



HAL
open science

OPTIMAL SIZE, SHAPE AND CONTROL DESIGN IN DYNAMICS OF PLANAR FRAME STRUCTURES UNDER LARGE DISPLACEMENTS AND ROTATIONS

Matija Gams, Miran Saje, Igor Planinc, Marko Kegl

► **To cite this version:**

Matija Gams, Miran Saje, Igor Planinc, Marko Kegl. OPTIMAL SIZE, SHAPE AND CONTROL DESIGN IN DYNAMICS OF PLANAR FRAME STRUCTURES UNDER LARGE DISPLACEMENTS AND ROTATIONS. *Engineering Optimization*, 2009, 42 (01), pp.69-86. 10.1080/03052150902998552 . hal-00546868

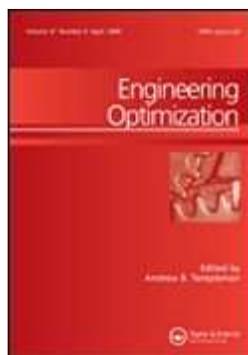
HAL Id: hal-00546868

<https://hal.science/hal-00546868>

Submitted on 15 Dec 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**OPTIMAL SIZE, SHAPE AND CONTROL DESIGN IN DYNAMICS
OF PLANAR FRAME STRUCTURES UNDER LARGE
DISPLACEMENTS AND ROTATIONS**

Journal:	<i>Engineering Optimization</i>
Manuscript ID:	GENO-2008-0170.R5
Manuscript Type:	Original Article
Date Submitted by the Author:	10-Apr-2009
Complete List of Authors:	Gams, Matija; Slovenian National Building and Civil Engineering Institute Saje, Miran; Faculty of Civil and Geodetic Engineering Planinc, Igor; Faculty of Civil and Geodetic Engineering Kegl, Marko; University of Maribor, Faculty of Mechanical Engineering
Keywords:	Structures, Dynamics, Nonlinear response, Optimization



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

OPTIMAL SIZE, SHAPE AND CONTROL DESIGN IN DYNAMICS OF PLANAR FRAME STRUCTURES UNDER LARGE DISPLACEMENTS AND ROTATIONS

M. GAMS, M. SAJE, I. PLANINC

Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, 1000 Ljubljana, Slovenia

M. KEGL

Faculty of Mechanical Engineering, University of Maribor, Smetanova 17, 2000 Maribor, Slovenia

ADDRESS:

M. Kegl
FACULTY OF MECHANICAL ENGINEERING
Smetanova 17
SI-2000 Maribor
SLOVENIA
Telephone: (+386 2) 220-7802
Telefax: (+386 2) 220-7990
E-mail: marko.kegl@uni-mb.si

SPONSOR:

Agency of research of the Republic of Slovenia, grant number 8311-03-831622

ABSTRACT

Size, shape, and drive optimization procedures are combined with an energy-conserving time integration scheme for the dynamic analysis of planar geometrically non-linear frame structures undergoing large overall motions. The solution method is based on the finite element formulation employing the classical displacement-based planar beam finite elements described in an inertial frame. Finite axial, bending and shear strains are taken into account. If the system is conservative, the energy and momenta conservation in the discrete system during motion is guaranteed. Size, shape, and drive design variables are introduced into the model. Shape parameterization is achieved by the design element technique, utilizing Bezier patches. The sensitivity analysis is performed by the discrete approach and the analytical

1
2
3
4 direct differentiation method. A gradient-based optimization method, utilizing an
5
6 automatically adjustable convex approximation technique, is employed. The efficiency and
7
8 the applicability of the approach are demonstrated via numerical examples. The shape and the
9
10 driving function of a load moving robot arm are optimized to reduce oscillations in its final
11
12 position. The shape of a steel frame is optimized to reduce oscillations after an idealized
13
14 ground motion jerk.
15
16
17
18
19
20

21 KEYWORDS: Structures; Manipulators; Non-linear dynamics; Structural optimization;
22
23 Bezier patch
24
25
26
27

28 1. INTRODUCTION

29
30 Shape or control optimization of flexible dynamic systems is a developing field of
31
32 science. Lighter, faster and more flexible robot manipulators, which use less power to operate,
33
34 and tall buildings and towers which are more resistant to dynamic loads such as wind or
35
36 earthquake, are just a few among many areas of application.
37
38

39
40 Comprehensive studies on optimization of structural systems with transient dynamic
41
42 response began in the 1970s. A thorough review of the subject, focusing on the optimization
43
44 of structures modeled by linearized equations, combined with gradient-based optimization
45
46 algorithms, is given in a recent report by Kang *et al.* (2006). In contrast, non- linear models
47
48 are rather rarely addressed in optimization. This is due to a complicated structure of
49
50 non-linear governing equations which require reliable numerical algorithms and a rather large
51
52 computational effort for both the response analysis and the sensitivity analysis. This gave the
53
54 push to improve gradient-based optimization methods and sensitivity analyses, see, *e.g.*
55
56 (Cardoso and Arora 1992, Arora and Dutta 1997). Kulkarni and Noor (1995) describe the
57
58
59
60

1
2
3
4 sensitivity calculation of 2D viscoplastic structures with respect to material parameters. The
5
6 calculation of approximate sensitivities in explicit time-integration scheme of dynamics is
7
8 addressed by Stupkiewicz (2001). The design sensitivity analysis and the gradient-based
9
10 optimization of transient dynamics of elastoplastic structures is addressed by Cho and Choi
11
12 (2000a, 2000b) for the case of size and configuration design variables. These works employ a
13
14 continuum-based design sensitivity, which simplifies the implementation of the sensitivity in
15
16 existing FEA software packages. Utilization of a mesh-free analysis method was proposed by
17
18 Kim and Choi (2001) in order to perform the response and shape sensitivity analysis for the
19
20 optimization of elastoplastic structures under impact with a rigid surface. A further insight
21
22 into optimization of elastoplastic structures is given in (Sousa *et al.* 1997) and (Pedersen
23
24 2004). The latter addresses topology optimization of elastoplastic 2D frames. Non-linear
25
26 elastic models are frequently employed in the optimization of flexible manipulators. The
27
28 article by Okubo and Tortorelli (2004) describes a general methodology to design open loop
29
30 controllers for non-linear dynamic systems. This methodology employs the displacement-
31
32 based finite elements, and the Newmark time integration scheme.
33
34
35
36
37
38
39

40 The reduction of material generally leads to more flexible systems possibly deforming
41
42 far out of the linear regime; consequently, geometrically non-linear models are required. The
43
44 present work employs the geometrically exact theory of Reissner (1972). Optimization of the
45
46 shape and/or drive of a mechanism/structure may be done quite efficiently, if adequate shape
47
48 parameterization is employed; in this work Bezier patches are used. In contrast to Cho and
49
50 Choi (2000a) and Kim and Choi (2001), the design sensitivity is here performed on the
51
52 discrete equations. This enables the use of software packages for the automatic analytical
53
54 differentiation and code generation, which substantially speeds up the determination of both
55
56 the sensitivity coefficients and the tangent stiffness matrices. A similar approach to
57
58
59
60

1
2
3
4 optimization of elastic manipulators was also adopted by Vohar *et al.* (2008). However, in
5
6 contrast to this work, in (Vohar *et al.* 2008) the focus was on the implementation of an ANCF
7
8 beam finite element in a gradient-based optimization process. ANCF finite elements are quite
9
10 different from standard finite element types and require special implementation procedures.
11
12

