(x) \le 0, \ i=1,...,m \tag{6.2}$$
$$h_j(x) = 0, \ j=1,...,p \tag{6.3}$$

In structural optimization problems, the evaluation of structural (static and/or dynamic) response is involved in at least one of the functions f, g, h as shown in (6.1), (6.2) and (6.3) respectively, either directly or indirectly. Semi – analytic

implementations of structural solvers within optimization algorithms that improve the overall computational performance have been published by Dafalias & Dupuis (1972) and by Chern et al. (1973). Abaqus2Matlab provides an interface between Matlab which serves as the optimization environment, and ABAQUS, which serves as the structural analysis solver. In such cases where interfaces are used for coupling the optimization software and the solver software, it is of paramount importance that the user takes into consideration two major components of the optimization procedure which are (1) formulation and way of solving the optimization problem and (2) sensitivity and approximation issues. These are described below, along with various recommendations to the Abaqus2Matlab user, in order to enjoy the best possible solutions to optimization problems using the Abaqus2Matlab software.

6.3.1.1   Formulation

The formulation of an optimization problem affects the success of the optimum design process in terms of computational effort and quality of results. Numerical difficulties result mainly from the following reasons:

- Usage of highly nonlinear objective or constraint functions. The gains in the convergence rate are apparent when linear formulations inside the objective and constraint functions of the optimization problem are used. In some cases, nonlinear formulations can be converted into equivalent linear and thus simplify the problem.

- Large differences between the magnitudes of the design variables, objective function(s) and constraint function(s). This problem can worsen if the software involved in the optimization procedure are not numerically robust. The best option in this case is to normalize the design variables, objective function(s) and equality constraint(s) to order 1 by scaling, and to normalize the inequality constraint(s) by the maximum or minimum value used to form them.

- Determination of the set of active constraints. If all the constraints are considered during the search process, the computational effort may be very high, whereas the consideration of only the constraints that are active or nearly active at a trial solution may lead to spurious oscillations and therefore inability for convergence. An appropriate and robust methodology for the consideration of the active set of constraints is a vital component of an optimization algorithm and must be carefully selected.

- The ABAQUS structural analysis model depends on the formulation of the optimization problem running in Matlab, in which it participates. While sometimes a detailed ABAQUS model is required to verify a case, if the same ABAQUS model is involved in an optimization procedure in Matlab, it needs to be appropriately simplified and/or reduced, so that the increase in the computational load is not prohibitively high, given the fact that a large number of ABAQUS analyses are induced by the optimization algorithm running in Matlab.

- The formulation of the optimization problem in Matlab sometimes depends on the ABAQUS model. If the sensitivity of the ABAQUS FE analysis results with respect to the design variables is low, then there is room for the use of a relatively simple and more straightforward optimization algorithm in Matlab; otherwise, depending on the complexity of the ABAQUS model, the optimization algorithm that will be used in Matlab must meet the ABAQUS model requirements.

- The ABAQUS FE analysis and Matlab optimization procedure can be integrated on a step-by-step basis, especially when the ABAQUS analysis in highly nonlinear. It is possible to combine the FE analysis and optimum design iterations in a single iterative process using Abaqus2Matlab.

6.3.1.2   Sensitivity

This aspect is related either to high gradients of the objective and/or constraint function(s) with respect to the design variables, or to the existence of jumps in the variation of these functions. The sensitivity of an optimization algorithm is influenced by the efficient calculation of derivatives of the objective and constraint function(s), with respect to the design variables. The importance of these derivatives is apparent, as they are usually used for:

- Approximate constraint evaluations to reduce the computational effort associated with exact evaluations

- Evaluation of the direction at which the optimization algorithm will proceed in each step to reach a solution which is better than the current

The sensitivity of the derivatives clearly affects the accuracy of the optimum solution as well as the stability of the optimization algorithm. In cases that the sensitivity of the derivatives is high, the following options are proposed:

- Suitable re-formulation of the optimization problem so that the optimization domain contains fewer or no irregular (singular) points. Quantitatively, estimation of the degree of irregularity of the search region is a matter of experience and can be crudely calculated by the ratio of the largest to the smallest eigenvalue of the Hessian matrix of the objective function at the optimal point, only in the case of unconstrained optimization problems.

- Substitute of the used optimization algorithm with another algorithm that uses fewer derivatives and/or has superlinear convergence rate, unless the computational load per iteration becomes prohibitively high. If a sequence $x_1, x_2, ..., x_n$ converges to a value r and if there exist real numbers $\lambda > 0$ and $a \geq 1$ such that

$$\lim_{n \to \infty} \frac{|x_{n+1} - r|}{|x_n - r|^a} = \lambda \tag{6.4}$$

then a is the rate of convergence of the sequence.

From the above it is apparent that the selection of the optimization algorithm is case-dependent, and it is nearly impossible to assert without ambiguity that an algorithm is generally better than another.

6.3.2   Structural analysis solver function using Abaqus2Matlab

The main purpose of Abaqus2Matlab as an interface within an optimization procedure is to play the role of the structural analysis solver in an automated and reliable way, non-prone to errors. For this purpose, Abaqus2Matlab follows a certain workflow, which is different among its various components, e.g. it is different between Abaqus2Matlab/fil2Matlab (responsible for processing ABAQUS results (*.fil) files), and Abaqus2Matlab/odb2Matlab (responsible for processing ABAQUS output data base (*.odb) files). In this thesis, the fil2Matlab component is used. The workflow is shown in Figure 6-1 and is executed as follows:

- The ABAQUS input file that corresponds to the current values of the design variables x is generated. This job is done by the Abaqus2Matlab function InpFileConstr.m. This function apparently depends on the ABAQUS model and the way that it is parameterized. x is a vector containing the current values of the design variables. The function InpFileConstr.m does not give any output. The following option with its parameter has to be specified in the ABAQUS input file in order for the results (*.fil) file to be generated:

...

*FILE FORMAT, ASCII

...

- After the ABAQUS input file is generated, it is run remotely by Abaqus2Matlab, through the function runABAQUSAnalysis.m. This function accepts the name of the ABAQUS input file to be run, an upper time limit t_{ub} and a lower time limit t_{lb}. ABAQUS execution starts and afterwards Matlab execution is halted, waiting until any of the following events happens:

(1) ABAQUS execution starts normally and the ABAQUS lock (*.lck) file is deleted. This is the normal case.

(2) ABAQUS execution starts normally and t_{ub} is exceeded (i.e. for some reason ABAQUS execution is lagged). In this case, ABAQUS analysis is automatically terminated and executed again.

(3) ABAQUS execution starts but the ABAQUS lock (*.lck) file is not generated, and t_{lb} is exceeded (most likely the execution of the process SMASimUtility.exe is lagged). In this case, the process SMASimUtility.exe is automatically killed and the ABAQUS analysis is executed again.

In this function Java variants of various Matlab commands are used where possible, in order to avoid memory leaks which may cause lags or crashes of ABAQUS execution. Furthermore, the Java commands are proven to be much more accurate than the corresponding Matlab commands.

- After the ABAQUS analysis has been completed, control is passed back to Matlab and then the Fil2str.m function is used to read the information contained in the ABAQUS results (*.fil) file in ASCII format and assemble it into a one-row string. The function Fil2str.m initially opens the results file with reading only permission, then reads the data included in this file as an assembly of strings within a cell structure, and finally it removes any newline and carriage return characters from these strings, eventually resulting in a single-row string that is passed as output (output argument s). Provisions are taken in order to ensure that the Fil2str.m function works for any Matlab version that the user may be running.
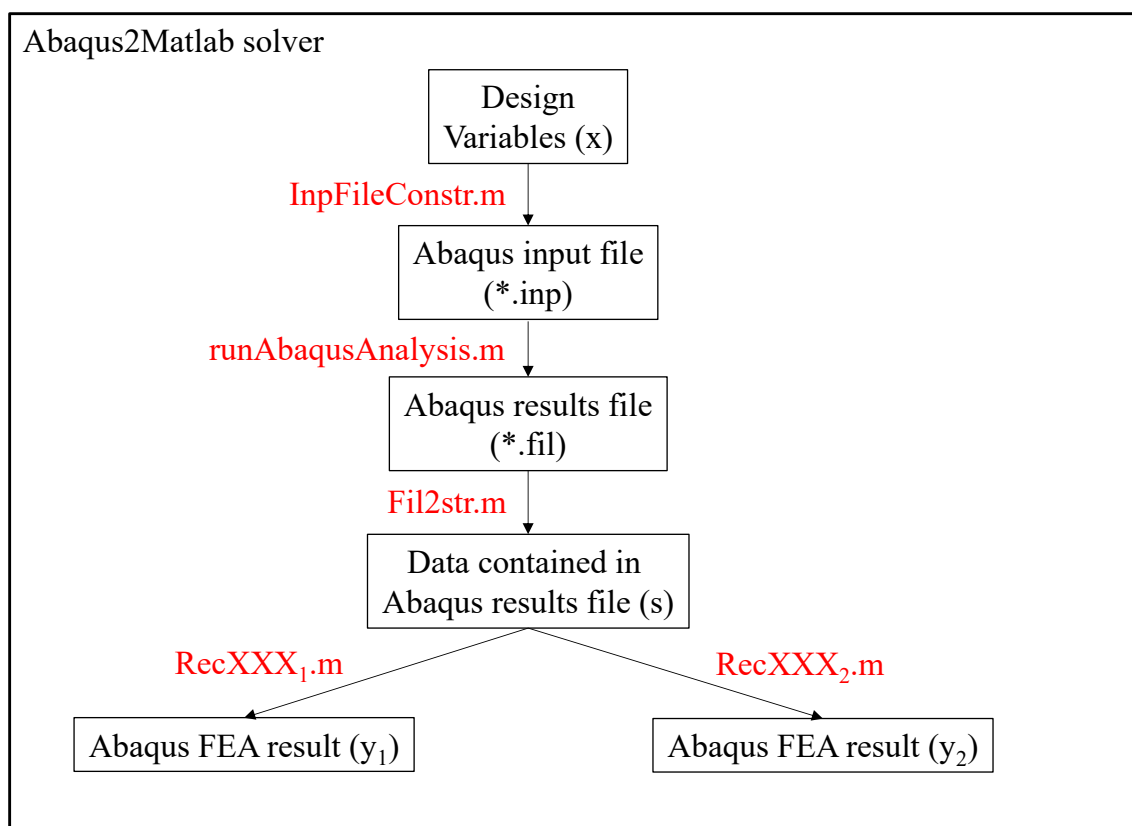
Figure 6-1: Flowchart of the Abaqus2Matlab application as a structural analysis solver function.

The single line string that is output by the function Fil2str.m is processed by another suitable function that is specified by the user, depending on the type of the desired ABAQUS results. Of course, the string may contain more than one types of data (such as nodal displacements and element stresses for example), for the extraction of which different functions must be selected by the user then using the open source version of Abaqus2Matlab (v.1.0). This feature has been improved in the second version of Abaqus2Matlab, where a single function is used for any type of ABAQUS results that are extracted, and the user has only to specify the record key as an input argument to this function. In the first (open

source) version of Abaqus2Matlab there is a standard naming convention of these functions, i.e. RecXXX.m as shown in Figure 6-1, where XXX is replaced with the record key corresponding to the desired result. For example, in this section the function Rec101.m was used to extract nodal displacement results (record key 101), whereas the function Rec13.m was used to extract element section forces (record key 13).

6.3.3   Use of Abaqus2Matlab for optimum structural design

The aforementioned description is applicable for the implementation of any structural analysis solver function using Abaqus2Matlab. Consequently Abaqus2Matlab can be used for any structural optimization problem that is solvable in ABAQUS. In the following the implementation of Abaqus2Matlab for the solution of structural optimization problems will be shown, for cases in which the Matlab fmincon function is used as the optimizer.

There are various patterns for using the Abaqus2Matlab solver within the framework of the Matlab fmincon function. Figure 6-2 presents the relevant schematic flow diagram of possible ways of inclusion of the Abaqus2Matlab solver. It is possible to use Abaqus2Matlab for the objective function evaluation (Case 1) or not (Case 2). The same happens with the constraint function evaluation. Therefore the Abaqus2Matlab solver that is presented in Figure 6-1 can be used as shown in Figure 6-2 for the evaluation of either the objective function, or the constraint function, or both of them. Since we are dealing with a structural optimization problem, it is necessary that at least one of the two functions (objective and constraint functions) implements the flowchart of Case 1. In the present section, the Abaqus2Matlab was used only in the constraint function evaluation, i.e. Case 2 and Case 1 were used for the objective and constraint functions respectively. Instructions about the proposed methodology
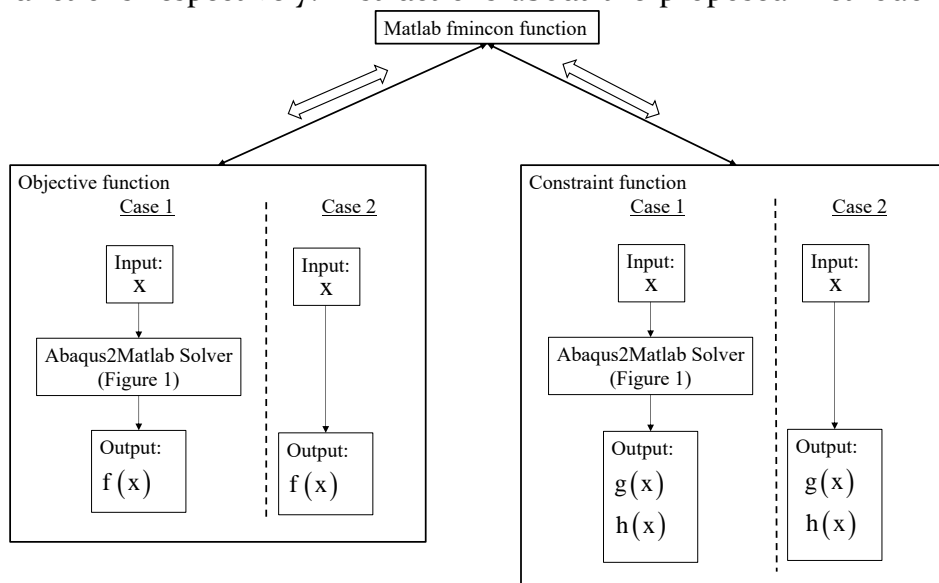
in order to

Figure 6-2: Flowchart of the possible uses of Abaqus2Matlab for the objective and constraint function evaluations within the framework of the Matlab fmincon function.

```
┌─────────────────────────────────┐
│ Define the design variables (x) of the │
│ optimization problem             │
└─────────────────────────────────┘
              │
┌─────────────────────────────────┐
│ Construct a Matlab function which │
│ generates the Abaqus input file for │
│ given values of the design variables (x) │
└─────────────────────────────────┘
              │
┌─────────────────────────────────┐
│ Define the objective function which │
│ returns  f(x)                    │
└─────────────────────────────────┘
              │
┌─────────────────────────────────┐
│ Define the constraint function which │
│ returns  g(x), h(x)              │
└─────────────────────────────────┘
              │
┌─────────────────────────────────┐
│ Define the upper and lower bounds of │
│ the design variables             │
└─────────────────────────────────┘
              │
┌─────────────────────────────────┐
│ Program the main Matlab optimization │
│ code                             │
└─────────────────────────────────┘
              │
┌─────────────────────────────────┐
│ Assign an initial value to the design │
│ variables within the upper and lower │
│ bounds and evaluate the objective and │
│ constraint functions. Check that both of │
│ the return the expected outputs  │
└─────────────────────────────────┘
```

Main optimization loop

Abaqus2Matlab solver (Figure 1)

Optimization algorithm (e.g. IPA) ← Run Abaqus analysis

Meeting termination criteria

NO → Generate new Abaqus input file → Run Abaqus analysis

YES

Finish

Figure 6-3: Suggested methodology for setting up and performing a structural optimization problem using Abaqus2Matlab.

setup and solve a structural optimization problem are shown in Figure 6-3.

6.4   Abaqus2Matlab – verification

Verification is a substantial process for any software, which is has to be done for the following reasons:

- It provides substantial evidence for the effectiveness, accuracy and robustness of a software
- It renders a software computationworthy, by building trust to its users
- It provides ways for the correct interpretation of the results from the usersand/or developers
- The cases that are used for the verification can be used as a template for various more complex postprocessing tasks.
- It provides hints on how to alleviate computationally intensive processing tasks.

In the following, three benchmark truss optimization problems will be solved with Abaqus2Matlab as a verification process.

6.4.1   2-bar truss

6.4.1.1   Description

Consider a 2-bar plane truss shown in Figure 1 with the following structural characteristics:

- Modulus of Elasticity: E = 68.948 GPa
- Material unit weight: ρ = 2767.990 kg/m3
- L = 9.144 m
- P = 444.974 kN

A concentrated force is applied at node 2 of the truss, whereas all degrees of freedom of nodes 1 and 3 are constrained (hinged). A linear elastic static analysis is performed to determine the displacements of the free node. The element numbers are shown near the middle of each element, whereas the node numbers are encircled.
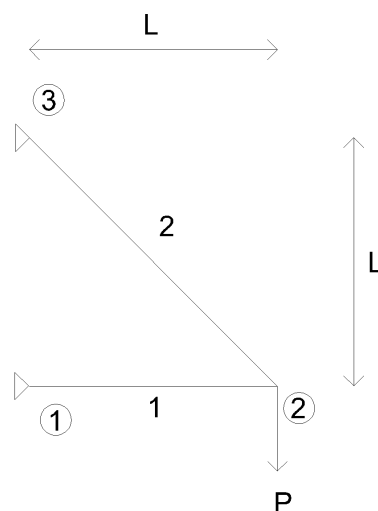
Figure 6-4: 2-bar truss.

The objective of the optimization procedure is to minimize the weight of the truss, which entails that minimum quantity of material will be used for its construction, meaning maximization of structural economy. Minimization of weight is associated with minimization of volume, since density is constant and the same for all members of the truss. Since the length of the various members of the truss does not change during the optimization process, minimization of weight means minimization of the members' cross-sectional area. The design variables are the cross section area of each member in the interval [0.003650822800775, 0.0225806] (m²). The truss structure with the minimum weight has to satisfy certain constraints which are imposed on stresses and displacements. The maximum allowable displacement in the ±x and ±y directions for each node is d_{max} = 0.0508 m, while the maximum allowable stress (absolute value) is σ_{max}= 172.369 MPa in tension or compression. The lower limit of cross section area is determined from the fact that the axial stresses cannot exceed the aforementioned maximum allowable value. Considering horizontal and vertical equilibrium of the truss, it is found that member 1 is loaded with compressive axial force equal to P, whereas member 2 is loaded with extensional axial force equal to $\sqrt{2}P$. Therefore, the critical member is the latter, and the stress inequality is written as:

$$\sigma \leq \sigma_{max} \Leftrightarrow \frac{\sqrt{2}P}{A_{min}} \leq \sigma_{max} \Leftrightarrow A_{min} \geq \frac{\sqrt{2}P}{\sigma_{max}} = 0.003650822800775 m^2 \qquad (6.5)$$

It is apparent that by specifying the lower limit of the member cross section area in this way, the constraints of maximum allowable stresses are automatically satisfied and need not be considered explicitly in the optimization procedure. Consequently, only constraints regarding displacements will be considered.

6.4.1.2 Implementation

The problem is solved by executing a main Matlab script in which various variables are firstly defined and finally the built in function fmincon.m is used to perform the optimization process. Apart from fmincon.m, other user defined external functions are used as well. All scripts and functions are explained in the following.

- Main script

The main Matlab script is shown in Listing 6-5. Initially, the variable tABAQUS is declared as global, in order to be "seen" by the function TrussConfun.m, without being among its input arguments. The syntax of the functions specified as input arguments to fmincon.m has a standard format and specific input arguments. Therefore, if a variable different from the default input arguments has to be passed inside such a function, it has to be declared as global. NumElements is equal to the number of elements of the truss. At line 5, NumElements is set to be

2. After this, an initial guess xo is made for the cross-sectional areas of the 2 bars of the truss. This is a vector of dimensions 2x1, containing these initial values. Next, the upper and lower limits of the design variables (cross-sectional areas of the bars of the truss) are specified. All design variables cannot be lower than 0.003650822800775 m2 and higher than 0.0225806 m2, as already mentioned in the previous section. AreaMin and AreaMax are the minimum and maximum cross-sectional area of the members respectively. lb and ub are the vectorized lower and upper limits respectively (size 2x1) only for purposes of suitability as input arguments for the fmincon function. At lines 15 – 18 nondefault options are specified for the fmincon function. Specifically, options.TolFun (FunctionTolerance) is a lower bound on the change in the value of the objective function (i.e. the weight of the truss) during a step and options.TolCon (ConstraintTolerance) is an upper bound on the magnitude of the constraint functions. options.Display is the level of display at the command window, specified to display output at each iteration and give the technical exit message. At lines 20 & 21 the timer and tABAQUS variable are initialized. At lines 23 and 24 the fmincon function is called with the following input arguments:

- Objective function: TrussObjfun.m
- Initial point: x_o
- Lower bounds: lb
- Upper bounds: ub
- Constraint function: TrussConfun.m
- Options: options

The objective and constraint functions will be explained in the following sections.

- TrussObjfun.m

The script of this function is shown in Listing 6-6. The input argument of the objective function is a vector of size 2x1, containing the current values of the design variables. The output of the objective function is the weight of the truss.

- TrussConfun.m

The script of this function is shown in Listing 6-7. The input argument of the constraint function is the same as that of the objective function. Initially the variable tABAQUS is declared as global, in order to be increased by the time ABAQUS takes to perform the static analysis of the truss. In lines 6 & 8 the maximum absolute value of horizontal displacement and the maximum absolute value of vertical displacement are set respectively. Next, the ABAQUS input file TrussABAQUS.inp is created, which is run by ABAQUS to give the results. The code of function TrussInpFileConstr.m is presented in Listing 6-8. After this, the time elapsed till the start of ABAQUS analysis is reported in variable t1. In the sequel, the input file TrussABAQUS.inp starts to be processed by ABAQUS at line 14. To halt Matlab execution at this point till ABAQUS analysis terminates, a while loop is inserted at lines 19 – 21, which is executed under the condition that the ABAQUS TrussABAQUS.lck lock file exists in the working directory

(indicating that ABAQUS is running). As soon as ABAQUS analysis finishes, this file is automatically deleted by ABAQUS and the while condition becomes false, so Matlab proceeds with the next commands in the script. To give ABAQUS enough time to create the lock file, a pause of duration 10 sec is specified before the while loop starts (line 17). After ABAQUS stops running, the time elapsed is reported in variable t2 at line 23. The time difference t2-t1 is equal to the time ABAQUS takes for the analysis and is added to tABAQUS at line 25. At line 27, using the Fil2str function which is part of the Abaqus2Matlab toolbox, the contents of the ABAQUS results file named TrussABAQUS.fil are converted to an one-row string (Rec) from which the results will be retrieved. The function to read nodal displacements from Rec is found to be Rec101.m, and is applied to obtain the nodal displacements (line 29). After the extraction of the results from the results file the TrussABAQUS.fil, TrussABAQUS.prt, TrussABAQUS.com and TrussABAQUS.sim files are deleted at lines 32-35, because ABAQUS cannot rewrite them in the next call of the constraint function, where a new analysis takes place. Next, postprocessing of the aforementioned results takes place, which concludes in the formation of the inequality and equality vectors required as output of the constraint function in Matlab (c and ceq respectively in Listing 6-7).

Line	Code
1	`% Declare ABAQUS time counter as global variable (used also in`
2	`% TrussConfun.m)`
3	`global tABAQUS`
4	`% Specify the number of elements of the truss.`
5	`NumElements=2;`
6	`% Make a starting guess for the solution.`
7	`x0 = [0.0037; 0.0049];`
8	`% Set the lower and upper limit of the cross section areas of the two`
9	`% members of the truss.`
10	`AreaMin=0.003650822800775; % P*sqrt(2)/maxstress`
11	`AreaMax=0.0225806;`
12	`lb=AreaMin*ones(1,NumElements);`
13	`ub=AreaMax*ones(1,NumElements);`
14	`% Set FunctionTolerance and StepTolerance`
15	`options=optimset('fmincon');`
16	`options.Display='iter-detailed';`
17	`options.TolFun=1e-3;`
18	`options.TolCon=1e-3;`
19	`% Start timer`
20	`tic`
21	`tABAQUS=0;`
22	`% Perform constrained optimization of the truss`
23	`[X,fval,exitflag,output,lambda]=fmincon(@TrussObjfun,x0,[],[],[],[],...`
24	`    lb ,ub,'TrussConfun',options)`
25	`% Report elapsed times (total, required by ABAQUS and required by Matlab`
26	`% respectively)`
27	`tTOTAL=toc`
28	`tABAQUS`
29	`tMATLAB=tTOTAL-tABAQUS`

Listing 6-5: Matlab code of the main script for the solution of the 2-bar truss problem.

Line	Code
1	`function f = TrussObjfun(x)`
2	`% Horizontal length`
3	`u=9.144;`
4	`% total weight`
5	`f = 9.81*2767.990471*x'*u*[1;sqrt(2)];`
6	`end`

Listing 6-6: Matlab code of the function TrussObjfun.m used for the optimization of the 2-bar truss.

Line	Code
1	`function [c,ceq] = TrussConfun(x)`
2	`% Declare ABAQUS timer as global variable`
3	`global tABAQUS`
4	`% Set the displacement limits of the 2-bar truss`
5	`% maximum absolute value of horizontal displacement (m)`
6	`Dmaxhor=0.0508;`
7	`% maximum absolute value of vertical displacement (m)`
8	`Dmaxver=0.0508;`
9	`% Construct the ABAQUS input file TrussABAQUS.inp`
10	`TrussInpFileConstr(x)`
11	`% Report time before ABAQUS analysis starts`
12	`t1=toc;`
13	`% Run the input file TrussABAQUS.inp with ABAQUS`
14	`!ABAQUS job=TrussABAQUS`
15	`% Pause Matlab execution to give ABAQUS enough time to create the`
16	`% TrussABAQUS.lck file`
17	`pause(10)`
18	`% If the TrussABAQUS.lck file exists then halt Matlab execution`
19	`while exist('TrussABAQUS.lck','file')==2`
20	`    pause(0.1)`
21	`end`
22	`% Report time after ABAQUS analysis terminates`
23	`t2=toc;`
24	`% Add elapsed time to ABAQUS time counter`
25	`tABAQUS=tABAQUS+t2-t1;`
26	`% Assign all lines of the TrussABAQUS.fil file in an one-row string`
27	`Rec = Fil2str('TrussABAQUS.fil');`
28	`% Obtain the nodal displacements`
29	`out2 = Rec101(Rec);`
30	`NodalDisplacements=out2(:,2:3);`
31	`% Delete the files of last ABAQUS run to avoid rewriting them`
32	`delete('TrussABAQUS.fil');`
33	`delete('TrussABAQUS.prt');`
34	`delete('TrussABAQUS.com');`
35	`delete('TrussABAQUS.sim');`
36	`% Calculate the maximum nodal displacements`
37	`maxNodDisplX1=max(abs(NodalDisplacements(:,1)));`
38	`maxNodDisplY1=max(abs(NodalDisplacements(:,2)));`
39	`% Assemble the constraints`
40	`c = [maxNodDisplY1-Dmaxver;`
41	`    maxNodDisplX1-Dmaxhor];`
42	`ceq = [];`
43	`end`

Listing 6-7: Matlab code of the function TrussConfun.m used for the optimization of the 2-bar truss.

- TrussInpFileConstr.m

In Listing 6-8 the code of TrussInpFileConstr.m function is shown. This function opens an empty notepad file named "TrussABAQUS.inp" (lines 3 & 4), prints the necessary input file options for the ABAQUS analysis (lines 6 – 48), and closes the file (line 50). As seen in Listing 6-8 the majority of options remain unchanged between ABAQUS analyses, except for lines 24 & 28, where the cross section area of the two truss members is specified as a function of the design variable x. This design variable is altered internally by fmincon function during the optimization procedure.

Line	Code
1	`function TrussInpFileConstr(x)`
2	`% Open input file`
3	`OutputFileName = 'TrussABAQUS.inp';`
4	`fileID = fopen(OutputFileName,'wt');`
5	`% Write options`
6	`fprintf(fileID,' *Heading\n');`
7	`fprintf(fileID,' *Preprint, echo=NO, model=NO, history=NO, contact=NO\n');`
8	`fprintf(fileID,' *Part, name=Part-1\n');`
9	`fprintf(fileID,' *End Part\n');`
10	`fprintf(fileID,' *Assembly, name=Assembly\n');`
11	`fprintf(fileID,' *Instance, name=Part-1-1, part=Part-1\n');`
12	`fprintf(fileID,' *Node\n');`
13	`fprintf(fileID,' 1,    0,   0\n');`
14	`fprintf(fileID,' 2,    9.144,  0\n');`
15	`fprintf(fileID,' 3,    0,   9.144\n');`
16	`fprintf(fileID,' *Element, type=FRAME2D\n');`
17	`fprintf(fileID,' 1, 1, 2\n');`
18	`fprintf(fileID,' 2, 2, 3\n');`
19	`fprintf(fileID,' *Elset, elset=_PickedSet2_#1, internal\n');`
20	`fprintf(fileID,' 1,\n');`
21	`fprintf(fileID,' *Elset, elset=_PickedSet2_#2, internal\n');`
22	`fprintf(fileID,' 2,\n');`
23	`fprintf(fileID,' *FRAME Section, elset=_PickedSet2_#1, PINNED\n');`
24	`fprintf(fileID,'     %s\n',   [num2str(x(1),20),',   6.6597028096E-10,   0, 3.7460828304E-10']);`
25	`fprintf(fileID,' 0.,0.,-1.\n');`
26	`fprintf(fileID,' 68947572932, 1e3\n');`
27	`fprintf(fileID,' *FRAME Section, elset=_PickedSet2_#2, PINNED\n');`
28	`fprintf(fileID,'     %s\n',   [num2str(x(2),20),',   6.6597028096E-10,   0, 3.7460828304E-10']);`
29	`fprintf(fileID,' 0.,0.,-1.\n');`
30	`fprintf(fileID,' 68947572932, 1e3\n');`
31	`fprintf(fileID,' *End Instance\n');`
32	`fprintf(fileID,'  *Nset,  nset=_PickedSet21,  internal,  instance=Part-1-1\n');`
33	`fprintf(fileID,' 2\n');`
34	`fprintf(fileID,'  *Nset,  nset=_PickedSet22,  internal,  instance=Part-1-1\n');`
35	`fprintf(fileID,' 1, 3\n');`
36	`fprintf(fileID,' *End Assembly\n');`

```
37    fprintf(fileID,' *Step, name=Step-1\n');
38    fprintf(fileID,' *Static\n');
39    fprintf(fileID,' 1., 1., 1e-05, 1.\n');
40    fprintf(fileID,' *FILE FORMAT, ASCII\n');
41    fprintf(fileID,' *Boundary\n');
42    fprintf(fileID,' _PickedSet22, 1, 1\n');
43    fprintf(fileID,' _PickedSet22, 2, 2\n');
44    fprintf(fileID,' *Cload\n');
45    fprintf(fileID,' _PickedSet21, 2, -444974.11497\n');
46    fprintf(fileID,' *NODE FILE\n');
47    fprintf(fileID,' RF, U\n');
48    fprintf(fileID,' *End Step\n');
49    % Close input file
50    fclose(fileID);
```

Listing 6-8: Matlab code of the function TrussInpFileConstr.m used for the optimization of the 2-bar truss.

6.4.1.3   Result

After the optimization analysis terminates, the results which appear in the Matlab command window are shown in Listing 6-9. From this, it is concluded that the local minimum has been reached, while satisfying the constraints. The design variables at the local minimum are [0.003651106365609, 0.004819002266391] m2 and the minimum truss weight is equal to 2.598717321129937 kN. The output structure contains information about the optimization process, e.g. the iterations taken and the number of objective function evaluations (6 and 21 respectively). Concerning running times which appear in the command window, it is evident that the majority of the running time is consumed by ABAQUS (99.79%) whereas Matlab (including the Abaqus2Matlab toolbox) takes the rest (0.21%).

```
Command Window
                                        First-order      Norm of
  Iter F-count            f(x)   Feasibility    optimality        step
     0       3    2.639299e+03    0.000e+00    1.731e+03
     1       6    2.222749e+03    9.909e-03    1.348e+02    1.155e-03
     2       9    2.221682e+03    9.944e-03    5.491e+01    2.938e-06
     3      12    2.233984e+03    9.533e-03    1.664e+03    3.466e-05
     4      15    2.519902e+03    1.695e-03    1.008e+05    8.157e-04
     5      18    2.595075e+03    7.575e-05    8.515e+03    2.146e-04
     6      21    2.598717e+03    1.718e-07    1.473e+02    1.135e-05


Optimization  completed:  The  relative  first-order  optimality  measure,
4.195356e-04,
is  less  than  options.TolFun  =  1.000000e-03,  and  the  relative  maximum
constraint
violation, 1.717630e-07, is less than options.TolCon = 1.000000e-03.

Optimization Metric                                        Options
relative  first-order  optimality  =     4.20e-04          TolFun  =     1e-03
(selected)
relative  max(constraint  violation)  =     1.72e-07       TolCon  =     1e-03
(selected)


X =
```

```
    0.003651106365609
    0.004819002266391


fval =

    2.598717321129937e+03


exitflag =

    1


output =

           iterations: 6
            funcCount: 21
        constrviolation: 1.717630013498006e-07
             stepsize: 1.134970839769077e-05
            algorithm: 'interior-point'
        firstorderopt: 1.473172563314147e+02
          cgiterations: 0
              message: 'Local minimum found that satisfies the constrai...'


lambda =

          eqlin: [0x1 double]
       eqnonlin: [0x1 double]
         ineqlin: [0x1 double]
           lower: [2x1 double]
           upper: [2x1 double]
      ineqnonlin: [2x1 double]


tTOTAL =

    5.431887016466463e+02


tABAQUS =

    5.420298590808136e+02


tMATLAB =

   1.158842565832742
```

Listing 6-9: Results shown in the Matlab command window after termination of the optimization process of the 2-bar truss.

6.4.2   10-bar truss

6.4.2.1   Description

Consider a 10 bar plane truss shown in Figure 6-5 with the following structural characteristics:

- Modulus of Elasticity E = 10,000 ksi
- material weight rho = 0.1 lb/in³
- length L = 360 in
- load P = 100 kip

The structural members are divided into 10 groups. The design variables are the cross section areas of each member group in the interval [0.1, 35] (in²). The constraints are imposed on stresses and displacements. The maximum allowable displacement in the ±x and ±y directions for each node is d_max= 2 in, while the maximum allowable stress (absolute value) is sigma_allow= 25 ksi in tension or compression and the objective is to minimize the weight of the structure under the specified constraints.

The implementation for the solution of the 10-bar truss optimization problem using Abaqus2Matlab is much the same as that used for the solution of the 2-bar truss optimization problem, and for this reason the relevant details are not presented.
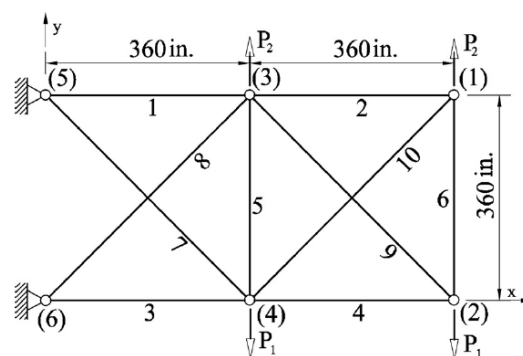
Figure 6-5: Geometry and applied load for 10-bar truss.

6.4.2.2   Results & comparison with literature

Table 6-6 presents the best optimum results found by the proposed optimization procedure and the corresponding number of function evaluations. The results of the present thesis are compared with corresponding results of some previous studies found in the literature. It is observed that the optimum weight and design variables obtained in this thesis are very close to those obtained from previous studies. However, it is clear that the proposed optimization procedure requires much lower structural analyses than other methods to reach the optimum designs.

Variables	Optimal cross section area (in²)					
Design name	M. Sonmez	Wu & Tseng	Li et al. (2007)	Degertekin &Hayalioglu	Degertekin (2012)	Kaveh et al. (2014)	Present thesis

	(2011)	(2010)		(2013)			
A1	30.548	30.378	30.704	30.429	30.394	30.208	30.5218
A2	0.1	0.1	0.1	0.1	0.1	0.1	0.1
A3	23.18	23.468	23.167	23.244	23.098	22.698	23.1999
A4	15.218	15.196	15.183	15.368	15.491	15.275	15.2229
A5	0.1	0.1	0.1	0.1	0.1	0.1	0.1
A6	0.551	0.533	0.551	0.575	0.529	0.529	0.5514
A7	7.463	7.437	7.46	7.440	7.488	7.558	7.4572
A8	21.058	21.084	20.978	20.967	21.189	21.559	21.0364
A9	21.501	21.433	21.508	21.533	21.342	21.491	21.5284
A10	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Weight (lb)	5060.88	5060.45	5060.92	5060.96	5061.42	5062.39	5060.9
Number of function evaluations	500,000	32,100	125,000	16,872	7,081	9,791	347

Table 6-6: Optimization results of the 10-bar truss.