13
14 In non-linear dynamics, stability of the numerical time integration is still a key issue
15
16 (Crisfield *et al.* 1997, Gams *et al.* 2007, Ibrahimbegovic and Mamouri 2002, Kuhl and Ramm
17
18 1996, Kuhl and Crisfield 1999, Sansour *et al.* 2004, Simo *et al.* 1995). Stability is particularly
19
20 problematic, when the governing system of differential equations is stiff, a situation which is
21
22 very often met in structural and mechanism analyses, where axial and bending stiffnesses are
23
24 very different or if there are sudden changes in loads or drives. Due to the unpredictable
25
26 solution path during the iteration process, very different stiffnesses might be encountered in
27
28 every iteration of the optimization; that is why the time integration scheme should be
29
30 unconditionally stable.
31
32
33

34
35 The stable implicit algorithms for the non-linear dynamics must satisfy the stability
36
37 criterion of energy conservation/decay. Roughly three main-stream groups of algorithms exist
38
39 aiming to do so (Kuhl and Crisfield 1999). The first group deals with the numerical
40
41 dissipation algorithms. These algorithms do not always guarantee dissipation of energy in the
42
43 non-linear regime, and energy in a time step can sometimes be created instead of dissipated
44
45 (Crisfield *et al.* 1997), hence they are not well suited for the stiff problems. The second group
46
47 of algorithms uses enforced constraint methodology, where energy and momenta conservation
48
49 requirements are introduced via the Lagrange multiplier method. The approach exactly
50
51 satisfies the requirements and hence guarantees perfect conservation, but problems with
52
53 stability may still emerge (Kuhl and Ramm 1996). The third group of algorithms, termed the
54
55 Energy Momentum Methods (EMM), is based on algorithmic conservation, *i.e.* the solution
56
57
58
59
60

1
2
3
4 algorithm inherently conserves energy and momenta. The earliest EMM schemes were
5
6 proposed by Simo *et al.* (1995). Stability of the EMM algorithms is generally better than the
7
8 one of the previously mentioned algorithms (Ibrahimbegovic and Mamouri 1999).
9

10
11 In the present article, a newly developed variant of EMM method (Gams *et al.* 2007) is
12
13 used in optimization problems. This method, described in detail in (Gams *et al.* 2007) and
14
15 briefly in Section 2, assumes the incremental form of the kinematic equations in calculating
16
17 the update of strains.
18
19

20
21 The outline of the article is as follows. Section 2 shortly describes the finite-element
22
23 formulation and the energy-conserving time-integration scheme. Section 3 sets the
24
25 optimization problem and the concepts of an efficient shape design. In Section 4 the solution
26
27 process is described; special attention is paid to the implementation of the time integration
28
29 scheme (Gams *et al.* 2007) into the optimization algorithm. Four numerical examples,
30
31 demonstrating the applicability and the efficiency of the presented approach, are analyzed in
32
33 Section 5.
34
35
36
37
38

39 2. GEOMETRICALLY EXACT BEAM FORMULATION AND ENERGY CONSERVING 40 41 TIME-INTEGRATION SCHEME 42

43
44 *The planar Reissner beam* (Reissner 1972). An initially straight planar elastic beam is
45
46 considered. The hypothesis of planar cross-sections holds, with the axial, shear and bending
47
48 strains being taken into account. The plane sections remain plane but not necessarily
49
50 perpendicular to the axis of the deformed beam.
51
52

53 *Kinematics.* The kinematic equations of the beam are given by
54

$$55 \quad \boldsymbol{\varepsilon} = \boldsymbol{\Lambda}(\mathbf{u} + \mathbf{x})' + \mathbf{c}_1, \quad (1)$$

56
57 where $\boldsymbol{\varepsilon}$ is the strain vector, \mathbf{u} the displacement vector and \mathbf{x} the initial position vector of the
58
59
60

centroid axis. Λ is the rotation matrix and \mathbf{c}_1 is a given vector constant:

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon \\ \gamma \\ \kappa \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u \\ v \\ \varphi \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_0 + s \cos \varphi_0 \\ y_0 + s \sin \varphi_0 \\ 0 \end{bmatrix},$$

$$\Lambda = \begin{bmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{c}_1 = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}.$$

ε , γ and κ are axial, shear and bending strain along the beam axis, respectively; u and v are the x and y components of the displacement vector of the axis, while φ is the rotation of the cross-section. The prime (') denotes the differentiation with respect to s .

Time integration and energy conservation. Energy conservation of the time-integration scheme is possible only in conservative systems and is here achieved by employing midpoint approximation rules (Ibrahimbegovic and Mamouri 1999, 2002, Simo *et al.* 1995) in conjunction with the kinematic relations in the rate form. For details, please refer to Gams *et al.* (2007) or for general continuum formulations to Sansour *et al.* (2004). The rate form implies that strains at t_{n+1} are calculated in additive manner:

$$\boldsymbol{\varepsilon}_{n+1} = \boldsymbol{\varepsilon}_n + \Delta \boldsymbol{\varepsilon} \quad (2)$$

$$\Delta \boldsymbol{\varepsilon} = \mathbf{W} \Lambda_m (\mathbf{u} + \mathbf{x})'_m \Delta \varphi + \Lambda_m \Delta \mathbf{u}' \quad (3)$$

Here the subscript m denotes the midpoint time configuration at $t_m = t_n + \Delta t / 2$. \mathbf{W} is a constant matrix, described in the sequel.

The algorithmic midpoint values of displacements and strains are calculated by the trapezoidal rule:

$$\boldsymbol{\varepsilon}_m = (\boldsymbol{\varepsilon}_n + \boldsymbol{\varepsilon}_{n+1}) / 2, \quad \mathbf{u}_m = (\mathbf{u}_n + \mathbf{u}_{n+1}) / 2. \quad (4)$$

The governing equations. The principle of virtual work proposed at midpoint time reads

$$\int_0^L \left[\mathbf{N}_m^T \mathbf{\Lambda}_m \delta \mathbf{u}'_m + \mathbf{N}_m^T \mathbf{W} \mathbf{\Lambda}_m (\mathbf{u} + \mathbf{x})'_m \delta \varphi_m - (\mathbf{p}_m^T - \ddot{\mathbf{u}}_m^T \mathbf{c}_\rho) \delta \mathbf{u}_m \right] ds - \mathbf{P}_m^T \delta \mathbf{\Psi}_m = 0. \quad (5)$$

Here \mathbf{N} is the vector of the cross-sectional stress resultants in the local basis (\mathbf{e}_n , \mathbf{e}_t , \mathbf{e}_z) (Figure 1), \mathbf{p} is the vector of external distributed loads (given per unit length of the undeformed axis), \mathbf{c}_ρ is the diagonal matrix of the cross-sectional inertia properties, \mathbf{P} is the vector of the generalized boundary loads and $\mathbf{\Psi}$ is the vector of the generalized boundary displacements:

$$\mathbf{N} = \begin{bmatrix} N \\ Q \\ M \end{bmatrix}, \quad \mathbf{c}_\rho = \text{diag}(A\rho, A\rho, I\rho), \quad \mathbf{p} = \begin{bmatrix} p_x \\ p_y \\ m_z \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{P}^T = [P_1, P_2, P_3, P_4, P_5, P_6],$$

$$\mathbf{\Psi}^T = [u(0), v(0), \varphi(0), u(L), v(L), \varphi(L)].$$

When material is elastic and homogenous, \mathbf{N} takes the form:

$$\mathbf{N} = \mathbf{c}_E \boldsymbol{\varepsilon} = \text{diag}(EA, GA_s, EI) \boldsymbol{\varepsilon}.$$

E and G are elastic and shear moduli, A and A_s are the area and the shear area of the cross-section of the beam, and I is its moment of inertia; ρ is density of material; δ denotes the variation; a superposed dot denotes the differentiation with respect to time.