6.4.3   25-bar truss

6.4.3.1   Description

The description of the benchmark 25-bar space truss optimization problem is as follows. Consider a 25 bar space truss shown in Figure 6-6 with the following structural characteristics:

Modulus of Elasticity E =10,000 ksi

Material density rho = 0.1 lb/in³

The structural members are divided into 8 groups. The design variables are the cross section areas of each member group in the range [0.01, 5] (in²). The constraints are imposed on stresses and displacements. The maximum allowable displacement in the ±x, ±y and ±z directions for each node is d_max= 0.35 in. Two load cases have been considered. The maximum and minimum allowable stress is shown in Table 6-7. The objective is to minimize the weight of the structure under the specified constraints for both load cases simultaneously. The members were grouped as follows: (1) element 1; (2) elements 2, 3, 4 and 5; (3) elements 6, 7, 8 and 9; (4) elements 10 and 11; (5) elements 12 and 13; (6)elements 14, 15, 16, 17 and 18; (7) elements 18, 19, 20 and 21; (8) elements 22, 23, 24 and 25.

6.4.3.2   Results & comparison with literature

Table 6-8 presents the optimum results of the proposed optimization procedure and compares these results with those previously reported in the literature. The difference among all the results is very small, hence all optimization algorithms indicated in Table 6-8 found almost the same optimum structural weight. It is apparent that the optimization procedure proposed in this thesis can reach optimum results much faster than the other algorithms.

DESIGN VARIABLE	MEMBER	ALLOWABLE TENSION STRESS (ksi)	ALLOWABLE COMPRESSION STRESS (ksi)
1	1	40	-35.092
2	2,3,4,5	40	-11.59
3	6,7,8,9	40	-17.305
4	10,11	40	-35.092
5	12,13	40	-35.092
6	14-17	40	-6.759
7	18-21	40	-6.759
8	22-25	40	-11.082

Table 6-7: Member families of the 25-bar truss optimization problem and corresponding stress limits.
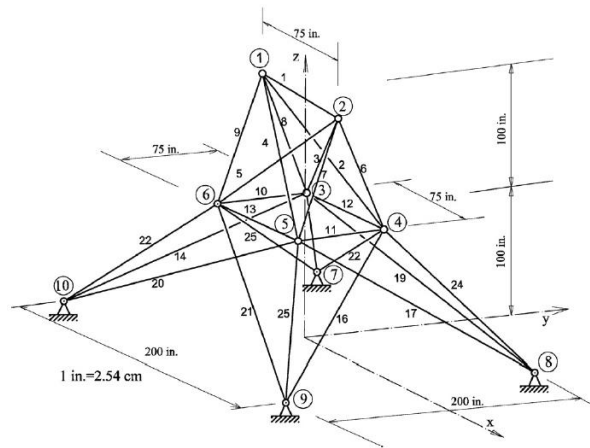
Figure 6-6: Geometry and applied load for 25-bar truss.

Variables		Optimal cross section area (in²)					
No	Design variables	M. Sonmez (2011)	Li et al. (2007)	Degertekin & Hayalioglu (2013)	Degertekin (2012)	Kaveh et al. (2014)	Present thesis
1	A1	0.011	0.01	0.01	0.01	0.01	0.01
2	A2-A5	1.979	1.970	2.071	2.074	1.9907	1.9856
3	A6-A9	3.003	3.016	2.957	2.961	2.9881	2.9969
4	A10-A11	0.01	0.01	0.01	0.01	0.01	0.01
5	A12-A13	0.01	0.01	0.01	0.01	0.01	0.01
6	A14-A17	0.69	0.694	0.6891	0.691	0.6824	0.679
7	A18-A21	1.679	1.681	1.6209	1.617	1.6764	1.6769
8	A22-A25	2.652	2.643	2.6768	2.674	2.6656	2.6676
Weight (lb)		545.193	545.19	545.09	545.12	545.164	545.166
Number of function evaluations		300,000	125,000	15,318	9,051	13,326	851

Table 6-8: Optimization results of the 25-bar truss.

6.5   Optimum design of cantilever walls retaining linear elastic backfill by use of genetic algorithm

6.5.1   Introduction

Cantilever retaining walls are among the simplest and most common geotechnical structures intended to support earth backfills. Their main representatives are retaining walls supporting deep excavations, bridge abutments, harbor-quay walls, anchored retaining walls, etc. Their design must satisfy two major requirements: internal and external stability. The former ensures the structural integrity of the various parts of the retaining wall; the latter ensures that the wall – soil system formed after construction will remain in equilibrium, except for some displacements of affordable magnitude.

Retaining walls have to satisfy constraints imposed by the norms, assumptions, preferences and the target to be accomplished, and simultaneously have to be as economical as possible. The design is based on a trial-and-error procedure, which renders the experience of the designer an important factor to reach a cost-effective design. This manual research for the optimum design may be very time-consuming and tedious, while it is not ensured that the final result will be the optimum possible. This necessitates the need for application of various optimization procedures in order to achieve the optimum design.

Relevant optimization methods range from relatively simple mathematical programming based (exact) methods to novel heuristic search techniques. The methods belonging to the first category are very efficient for cases with a few design variables. A design aid is compiled by Rhomberg & Street (1981) from results of an exhaustive search, with which simple rules of thumb were developed to provide for minimum cost design of cantilever retaining walls. Optimization of reinforced concrete cantilever retaining walls was performed by Saribas & Erbatur (1996) and the optimum design problem is posed as a constrained non-linear programming problem with seven design variables. Cost and weight of the walls were used as objective functions and overturning failure, sliding failure, no tension condition in the foundation base, shear and moment capacities of toe slab, heel slab, and stem of wall as constraints. The problem of optimal cost design of cantilever retaining walls is formulated by Basudhar et al. (2006) as a non-linear programming problem and a sequential unconstrained minimization technique is adopted. Sivakumar Babu & Munwar Basha (2008) presented optimum reliability-based design of cantilever retaining walls by considering the parameter uncertainties and evaluating the safety in terms of reliability index and not merely by calculating the safety factor.

However, exact methods require large computational effort when the number of design variables increases, and apart from this, they require gradient information and seek to improve the solution in the neighborhood of a starting point. So, in order to attain an optimum design, one has to resort to more robust optimization techniques, which are capable of searching effectively the whole design variable domain and not being trapped into local optima. Recently

developed heuristic methods, such as genetic algorithms, simulated annealing, threshold accepting, tabu search, ant colonies, particle swarm, etc. provide more attractive alternatives. Although these methods use simple algorithms, they require great computational effort. Representative studies of optimum design of retaining walls by use of heuristic methods are presented in the following. An application of a simulated annealing algorithm is reported by Ceranic et al. (2001) to minimum cost design of reinforced concrete cantilever retaining walls that are required to resist a combination of earth and hydrostatic loading by using only geometric design variables, whereas Yepes et al. (2008) used simulated annealing for optimum design of RC cantilever retaining walls utilized in road construction by using more design variables, effectively leading to more detailed simulations. Khajehzadeh et al. (2010, 2011) proposed a modified particle swarm optimization (MPSO) based on PSO with passive congregation to find the optimum cost design of a cantilever RC retaining wall. Ghazavi & Bazzazian Bonab (2011) applied an Ant Colony Optimization (ACO) algorithm to arrive at optimal design of a RC retaining wall (designed as a gravity wall, i.e. structural integrity is not taken into account while imposing the constraints). Ghazavi & Salavati (2011) presented a bacterial foraging optimization algorithm whereas harmony search based algorithms were proposed by Kaveh & Abadi (2010). Donkada & Menon (2012) applied a genetic algorithm to reach minimum cost design of three types of retaining walls: cantilever retaining wall, counterfort retaining wall and retaining wall with relieving platforms. Pei & Xia (2012) applied a random direction search complex method and three heuristic algorithms (genetic algorithm, particle swarm optimization and simulated annealing) are used to obtain the minimum cost design of a reinforced concrete cantilever retaining wall. Finally, Talatahari et al. (2012) performed optimum design of gravity retaining walls subject to dynamic loading using a charged system search algorithm, while the Mononobe-Okabe method was used to determine the dynamic earth pressures.

Common feature of all the aforementioned studies is the fact that for the design of the retaining wall, dynamic earth pressures are ignored (except for the study by Talatahari et al. (2012) in which they are taken into account in a simplistic way through a pseudostatic approach). In addition, the static earth pressures (resulting from gravity and/or surcharge load) are calculated according to Rankine or Coulomb earth pressure theories which assume that a state of plastic equilibrium is developed in the retained backfill. Moreover, to the authors' knowledge, no suitable constraint has been imposed to any retaining wall optimum design case to ensure that the deformations of the retaining wall and the backfill are within acceptable limits. In most studies, this is ensured implicitly by avoiding the possibility of overturning and sliding, by controlling the stresses within allowable limits and by securing the stability of the retaining wall – retained soil system.

Apart from these, the seismic response of retaining systems is still a matter of ongoing experimental, analytical and numerical research. The dynamic interaction between a wall and a retained soil layer makes the response complicated. The dynamic analysis becomes much more complex, as usually material and/or geometry non-linearities have to be taken into account (Kramer, 1996, Wu & Finn, 1999). Depending on the expected material behavior of the

retained soil and the possible mode of the wall displacement, there exist two main categories of analytical methods used in the design of retaining walls against earthquakes: (a) the pseudo-static limiting-equilibrium solutions which assume yielding walls resulting in plastic behavior of the retained soil (Okabe, 1926, Mononobe & Matsuo, 1929, Seed & Whitman, 1970), and (b) the elasticity-based solutions that regard the retained soil as a visco-elastic continuum (Veletsos & Younan, 1997, Scott, 1973, Wood, 1975). In most studies presented so far, in order to perform optimum design of retaining walls, the assumption of pseudo-static limiting equilibrium is made; there-fore the design is performed in a simplistic way, ignoring the possibility of a linear elastic or viscoelastic soil backfill.

This section is concerned with the optimum design of cantilever retaining walls which are subject to earthquake loading and are responding in a linear elastic way. The objective function which is optimized is the weight of the retaining wall. This is roughly proportional to its construction cost, as the latter is generally an increasing function of the weight of the material used. This function is minimized subject to design constraints. Apart from the usual constraints imposed in most optimization studies, in this section a direct design constraint is imposed which controls the rocking response of the retaining wall. The optimization analysis is conducted via the use of a genetic algorithm, since Pei & Xia (2012) have shown that GA can be successfully applied for the optimal solution of structural optimization problems with many design variables and complex constraints. Two numerical examples are presented, in which optimum designs are performed for two values of the height of the soil layer to be retained.

6.5.2   Numerical modeling

In this section the numerical model used to simulate the dynamic response of a cantilever retaining wall is described. This model consists of an infinite soil layer with horizontal base and free surface which is at higher elevation towards $+\infty$ than $-\infty$. These two elevations result in the existence of a vertical slope of height H which is retained by a cantilever wall. The wall's foundation is at a depth equal to h_{emb}, relative to the downstream soil surface. Consequently, the overall height of the retaining wall stem is $H + h_{emb}$. The retaining wall is considered to rest on a strip foundation which consists of the toe, which is the portion of the foundation extending downstream from the wall, and of the heel, extending in the opposite direction (upstream). The depth of the rigid bedrock from the foundation of the wall is $1.5H$, where H is the thickness of the horizontal layer to be retained, as seen schematically in Figure 6-7. The distance from the wall toe tip to the far field (downstream) vertical boundary of the model is $10H$; the same happens with the distance from the wall heel tip to the far field boundary of the model in the upstream direction. Shown in Figure 6-7 is also the local coordinate system to which all calculations are referred. Text in bold denotes the design variables whereas the others either denote the variables which are dependent on the

design variables or are problem parameters which remain fixed during optimum design.
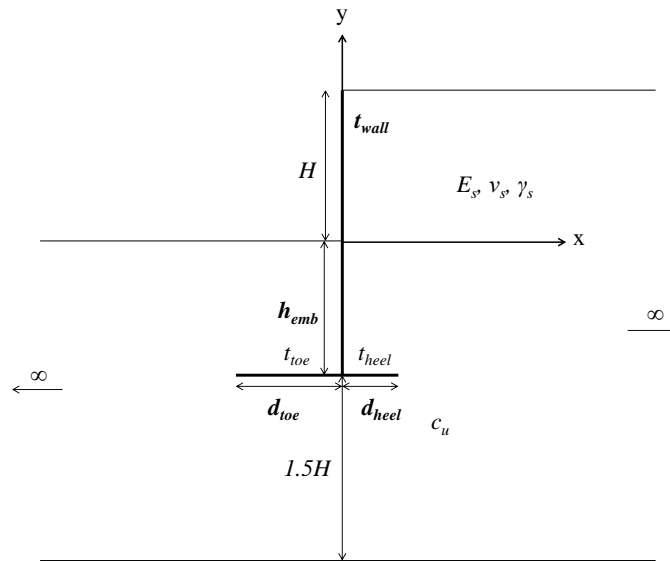
Figure 6-7: Cantilever reinforced concrete retaining wall model.

The soil layer is fixed on rigid bedrock and along the soil – rock interface horizontal and vertical fixity is imposed. In order to simulate sufficiently the one dimensional dynamic soil response, vertical kinematic constraints were used at the two vertical ends of the model. These constraints are different from the corresponding kinematic constraints imposed for gravity loading at the same boundaries, which were in the horizontal direction to simulate one dimensional compression. The two vertical boundaries of the model were placed relatively far from the wall to minimize the influence of the difference between the model response in these regions and one dimensional soil response. The whole model is considered to respond in plane strain condition, an assumption fairly accurate for cantilever retaining walls with length much higher than their width, height and thickness. The wall – soil and the foundation – soil interfaces are considered to be tied, an assumption generally valid for cohesive soils. This means there is no separation or relative slip along these interfaces. Initially, gravity acceleration (body force) is applied to the whole model and in a second step of the analysis, the transverse ground acceleration record recorded during the December 11, 1967 Koyna earthquake, which was of magnitude 6.5 on the Richter scale, is imposed along the base of the soil layer. The time history graph of this record is shown in Figure 6-8.
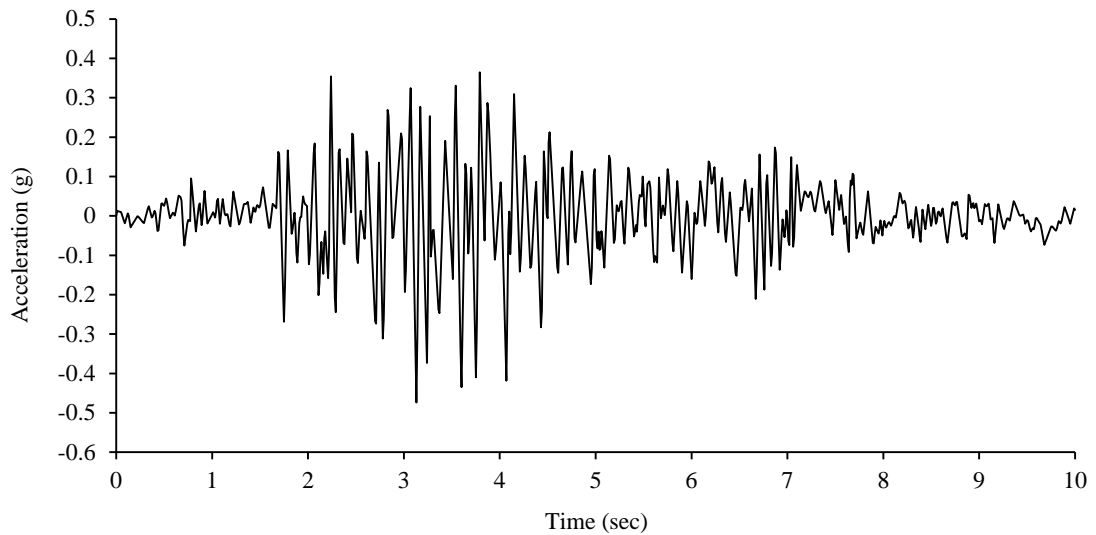
Figure 6-8: Transverse acceleration time history record of the December 11, 1967 Koyna earthquake, of magnitude 6.5 on the Richter scale.
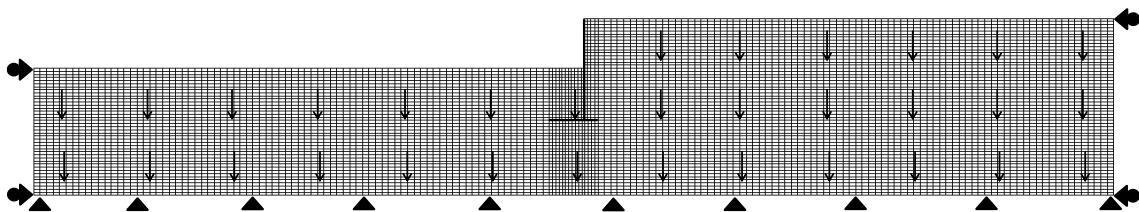
Figure 6-9: Numerical model analyzed for the 1st case (H=8m). Loading and boundary conditions for the initial gravity step are shown.
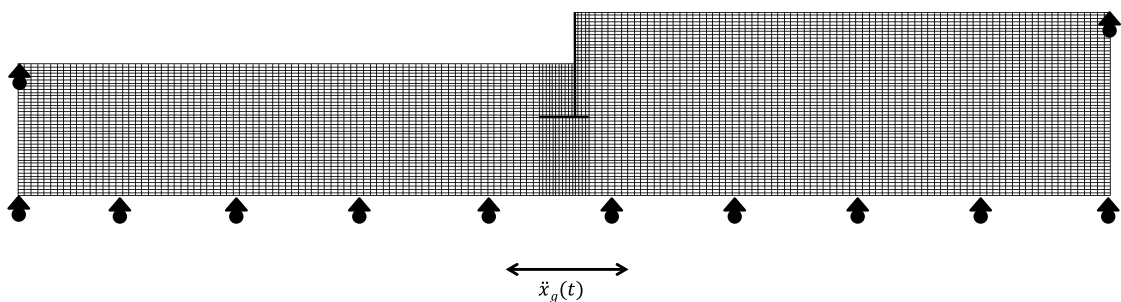
Figure 6-10: Numerical model analyzed for the 1st case (H=8m). Loading and boundary conditions for the main dynamic time – history analysis step are shown.

In this section, in order to minimize the weight of the retaining wall, two-dimensional numerical simulations were performed for the wall-soil systems depicted in Figure 6-9 and Figure 6-10, utilizing the finite element software ABAQUS. The soil layer is discretized with 8-node bi-quadratic plane strain solid elements (CPE8). 3-node quadratic interpolation beam elements in plane (B22) are used for modeling the retaining wall and its foundation. These elements allow

for transverse shear deformation according to Timoshenko theory and in their shear flexible formulation it is assumed that the transverse shear behavior is linear elastic with a fixed shear modulus and, thus, independent of the response of the beam section to axial stretch and bending. For Timoshenko beam elements a lumped mass formulation with a 1/6, 2/3, 1/6 distribution is used. The mesh gets coarser for the part of the soil layer which is left and right of the wall (the horizontal dimension of the elements is double). This is apparent in Figure 6-9 and Figure 6-10.

The eigenmodes used for the modal dynamic analysis are extracted in a previous frequency step, in which the Lanczos eigensolver is used, which is a powerful tool for extraction of the extreme eigenvalues and the corresponding eigenvectors of a sparse symmetric generalized eigenproblem. For the Lanczos eigensolver, the minimum and maximum frequencies of interest are specified and all eigenmodes with eigenfrequencies falling in this range are extracted. These modes are subsequently used for the calculation of the dynamic response during the modal dynamic analysis. Energy dissipation due to damping mechanisms is not modeled explicitly as a material property (e.g. through the simplistic Rayleigh damping approximation), but it is specified as a fraction of critical damping assigned at all eigenmodes included for the calculation of the dynamic response, equal to 5%. Thus the damping fraction remains constant along the frequency range of interest and energy dissipation is of the same intensity for lower and higher frequencies.

6.5.3   Formulation of the optimization problem

In this section the optimization problem to be solved is explained in detail. The design variables, the parameters, the constraints, the objective function and the optimum design process are presented.

6.5.3.1   Design variables

The design variables of the problem are shown in bold in Figure 6-7. These are the depth of the wall embedment denoted by h_{emb}, the width of the toe denoted by d_{toe}, the width of the heel denoted by d_{heel} and the thickness of the wall stem denoted by t_{wall}. The thickness of the wall toe and heel (t_{toe} and t_{heel} respectively) are selected to be the same and equal to the minimum between the wall stem thickness t_{wall} and one tenth of the corresponding widths ($d_{toe}/10$ and $d_{heel}/10$), so that beam modeling for these components is reasonable.

Design variable	Lower limit (m)	Upper limit (m)
h_{emb}	0.2	16

d_{toe}	2	12
d_{heel}	2	12
t_{wall}	0.2	2.5

Table 6-9: Design variables of the optimization problem and their lower and upper bounds.

In the aforementioned design variables upper and lower limits are set, in order to prevent the algorithm from giving technically infeasible solutions. Table 6-9 shows the design variables and their corresponding upper and lower limits.

6.5.3.2   Parameters

The parameters of the wall-soil layer system are all the quantities that remain fixed during a particular optimization search. The parameters of the problem are summarized in Table 6-10. These are the physical properties of the soil and the wall. All materials involved in the model are linear elastic, leading thus to a linear dynamic response. The soil has density γ_s=1800 kg/m3, modulus of elasticity E_s=100 MPa and Poisson's ratio ν_s=0.3. The retaining wall is modeled as a reinforced concrete beam with a general section, density γ_w=2500 kg/m3, and modulus of elasticity E_w=30.5 GPa which corresponds to C25/30. Although the retaining wall has the inertial and stiffness characteristics of concrete, it deforms in a linear elastic way, which implies that its stiffness in tension and compression is equal. Another parameter of the problem is the frequency range used for the modal dynamic analysis; this is selected to be in the range [0.01 Hz, 29 Hz]. The lower limit is selected so that the very low frequency spurious eigenmodes are excluded from the analysis; these are associated with very large modal mass. The higher limit is selected based upon the fact that the lowest wavelength of the waves propagating into the soil (lowest velocity of propagation and highest frequency) has to be at least ten times the internodal interval of the mesh; this is approximately the distance between adjacent nodes, and it increases as the mesh gets coarser.

Parameter	Assigned value
γ_s	1800 kg/m³
E_s	100 MPa
ν_s	0.3
γ_w	2500 kg/m³
E_w	30.5 GPa
f_{min}	0.01 Hz
f_{max}	29 Hz

Table 6-10: Parameters of the optimization problem and their fixed values.

6.5.3.3   Constraints

The constraints of the optimization problem at hand are divided into structural constraints and geotechnical constraints. The satisfaction of the former ensures that the retaining wall does not fail as regards its structural integrity,

whereas the latter ensures that the soil retained by and supporting the wall does not fail. The constraints are shown in Table 6-11, which includes the formulas for the calculation of the limiting quantities and the constraint inequalities imposed for the optimization problem. As far as the structural constraints are concerned, the maximum tensile (and the minimum compressive) stress which develop due to axial force and bending moment at the wall stem, toe and heel must not be higher than (respectively lower than) the material strength. For concrete C25/30 this is 25 MPa by definition, without taking into account the partial safety factor for concrete strength (EC2, 2004). The material model used for the wall in this section is linear elastic and this leads to the essential assumption that the distribution of strains and stresses along the section of the wall and its foundation is linear which results in the presence of "theoretical" tensile stresses which are not present in practice. In practice, there are no tensile concrete stresses and the necessary tensile forces for the equilibrium of the section are provided by the steel reinforcement. In any case, the tensile stress constraint is not active in the final optimum design, as will be described in detail later. Shear stiffness is ignored since shear strength of reinforced concrete cannot be calculated in a theoretically sound basis; the procedure of calculation and the final result is very norm specific in general. Except for this, it depends highly on the reinforcement and its distribution into the beam. Concerning the geotechnical constraints, the following are specified:

a)   It is ensured that the normalized displacement at the top of the wall stem θ does not exceed 0.33%. The normalized displacement is given by the ratio of the horizontal displacement at the top of the wall due to tilting or horizontal translation, divided by the height of its stem including embedded part ($H + h_{emb}$). The above inequality is specified to prevent the development of a limit state or the initiation of a failure plane in the retained soil (Clough & Duncan, 1991). In the opposite case the assumption of a linear elastic soil would not be accurate. It is assumed that, as far as its strength is concerned, the supporting and retained soil behaves like compacted clay. According to Clough & Duncan (1991), the values of normalized displacement required to reach active and passive earth pressure conditions are 1% and 5% respectively. By ensuring that the normalized displacement is lower than 0.33% (conventionally taken as one third of the normalized displacement required for active conditions) neither active nor passive states will develop in the soil.

Quantity	Formula	Constraint
Normalized displacement of wall stem	$\theta = \max[abs\{Displ_{top}/(H+h_{emb})\}]$	$\theta \leq 0.33\%$
Undrained shear strength	$c_u = E_u/850 = 3 \cdot E_s/\{2 \cdot (1+v_s)\}/850 = 136$ kPa	$\max(\tau) \leq c_u$ $\min(\tau) \geq -c_u$
Soil bearing capacity	$q_u = 5.14 c_u + \gamma_s \cdot h_{emb}$	$-\min(\sigma_{yy}) \leq q_u$

Foundation uplift		$\max(\sigma_{yy}) \leq 0$
Max bending stress*	$\sigma_{max} = \max(N/t + 6M/t^2)$	$\sigma_{max} \leq 25$ MPa
Min bending stress*	$\sigma_{min} = \min(N/t - 6M/t^2)$	$\sigma_{min} \geq -25$ MPa

* For the wall stem, toe and heel. t denotes t_{stem}, t_{toe} and t_{heel}.

Table 6-11: Constraints of the optimization problem.

b)   Regarding its strength, the soil is assumed to behave as a cohesive soil in undrained conditions. So, its shear strength in terms of total stresses is equal to its undrained shear strength cu, i.e. the ϕ=0 approach is followed. Thus, it is specified that the maximum and minimum shear stress along the wall foundation must not exceed the undrained shear strength of the underlying soil, equal to 136 kPa. This value is calculated as follows: the undrained modulus of elasticity E_u of the soil is calculated according to Table 6-11 to be E_u=115.38 MPa. The fraction E_u/c_u according to data available in the literature (Jamiolkowki et al., 1979 and Jardine et al., 1985) is selected to be roughly 850.

c)   The bearing capacity of the soil underlying the foundation must not be surpassed. For this purpose, the bearing capacity under undrained loading (c = c_u, ϕ = 0) is calculated according to the Meyerhof formula for vertical and central loading of horizontal strip foundation at a depth equal to h_{emb}. The maximum vertical normal stress at the lower interface of the wall foundation must not get larger than this value.

d)   Along the interface where the aforementioned constraint (c) is imposed, there must also be no tension, otherwise there would be foundation uplift which would render the dynamic response of the retaining wall geometrically non-linear.

The approach followed to impose the constraints is the penalty method. Penalty methods add a penalty to the objective function to decrease the quality of infeasible solutions. The penalty quantities that are added are virtually the product of the constraint violation and a penalty factor which is fixed for each constraint and adjusted to take into account the relative importance of the constraint violations.

6.5.3.4   Objective function

The objective function to be minimized is the volume of the retaining wall per meter in the longitudinal direction. This is proportional to its weight and indirectly related to its cost of construction. The fitness function which is minimized by the genetic algorithm used in this section results from the objective function after the application of the penalties due to constraint violations, if any.

6.5.4   Optimization algorithm

The Genetic Algorithm (GA) is a stochastic global search optimization method that emulates natural biological evolution. GAs apply on a population of potential solutions the principle of survival of the fittest to produce better approximations to a solution. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics (crossover, mutation, etc.). This process leads to the evolution of individuals that are better suited to their environment than the individuals that they were created from, just as in natural evolution process. In order to minimize the objective function, the genetic algorithm implemented in MATLAB software was used.

The encoding strategy followed is real-valued representation. The use of real-valued genes in GAs offers a number of advantages in numerical function optimization over binary encodings: (a) efficiency of the GA is increased as there is no need to convert chromosomes to phenotypes before each function evaluation, (b) less memory is required as efficient floating point internal computer representations can be used directly, (c) there is no loss in precision by discretization to binary or other values and (d) there is greater freedom to use different genetic operators.

The population size (number of individuals in each generation) is equal to 20. The initial population with which the GA begins is created as a random initial population with uniform distribution. Fitness scaling was implemented by using a rank function, which scales the raw scores based on the rank of each individual instead of its score. The rank of an individual is its position in the sorted scores. An individual with rank r has scaled score proportional to $\sqrt{r}$. Rank fitness scaling removes the effect of the spread of the raw scores. The square root makes poorly ranked individuals more nearly equal in score, compared to rank scoring.

Regarding the basic genetic operators, stochastic uniform selection is used for the selection process. In this function each parent corresponds to a section of the line of length proportional to its scaled value. The algorithm moves along the line in steps of equal size. At each step, the algorithm allocates a parent from the section it lands on. The first step is a uniform random number less than the step size. For reproduction, the number of individuals that are guaranteed to survive to the next generation (elite children) is 2 and the fraction of the next generation, other than elite children, that is produced by crossover (crossover fraction) is equal to 0.8. The mutation function used is Gaussian, which adds a random number taken from a Gaussian distribution with mean 0 to each entry of the parent vector. For the combination of parents to produce the next generation offspring (crossover), scattered crossover is used, which applies in problems without linear constraints. It creates a random binary vector and selects the

genes where the vector entry is 1 from the first parent, and the genes where the vector entry is 0 from the second parent, and combines the genes to form the child. In the GA no migration occurs, as there are no subpopulations.

As stopping criteria for the algorithm the following were specified: (a) the maximum number of iterations for the genetic algorithm to perform is equal to 100, (b) the algorithm stops if the weighted average relative change in the best fitness function value over 50 generations is less than or equal to the function tolerance (equal to 10^{-6}).

The following outline summarizes how the GA procedure works:

a) The algorithm begins by creating a random initial population.

b) The algorithm then creates a sequence of new populations. At each step, the algorithm uses the individuals in the current generation to create the next population. To create the new population, the algorithm performs the following steps:

1. Scores each member of the current population by computing its fitness value.

2. Scales the raw fitness scores to convert them into a more usable range of values.

3. Selects members, called parents, based on their fitness.

4. Some of the individuals in the current population that have better fitness are chosen as elite. These elite individuals are passed to the next population.

5. Produces offspring from the parents. Offspring are produced either by making random changes to a single parent (mutation) or by combining the vector entries of a pair of parents (crossover).

6. Replaces the current population with the offspring to form the next generation.

c) The algorithm stops when one of the stopping criteria is met.

The GA optimizer of Matlab is properly coupled with the Abaqus analysis solver in order to take the modal dynamic analysis results, using Abaqus2Matlab (Papazafeiropoulos et al., 2017). This is done inside the objective function in which the analysis solver is called to perform the necessary analyses. Except for this, suitable functions are called to create the necessary input (*.inp) files to conduct the analyses and read the results of the analyses from the corresponding results (*.fil) files. While the analysis solver is running the optimizer is halted and its execution is continued after the ABAQUS lock (*.lck) file has been deleted. Constraint enforcement is applied through an advanced penalty method and not by the default constraint handlers developed in MATLAB.

6.5.5   Conventional seismic design of cantilever retaining walls

Conventional seismic design of cantilever reinforced concrete retaining walls is achieved with use of the well-known Mononobe-Okabe (M-O) theory of seismic earth pressures (Okabe, 1926 and Mononobe & Matsuo, 1929). Design is performed regarding the wall stability (sliding and overturning about the tip of its toe), and the design variables are the wall embedment (h_{emb}), the width of the wall toe (d_{toe}) and the width of the wall heel (d_{heel}). The thickness of the wall stem and foundation (toe and heel) are selected based on general guidelines for initial wall dimension proportioning, i.e. they are set equal to 1/10 of the total wall height ($H + h_{emb}$). According to the dimension proportioning practice, the inequality $0.3(H + h_{emb}) \leq d_{heel} \leq 0.5(H + h_{emb})$ must hold for the width of the wall heel. The minimum values of the wall embedment, toe width and heel width are set equal to 0.2 m, 2 m and 2 m respectively. The conventional seismic design method implemented in this section involves also some kind of optimization procedure which leads to the minimum total weight of the wall by strict abidance by all of the constraints mentioned above.

A necessary step during the design process is the evaluation of the soil internal friction angle φ and the soil – wall interface friction angle δ. These are set to the following typical values: $\varphi = 30^o$, $\delta = 18^o$. The horizontal acceleration coefficient is taken as $k_h = 0.48$ (resulting from the maximum acceleration of the earthquake record which is 0.48g). These properties are assigned to the soil lying over the wall foundation. For the soil under the wall foundation undrained response is assumed, namely its internal friction angle is taken equal to zero and its undrained shear strength c_u is specified in section 6.5.3. The inertial forces and moments of the wall are also taken into account for the design whereas, regarding overturning checks, only the upstream and downstream soil portions lying over the wall foundation are considered to contribute to the wall stability.

6.5.6   Optimization results

Two retaining wall weight optimization cases were examined: in the first case (Case 1) the height of the soil layer to be retained by the wall is equal to 8 m and in the second case (Case 2) the height is 12 m. The results of the GA optimization procedure as analyzed in the previous sections are shown in Table 6-12. It is observed in general that the embedment and foundation dimensions required to retain the soil layer with greater height (Case 2) are larger than those in Case 1. The optimum value of the objective function increases as well. In both cases, the length of the wall heel (d_{heel}) which leads to optimum design is the minimum possible, i.e. equal to its lower bound. This means that for the parameter values and earthquake record considered in this section the heel does not contribute significantly to the retaining wall stability and/or structural integrity. The thickness of the heel is constrained by the requirement that it is not more than

one tenth of its length (to justify its modeling as a beam), whereas the thicknesses of the other components (stem and toe) are equal.

Table 6-13 shows the results of the conventional seismic design according to the M-O method. It is observed that the conventional design which is adopted in most seismic norms worldwide leads to larger weight of the retaining wall. Although the two methods stem from essentially different assumptions, their comparison shows clearly the fact that more economical and simultaneously safe designs can be achieved by applying detailed optimization methods for the seismic design of retaining walls; current seismic code practices can lead to unreasonably conservative designs.

As far as the constraints are concerned, the maximum and minimum normal stresses of the heels of the two walls do not differ much. On the contrary, the maximum normal stress of the toe differs by a factor greater than 2 between the two wall cases. Furthermore, the minimum shear stress along the lower interface between the wall foundation and the supporting soil is roughly the same for the

	Case 1 (H=8m)	Case 2 (H=12m)
Design variables		
h_{emb} (m)	7.76	7.16
d_{toe} (m)	4.57	6.57
d_{heel} (m)	2.00	2.00
t_{wall} (m)	0.20	0.22
t_{toe} (m)	0.20	0.22
t_{heel} (m)	0.20	0.20
Constraint quantities		
θ	**0.328%**	0.246%
maxτ (kPa)	78.42	88.46
minτ (kPa)	-131.96	**-135.73**
minσ_{yy} (kPa)	-505.98	-592.45
maxσ_{yy} (kPa)	-122.07	-124.53
$\sigma_{b,s,max}$ (kPa)	22570.73	21476.85
$\sigma_{b,t,max}$ (kPa)	4265.33	1183.35
$\sigma_{b,h,max}$ (kPa)	2029.97	2180.68
$\sigma_{b,s,min}$ (kPa)	-23795.84	-23385.15
$\sigma_{b,t,min}$ (kPa)	-7336.37	-8128.82
$\sigma_{b,h,min}$ (kPa)	-2176.36	-2501.06
Algorithm details		
Min. value of obj. fun. (m²)	4.47	6.12
Number of generations	73	64
Number of fun. evaluations	1480	1300

Table 6-12: Results of the optimization procedure of the two retaining wall cases.

two wall cases. This observation implies that minimum shear stress along the lower interface of the wall foundation is independent of the wall height.

Furthermore, the constraint imposed for this quantity is active in Case 2. The above may provide a hint for controlling the optimization process.

For more details about the problem discussed in this section, the reader is referred to publication [19] in the Appendix A of this thesis.

	Case 1 (H=8m)	Case 2 (H=12m)
h_{emb} (m)	0.20	0.20
d_{toe} (m)	5.58	8.43
d_{heel} (m)	4.10	6.10
t_{wall} (m)	0.73	1.32
t_{toe} (m)	0.73	1.32
t_{heel} (m)	0.73	1.32
Area (m²)	13.05	35.28

Table 6-13: Results of the conventional seismic design of the two retaining wall cases.