Spatial discretization. The spatial discretization employs Lagrangian polynomials $P_i(s)$ of an arbitrary order K to interpolate the displacements and rotations and their variations:

$$\delta \mathbf{u}(s, t) = \sum_{i=1}^{K+1} \delta \mathbf{U}_i(t) P_i(s), \quad \mathbf{u}(s, t) = \sum_{j=1}^{K+1} \mathbf{U}_j(t) P_j(s), \quad (6)$$

where $\mathbf{U}_i = [U_i, V_i, \theta_i]^T$ is the vector of discrete nodal displacements. Upon the discretization of the variations $\delta \mathbf{u}_m$ in Equation (5), one obtains (subscript m at the discrete variations is omitted for brevity reasons)

$$\sum_{i=1}^{K+1} \int_0^L \left[\mathbf{N}_m^T \boldsymbol{\Lambda}_m \delta \mathbf{U}_i P_i' + \mathbf{N}_m^T \mathbf{W} \boldsymbol{\Lambda}_m (\mathbf{u} + \mathbf{x})_m' \delta \theta_i P_i - (\mathbf{p}^T - \ddot{\mathbf{u}}^T \mathbf{c}_\rho)_m \delta \mathbf{U}_i P_i \right] ds - \mathbf{P}_m^T \delta \boldsymbol{\Psi}_m = 0. \quad (7)$$

Equation (7) constitutes a system of $3(K+1)$ equations for $3(K+1)$ unknown nodal displacement and rotation components. A linear mapping $s = Lz$, $z \in [0, 1]$ is introduced, in order to change the integration boundaries. This will be important when the differentiation with respect to the design variables is performed. The Lagrangian polynomials and the prime (') are now related to the coordinate z . When the mass matrices \mathbf{M}_{ij} of elements are recovered from Equation (7),

$$\mathbf{M}_{ij} = \int_0^1 \mathbf{c}_\rho P_i P_j L dz, \quad i, j = 1, \dots, K+1, \quad (8)$$

the matrices are assembled into a global mass matrix \mathbf{M} . After considering the arbitrariness of the variations, Equation (7) can be written in the form of the ordinary differential equation of the 2nd order in time

$$\mathbf{Q} \equiv \mathbf{M} \ddot{\mathbf{U}} + \mathbf{G} = \mathbf{0}, \quad (9)$$

where \mathbf{G} is a global force vector, assembled from the element force vectors \mathbf{G}_i :

$$\mathbf{G}_i = \int_0^1 \left[\boldsymbol{\Lambda}_m \mathbf{N}_m P_i' + (\mathbf{N}_m^T \mathbf{W} \boldsymbol{\Lambda}_m (\mathbf{u} + \mathbf{x})_m)' \right] \mathbf{c}_2 P_i - \mathbf{p}_m P_i L dz, \quad i = 1, \dots, K+1 \quad (10)$$

where $\mathbf{c}_2 = [0, 0, 1]^T$. Note that, that the product $\mathbf{N}_m^T \mathbf{W} \boldsymbol{\Lambda}_m (\mathbf{u} + \mathbf{x})_m'$ is a scalar and that the generalized boundary loads \mathbf{P}_m and/or the degrees of freedom could be prescribed at nodes $i=1$ and $i=K+1$. The problem to be solved is one of finding nodal displacements $\mathbf{U}(t)$ satisfying Equation (9) and the initial conditions $\mathbf{U}(0) = \mathbf{U}_0$ and $\dot{\mathbf{U}}(0) = \dot{\mathbf{U}}_0$.

3. OPTIMIZATION PROBLEM

3.1 Problem formulation

Optimization of dynamic systems can loosely be described as seeking a combination of geometrical, material and loading data from within an allowed set, so that some specific property of the system at some specific time is minimized or maximized, usually in the presence of time-dependent constraints. Geometrical, material and loading data are made dependent on a group of variables, assembled in the vector \mathbf{b} , termed the *design variables*. A non-linear mathematical programming problem now reads

$$\min f_0, \quad (11)$$

$$\begin{aligned} f_i &\leq 0, \quad i=1, \dots, C \\ \mathbf{b}^L &\leq \mathbf{b} \leq \mathbf{b}^U. \end{aligned} \quad (12)$$

Here f_0 is a scalar objective function at an arbitrary time station; $f_i, i=1, \dots, C$ are constraints, generally each at its own arbitrary time station. \mathbf{b}^L and \mathbf{b}^U are the lower and the upper limit values of the design variables.

Since the defining data of the problem depend explicitly on the design variables, any change of the design variables causes a change in the response of the structure. Hence, by taking into account Equation (9), the system response implicitly depends on the design variables, i.e. $\mathbf{U} = \mathbf{U}(t, \mathbf{b})$.

3.2 Types of the design variables employed in the formulation

Size design variables. In this work, the cross-sectional dimensions (such as flange width or thickness) are assumed to vary linearly along the beam element. Let $\zeta^{z=0}$ and $\zeta^{z=1}$ denote the values of a generic cross-sectional dimension, corresponding to both ends of the beam. Its value along the element is given by

$$\zeta = (1-z)\zeta^{z=0} + z\zeta^{z=1}. \quad (13)$$

By defining $\zeta^{z=0}$ and $\zeta^{z=1}$ in terms of design variables \mathbf{b} , one gets a linearly varying and design-dependent cross-section, *i.e.*

$$A = A(\mathbf{b}), \quad A_s = A_s(\mathbf{b}), \quad I = I(\mathbf{b}). \quad (14)$$

Across the element boundaries, the C^0 continuity of the cross-sectional dimensions can easily be achieved by making the corresponding end-values dependent on the same design variable.