6.6 Calibration of tyre material properties based on an optimization procedure

6.6.1 Introduction

Handling low frequency interior noise and vibrations which transmits through subframe components on vehicles is a main issue regarding their design. The importance of this aspect is apparent from the related legislation which limits the level of noise a vehicle is allowed to produce. The main source of the vehicle noise is the vibrations induced by the tyres. These, after being transmitted from the tyre to the wheel axle, and through that to the passengers in the vehicle, can have various undesirable effects, some of which are the passengers' inconvenience or body distress, the low performance of the vehicle and its suspension system, etc. A tyre is subjected to dynamic forces mainly from two main sources: (a) road surface irregularities, potholes, bumps and various other obstacles which impose dynamic loads to the tyre, and (b) dynamic loads originating from various nonuniformities of the tyre, such as slight imbalances or asymmetric tread pattern designs.

It is essential to consider the dynamic characteristics of the tyres of a vehicle, to minimize the aforementioned negative consequences. For this purpose, there is need for detailed knowledge of the dynamic response of a tyre, which is associated with the energy that is being transmitted to the vehicle from various external dynamic events. The dynamic response of a tyre is characterized by its vibration modes, or eigenmodes, namely the natural frequencies of the tyre and the corresponding mode shapes. These, apart from their significance for the design process and troubleshooting of various problems, can constitute a basis for the computational efficiency of the various numerical models of tyres used by both tyre and automotive industries for prediction of performance.

Tyre vibration modes are widely used over the years to represent dynamics in tyre models. The dynamic response of tyre models has been studied analytically, experimentally or semi-empirically, and numerically, however due to the limitations of the analytical and experimental studies, many studies in the literature employ numerical (often finite element) models, which can simulate complex geometries as well as material, geometric and boundary nonlinearities. Relevant studies about tyre dynamics, as well as optimization procedures are mentioned in the next.

Experimental studies about the eigenmodes analysis of tyres have been presented by Scavuzzo et al. (1993), Bandel & Monguzzi (1983) and Matsuoka & Okuma (2002). In the study by Scavuzzo et al. (1993), the dynamic response of the vehicle in terms of accelerations was mon-itored at the wheel axis and the passenger compartment. The tyre vibration modes were identi-fied from the peaks in the response. Bandel & Monguzzi (1983) developed a lumped parameter model to study the behaviour of a tyre running on a road surface with irregularities charac-terized by short wave-length spectrum components. However, the parameters of the lumped model are given by empirical relations, which have resulted from an experimental methodolo-gy. Matsuoka & Okuma (2002) presented an experimental modal parameter estimation method in which the frequency response function (FRF) of a tyre is decomposed into the components of individual modes based on the Fourier transform algorithm.

The analytical models developed for the estimation of the eigenproperties of a tyre range from simple mass/spring systems to various forms of idealized, spring supported, flexible rings. Representative studies are these by Vinesse (1996), where a rotating and vibrating tyre coupled at its spindle to a secondary structure is simulated. A model of a membrane on an elastic foundation is used for the description of the vibration of a rolling tyre, as well as models for the calculation of the forces at the spindle of a tyre rolling over a small cleat. In the study by Molisani (2004) the tyre is modelled as a shell structure in contact with the road surface. The contact patch is simulated as a prescribed deformation, and the coupled tyre-cavity governing equation of motion is solved analytically to obtain the tyre structural and acoustic responses.

Representative numerical studies regarding the modal analysis of tyres are those byWheeler et al. (2005), Dorfi et al. (2005), Chatterjee & Ranjan (2012) and Bolarinwa & Olatunbosun (2015). Wheeler et al. (2005) presented the vibration modes of radial tyres on a fixed spindle and investigated the effect of the tyre components and their contribution in the mode shapes. Following that, the corresponding tyre model under rolling conditions was considered by Dorfi et al. (2005) and it was shown that non-rolling tyre models are subordinate to their roll-ing counterparts, as they do not take into account the proper kinematics. The finite element commercial software ANSYS was used by Chatterjee & Ranjan (2012) to study the effects of the inflation pressure, the ply angle, the tread pattern and the thickness of the belt on the natu-ral frequencies of the tyre. A

basic assumption in this section was that the rubber was simulated as a linear elastic material. Another commercial finite element software (ABAQUS) has been used by Bolarinwa & Olatunbosun (2015) where by using various capabilities of ABAQUS, the footprint under purely vertical load was obtained for a vertically loaded tyre. Afterwards, the nodes (node coordinates) being in contact with the road were maintained in contact by applying an equivalent distributed vertical load, whereas the centre of the wheel was set free in all degrees of freedom. In this condition of the model, a frequency analysis was performed and it was found that the boundary conditions on the tyre model can have large impact on its eigenmode response.

In this section the eigenmodes at the low frequency range are considered for the development of a realistic tyre model, based on numerical data published in the literature. This is achieved through an optimization process which efficiently adjusts various tyre parameters, so that the eigenmodes of the final tyre model reach the corresponding data as close as possible. Optimization methods that are based on simple mathematical programming (exact) are very efficient for cases with a few design variables. Methods belonging to this category are those using the sequential quadratic programming procedure for nonlinear optimization (used in this section), as well as others. More details regarding these methods are presented by Nocedal & Wright (2006).

6.6.2   Numerical modeling

The tyre considered for the optimization study is modelled in the commercial finite element code ABAQUS 6.13. Implicit integration was performed using ABAQUS/Standard, which was also used for the eigenfrequency and eigenmode extraction of the tyre. The optimization procedure, as well as the necessary coupling with ABAQUS, was implemented in MATLAB programming language.

The cross section of the tyre, P235/75R17, is shown in Figure 6-11. The tyre is comprised of the belt region, the tread region and the side walls which are being modelled with a hyperelastic material, representative of rubber. The hyperelastic material is simulated by the one term polynomial strain energy potential (Mooney-Rivlin model) with one term Prony series to account for viscoelasticity (Bekakos et al., 2016). The belt region contains reinforcement of two layers (illustrated as Belt layer 1 & 2 in Figure 6-11), and the reinforcement of carcass. The last extends over the belt region and it covers the side walls. Both belt layers and the carcass are discretized with surface elements with twist (SMFGAX1). The rim is discretized with 2-node, linear links for axisymmetric planar geometries (RAX2), and the belt, bead, sidewall and tread regions are discretized with 4-node bi-linear, reduced integration elements with hourglass control (CGAX4R). The nodes of the surface elements of the carcass share the same nodes with those of the belt region elements. If separate nodes are used for these two sections (which have the same coordinates) numerical instabilities may occur during the analysis.
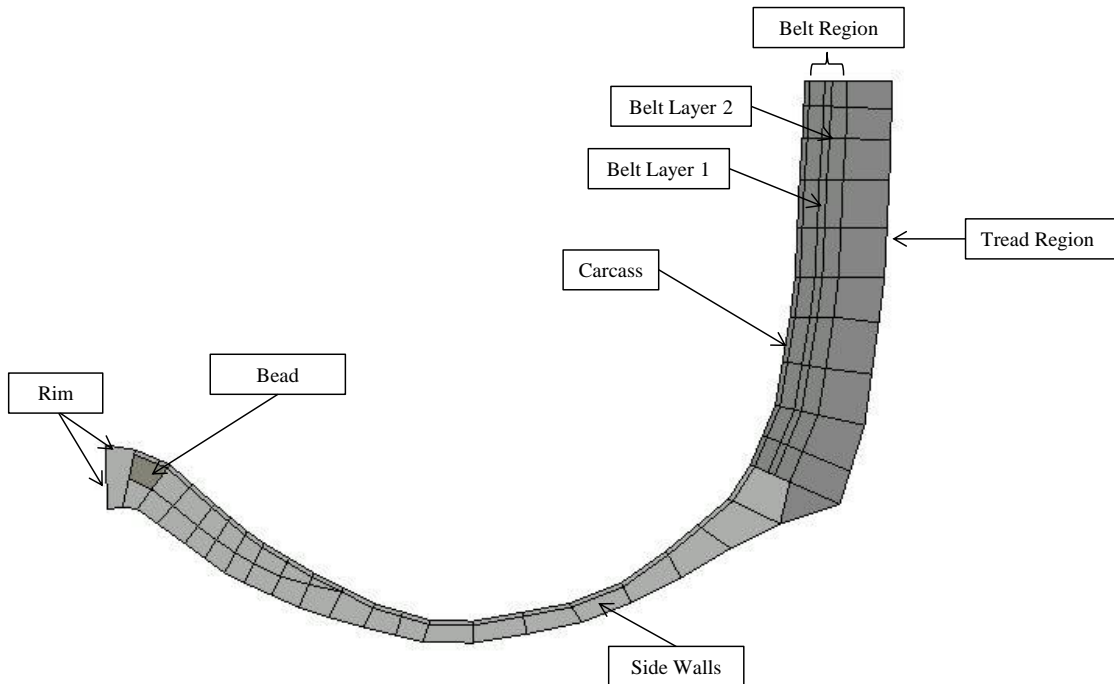
Figure 6-11: Tyre half-cross section geometry.
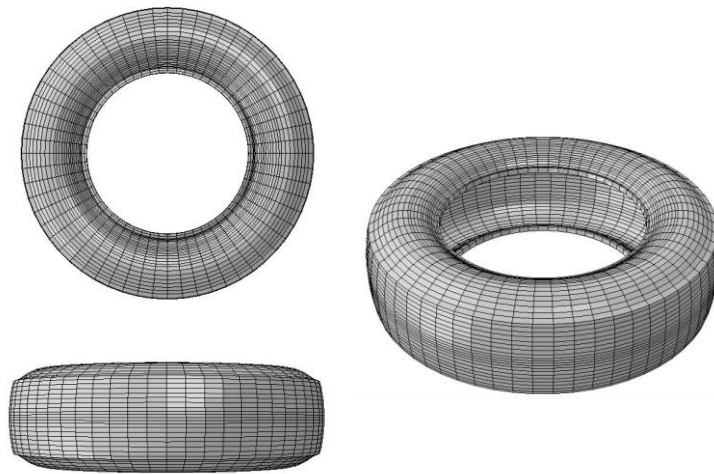
Figure 6-12: Illustration of the tyre model.

By utilizing the capabilities of ABAQUS with regard to symmetric model generation (SMG), symmetric results transfer (SRT) and restart option, the full 3d numerical model of the tyre is developed, as shown in Figure 6-12. Inflation pressure is imposed on the inner surface of the tyre as a distributed load. Regarding the boundary conditions, two cases can be distinguished: (a) for the unloaded tyre, the boundary conditions are imposed on the six degrees of freedom of the wheel centre (fixed-spindle), and (b) for the loaded tyre, the road is considered to be fixed and the tyre centre is constrained along all degrees of freedom except for the degree of freedom along which the vertical load is

imposed. The rim is rigidly constrained to the tyre centre. The friction between the tyre and the road (in the case of the loaded tyre) is assumed to be of Coulomb type, with coefficient equal to 0.5.

6.6.3   Formulation of the optimization problem

6.6.3.1   Design variables

The geometric properties of the belts and carcass reinforcement, as well as the hyperelastic Mooney-Rivlin C_{10} constant are selected as design variables. The reinforcement layers are defined in ABAQUS as smeared layers with a thickness equal to the ratio of the area of each reinforcing bar to the reinforcing bar spacing. This calculated thickness is assumed to remain constant all over the extent of the layer. This consideration has a considerable effect on the selection of the design variables, since the stiffness of each reinforcement layer contributes to the eigenproperties of the tyre. Due to the fact that the rebar stiffness is given by a fraction of two separate input parameters, for constant layer stiffness they become dependent on each other. Therefore, it is objective that only one of the two parameters for each layer is selected as an independent design variable, and the other remains fixed. The variable to remain fixed is the easier to be measured, in terms of order of magnitude. Another point to be mentioned is that, because the two belt layers have symmetric orientation with respect to the plane of the tyre, and the tyre is a centre symmetric structure, its eigenmodes are expected to be also symmetric; this means that the cross section areas of the two belt reinforcements have to be equal, and therefore the belt reinforcement cross sectional area was considered as a single design variable. The design variables of the optimization problem, along with their upper and lower bounds are shown in Table 6-14.

Design variable	Lower bound	Upper bound
A_{belt}	10^{-7}	10^{-5}
$A_{carcass}$	10^{-8}	10^{-5}
C_{10}	10^{5}	10^{7}

Table 6-14: Design variables of the optimization problem and their lower and upper bounds.

6.6.3.2   Parameters

The parameters of the optimization problem are the design input data that remain fixed during the optimization process. These include, as mentioned in the previous section, the spacing of the rebar layers, which is set to be equal to 0.00116m for the belts and 0.001m for the carcass. Furthermore, the constants of the Mooney-Rivlin strain energy potential are $C_{01}=0$ and $D_{1}=5.085E-8Pa^{-1}$. The cord angles are 70 and 110 degrees for the two belt layers, and 0 degrees for the

carcass. The material properties of the belts and the carcass are also held fixed during the optimization process. More details about these properties can be found in Bekakos et al. (2016). The inflation pressure with which the tyre is inflated is 240kPa.

6.6.3.3 Constraints

No constraints are imposed to the model being optimized, apart from the upper and lower limits of the design variables. The latter require some experience to be specified, because large upper bounds or small lower bounds can lead to numerical instabilities in the solver, such as excessive element distortion, etc, which result in the premature termination of the optimization procedure.

6.6.3.4 Objective function

The objective function for the optimization problem has to be of an appropriate form, so that it becomes minimum if the numerically calculated eigenfrequencies coincide with the ones available from the literature. The first 16 eigenfrequencies of the tyre are considered in the objective function, which is given by the following equation:

$$obj = \sqrt{\sum_{i=1}^{16}\left(f_{i,num} - f_{i,lit}\right)^2} \tag{6.6}$$

where $f_{i,num}$ is the ith eigenfrequency calculated by the numerical model in every iteration of the algorithm and $f_{i,lit}$ is the corresponding ith eigenfrequency available in the literature. The correspondence between the various eigenfrequencies is made by taking into account the deformed configurations of the various eigenmodes.

6.6.4 Optimization algorithm

The optimization algorithm used in this section uses a sequential quadratic programming (SQP) method. In this method, a quadratic programming (QP) subproblem is solved at each iteration. For this purpose the MATLAB built in function fmincon is used. This function used an active set strategy and updates an estimate of the Hessian of the Lagrangian at each iteration using the BFGS formula. An active-set method initializes by making a guess of the optimal active set, and if this guess is incorrect, it repeatedly uses gradient and Lagrange multiplier information to proceed towards the optimum solution.

The fmincon optimizer (MATLAB) is properly coupled with the analysis solver (ABAQUS) in order to take the frequency analysis results using Abaqus2Matlab. This is done inside the objective function in which ABAQUS is called to perform the necessary analyses. Except for this, the necessary input (*.inp) files for the ABAQUS runs are created by suitable MATLAB functions. To read the results of the analyses from the corresponding ABAQUS results (*.fil) files, special MATLAB functions are used. While the analysis solver is running the optimizer is halted and its execution is continued after the lock (*.lck) file has been deleted.

6.6.5   Optimization results

The results of the optimization process as presented in the previous sections are shown in Table 6-15.

	Initial model	Optimised model	Wheeler et al. (2005)	Deviation (%)
Design Variables				
A_{belt} (m²)	$2.11868*10^{-7}$	$3.64826*10^{-7}$	N/A	-
$A_{carcass}$ (m²)	$4.20835*10^{-7}$	$8.01133*10^{-8}$	N/A	-
C_{10} (Pa)	10^6	$10^6 +0.01489$	N/A	-
Eigenfrequencies				
f_1 [0,0] (Hz)	36.85	30.86	31.7	2.66
f_2 [0,0] (Hz)	37.17	35.85	35	2.43
f_3 [1,1] (Hz)	43.85	36.92	37.8	2.33
f_4 [1,1] (Hz)	43.85	36.92	37.8	2.33
f_5 [1,0] (Hz)	65.07	58.75	58.5	0.43
f_6 [1,0] (Hz)	65.07	58.75	58.5	0.43
f_7 [2,1] (Hz)	76.33	68.41	66.1	3.49
f_8 [2,1] (Hz)	76.33	68.41	66.1	3.49
f_9 [2,0] (Hz)	86.65	78.67	79.5	1.04
f_{10} [2,0] (Hz)	86.65	78.67	79.5	1.04
f_{11} [3,0] (Hz)	104.36	96.42	97.6	1.21
f_{12} [3,0] (Hz)	104.36	96.42	97.6	1.21
f_{13} [3,1] (Hz)	117.07	107.9	102.7	5.06
f_{14} [3,1] (Hz)	117.07	107.9	102.7	5.06
f_{15} [4,0] (Hz)	122.65	114.9	115.9	0.83
f_{16} [4,0] (Hz)	122.65	114.9	115.9	0.83
Algorithm Details				
Min. value of obj. function	-	8.59	-	-
Number of obj. function evaluations	-	25	-	-

Table 6-15: Results of the optimization procedure of the tyre frequency analysis considered in this section.

It is noted that each natural frequency corresponds to a pair of integers enclosed in brackets ([c, m]). The first integer denotes the number of sinusoidal waves in the circumferential direction of the wheel, whereas the second integer shows the number of waves in the meridional direction at a specific location, where the deformation of the eigenmode shape is maximum. In addition, only the first 16 eigenmodes were considered for the development of the realistic tyre model, in order to reduce the computational cost.

The first column of Table 6-15 shows the data of the initial model, used as the starting point of the optimization process. It is evident that the eigenfrequencies of the initial model have large difference from the eigenfrequencies of the model published by Wheeler et al. (2005). In the second column, the parameters of the optimum model are shown, as well as the values of the design variables corresponding to it. Regarding the eigenfrequencies, it is observed that they are much closer than those of the initial model, leading thus to a numerical model that conforms more to the available numerical data, and therefore it is more realistic. The maximum deviation of the eigenfrequencies is noted to be roughly 5%. The optimum model has higher cross section of the reinforcement of the belts, and lower cross section area of the reinforcement of the carcass than the initial model. The hyperelastic constant C_{10} is only slightly increased after the optimization. Regarding the algorithm output, the minimum value of the objective function is equal to approximately 8.59Hz, and the algorithm converged after 25 objective function evaluations. The reason for the termination of the algorithm is that the magnitude of the search direction was less than the corresponding tolerance. The most important factor affecting the tyre modal behaviour during the optimization procedure is proved to be the cross section area of the carcass ($A_{carcass}$). Due to the fact that the initial model has generally higher eigenfrequencies than those of the target model (Wheeler et al., 2005), its stiffness had to be decreased, in order for the model to approach the latter. The decrease in stiffness is achieved with a relatively large decrease in the cross sectional area of the carcass, although the cross section area of the belt reinforcement increases.