Shape design variables. To enable efficient shape variation description, the design element technique is utilized in the way as presented in Kegl (2000, 2005) and Kegl and Brank (2006). Planar Bezier patches are employed as design elements. Thus, each design element is defined by $I \times J$ control points, whose positions are defined by the vectors \mathbf{q}_{ij} . Points in the design element are related to their global position in the global coordinate system by the mapping

$$\mathbf{r} = \sum_{i=1}^I \sum_{j=1}^J B_i^I B_j^J \mathbf{q}_{ij}.$$

\mathbf{r} is the position vector with respect to the global coordinate system, B_i^I and B_j^J are the i th and j th univariate Bernstein blending polynomials defined in unit space. By defining the position of the beam element in the domain of the design element and by making the control point positions \mathbf{q}_{ij} design-dependent, one gets a design-dependent shape of the finite element mesh. Thus, the position vector \mathbf{x} of a point on the centroid axis of the beam becomes design-dependent, *i.e.*

$$\mathbf{x} = \mathbf{x}(\mathbf{b}). \quad (15)$$

Actuation design variables. By introducing the design variables into the actuating functions, it is possible to determine optimal actuating conditions (Ibrahimbegovic *et al.* 2004), *e.g.* with

the aim to minimize oscillations or maneuver time of a flexible robot. For the reasons given in (Vohar *et al.* 2008), a Bezier function is selected as a general actuating function. In contrast to (Vohar *et al.* 2008), however, a 5th order function χ is adopted here, so that one has

$$\chi = \sum_{i=1}^6 B_i(v) q_i, \quad v \in [0,1]$$

where B_i is the corresponding Bernstein blending polynomial. Since its parameterization is not very convenient for practical optimization tasks, special mappings are introduced as follows: at first, the defining interval $[0,1]$ is mapped into the required time interval $[t_0, t_1]$ by

$$v = \mathcal{G} \xi^2 + (1 - \mathcal{G})(2\xi - \xi^2), \quad \xi = \frac{t - t_0}{t_1 - t_0}, \quad t \in [t_0, t_1]$$

where $\mathcal{G} \in [0,1]$ represents a shape parameter, introduced to improve the adaptability of the curve during optimization. It shifts the ‘interior’ of the function to the left or right, without influencing any of its end points. Secondly, the ‘control values’ q_i are defined as follows

$$q_1 = \chi_0, \quad q_2 = \chi_0 + \frac{\chi_{t_0}}{5v_{t_0}}, \quad q_3 = \chi_0 + \alpha(\chi_1 - \chi_0)$$

$$q_4 = \chi_0 + \beta(\chi_1 - \chi_0), \quad q_5 = \chi_1 - \frac{\chi_{t_1}}{5v_{t_1}}, \quad q_6 = \chi_1$$

where χ_0 and χ_1 are the required starting and end values, χ_{t_0} and χ_{t_1} are the required starting and end velocities, while v_{t_0} and v_{t_1} represent the corresponding derivatives dv/dt , Figure 2. The parameters α and β determine the shape of the curve. By adopting this setup, one gets a very convenient curve, parameterized by meaningful and convenient parameters: boundary values (positions) χ_0 and χ_1 , boundary slopes (velocities) χ_{t_0} and χ_{t_1} , start and end times t_0 and t_1 , and ‘interior’ shape parameters α , β , and \mathcal{G} . Any of these parameters may become design-dependent for the purpose of optimization. In the present work, this curve is utilized to model actuating displacements and rotations, \mathbf{U}^a , or actuating forces and

moments, \mathbf{P}_m . Thus, in general one has

$$\mathbf{U}^a = \mathbf{U}^a(\mathbf{b}), \quad \mathbf{P}_m = \mathbf{P}_m(\mathbf{b}) \quad (16)$$

To further increase flexibility, several curves may easily be combined into a single curve, exhibiting C^1 continuity. This can be achieved by making the corresponding boundary parameters dependent on the same design variable.

4. SOLUTION PROCESS

4.1 Response and sensitivity analysis

The response analysis involves solving the governing equations of the mechanical problem, in our case Equation (9), and evaluating $f_i, i = 0, \dots, C$ of the optimization problem, see Michaleris *et al.* (1994). The most difficult part is the solution of Equation (9), which has to be discretized in time. After the implementation of the midpoint scheme has been performed, the only unknown variables are the discrete displacements \mathbf{U}_{n+1} . Newton's method is used to solve the corresponding equations.

For the evaluation of the governing equations, the known values of \mathbf{U}_n , $\dot{\mathbf{U}}_n$ and $\boldsymbol{\varepsilon}_n$ from the previous time station are needed. As discussed in Section 2, the specialty of the present approach is the use of the rate form of the kinematic equations, Equation (3). This fact has to be taken into account in deriving the linearization of the governing equations.

Once the unknowns \mathbf{U}_{n+1} are calculated, the corresponding velocities $\dot{\mathbf{U}}_{n+1}$ are obtained by the midpoint rules, *i.e.*

$$\dot{\mathbf{U}}_{n+1} = \frac{2}{\Delta t}(\mathbf{U}_{n+1} - \mathbf{U}_n) - \dot{\mathbf{U}}_n. \quad (17)$$

The most difficult part of the sensitivity analysis is the computation of the derivatives of

the response variables with respect to the design variables. These are obtained by the differentiation of Equation (9) with respect to \mathbf{b} . By taking into account the design variable types and the dependencies (14), (15) and (16), it follows from Equation (9)

$$\frac{d\mathbf{M}}{d\mathbf{b}} \ddot{\mathbf{U}} + \mathbf{M} \frac{d\ddot{\mathbf{U}}}{d\mathbf{b}} + \frac{\partial \mathbf{G}}{\partial \mathbf{U}} \frac{d\mathbf{U}}{d\mathbf{b}} + \frac{\partial \mathbf{G}}{\partial \mathbf{b}} = \mathbf{0}. \quad (18)$$

According to Equation (8), the derivative $d\mathbf{M}/d\mathbf{b}$ is assembled from

$$\frac{d\mathbf{M}_{ij}}{d\mathbf{b}} = \int_0^1 \left(\frac{d\mathbf{c}_\rho}{d\mathbf{b}} L + \mathbf{c}_\rho \frac{dL}{d\mathbf{b}} \right) P_i P_j dz. \quad (19)$$

Similarly, according to Equation (10), the derivatives $\partial \mathbf{G}/\partial \mathbf{U}$ and $\partial \mathbf{G}/\partial \mathbf{b}$ are assembled from

$$\begin{aligned} \frac{\partial \mathbf{G}_i}{\partial \mathbf{U}} = & \int_0^1 \left[\frac{\partial \mathbf{N}_m^T}{\partial \mathbf{U}} \left(\Lambda_m P_i' + \mathbf{W} \Lambda_m (\mathbf{u} + \mathbf{x})_m' \mathbf{c}_2 P_i \right) \right. \\ & \left. + \mathbf{N}_m^T \left(\frac{\partial \Lambda_m}{\partial \mathbf{U}} P_i' + \mathbf{W} \frac{\partial \Lambda_m}{\partial \mathbf{U}} (\mathbf{u} + \mathbf{x})_m' \mathbf{c}_2 P_i + \mathbf{W} \Lambda_m \left(\frac{\partial \mathbf{u}}{\partial \mathbf{U}} + \mathbf{x} \right)_m' \mathbf{c}_2 P_i \right) \right] dz \end{aligned} \quad (20)$$

and

$$\begin{aligned} \frac{\partial \mathbf{G}_i}{\partial \mathbf{b}} = & \int_0^1 \left[\frac{\partial \mathbf{N}_m^T}{\partial \mathbf{b}} \left(\Lambda_m P_i' + \mathbf{W} \Lambda_m (\mathbf{u} + \mathbf{x})_m' \mathbf{c}_2 P_i \right) - \mathbf{p}_m^T P_i \frac{dL}{d\mathbf{b}} \right. \\ & \left. + \mathbf{N}_m^T \left(\frac{\partial \Lambda_m}{\partial \mathbf{b}} P_i' + \mathbf{W} \frac{\partial \Lambda_m}{\partial \mathbf{b}} (\mathbf{u} + \mathbf{x})_m' \mathbf{c}_2 P_i + \mathbf{W} \Lambda_m \left(\mathbf{u} + \frac{d\mathbf{x}}{d\mathbf{b}} \right)_m' \mathbf{c}_2 P_i \right) \right] dz. \end{aligned} \quad (21)$$

After the midpoint rules and its derivatives are inserted into Equations (20)-(21), the sensitivity Equation (18) represents a linear system of equations with the unknown displacement derivatives $d\mathbf{U}_{n+1}/d\mathbf{b}$. Once this is solved, the derivatives of the velocities, $d\dot{\mathbf{U}}_{n+1}/d\mathbf{b}$, are found by the differentiation of the midpoint rule, *i.e.*

$$\frac{d\dot{\mathbf{U}}_{n+1}}{d\mathbf{b}} = \frac{2}{\Delta t} \left(\frac{d\mathbf{U}_{n+1}}{d\mathbf{b}} - \frac{d\mathbf{U}_n}{d\mathbf{b}} \right) - \frac{d\dot{\mathbf{U}}_n}{d\mathbf{b}}. \quad (22)$$

Due to the incremental-type of the strain update, the strain derivative with respect to the design variables is calculated as

$$\frac{d\boldsymbol{\varepsilon}_{n+1}}{d\mathbf{b}} = \frac{d\boldsymbol{\varepsilon}_n}{d\mathbf{b}} + \frac{\partial(\Delta\boldsymbol{\varepsilon})}{\partial\mathbf{U}_{n+1}} \frac{d\mathbf{U}_{n+1}}{d\mathbf{b}} + \frac{\partial(\Delta\boldsymbol{\varepsilon})}{\partial\mathbf{U}_n} \frac{d\mathbf{U}_n}{d\mathbf{b}} + \frac{\partial(\Delta\boldsymbol{\varepsilon})}{\partial\mathbf{b}}. \quad (23)$$

Here, $d\boldsymbol{\varepsilon}_n/d\mathbf{b}$ is given from the previous time station, while the remaining terms still have to be computed.

The analytical, exact linearization is carried out symbolically by the computer employing the automatic differentiation software package AceGen (Korelc 1997).

4.2 Optimization using adjustable approximation with an additive convex term

If the functions defined in Equations (11) and (12) are analytically differentiable with respect to \mathbf{b} , the optimization problem can often be very efficiently solved by gradient-based methods of mathematical programming. This exactly agrees with the situation in the present case. Therefore, the gradient-based convex approximation optimization technique (Kegl and Oblak 1997, Kegl *et al.* 2002) is here used to solve the problem, defined in Equations (11) and (12).

The formulations utilize an optimizer based on adjustable approximation with an additive convex term. Whilst solving the optimization problem, the arbitrary functions of the optimization problem f_i are approximated by their corresponding approximations ${}^k g_i$ at design points ${}^k \mathbf{b}$, $k=0,1,2,\dots$, generated during the optimization process. The left hand superscript k denotes the iteration number of the optimization loop.

The optimizer employed the following form for the approximate functions ${}^k g_i$

$${}^k g_i = {}^k c_i + \sum_{j=1}^M {}^k e_{ij} (b_j - {}^k b_j) + \sum_{j=1}^M {}^k \alpha_{ij} (b_j - {}^k b_j)^2,$$

where

$${}^k c_i = f_i \Big|_{\mathbf{b}={}^k \mathbf{b}}, \quad {}^k e_{ij} = \frac{df_i}{db_j} \Big|_{\mathbf{b}={}^k \mathbf{b}},$$

and ${}^k \alpha_{ij}$ is a positive parameter, which influences the conservativeness of the approximation.

Note that this form of ${}^k g_i$ assures strict convexity and separability of the approximate problem. Even more, the Lagrange multipliers of its corresponding dual problem can be expressed explicitly from the Karush-Kuhn-Tucker conditions. This greatly simplifies the solution of the dual problem, from which the solution of the primal approximate problem can be extracted.

Regardless of the values of ${}^k \alpha_{ij}$, ${}^k g_i$ is the first-order approximation of f_i at ${}^k \mathbf{b}$, since it can be easily verified that

$${}^k g_i \Big|_{\mathbf{b}={}^k \mathbf{b}} = {}^k c_i, \quad \frac{d^k g_i}{db_j} \Big|_{\mathbf{b}={}^k \mathbf{b}} = {}^k e_{ij}.$$

Thus, the parameters ${}^k \alpha_{ij}$ can be determined in such a way that ${}^k g_i$ is matched to f_i even better. Their actual values are obtained by imposing the equality of design derivatives of ${}^k g_i$, computed at the previous design point ${}^{k-1} \mathbf{b}$ to ${}^{k-1} e_{ij}$. The ‘trial’ parameters ${}^k \bar{\alpha}_{ij}$ are hence obtained directly from the linear equations

$$\frac{d^k g_i}{db_j} \Big|_{\mathbf{b}={}^{k-1} \mathbf{b}, {}^k \alpha_{ij}={}^k \bar{\alpha}_{ij}} = {}^{k-1} e_{ij}.$$

To assure numerical stability, ${}^k \bar{\alpha}_{ij}$ are not used directly. Instead, the parameters actually used for the approximation, are obtained from

$${}^k\alpha_{ij} = \max\left(\beta^{k-1}\alpha_{ij} + (1-\beta){}^k\alpha_{ij}, \tilde{\varepsilon}\right),$$

where $\tilde{\varepsilon}$ is a small numerical positive value, which assures that ${}^k\alpha_{ij} > 0$, and $\beta \in [0,1]$ is a kind of a damping parameter, which assures that the gradient history (information from previous design points) is also taken into account. This additionally stabilizes the history of the conservativeness parameters ${}^k\alpha_{ij}$.

5. NUMERICAL SIMULATIONS

In this section four numerical examples are analyzed to demonstrate the performance of the present optimization approach. The finite elements with the quadratic interpolation of displacements were used. In order to alleviate locking, reduced numerical integration is used for the stiffness terms and full integration for the inertia terms. The complete finite-element computer code was produced within the AceGen (Korelc 1997). Load moving robot arm simulations were inspired by articles by Moallem *et al.* (2003) and Kane *et al.* (1987).

5.1 Load moving robot arm: geometry optimization

This example addresses the task of moving a mass using a flexible robot arm. The relevant data are presented in Figure 3 and Table 1. The 5 m long hollow rectangular steel robot arm is loaded with a 250 kg point mass at the free end. The arm is rotated about the out-of-plane axis at the pinned end for an angle of $\pi/2$ in $t_1 = 0.5$ s; the motion is driven by the prescribed rotation $\varphi(t)$ about the pinned end of the beam.

The robot arm is modeled by the finite-element mesh consisting of 8 elements of equal length. Each element has linearly varying width and height of the cross section along the element, whereas the wall thickness is constant. Hence, each element has five design variables, *i.e.* two for width and height at each of the two ends, and one for the wall thickness.