For more details about the problem discussed in this section, the reader is referred to publications [8,10,11,15,16,17] in the Appendix A of this thesis.

6.7   Other applications of Abaqus2Matlab

In this chapter the use of Abaqus2Matlab software has been illustrated for the solution of optimization problems which involve the dynamic properties or dynamic response of structures. However, its use is not limited in this field, since Abaqus2Matlab provides a convenient integration framework between ABAQUS and Matlab and vice versa which can be used in many more engineering disciplines. Abaqus2Matlab has already been used in the literature for the following topics:

6.7.1   Optimum design against buckling of plate girders with multiple longitudinal stiffeners subject to combined bending and shear

Abaqus2Matlab is used to optimize the position of the longitudinal stiffeners of a multi-stiffened steel plate subject to combined bending and shear, so that the buckling coefficient of the plate is maximized. For this purpose, a model of the

steel plate is developed in ABAQUS, and an optimization procedure is implemented in Matlab. The ABAQUS model performs buckling analysis of the steel plate model which is automatically generated by suitable Abaqus2Matlab scripts for specific values of the design variables. The Matlab codes use a gradient-based interior point algorithm (IPA) which involves a direct Newton step and a Conjugate Gradient step to proceed towards the optimum solution. Abaqus2Matlab effectively coupled the solver (ABAQUS) and the optimizer (Matlab) functions of this complex optimization model by providing a successful linking between the various routines. This optimization procedure not only verified existing relevant results in the literature (Rockey & Cook, 1965 and Alinia & Moosavi, 2008) but also led to successful optimum designs for various configuration and loading conditions of the steel plate considered. The new designs achieved lead to significantly lower amounts of steel material required for construction, reduced by at least 61.76%), while at the same time structural safety is increased by as much as 180%. For more details about this application, the reader is referred to publications [2,3,4] in the Appendix A of this thesis.

6.7.2   Prediction of buckling coefficient of stiffened plate girders using deep learning algorithm

A Deep Learning- (DL-) based procedure for the prediction of the critical buckling coefficient of longitudinally stiffened web plate girders subjected to pure bending is developed. Datasets, consisting of input data (various geometric dimensions of the girder) and output data (critical buckling coefficient), are generated from eigenvalue buckling analyses in ABAQUS. In this procedure Abaqus2Matlab is used for the data transfer between Matlab and ABAQUS and vice versa, within a looped procedure for the generation of the training data of the Deep Learning network that is developed and verified. 2,200 training data are employed to establish the model to predict the buckling coefficient using deep learning. The number of hidden layers and the number of neurons in each layer, optimizer and activation function are chosen so that the metamodel is optimized for the given training data. Finally, 200 test data are utilized to estimate the model accuracy. The efficiency of the DL model is verified by comparison of its results with those obtained from the literature which showed a good agreement. For more details about this application, the reader is referred to publication [12] in the Appendix A of this thesis.

6.7.3   A computational method for performing nonlinear adaptive pushover analysis of structures through ABAQUS simulation

A computational method which uses the nonlinear adaptive static (pushover) analysis to evaluate the static force-deformation response of a planar moment resisting frame (MRF) for both monotonic and cyclic response has been developed. The MRF considered in this section is a seismic-resisting frame of a prototype five-storey five-bay steel building structure designed based on the

Japanese seismic design code. The frame model is simulated by the finite element software ABAQUS. Failure on members is captured by adopting shell elements for beams and columns combined with a refined meshing. The lateral force distribution is adapted during the pushover analysis according to the first eigenmode of the structure by utilizing a novel inverse optimization algorithm. A new stop analysis criterion is introduced that overcomes the numerical difficulty of the available static-solution algorithms to terminate the analysis in the degrading region of the load-deformation response. The effects of load distribution and first (fundamental) small strain eigenperiod on the force-deformation pushover curve are studied. The monotonic adaptive pushover procedure is implemented using the programming language MATLAB. The new tool effectively combines the advanced modeling and analysis capabilities of ABAQUS with the programming simplicity of MATLAB, thus leading to a user-friendly environment. The last offers a robust implementation of pushover analysis and superior numerical results can be obtained, especially the descending branch of the pushover curve and collapse mechanism. For more details about this application, the reader is referred to publication [13] in the Appendix A of this thesis.

6.8   Notation

A_{min} : Minimum cross section area of truss members

d_{heel} : Width of the wall heel

d_{toe} : Width of the wall toe

$f(x)$: Objective function

$f_{i,lit}$: Experimental value of the ith eigenfrequency of the tyre

$f_{i,num}$: The ith eigenfrequency of the tyre calculated numerically

$g_i(x)$: Inequality constraint function ($i = 1,...,m$)

H : Height of vertical slope retained by the retaining wall

h_{emb} : Depth of the wall's foundation

$h_j(x)$: Equality constraint function ($j = 1,...,p$)

k_h : Horizontal acceleration coefficient

obj : objective function for fitting dynamic properties of the Abaqus tyre model

P : Axial force of truss

r : rank of an individual in a population of the genetic algorithm

t_{heel} : Thickness of the wall heel

t_{toe} : Thickness of the wall toe

t_{wall} : Thickness of the wall stem

x_k : Design variables ($k = 1,...,n$)

δ : Soil – wall interface friction angle.

λ : Convergence parameter

σ : Axial stress

σ_{max} : Maximum (absolute) axial stress

φ : Soil internal friction angle

Chapter 7

■ ■ ■ ■ ■

7 Conclusions

7.1   Original contribution of the thesis

The objective of the present thesis was to develop algorithms and methodologies for the optimum design of structures which respond due to dynamic (seismic) loading. Also, the necessary numerical tools for the implementation of the new computational techniques are provided. These problems have some unique characteristics that are described in the following. Also, the contribution of the present thesis to each one of these problems is described in detail in the next sections.

7.1.1   Development of a family of advanced direct time integration algorithms for nonlinear dynamic analysis

Direct time integration (or time stepping, or step by step) methods are a widely used approach to solve dynamic linear or nonlinear response analysis problems. They have to satisfy certain criteria in order to be suitable for the integration of the differential equation of motion in the linear or nonlinear regime. In linear dynamic response, emphasis is given in accuracy, whereas in nonlinear dynamic response numerical stability is of primary interest. The large number of criteria that have to be satisfied has led to the development of problem-suited dynamic time integration algorithms, i.e. while any algorithm may be suitable for dynamic analysis involving a specific time stepping and/or constitutive model, it may be inappropriate for dynamic analysis involving different characteristics of the three aforementioned factors. From the aforementioned points, it is obvious that there is the need for the development of a direct time integration algorithm that will be able to be universally applied to any dynamic structural analysis problem. The General Single Step Single Solve (GSSSS) algorithm family which consists of advanced time integration algorithms that allow for controlled dissipation, dispersion and overshooting properties has been developed in Chapter 2 to cover this need

7.1.2   New software for strong ground motion data processing

Most software that are used for processing of raw ground motion data nowadays employ rather elementary dynamic direct time integration algorithms. Taking advantage of the advanced GSSSS algorithm family that is presented in Chapter 1 for the evaluation of the dynamic response, a new software for ground motion data processing was developed in this thesis (OpenSeismoMatlab). A numerical investigation is made which showed that OpenSeismoMatlab provides generally more accurate results than SeismoSignal, another reliable commercial proprietary software when the same integration step size is used for both software. This is attributed to the fact that OpenSeismoMatlab uses advanced time integration algorithms of the GSSSS family. OpenSeismoMatlab is a unique software that combines innovative numerical algorithms, high quality and robustness and is provided as an open-source tool to the research and professional engineering communities for the seismic design of structures as well as the processing of strong ground motions. The new software can be used for free by students and/or programmers for the seismic design of structures as well as general processing of strong ground motions. Thanks to its open source nature, it can be of high educational value for related university courses and can be easily extended or modified in order to be incorporated in higher level software.

7.1.3   Novel gradient – based optimization concept related to dynamics of structures

The gradient-based algorithms for optimum structural design generally require considerable computational effort, especially when direct time integration for the calculation of the structural response is involved in the objective function. Most part of the computations is carried out for the calculation of the gradient of the objective function at each iteration of the optimization algorithm. This gradient determines the direction to which the algorithm will proceed for the next evaluation of the objective function. The concept of the equivalent structure for the calculation of the gradient of the objective function is one of the main contributions of the present thesis. According to this concept, while the objective function is evaluated using the model of the real structure that is normally considered, the gradient of the objective function is calculated by the consideration of an equivalent simplified version of the structural model considered, that allows for a significantly reduced computational effort. It has been proved in this thesis that this highly accelerates computations (up to 10x) saving significant amounts of computational effort. As an example to illustrate the application of this concept the problem of uniform distribution of the dissipated seismic input energy along the height of a shear planar building has been considered.

7.1.4   Formulation of a new artificial ground motion generation algorithm matching both acceleration and energy spectra

The various seismic norms worldwide require the selection of suites of representative acceleration time histories to be used for the dynamic analysis of a structural model, in order to carry out the seismic design. In cases that there is scarcity of the available acceleration time history data for a specific site, artificial ground motion data have to be generated ensuring that the last are representative of the seismic activity at the site where construction will take place. However, the current status of the various norms regarding the selection of suitable ground motion records that meet specific requirements is rather simplified, which, despite the robustness of the various finite element models available for seismic design, may account for significant source of error in structural design. On the other hand, it has been proven that the destructiveness of an earthquake is associated more to the energy absorbed by the structures, rather than the acceleration imposed on the latter. The energy-based design (EBSD) approach accounts for the effects of duration of the earthquake ground motion and conveys information about various characteristics of the seismic motion (e.g. impulse, etc.). Taking into account the aforementioned points, it is obvious that the artificial ground motions that are generated to match both target acceleration and target energy spectra will be much more realistic than those generated based merely on a target acceleration spectrum. In this thesis, a mixed real – integer genetic algorithm with appropriately customized genetic operators is developed for the generation of ground motion acceleration time histories that are compatible with both acceleration and input energy spectra, enabling in this way a more realistic seismic design of structures against earthquakes.

7.1.5   Development of a new software for linking Abaqus and Matlab

Usually there are some cases in which, although an advanced FE software and a programming language are available, they cannot perform alone a high level computational task, such as an optimization procedure. In order to achieve the computational task, they most likely need to be suitably "combined". In the case of the Abaqus FEA software and Matlab programming language, Abaqus2Matlab provides a solution for combining the two first software by transferring model data and results from one of them to the other and vice versa. Abaqus2Matlab has been used successfully in this thesis to solve an optimization problem, conduct an inverse analysis, perform a static monotonic or cyclic pushover analysis, optimize the design of girder plates for maximum critical buckling coefficient and finally train a Deep Neural Network for the estimation of the buckling coefficient for various geometric configurations of a girder plate. In all the aforementioned cases, Abaqus2Matlab has been successfully executed without flaws or bugs, while it has preserved a high level of accuracy in the

results. This proves its robustness and applicability for solving a large variety of advanced engineering problems.

7.2   Overall conclusions

Apart from the conclusions discussed in detail in the various chapters, the research work done for the thesis led to the following fundamental overall conclusions:

- The family of nonlinear generalized single step single solve (GSSSS) algorithms developed in Chapter 2 of this thesis is very efficient, accurate and stable, even with increased size of the time step, while the Continuous Acceleration methods, which include the HHT-a method as a special case, exhibit the most accurate response for most of the cases studied.

- In some cases, the quality of the results of OpenSeismoMatlab, a new software that is developed in Chapter 3 of this thesis, is superior to that of SeismoSignal, a reliable commercial proprietary software, due to the fact that the former uses advanced time integration algorithms that allow for controlled dissipation, dispersion and overshooting properties.

- It is shown that there exist unique optimum stiffness distributions, of quasi-linear shapes, which correspond to equidistributed viscous damping and hysteretic energy dissipation for linear elastic and elastoplastic planar shear building structures, respectively. Their shapes are generally independent of the earthquake excitation and offer the possibility for the development of simple methods for the calculation of the optimum stiffness distribution in shear buildings. Uniform distribution of energy along the structural height provides increased protection against global collapse and loss of life during strong earthquake events.

- The novel concept of linear directions equipped with a stabilizer for optimization of nonlinear problems, as applied for the modification of a simple full N-R method in Chapter 4 of this thesis, leads to substantial computational savings. The new modified N-R algorithm is robust and efficient.

- This consideration is leading therefore to optimized acceleration time histories, which represent actual motions in a much more realistic way. In order to produce elastic spectra that match as closely as possible to a given target spectrum, the procedure of selection and scaling of a suite of ground motion records to fit a given target spectrum is formulated as an optimization problem. Three characteristic ground motion records of different inherent nature are selected as target spectra, to verify the effectiveness of the proposed algorithm, ensuring that its performance is not ground motion record-dependent assuming different matching scenarios.

- Regarding the proposed novel spectra-matching framework developed in Chapter 5 of this thesis, it is shown that there exists a good agreement between the target and optimized spectra for the various cases examined, regardless of the nature of target spectrum, demonstrating thus the effectiveness of the algorithm.

- It is proved that the records which are generated artificially by the spectra-matching framework proposed in Chapter 5 are much more realistic and suitable for the seismic design of structures, since they reproduce better the real slightly nonlinear structural inelastic response in terms of the damping energy.

- It is proved that Abaqus2Matlab, the new software that has been developed during the preparation of the present thesis primarily for solving optimization problems, is robust and accurate, and applicable in a large variety of high level engineering problems.

The main objective of this thesis is to establish some methodologies for the design of structures based on their dynamic properties and seismic response by employing innovative computational techniques and present their advantages. Also, this thesis tries to establish the use of these energy concepts for the seismic design of structures as the state of the art in the near future and try to encourage practice towards that direction, away from the current use of safety factors and trial and error processes for the design of structural systems.

7.3   Future work

Following the research work done in this thesis, there are some natural extensions to this that would help expand and strengthen the methodologies proposed and the obtained results:

- Further research has to be made to investigate the relation between the stable time increment of the generalized single step single solve algorithms applied in nonlinear problems and various other problem-dependent input data. Apart from this, the numerous integration constants of the algorithms belonging to the GSSSS family, allows for optimization of the values of the integration constants, so that certain difficult dynamic nonlinear problems can be efficiently time-integrated

- OpenSeismoMatlab can be used for free by students and/or programmers for the seismic design of structures as well as general processing of strong ground motions. Thanks to its open source nature, it can be of high educational value for related university courses and can be easily extended or modified in order to be incorporated in higher level software.

- The new optimization concept presented in Chapter 4 of this thesis can be applied, apart from the Newton-Raphson algorithm, to other commonly used optimization algorithms, accelerating the optimization procedure.

Bibliography

ABAQUS, Analysis User's Manual Version 2016, Dassault Systèmes Simulia Corp., Providence, Rhode Island, USA.

AIJ Recommendations for Loads on Buildings, Architectural Institute of Japan, 1996.

Akiyama, H., Earthquake-resistant limit-state design for buildings, Univ of Tokyo Pr., 1985.

Alinia M.M. and Moosavi S.H., A parametric study on the longitudinal stiffeners of web panels, Thin-Walled Struct. 46 1213–1223, 2008.

Ambraseys, N., Smit, P., Douglas, J., Margaris, B., Sigbjörnsson, R., & Olafsson, S., Internet site for European strong-motion data. Bollettino di geofisica teorica ed applicata, 45(3), 113-129, 2004.

Antoniou S., Pinho R. and Bianchi F., SeismoSignal, Pavia, Italy: Seismosoft Ltd, 2012.

Akkar S., An introduction to utility software for data processing (USDP), 4th BSHAP Project Workshop, Budva, 2008.

Argyris J.H. and Scharpf D.W., Finite elements in time and space, Nucl. Eng. Des. 10, pp. 45-64, 1969.

Argyris J.H., Dunne P.C., Angelopoulos T., Dynamic response by large step integration. Earthquake Engineering & Structural Dynamics, 2, pp. 185-203, 1973.

Arias A., A Measure of Earthquake Intensity, Seismic Design for Nuclear Power Plants, R. J. Hansen, ed., Cambridge, Mass., 1970.

Armero F., Petöcz E., A new class of conserving algorithms for dynamic contact prob-lems. Désidéri J.A., LeTallec P., Oñate E., Périaux J., Stein E. eds. 2nd ECCOMAS Conf. on Numerical Methods in Engineering, Paris, France, September 9-13, 1996.

Ascher U.M., Petzold L.R., Computer methods for ordinary differential equations and differential-algebraic equations, SIAM, 1998.

Atkinson, G. M., & Goda, K., Inelastic Seismic Demand of Real versus Simulat-ed Ground-Motion Records for Cascadia Subduction EarthquakesInelastic Seismic Demand of Real versus Simulated Ground-Motion Records for Subduction Earth-quakes. Bulletin of the Seismological Society of America, 100(1), 102-115, 2010.

Bandel P., Monguzzi C., Simulation model of the dynamic behavior of a tire running over an obstacle. 2nd annual meeting of The Tire Society, The University of Akron, Akron, Ohio, March 23-24, 1983.

Bar-Yoseph P.Z., Time finite element methods for initial value problems, App. Numer. Math. 33, pp. 435-445, 2000.

Basudhar P.K., Lakshman B., Dey A., Optimal cost design of cantilever retaining walls. IGC 2006, Chennai, India, December 14-16, 2006.

Bathe K.J., Conserving energy and momentum in nonlinear dynamics: a simple im-plicit time integration scheme. Computers and Structures, 85(7-8), pp. 437–445, 2007.

Bathe K.J. and Baig M.M.I., On a composite implicit time integration procedure for nonlinear dynamics, Computers and Structures, 83(31-32), pp. 2513–2524, 2005.

Bathe K.J. and Noh G., Insight into an implicit time integration scheme for structural dynamics, Computers and Structures, vol. 98-99, pp. 1–6, 2012.

Bekakos C.A., Papazafeiropoulos G., O'Boy D.J., Prins J., Pneumatic tyres interacting with deformable terrains. 13th International Conference on Motion and Vibration Control, University of Southampton, July 3-6, 2016.

Berg, G.V. and Thomaides, S.S., Energy consumption by structures in strong-motion earthquakes, Proceedings of the second world conference on earthquake engineering, 1960.

Bolarinwa E., Olatunbosun O., On Finite Element Tyre Modal Analysis, SAE Technical Paper 2015-01-1518, 2015

Boore D.M., TSPP - A collection of FORTRAN programs for processing and manipu-lating time series, U.S. Geological Survey, 2001.

Boore, D. M., Comparing Stochastic Point--Source and Finite--Source Ground--Motion Simulations: SMSIM and EXSIM. Bulletin of the Seismological Society of America, 99(6), 3202-3216, 2009.

Bornemann P.B., Galvanetto U. and Crisfield M.A., Some remarks on the numerical time integration of nonlinear dynamical systems, J. Sound Vib. 252, pp. 935-944, 2002.

Carlton B., seismicparam, online: https://www.mathworks.com/matlabcentral/fileexchange/54188-seismicparam, 2015.

CEN, EN 1998-1: Eurocode 8, Design of structures for earthquake resistance, Part 1, Eu-ropean Committee for Standardization (CEN), 1998.

Ceranic B., Fryer C., Baines R.W., An application of simulated annealing to the optimum design of reinforced concrete retaining structures. Computers and Structures, 79, 1569–1581, 2001.

Chapman, M. C., On the use of elastic input energy for seismic hazard analysis. Earthquake Spectra, 15(4), 607-635, 1999.

Chatterjee A., Ranjan V., Free vibration analysis of radial pneumatic tire using FEM. Int. J. Emerging Technology & Advanced Engineering, 2(8), 319-324, 2012.

Chern, J.M., Dafalias, Y.F., and Martin, J.B., Structural Design for Bounds on Dynamic Response, J. Eng. Mechanics, ASCE, Vol. 99, pp. 261-270, 1973.

Chien C.C. and Wu T.Y., An improved predictor/multi-corrector algorithm for a time discontinuous Galerkin finite element method in structural dynamics, Comput. Mech., 25(5), pp. 430-437, 2000.

Chopra A.K., Dynamics of Structures: Theory and Applications to Earthquake Engi-neering, 4th ed., Englewood Cliffs, N.J.: Prentice Hall, 2012.

Chopra A.K., Dynamics of Structures, 5th Edition, University of California at Berkeley, Pearson, 2017.

Chou, C.C. and Uang, C.M., A procedure for evaluating seismic energy demand of framed structures, Earthquake Engineering & Structural Dynamics 32(2), pp. 229-244, 2003.

Chung J., Hulbert G.M., A time integration method for structural dynamics with im-proved numerical dissipation: the generalized a-method. Journal of Applied Mechan-ics, 60(2), pp. 371-375, 1993.

Cimellaro G.P. and Marasco S., A computer-based environment for processing and selection of seismic ground motion records: OPENSIGNAL, Frontiers in Built Envi-ronment, vol. 1, pp. 1-17, 2015.

Cimellaro G.P. and Marasco S., OPENSIGNAL: a complete software framework for earthquake record processing and selection, 2nd European Conference on Earthquake Engineering and Seismology, Istanbul, 2014.

Clough G.W., Duncan J.M., Earth pressures. Foundation Engineering Handbook, ed. H-Y. Fang, 224–235, Van Nostrand Reinhold, New York, 1991.

Clough R.W., Penzien J., Dynamics of structures, 3th Edition. Computers & Structures, Inc, 2003.

Connor, J.J., Wada, A., Iwata, M. and Huang, Y., Damage-controlled structures. I: Pre-liminary design methodology for seismically active regions, Journal of Structural En-gineering 123(4), pp. 423-431, 1997.

Dafalias, Y.F. and Dupuis, G., communicated by William Prager, Minimum-Weight Design of Continuous Beams Under Displacement and Stress Constraints, Journal of Optimization Theory and Applications, Vol. 9, pp. 137-154, 1972.

Dahlquist G., A special stability problem for linear multistep methods. BIT, 3(1), pp. 27-43, 1963.

Decanini, L.D. and Mollaioli, F., An energy-based methodology for the assessment of seismic demand, Soil Dynamics and Earthquake Engineering 21(2), pp. 113-137, 2001.

Degertekin S.O., Improved harmony search algorithms for sizing optimization of truss structures. Computers & Structures, 92-93: 229-241, 2012.

Degertekin S.O. and Hayalioglu M.S., Sizing truss structures using teaching-learning-based optimization. Computers & Structures, 119: 177-188, 2013.

Dindar, A. A., Yalçın, C., Yüksel, E., Özkaynak, H., & Büyüköztürk, O., Development of earthquake energy demand spectra. Earthquake Spectra, 31(3), 1667-1689, 2015

Dobry R., Idriss I.M. and Ng E., Duration Characteristics of Horizontal Components of Strong-Motion Earthquake Records, Bulletin of the Seismological Society of America, vol. 68, no. 5, pp. 1487-1520, 1978.

Donkada S., Menon D., Optimal design of reinforced concrete retaining walls. The Indian Concrete Journal, 9-18, 2012.

Dorfi H.R., Wheeler R.L., Keum B.B., Vibration Modes of Radial Tires: Application to Non-rolling and Rolling Events. SAE Noise and Vibration Conference and Exhibition, Traverse City, Michigan, May 16-19, 2005.

Dorigo, M., Ant Colony Optimization-new optimization techniques in engineering. Onwubolu, GC, and BV Babu, Springer-Verlag Berlin Heidelberg, 101-117, 1991.

Duffing G., Forced oscillations with variable natural frequency and their technical relevance, (in German), Heft 41/42, Braunschweig: Vieweg, 1918.

EC2, Eurocode 2: Design of concrete structures - Part 1-1: General rules and rules for buildings. European standard CEN-EN-1992-1, European Committee for Standardization, Brussels, 2004.

Federal Emergency Management Agency, State of the art report on systems performance of steel moment frames subject to earthquake ground shaking. FEMA 355C, 2000.

FFTW, Online: http://www.fftw.org.

Fox L., Goodwin E.T., Some new methods for the numerical integration of ordinary differential equations. Mathematical Proceedings of the Cambridge Philosophical So-ciety, 45, pp. 373-388, 1949.

Frigo, M., & Johnson, S. G., FFTW: An adaptive software architecture for the FFT. Pa-per presented at the Proceedings of the 1998 IEEE International Conference on Acous-tics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181), 1998.

Fung T.C., Complex-time-step Newmark methods with controllable numerical dissi-pation, International Journal for Numerical Methods in Engineering, vol. 41, no. 1, pp. 65–93, 1998.

Fung T.C., Fan S.C. and Sheng G., Mixed time finite elements for vibration response analysis, J. Sound Vib. 213, pp. 409-428, 1998.

Ganjavi, B. and Hao, H., Effect of structural characteristics distribution on strength demand and ductility reduction factor of MDOF systems considering soil-structure interaction, Earthquake Engineering and Engineering Vibration 11, pp. 205-220, 2012(a).

Ganjavi, B. and Hao, H., A parametric study on the evaluation of ductility demand distribution in multi-degree-of-freedom systems considering soil–structure interac-tion effects, Engineering Structures 43, pp. 88-104, 2012(b).

Ganjavi, B. and Hao, H., Optimum lateral load pattern for seismic design of elastic shear-buildings incorporating soil-structure interaction effects, Earthquake Engineer-ing & Structural Dynamics 42(6), pp. 913-933, 2013.

Gellert M., A new algorithm for integration of dynamic systems, Comp. & Struct. 9, pp. 401-408, 1978.

Ghazavi M., Bazzazian Bonab S., Optimization of Reinforced Concrete Retaining Walls Using Ant Colony Method. 3rd International Symposium on Geotechnical Safety and Risk (ISGSR), 2011.

Ghazavi M., Salavati V., Sensitivity analysis and design of reinforced concrete cantilever retaining walls using bacterial foraging optimization algorithm. 3rd International Symposium on Geotechnical Safety and Risk (ISGSR), 2011.

Ghosh, S. and Collins, K.R., Merging energy-based design criteria and reliability-based methods: exploring a new concept, Earthquake Engineering & Structural Dy-namics 35(13), pp. 1677-1698, 2006.

Glover, F., & Laguna, M., Tabu search. In Handbook of combinatorial optimi-zation (pp. 2093-2229): Springer, 1998.

Goldberg, D. E., & Holland, J. H., Genetic algorithms and machine learning. Machine learning, 3(2), 95-99, 1988.

Golley B.W., A time-stepping procedure for structural dynamics using gauss point collocation, Int. J. Numer. Meth. Eng. 39, pp. 3985-3998, 1996.

Graves, R. W., & Pitarka, A., Broadband ground-motion simulation using a hybrid approach. Bulletin of the Seismological Society of America, 100(5A), 2095-2123, 2010

Haddock, J., & Mittenthal, J., Simulation optimization using simulated annealing. Computers & industrial engineering, 22(4), 387-395, 1992.

Hajirasouliha, I., Asadi, P. and Pilakoutas, K., An efficient performance-based seismic design method for reinforced concrete frames, Earthquake Engineering & Structural Dynamics 41(4), pp. 663-679, 2012.

Hajirasouliha, I. and Moghaddam, H., New lateral force distribution for seismic de-sign of structures, Journal of Structural Engineering 135(8), pp. 906-915, 2009.

Hajirasouliha, I. and Pilakoutas, K., General seismic load distribution for optimum performance-based design of shear-buildings, Journal of Earthquake Engineering 16(4), pp. 443-462, 2012.

Hilber H.M., Hughes T.J.R., Taylor R.L., Improved numerical dissipation for time in-tegration algorithms in structural dynamics. Earthquake Engineering & Structural Dynamics, 5, pp. 283-292, 1977.

Holland, J. H., Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence: MIT press, 1992.

Housner G.W., Behaviour of structures during earthquakes, J. Engrg. Mech. Div., ASCE, vol. 85, no. 4, pp. 109-129, 1959.

Housner G.W. and Jennings P.C., The Capacity of Extreme Earthquake Motions to Damage Structures, Structural and Geotechnical Mechanics, W. J. Hall, ed., Eng-lewood Cliffs, N.J., Prentice-Hall, Inc., pp. 102-116, 1977.

Hughes T.J.R., Caughey T.K., Liu W.K., Finite-element methods for nonlinear elasto-dynamics which conserve energy. Journal of Applied Mechanics, Transactions of the ASME, 45, pp. 366-370, 1978.

IBC (International Building Code), International Code Council, Inc. (formerly BOCA, ICBO and SBCCI), 4051, pp. 60478-65795, 2006.

Iervolino, I., Maddaloni, G., & Cosenza, E., Eurocode 8 compliant real record sets for seismic analysis of structures. Journal of Earthquake Engineering, 12(1), 54-90, 2008.

Iervolino, I., Galasso, C. and Cosenza, E., REXEL: computer aided record selection for code-based seismic structural analysis. Bulletin of Earthquake Engineering, 8, 339-362, 2010.

Irikura, K., & Kamae, K., Estimation of strong ground motion in broad-frequency band based on a seismic source scaling model and an empirical Green's function technique. Annals of Geophysics, 37(6), 1721-1743, 1994.

Jamiolkowki M., Lancellotta R., Marchetti S., Nova R., Pasqualini E., Design Parameters for Soft Clays. Proc. 7th Eur. Conf. on Soil Mech. Found. Engrg., 5, 21-54, Brighton, 1979.

Jardine R.J., Brooks N.J., Smith P.R., The use of electrolevel gauges in triaxial tests on weak rock. Int. J. Rock Mech. Min. Sci. and Geomech., 22(5), 331-337, 1985.

Jeong, S.H., Lee, K.H., and Jang, W.S., PRISM for Earthquake Engineering, Computer Software. Department of Architectural Engineering, INHA University, 2016.

Jiao, Y., Yamada, S., Kishiki, S., & Shimada, Y., Evaluation of plastic energy dissipation capacity of steel beams suffering ductile fracture under various loading his-tories. Earthquake Engineering & Structural Dynamics, 40(14), 1553-1570, 2011.

Jones J., Kalkan E. and Stephens C., Processing and Review Interface for Strong Mo-tion Data (PRISM), version 1.0.0-Methodology and Automated Processing, U.S. Geo-logical Survey Open-File Report 2017–1008, 81 p., 2017.

Jones J., Kalkan E., Stephens C. and Ng P., PRISM Software: Processing and Review In-terface for Strong-Motion Data, Seismological Research Letters, vol. 88, no. 3, pp. 851-866, 2017.

Jones J., Kalkan E., Stephens C. and Ng P., PRISM Software-Processing and Review In-terface for Strong-Motion Data, U.S. Geological Survey Techniques and Methods, book 12, chap. A2, 4 p., 2017.

Jones, J., Kalkan, E., Stephens, C. and Ng, P., PRISM: Processing and Review Interface for Strong-Motion Data Software, Proc. of the 11th National Earthquake Engineering Conference, Los Angeles, CA, 2018.

Kalkan E., Nonlinear-inelastic constant yield displacement response spectra, online: https://www.mathworks.com/matlabcentral/fileexchange/58856-nonlinear-inelastic-constant-yield-displacement-response-spectra, 2016.

Kalkan E., Nonlinear multi-axial constant ductility response spectra for horizontal, vertical and tilt motions, online: https://www.mathworks.com/matlabcentral/fileexchange/62796-nonlinear-multi-axial-constant-ductility-response-spectra-for-horizontal-vertical-and-tilt-motions, 2017.

Kalkan E., Pseudo spectral acceleration, velocity and displacement spectra, online: https://www.mathworks.com/matlabcentral/fileexchange/57906-pseudo-spectral-acceleration-velocity-and-displacement-spectra, 2017.

Kalkan, E., & Kunnath, S. K., Effects of fling step and forward directivity on seismic response of buildings. Earthquake Spectra, 22(2), 367-390, 2006.

Kalkan E. and Stephens C., Systematic comparisons between PRISM version 1.0.0, BAP, and CSMIP ground-motion processing, U.S. Geological Survey Open-File Report 2017–1020, 108 p., 2017.

Kashima T., Viewwave 2.2.0 manual, International Institute of Seismology and Earth-quake Engineering (IISEE), Building Research Institute (BRI), 2016.

Katsanos, E., Sextos, A. G., & Manolis, G. D., Selection of earthquake ground motion records: A state-of-the-art review from a structural engineering perspective. Soil Dynamics and Earthquake Engineering, 30(4), 157-169, 2010.

Katsanos E.I. and Sextos A.G., ISSARS: An integrated software environment for struc-ture-specific earthquake ground motion selection, Advances in Engineering Software, vol. 58, pp. 70-85, 2013.

Kaveh A., Abadi A.S.M., Harmony Search Based Algorithm for the Optimum Cost Design of Reinforced Concrete Cantilever Retaining Walls. International Journal of Civil Engineering, 4(8), 336-357, 2010.

Kaveh A., Bakhshpoori T., Afshari E., An efficient hybrid Particle Swarm and Swallow Swarm Optimization algorithm. Computers & Structures, 143: 40-59, 2014.

KBC (Korean Building Code), Ministry of Infrastructure and Transport, 2009.

Khajehzadeh M., Raihan Taha M., El-Shafie A., Eslami M., Economic Design of Retaining Wall Using Particle Swarm Optimization with Passive Congregation. Australian Journal of Basic and Applied Sciences, 4(11), 5500-5507, 2010.

Khajehzadeh M., Raihan Taha M., El-Shafie A., Eslami M., Modified particle swarm optimization for optimum design of spread footing and retaining wall. Journal of Zhejiang University-SCIENCE A (Applied Physics & Engineering), 12(6), 415-427, 2011.

Kramer S., Geotechnical Earthquake Engineering, Prentice Hall, 1996.

Kuhl D., Crisfield M.A., Energy-conserving and decaying algorithms in non-linear structural dynamics. Int. J. Numer. Meth. Engng, 45, pp. 569-599, 1999.

Kuhl D., Ramm E., Constraint energy momentum algorithm and its application to nonlinear dynamics of shells. Computer Methods in Applied Mechanics and Engi-neering, 136, pp. 293–315, 1996.

Kuwamura, H., & Galambos, T. V., Earthquake load for structural reliability. Journal of Structural Engineering, 115(6), 1446-1462, 1989.

Krawinkler H. and Nassar A.A., Seismic design based on ductility and cumulative damage demand and capacities, Nonlinear Seismic Analysis and Design of Reinforced Concrete Buildings, New York, NY, USA, Elsevier Applied Science, 1992.

Lee J.H., Finite element modelling of interfacial forces and contact stresses of pneumatic tire on fresh snow for combined longitudinal and lateral slips, Journal of Terramechanics, 48, 171-197, 2011.

Leelataviwat, S., Saewon, W., & Goel, S. C., Application of energy balance concept in seismic evaluation of structures. Journal of Structural Engineering, 135(2), 113-121, 2009.

Léger, P. and Dussault, S., Seismic-energy dissipation in MDOF structures, Journal of Structural Engineering 118(5), pp. 1251-1269, 1992.

Li L.J., Huang Z.B. , Liu F., Wu Q.H., A heuristic particle swarm optimizer for optimization of pin connected structures. Computers & Structures, 85: 340-349, 2007.

Li X.D. and Wiberg N.E., Structural dynamic analysis by a time-discontinuous Ga-lerkin finite element method, Int. J. Numer. Meth. Eng. 39, pp. 2131-2152, 1996.

Liu T. Y., Zhao C. B., Li Q. B., and Zhang L. H., An efficient backward Euler time-integration method for nonlinear dynamic analysis of structures, Computers and Structures, vol. 106- 107, pp. 20–28, 2012.

López Almansa, F., Yazgan, A. U., & Benavent Climent, A., Design energy input spectra for high seismicity regions based on Turkish registers. Bulletin of earthquake engineering, 11(4), 885-912, 2013.

Macedo L. and Castro J.M., SelEQ: An advanced ground motion record selection and scaling framework, Advances in Engineering Software, vol. 114, p. 32-47, 2017.

Matias Silva W. T. and Mendes Bezerra L., Performance of composite implicit time integration scheme for nonlinear dynamic analysis, Mathematical Problems in Engi-neering, Article ID 815029, 16 pages, 2008.

MATLAB, Global Optimization Toolbox Version 9 (R2017b), The MathWorks Inc., Natick, MA, USA.

Matsuoka H., Okuma M., A New Experimental Modal Parameter Estimation Method for Cylindrical Structures by Standing Wave Decomposition and Application to Tire Modelling. Proceedings of ISMA at The International Conference on Noise and Vibration Engineering, V1913 – 1919, 2002.

Mezgebo, M. G., Estimation of Earthquake Input Energy, Hysteretic Energy and its Distribution in MDOF Structures, PhD dissertation, Dissertations - ALL. 228, Syra-cuse University, NY, 2015.

Miranda E. and Bertero V.V., Evaluation of strength reduction factor for earthquake-resistance design, Earthquake Spectra, vol. 10, no. 2, pp. 357-379, 1994.

Moghaddam, H. and Hajirasouliha, I., A new approach for optimum design of struc-tures under dynamic excitation, Asian Journal of Civil Engineering 5(1-2), pp. 69-84, 2004.

Moghaddam, H. and Hajirasouliha, I., Toward more rational criteria for determina-tion of design earthquake forces, International journal of solids and structures 43(9), pp. 2631-2645, 2006.