1
2
3
4 The entire operation of moving the mass is performed in a horizontal plane on an
5 imaginary frictionless surface. Hence the gravity acts only perpendicularly to the planar
6 movement of the arm and has no effect on it. The mass of the beam and the mass of the
7 weight act as inertial masses.
8
9

10
11
12
13 The optimization task is defined as: move the mass in such a way, that the remaining
14 oscillations after $t_1 = 0.5$ s, when the arm is in the final position, are minimal and use the least
15 amount of material possible.
16
17
18

19
20
21 The first issue is the choice of the objective function. The objective function could be
22 the amount of material or the mechanical energy (sum of kinetic and potential energy) of the
23 arm at time t_1 ; the latter would result in the robot arm standing still at t_1 . Alternatively, the
24 problem could be treated as a multi-objective one and have a little bit of both. Since the main
25 objective is that the system stops at t_1 , the mechanical energy has been chosen to be a
26 minimum at t_1 . Any multi-objective approach would be a compromise between enforcing
27 stopping and minimizing the volume, and would thus leave us with a more oscillating arm at
28 t_1 .
29
30
31
32
33
34
35
36
37
38
39
40

41 This seemingly leaves the demand regarding the minimal amount of material
42 unaddressed. By choosing the mechanical energy to be the objective function, yet setting no
43 constraint regarding the volume, the cross-section could expand uncontrollably. In order to
44 prevent this, a constraint on volume of the material is set.
45
46
47
48
49
50

51 In addition to the volume constraint, two further constraining criteria are imposed. They
52 limit the maximal offsets of the tip of the beam at t_1 to be less than 0.01 m and 0.001 m for
53 the x and y components of displacement, respectively. These criteria might seem redundant;
54 yet they proved helpful in directing the iteration procedure to the desired solutions.
55
56
57
58
59
60

The strategy of the solution is (i) to set some relatively large value for the maximal

1
2
3
4 volume allowed, (ii) fully optimize the design, and then (iii) reduce the allowed volume. A
5
6 bisection-like method is used to determine the minimal amount of material at which the
7
8 optimization is successful. The optimization was considered to be successful, if all the
9
10 constraints were met and the value of the objective function was less than 1 J.
11
12

13
14 As the total mechanical energy of the system is the sum of energies of the beam, Π_{beam} ,
15
16 and the point mass, Π_{mass} , it could be adopted as the objective function (approach A in Figure
17
18 4). This was not the approach employed here, however. Instead, the mechanical energy of the
19
20 beam and the point mass are treated separately, as if each would be an objective function by
21
22 itself. The minimization of the two functions was carried out by constraining both of them to
23
24 a new design variable b_{new} , which assumed the role of the objective function (approach B in
25
26 Figure 4).
27
28
29
30

31
32 The volume of the material and the mechanical energy of the initial design at time t_1 are
33
34 0.01638 m^3 and $1.79 \times 10^4 \text{ J}$, respectively. On the other hand, the corresponding values for
35
36 the optimized design are 0.01251428 m^3 and 0.0516 J . The shape of the optimized design in
37
38 2D view is shown in Figure 5. The width and height of the cross-section of the optimized
39
40 robot arm are the largest at the pinned end, and decrease to the minimal allowed dimensions
41
42 near the free end. The optimal wall thickness of the cross-section is found to be the minimal
43
44 allowed value (5 mm). A true-scale 3D axonometric view of the optimized arm is shown in
45
46 Figure 6.
47
48
49

50
51 The optimized system's energy comes to a minimum value at t_1 and remains unchanged
52
53 for the rest of the analysis, which lasts 0.75 s. This is to be contrasted to the initial design,
54
55 which oscillates violently about the anticipated final position, as can clearly be seen in Figure
56
57 7.
58
59
60

1
2
3
4 The driving torque of the initial and the optimized design solutions, corresponding to
5 the prescribed rotation of the pinned end, is shown in Figure 8.
6
7

8
9 The results for the mechanical energy, shown in Figure 9, confirm theoretical
10 deductions of energy conservation. The energy of the system is conserved after non-
11 conservative loads are removed from the system.
12
13
14

15 16 17 *5.2 Load moving robot arm: drive optimization*

18
19 This time the cross-section of the arm is taken to be constant along the entire length of
20 the robot arm ($w/h/t = 0.1/0.2/0.005$ m), and not subjected to optimization. A driving
21 function is sought that results in the arm stopping dead at t_1 . The prescribed time-function of
22 the rotation of the drive is now a six point Bezier curve. The first two and the last two points
23 of the curve are fixed to assure zero angular velocity at $t = 0$ s and at $t_1 = 0.5$ s. The remaining
24 two points are the design variables. The objective function is the mechanical energy at time
25 $t_1 = 0.5$ s, with the constraining criteria remaining as in Section 4.1.
26
27
28
29
30
31
32
33
34
35

36
37 Figure 10, left, compares the graphs of the optimized driving function and the wave up
38 function used in the previous numerical example. It is interesting to see that the two functions
39 are only slightly different. The response, particularly for $t > 0.3$ s, is, on the other hand, a lot
40 different, see Figure 10, right.
41
42
43
44
45

46
47 The energy of the optimized system at t_1 is about 2 J, i.e. very much less than the
48 maximal energy of the system, which is roughly 1.9×10^5 J at $t = 0.25$ s.
49
50
51

52 53 *5.3 Load moving robot arm: time optimization*

54
55 This numerical example attempts to find the driving function that minimizes the time
56 needed to move the mass. Thus the objective function is the lifting time. In addition, a
57 constraint is added, which sets the upper limit for mechanical energy at the final position to be
58
59
60

1
2
3
4
5 J. The geometry of the arm is the same as in Section 4.2.
6

7 The optimized driving function completes the movement in $t_1 = 0.4499$ s, which is a
8
9 10% improvement with respect to the initial design. The optimized driving function is shown
10
11 in Figure 11, left. The tip x coordinate, which includes rigid body motion of the arm, is
12
13 compared with the displacement of the rigid arm in Figure 11, right, being equal to $L \cos \varphi$.
14
15 Note that the rigid arm is a possible solution for a dead-stop type of a robot arm. Initially the
16
17 flexible arm lags behind the rigid-arm solution but the situation reverses for $t > 0.224$ s, as
18
19 the flexible bar overtakes the rigid-arm. The two solutions agree completely for $t > t_1$.
20
21
22
23
24

25 5.4 Jerk resistant elastic steel frame 26

27 The purpose of this numerical example is to show the application of the design element
28
29 strategy in the structural analysis. The structure under investigation is a seven storey, three
30
31 bay steel frame (Figure 12). The height of the upper storeys is 3 m, and is not subjected to
32
33 optimization. The initial design height of the first storey is 6 m, but can be reduced to 3 m
34
35 during optimization. Masses $M_1 = 20$ tons and $M_2 = 5$ tons are placed at inner and outer
36
37 column-beam intersections, respectively (Figure 12).
38
39
40
41

42 The properties of steel are the same as in example 4.1. All members of the frame have a
43
44 hollow rectangular cross-section with dimensions $w/h/t = 0.4/0.4/0.006$ m. The frame is
45
46 subjected to a sudden movement at its base, which imitates a single, highly idealized jerk. The
47
48 base of the structure moves 0.5 m to the right, peaks at $t_1 = 0.5$ s, and returns to the initial
49
50 position at $t_2 = 1$ s. The base displacement history is shown in Figure 13. The time step
51
52 $\Delta t = 0.02$ s is used in the analysis.
53
54
55
56

57 Each beam or column of the frame is modeled by a single quadratic finite element. Two
58
59 design elements with four control points along the height of the structure are introduced to
60