Molisani L., A coupled tire structure-acoustic cavity model, PhD Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, US, 2004.

Mononobe N., Matsuo H., On the determination of earth pressures during earthquakes. Proceedings of the World Engineering Congress, Tokyo, Japan, Vol. 9, Paper 388, 1929.

Murakami, Y., Noshi, K., Fujita, K., Tsuji, M. and Takewaki, I., Simultaneous optimal damper placement using oil, hysteretic and inertial mass dampers, Earthquakes and Structures 5(3), pp. 261-276, 2013.

Mustafa Sonmez. Artificial Bee Colony algorithm for optimization of truss structures. Applied Soft Computing, 11(2): 2406-2418, 2011.

Nakamura, T. and Yamane, T., Optimum design and earthquake-response con-strained design of elastic shear buildings, Earthquake Engineering & Structural Dy-namics 14(5), pp. 797-815, 1986.

Nakashima, M., Saburi, K. and Tsuji, B., Energy input and dissipation behaviour of structures with hysteretic dampers, Earthquake Engineering & Structural Dynamics 25(5), pp. 483-496, 1996.

Newmark N.M., A method of computation for structural dynamics. Journal of Engi-neering Mechanics, 85(EM3), 67–94, 1959.

Newmark N.M., Blume J.A. and Kapur K.K., Seismic design spectra for nuclear power plants, J. Power Div., ASCE, vol. 99, no. 2, pp. 287-303, 1973.

Newmark N.M. and Hall W.J., Seismic design criteria for nuclear reactor facilities, Proc. of the Fourth World Conf. on Earthquake Engrg., Santiago, Chile, 1969.

Newmark N.M. and Hall W.J., Earthquake Spectra and Design, EERI Monograph Se-ries, 1982.

Nocedal J., Wright S.J., Numerical Optimization, 2nd Edition. Springer Series in Operations Research, Springer Verlag, 2006.

NZS1170 (New Zealand Standard 1170), Structural Design Actions, Part 5: Earthquake Actions, 2004.

Okabe S., General theory of earth pressures. Journal of the Japan Society of Civil Engineering, 12(1), 1926.

Papazafeiropoulos G., Plevris V. and Papadrakakis M., A generalized algorithm framework for non-linear structural dynamics, Bulletin of Earthquake Engineering, vol. 15, no. 1, pp. 411-441, 2017.

Papazafeiropoulos G., Plevris V. and Papadrakakis M., A New Energy-Based Structur-al Design Optimization Concept under Seismic Actions, Front. Built Environ, vol. 3, no. 44, 2017.

Papazafeiropoulos G., OpenSeismoMatlab, online: https://www.mathworks.com/matlabcentral/fileexchange/67069-openseismomatlab, 2018.

Papazafeiropoulos G., V. Plevris, OpenSeismoMatlab, online: https://www.researchgate.net/publication/325529014_OpenSeismoMatlab, 2018.

Park, K. and Medina, R. A., Conceptual seismic design of regular frames based on the concept of uniform damage, Journal of Structural Engineering 133(7), pp. 945-955, 2007.

Pei Y., Xia Y., Design of Reinforced Cantilever Retaining Walls using Heuristic Optimization Algorithms. 2012 International Conference on Structural Computation and Geotechnical Mechanics, Procedia Earth and Planetary Science 5, 32 – 36, 2012.

Penzien, J., Dynamic response of elasto-plastic frames, Transactions of the American Society of Civil Engineers 127(2), pp. 1-13, 1960.

Prasanth, T., Ghosh, S. and Collins, K.R., Estimation of hysteretic energy demand us-ing concepts of modal pushover analysis, Earthquake Engineering & Structural Dy-namics 37(6), pp. 975-990, 2008.

Rathje E.M., Faraj F., Russell S. and Bray J.D., Empirical Relationships for Frequency Content Parameters of Earthquake Ground Motions, Earthquake Spectra, vol. 20, no. 1, pp. 119-144, 2004.

Rezaiee-Pajand M., Sarafrazi S. R., and Hashemian M., Improving stability domains of the implicit higher order accuracy method, International Journal for Numerical Methods in Engineering, vol. 88, no. 9, pp. 880–896, 2011.

Rhomberg E.J., Street W.M., Optimal design of retaining walls. Journal of Structural Division, ASCE, 107, 992-1002, 1981.

Riff R. and Baruch M., Stability of time finite elements, AIAA J. 22, pp. 1171-1173, 1984.

Riff R. and Baruch M., Time finite element discretization of Hamilton's law of varying action, AIAA J. 22, pp. 1310-1318, 1984.

Rockey K.C. and Cook I.T., Optimum reinforcement by two longitudinal stiffeners of a plate subjected to pure bending, Int. J. Solids Struct. 1 79–92, 1965.

Rodriguez, M., A measure of the capacity of earthquake ground motions to damage structures, Earthquake Engineering & Structural Dynamics 23(6), pp. 627-643, 1994.

Rosales M.B., Filipich C.P., Time integration of non-linear dynamic equations by means of a direct variational method. Journal of Sound and Vibration, 254(4), pp. 763–775, 2002.

Saribas A., Erbatur F., Optimization and sensitivity of retaining structures. ASCE Journal of Geotechnical Engineering, 122(8), 649–656, 1996.

Scavuzzo R.W., Richards T.R., Charek L.T., Tire Vibration Modes and Effects on Vehicle Ride Quality. Tire Science and Technology, TSTCA, 21(1), pp.23-39, 1993.

Scott R.F., Earthquake induced pressures on retaining walls. Proceedings of the 5th World Conference on Earthquake Engineering, 2, 1611–1620, 1973.

Seed H.B., Whitman R.V., Design of earth retaining structures for dynamic loads. Proceedings of the ASCE Special Conference on Lateral Stresses in the Ground and Design of Earth Retaining Structures, 103–147, 1970.

Shargh, F. H. and Hosseini, M., A Study on the Existence of an Optimal Distribution of Stiffness over the Height of Mid-to High-Rise Buildings to Minimize the Seismic Input Energy, Journal of Applied Sciences 10(1), pp. 45-51, 2010.

Shargh, F. H. and Hosseini, M., An optimal distribution of stiffness over the height of shear buildings to minimize the seismic input energy, Journal of Seismology and Earthquake Engineering 13(1), pp. 25-32, 2011.

Shargh, F. H., Hosseini, M. and Daneshvar, H., An Optimal Distribution of Stiffness over the Height of Buildings and its Buildings and its Influence on the Degree and Distribution of Earthquake induced Damages and Distribution of Earthquake induced Damages, 15th World Conference on Earthquake Engineering, Lisbon, 24-28 Sept., 2012.

Shi, Y., Particle swarm optimization: developments, applications and resources. Paper presented at the Proceedings of the 2001 Congress on Evolutionary Computa-tion (IEEE Cat. No. 01TH8546), 2001.

Shoop S.A., Finite Element Modelling of Tire-Terrain Interaction, Report number: ERDC/CRREL TR-01-16, University of Michigan, 2001.

Simo J.C., Tarnow N., The discrete energy-momentum method. Conserving algo-rithms for nonlinear elastodynamics. Journal of Applied Mathematics and Physics, 43(5), pp. 757–792, 1992.

Sivakumar Babu G.L., Munwar Basha B., Optimum design of cantilever retaining walls using target reliability approach. International Journal of Geomechanics, ASCE, 8(4), 240-252, 2008.

Soroushian A., A technique for time integration analysis with steps larger than the excitation steps, Commun. Numer. Meth. Engng, vol. 24, p. 2087–2111, 2008.

Soroushian A., Integration Step Size and Its Adequate Selection in Analysis of Struc-tural Systems Against Earthquakes, Computational Methods in Earthquake Engineer-ing, Springer, Cham, pp. 285-328, 2017.

Surahman, A., Earthquake-resistant structural design through energy demand and capacity. Earthquake Engineering & Structural Dynamics, 36(14), 2099-2117, 2007.

Takewaki, I., Building control with passive dampers: optimal performance-based de-sign for earthquakes, John Wiley & Sons, 2011.

Talatahari S., Sheikholeslami R., Shadfaran M., Pourbaba M., Optimum Design of Gravity Retaining Walls Using Charged System Search Algorithm. Mathematical Problems in Engineering, Vol. 2012, Article ID 301628.

Tazarv M., Elastic Response Spectrum, online: https://www.mathworks.com/matlabcentral/fileexchange/31254-elastic-response-spectrum, 2011.

Trifunac M.D. and Brady A.G., A Study on the Duration of Strong Earthquake Ground Motion, Bulletin of the Seismological Society of America, vol. 65, no. 3, pp. 581-626, 1975.

Tselentis, G., Danciu, L., & Sokos, E., Probabilistic seismic hazard assessment in Greece–Part 2: Acceleration response spectra and elastic input energy spectra. Nat-ural Hazards and Earth System Sciences, 10(1), 41-49, 2010.

UBC (Uniform Building Code), International conference of building officials, Whittier, CA, 1997.

Uang, C.-M., & Bertero, V. V., Use of energy as a design criterion in earth-quake-resistant design (Vol. 88): Earthquake Engineering Research Center, University of California Berkeley, 1988.

Uang, C.-M., & Bertero, V. V., Evaluation of seismic energy in structures. Earthquake Engineering & Structural Dynamics, 19(1), 77-90, 1990.

Uetani, K., Tsuji, M. and Takewaki, I., Application of an optimum design method to practical building frames with viscous dampers and hysteretic dampers, Engineering Structures 25(5), pp. 579-592, 2003.

Veletsos A.S. and Newmark N.M., Effect of inelastic behavior on the response of sim-ple systems to earthquake motions, Proc., II World Conf. on Earthquake Engineering, Tokyo, Japan, 1960.

Veletsos A.S., Newmark N.M. and Chelapati C.V., Deformation spectra for elastic and elastoplastic systems subjected to ground shock and earthquake motions, Proc., III World Conf. on Earthquake Engineering, Auckland, New Zealand, 1965.

Veletsos A.S. and Vann, W.P., Response of ground-excited elastoplastic systems, Journal of the Structural Division, vol. 97, no. 4, pp. 1257-1281, 1971.

Veletsos A.S., Younan A.H., Dynamic Response of Cantilever Retaining Walls. ASCE Journal of Geotechnical and Geoenvironmental Engineering, 123(2), 161–172, 1997.

Vinesse E.P., Substructuring method and eigenfunction representation for a rolling tyre coupled to a secondary mechanical system. Proc Instn Mach Engrs, 210, 313-325, 1996.

Wang, F. and Yi, T., A Methodology for Estimating Seismic Hysteretic Energy of Buildings, Civil Engineering and Urban Planning, pp. 17-21, 2012.

Wheeler R.L., Dorfi H.R., Keum B.B., Vibration modes of radial tires: measurement, prediction, and categorization under different boundary and operating conditions. SAE Noise and Vibration Conference and Exhibition, Traverse City, Michigan, May 16-19, 2005.

Wood J.H., Earthquake – Induced Pressures on a Rigid Wall Structure. Bulletin of New Zealand National Earthquake Engineering, 8, 175–186, 1975.

Wood W.L., Practical time stepping schemes. Oxford, USA, 1990

Wood W.L., Bossak M., Zienkiewicz O.C., An alpha modification of Newmark's method. International Journal for Numerical Methods in Engineering, 15(10), pp. 1562-1566, 1980.

Wu G., Finn W.D.L., Seismic lateral pressures for design of rigid walls. Canadian Geotechnical Journal, 36(3), 509-522, 1999.

Wu C.Y. and Tseng K.Y.,. Truss structure optimization using adaptive multi-population differential evolution. Structural and Multidisciplinary Optimization, 42: 575-590, 2010.

Yepes V., Alcala J., Perea C. and González-Vidosa F., A parametric study of optimum earth retaining walls by simulated annealing. Engineering Structures, 30(3), 821-830, 2008.

Zahrah, T.F. and Hall, W.J., Seismic energy absorption in simple structures, Universi-ty of Illinois Engineering Experiment Station, College of Engineering, University of Illinois at Urbana-Champaign, 1982.

Zahrah, T. F., & Hall, W. J., Earthquake energy absorption in SDOF structures. Journal of Structural Engineering, 110(8), 1757-1772, 1984.

Zhou X., Tamma K.K., Design, analysis, and synthesis of generalized single step sin-gle solve and optimal algorithms for structural dynamics. Int. J. Numer. Meth. Engng, 59, pp. 597–668, 2004.

Ziang H., Fu B. and Chen L., Constant-Damage Yield Strength Spectra for RC Struc-tures Using Modified Park-Ang Damage Model, Advances in Structural Engineering, vol. 18, no. 6, pp. 837 - 851, 2016.

Appendix A.
Listing of Publications

This appendix contains the publications that are related to the present thesis and were published during its preparation.

Publications in peer-reviewed journals

[1] G. Papazafeiropoulos, M.S. Georgioudakis, M.Papadrakakis. Selecting and Scaling of Energy-Compatible Ground Motion Records. Frontiers in Built Environment 5 (2019): 140, DOI: 10.3389/fbuil.2019.00140.

[2] Q.-V. Vu, V.-H. Truong, G. Papazafeiropoulos: Effect of multiple longitudinal stiffeners on ultimate strength of steel plate girders. DOI:10.1016/j.istruc.2019.09.002

[3] Q.-V. Vu, V.-H. Truong, G. Papazafeiropoulos, C. Graciano, S.-E. Kim: Bend-buckling strength of steel plates with multiple longitudinal stiffeners. Journal of Constructional Steel Research 03/2019; 158(1):41-52., DOI:10.1016/j.jcsr.2019.03.006

[4] Q.-V. Vu, G. Papazafeiropoulos, C. Graciano, S.-E. Kim: Optimum linear buckling analysis of longitudinally multi-stiffened steel plates subjected to combined bending and shear. Thin-Walled Structures 12/2018; 136:235-245., DOI:10.1016/j.tws.2018.12.008

[5] G. Papazafeiropoulos, V. Plevris: OpenSeismoMatlab: A new open-source software for strong ground motion data processing. Heliyon 09/2018; 4(9):e00784., DOI:10.1016/j.heliyon.2018.e00784

[6] G. Papazafeiropoulos, V. Plevris, M. Papadrakakis: A New Energy-Based Structural Design Optimization Concept under Seismic Actions. Frontiers in Built Environment 07/2017; 3., DOI:10.3389/fbuil.2017.00044

[7] G. Papazafeiropoulos, M. Muñiz-Calvente, E. Martínez-Pañeda: Abaqus2Matlab: A suitable tool for finite element post-processing. Advances in Engineering Software 03/2017; 105:9-16., DOI:10.1016/j.advengsoft.2017.01.006

[8] C. A. Bekakos, G. Papazafeiropoulos, D. J. O'Boy, J. Prins: Finite element modelling of a pneumatic tyre interacting with rigid road and deformable terrain. 01/2017; 3(2):142., DOI:10.1504/IJVP.2017.083359

[9] G. Papazafeiropoulos, V. Plevris, M. Papadrakakis: A generalized algorithm framework for non-linear structural dynamics. Bulletin of Earthquake Engineering 01/2017; 15(1):411–441., DOI:10.1007/s10518-016-9974-8

[10] C. A. Bekakos, G. Papazafeiropoulos, D. J. O'Boy: Off-Road Tire-Terrain Interaction: An Analytical Solution. SAE International Journal of Commercial Vehicles 09/2016; 9(2):244-251., DOI:10.4271/2016-01-8029

[11] C. A. Bekakos, G. Papazafeiropoulos, D. J. O'Boy, J. Prins: Pneumatic tyres interacting with deformable terrains. Journal of Physics Conference Series 07/2016; 744(1)., DOI:10.1088/1742-6596/744/1/012213

Publications in conferences after review

[12] G. Papazafeiropoulos, Q.-V. Vu, V.-H. Truong, Minh-Chinh Luong, Van-Trung Pham: Prediction of buckling coefficient of stiffened plate girders using deep learning algorithm. CIGOS 2019, Innovation for Sustainable Infrastructure, Proceedings of the 5th International Conference on Geotechnics, Civil Engineering Works and Structures, 10/2019, pages 1143-1148, ISBN: 978-981-15-0801-1, DOI:10.1007/978-981-15-0802-8_183

[13] K. A. Skalomenos, G. Papazafeiropoulos. A Computational Method for Performing Nonlinear Adaptive Pushover Analysis of Structures Through Abaqus Simulation, 7th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering Methods in Structural Dynamics and Earthquake Engineering, 24-26 June 2019, Crete, Greece, DOI: 10.7712/120119.7199.19194

[14] E. Martínez Pañeda, G. Papazafeiropoulos, M. Muñíz-Calvente. Abaqus2MatLab: A Novel Tool for Finite Element Post-Processing. 7th International Conference on Mechanics and Materials in Design. INEGI-Instituto De Ciência E Inovação Em Engenharia Mecânica E Engenharia Industrial, 2017.

[15] C.-A. Bekakos, G. Papazafeiropoulos, D. J. O' Boy, Jan Prins. Pneumatic tyres interacting with deformable terrains. In Journal of Physics: Conference Series, vol. 744, no. 1, p. 012213. IOP Publishing, 13th International Conference on Motion and Vibration Control, Southampton, UK; 07/2016, DOI: 10.1088/1742-6596/744/1/012213.

[16] C. A. Bekakos, G. Papazafeiropoulos, D. J. O'Boy, J. Prins: Development of accurate pneumatic tyre finite element models based on an optimization procedure. VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete Island, Greece; 06/2016, DOI:10.7712/100016.2411.5279

[17] C. A. Bekakos, G. Papazafeiropoulos, D. J. O'Boy, J. Prins: Dynamic response of rigid wheels on deformable terrains. 13th European Conference of the

International Society for Terrain and Vehicle Systems (ISTVS), Rome, Italy, 21–23/10/2015; 10/2015

[18] G. Papazafeiropoulos, V. Plevris, M. Papadrakakis: Nonlinear Dynamic Response of Hardening, Softening and Elastoplastic SDOF Systems Using Generalized Single Step Algorithms with Newton-Raphson Iterations. COMPDYN 2015 5th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering M. Papadrakakis, V. Papadopoulos, V. Plevris (eds.) Crete Island, Greece, 25–27 May 2015; 05/2015, DOI:10.7712/120115.3582.3606

[19] G. Papazafeiropoulos, V. Plevris, M. Papadrakakis: Optimum design of cantilever walls retaining linear elastic backfill by use of genetic algorithm. COMPDYN 2013 4th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering M. Papadrakakis, V. Papadopoulos, V. Plevris (eds.) Kos Island, Greece, 12–14 June 2013; 06/2013, DOI:10.7712/120113.4700.C1746