1
2
3
4 optimize the shape of the upper storeys. x coordinates of the design elements are adopted as
5
6 design variables. The further design variable is the height of the first storey.
7
8

9
10 The objective of the optimization is to minimize the oscillations of the frame after the
11 base movement subsides. In other words, the frame should return to its initial position with a
12 minimal residual velocity and post-jerk oscillations. Consequently, a series of subsequent
13 equal jerks would also result in a non-oscillating frame. The width of the frame and the height
14 of the first storey must remain within 12 to 24 m and 3 to 6 m, respectively. The span of the
15 centre bay beams must remain within 4 to 10 m.
16
17
18
19
20
21
22

23
24 The most suitable objective function for minimizing the oscillations of a structure at a
25 given time is the mechanical energy. Unfortunately, this requirement alone failed to yield an
26 acceptable result. In order to resolve the issue, the solution procedure is divided into two
27 phases. The first phase of the procedure starts with choosing the horizontal displacement at
28 the top of the frame at times $t_A = 1$ s and $t_B = 1.2$ s as the objective function and using the
29 approach B from Figure 4. Once a corresponding optimum is found, the horizontal
30 displacement at time $t_C = 1.4$ s is added to the objective function, and the related optimum is
31 found. With this optimum in hand, the time history of the top displacement, shown in Figure
32 14, left, meets the optimization task criteria quite well, however the strong oscillations of the
33 mid-height stories still remain.
34
35
36
37
38
39
40
41
42
43
44
45
46
47

48 In the second phase of the solution procedure, these oscillations are removed by setting
49 the mechanical energy of the structure to be the objective function. A substantial
50 improvement compared to the initial design is obvious from Figure 14.
51
52
53
54
55
56

57 CONCLUSIONS

58
59 The concepts of the recently developed energy-conserving time-integration scheme
60

1
2
3
4 (Gams *et al.* 2007) and the efficient optimal design approach (Kegl and Oblak 1997, Kegl *et*
5
6
7 *al.* 2002) were briefly presented. They were employed in the size, shape, and drive
8
9 optimization of dynamically loaded and geometrically exact planar frame structures. The
10
11 unifying concept of design variables introduced here allows addressing a variety range of
12
13 engineering problems, from the size/shape/drive optimization of very flexible manipulators to
14
15 the size/shape optimization of structures in the transient dynamic response undergoing finite
16
17 displacements and rotations. Efficient shape optimization is enabled by the utilization of the
18
19 design element technique and the Bezier patches.
20
21

22
23 Four numerical examples of the dynamic optimization are presented. The numerical
24
25 results confirm the practical value of the proposed approach and indicate that optimization is
26
27 still a task largely dependent on the experience of the analyst. The following issues require
28
29 further attention. One can see, with no exception even from simplest examples, that the
30
31 optimization problem in transient dynamics is essentially multi-objective. It is well known
32
33 that there is no universal way of tackling multi-objective problems. Our solution was to
34
35 introduce an additional design variable; this worked very well. It should be noted, though, that
36
37 the choice of appropriate normalization constants (weights) is completely left to the judgment
38
39 of an analyst. The second issue concerns handling the time-dependent constraint functions.
40
41 Often discussed fixed-point-wise treatment works very well for the present range of numerical
42
43 examples. The numerical experimentations show that the number of time-points to impose the
44
45 time constraints needs to be rather small. Their choice can be efficiently introduced
46
47 interactively during the progress of the optimization procedure. The third issue deals with
48
49 virtually redundant displacement constraints (Example 4.1) and the two-stage optimization
50
51 procedures (Example 4.4). It was found that, in the iteration process, the gradient-based
52
53 optimizer can get trapped into undesirable (either unfeasible or unacceptable from engineer's
54
55
56
57
58
59
60

1
2
3
4 point of view) regions of the design space, if the initial design points were far from the
5
6 solution and/or corresponded to some violently oscillating structures. This might be a
7
8 consequence of very rough or jagged design spaces. The displacement constraints and the two
9
10 stage optimizations are hence both empirically motivated and very helpful in obtaining a near-
11
12 optimal design point at the first stage of optimization. Once in the vicinity of the optimum, the
13
14 optimization typically progresses very quickly. The last issue, but not least important in
15
16 developing computer tools for dynamics optimization, is the use of the symbolic, automatic
17
18 analytical differentiation and computer code generating tools (Korelc 1997). Such a tool
19
20 makes tedious and time consuming tasks of obtaining the tangent stiffness matrix and the
21
22 sensitivity coefficients as well as coding computer subroutines much easier.
23
24
25
26
27
28
29

30 ACKNOWLEDGEMENT

31
32 The work of M. Gams was financially supported by the Agency of research of the Republic of
33
34 Slovenia under contract 8311-03-831622. The support is gratefully acknowledged.
35
36
37
38

39 REFERENCES

40
41 Arora, J.S. and Dutta, A., 1997. Explicit and implicit methods for design sensitivity analysis
42
43 of nonlinear structures under dynamic loads. *Applied Mechanics Reviews*, 50 (11 pt 2),
44
45 S11-S19.
46
47
48
49

50
51 Cardoso, J.B. and Arora, J.S., 1992. Design sensitivity analysis of nonlinear dynamic
52
53 response of structural and mechanical systems. *Structural Optimization*, 4 (1), 37–46.
54
55
56
57

58
59 Cho, S. and Choi, K.K., 2000. Design sensitivity analysis and optimization of non-linear
60
transient dynamics. Part I – Sizing design. *International Journal for Numerical Methods in*

1
2
3
4
5 *Engineering*, 48 (3), 351-373.
6
7

8
9 Cho, S. and Choi, K.K., 2000. Design sensitivity analysis and optimization of non-linear
10 transient dynamics. Part II – Configuration design. *International Journal for Numerical*
11 *Methods in Engineering*, 48 (3), 375-399.
12
13

14
15
16
17
18 Crisfield, M.A., Galvanetto, U. and Jelenić, G., 1997. Dynamics of 3-D co-rotational beams.
19
20 *Computational Mechanics*, 20 (6), 507-519.
21
22

23
24
25
26 Gams, M., Planinc, I. and Saje, M., 2007. Energy conserving integration scheme for
27 geometrically exact beam. *Computer Methods in Applied Mechanics and Engineering*, 196
28 (17-20) 2117-2129.
29
30
31

32
33
34
35 Ibrahimbegovic, A. and Mamouri, S., 1999. Nonlinear dynamics of flexible beams in planar
36 motion: formulation and time-stepping scheme for stiff problems. *Computers and Structures*,
37
38 70 (1), 1-22.
39
40
41

42
43
44 Ibrahimbegovic, A. and Mamouri, S., 2002. Energy conserving/decaying implicit time-
45 stepping scheme for three-dimensional beams undergoing finite rotations. *Computer Methods*
46 *in Applied Mechanics and Engineering*, 191 (37-38), 4241-4258.
47
48
49

50
51
52
53 Ibrahimbegovic, A., Knopf-Lenoir, C., Kucerova, A. and Villon, P., 2004. Optimal design and
54 optimal control of elastic structures undergoing finite rotations and deformations.
55
56 *International Journal for Numerical Methods in Engineering*, 61 (14), 2428-2460.
57
58
59
60

1
2
3
4
5
6
7 Kane, T.R., Ryan, R.R. and Banerjee, A.K., 1987. Dynamics of a cantilever beam attached to
8 a moving base. *Journal of Guidance, Control, and Dynamics*, 10 (2), 139-151.
9

10
11
12
13
14 Kang, B.S., Park, G.J. and Arora, J.S., 2006. A review of optimization of structures subjected
15 to transient loads, *Structural and Multidisciplinary Optimization*, 31 (2), 81–95.
16
17

18
19
20
21 Kegl, M. and Oblak, M.M., 1997. Optimization of mechanical systems: On non-linear first-
22 order approximation with an additive convex term. *Communications in Numerical Methods in*
23 *Engineering*, 13 (1), 13-20.
24
25
26

27
28
29
30 Kegl, M., 2000. Shape optimal design of structures: an efficient shape representation concept.
31 *International Journal for Numerical Methods in Engineering*, 49 (12), 1571-1588.
32
33

34
35
36
37 Kegl, M., Butinar, B. and Kegl, B., 2002. An efficient gradient-based optimization algorithm
38 for mechanical systems. *Communications in Numerical Methods in Engineering*, 18 (5),
39 363-371.
40
41
42

43
44
45
46 Kegl, M., 2005. Parameterization based shape optimization: theory and practical
47 implementation aspects. *Engineering Computations*, 22 (5/6) 646–663.
48
49
50

51
52
53
54 Kegl, M. and Brank, B., 2006. Shape optimization of truss-stiffened shell structures with
55 variable thickness. *Computer Methods in Applied Mechanics and Engineering*, 195 (19-22),
56 2611-2634.
57
58
59
60

1
2
3
4
5
6
7 Kim, N.H. and Choi, K.K., 2001. Design sensitivity analysis and optimization of nonlinear
8 transient dynamics. *Mechanics of Structures and Machines*, 29 (3), 351-371.

9
10
11
12
13
14 Korelc, J., 1997. Automatic generation of finite-element code by simultaneous optimization of
15 expressions. *Theoretical Computer Science*, 187 (1-2), 231-248.

16
17
18
19
20
21 Kuhl, D. and Ramm, E., 1996. Constraint energy momentum algorithm and its application to
22 non-linear dynamics of shells. *Computer Methods in Applied Mechanics and Engineering*,
23
24 136 (3-4), 293-315.

25
26
27
28
29
30 Kuhl, D. and Crisfield, M.A., 1999. Energy-conserving and decaying algorithms in non-linear
31 structural dynamics. *International Journal for Numerical Methods in Engineering*, 45 (5),
32
33 569-599.

34
35
36
37
38
39 Kulkarni, M. and Noor, A.K., 1995. Sensitivity analysis of the nonlinear dynamic viscoplastic
40 response of 2-D structures with respect to material parameters. *International Journal for*
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
Numerical Methods in Engineering, 38 (2), 183-198.

59
60 Michaleris, P., Tortorelli, D.A. and Vidal, C.A., 1994. Tangent operators and design
sensitivity formulations for transient nonlinear coupled problems with applications to elasto-
plasticity. *International Journal for Numerical Methods in Engineering*. 37 (14), 2471-2499.

Moallem, M., Patel, R.V. and Khorasani, K., 2003. Design of an actuated flexible arm for

1
2
3
4 improved vibration properties. *Journal of Vibration and Control*, 9 (9), 1041-1056.
5
6
7

8
9 Okubo, S. and Tortorelli, D.A., 2004. Control of nonlinear, continuous, dynamic systems via
10 finite elements, sensitivity analysis and optimization. *Structural and Multidisciplinary*
11 *Optimization*, 26 (3-4), 183-199.
12
13
14

15
16
17
18 Pedersen, C.B.W., 2004. Crashworthiness design of transient frame structures using topology
19 optimization. *Computer Methods in Applied Mechanics and Engineering*, 193 (6-8), 653-678.
20
21
22

23
24
25
26 Reissner, E., 1972. On one-dimensional finite-strain beam theory: The plane problem,
27 *Zeitschrift für angewandte Mathematik und Physik ZAMP*, 23 (5), 795-804.
28
29

30
31
32
33 Sansour, C., Wriggers, P. and Sansour, J., 2004. On the design of energy-momentum
34 integration schemes for arbitrary continuum formulations. Applications to classical and
35 chaotic motion of shells. *International Journal for Numerical Methods in Engineering*, 60
36
37
38 (15), 2419-2440.
39
40
41

42
43
44
45 Simo, J.C., Tarnow, N. and Doblare, M., 1995. Non-linear dynamics of three-dimensional
46 rods: exact energy and momentum conserving algorithms. *International Journal for*
47 *Numerical Methods in Engineering*, 38 (9), 1431-1473.
48
49
50

51
52
53
54 Sousa, L.G., Cardoso, J.B. and Valido, A.J., 1997. Optimal cross-section and configuration
55 design of elastic-plastic structures subject to dynamic cyclic loading. *Structural Optimization*,
56
57
58 13 (2-3), 112-118.
59
60

1
2
3
4
5
6
7 Stupkiewicz, S., 2001. Approximate response sensitivities for nonlinear problems in explicit
8 dynamic formulation. *Structural and Multidisciplinary Optimization*, 21 (4), 283-291.
9
10

11
12
13
14 Vohar, B., Kegl, M. and Ren, Z., 2008. Implementation of an ANCF beam finite element for
15 dynamic response optimization of elastic manipulators. *Engineering Optimization*, 40 (12)
16 1137–1150.
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

TABLES

Table 1. Data for cross-section, material parameters and load.

Cross-section (initial)	$w/h/t = 0.095 / 0.19 / 0.006 \text{ m}$
Cross-section (minimal)	$w/h/t = 0.05 / 0.1 / 0.005 \text{ m}$
Cross-section (maximal)	$w/h/t = 0.5 / 1 / 0.015 \text{ m}$
Material properties	$E = 210 \text{ GPa}, \nu = 0.3, \rho = 7800 \text{ kg/m}^3$
Prescribed rotation φ (the wave up function)	$\varphi(t) = \begin{cases} \frac{\pi}{4} \left(1 - \cos\left(\frac{\pi t}{t_1}\right) \right); & 0 \leq t \leq t_1 \\ \frac{\pi}{2}; & t > t_1 \end{cases}$
Lifting time / time step	$t_1 = 0.5 \text{ s} / \Delta t = 0.001 \text{ s}$

FIGURE CAPTIONS

Figure 1. Beam configurations, left. Stress resultants, right.

Figure 2. Parameters of the curve, used for actuation of dynamic systems.

Figure 3. Load moving robot arm: geometrical data.

Figure 4. Two approaches to minimizing the mechanical energy of the structure.

Figure 5. The width (left), and the height (right) of the optimized beam cross-section. x and y axes scales are different for better view.

Figure 6. 3D view of the optimized shape of the robot arm.

Figure 4. Tip x coordinate of the initial (dashed line) and the optimized design (solid line).

Figure 5. Driving torque of the initial (dashed line) and the optimized design (solid line).

Figure 6. Mechanical energy of the robot arm – point mass system. Oscillations of energy after load is removed (from 0.5 s to 0.75 s) are shown in the magnification.

Figure 7. Rotation of the pinned end (left) and tip x coordinate (right) versus time. The solid line represents the optimized drive solution, the dashed line the solution with the wave up function.

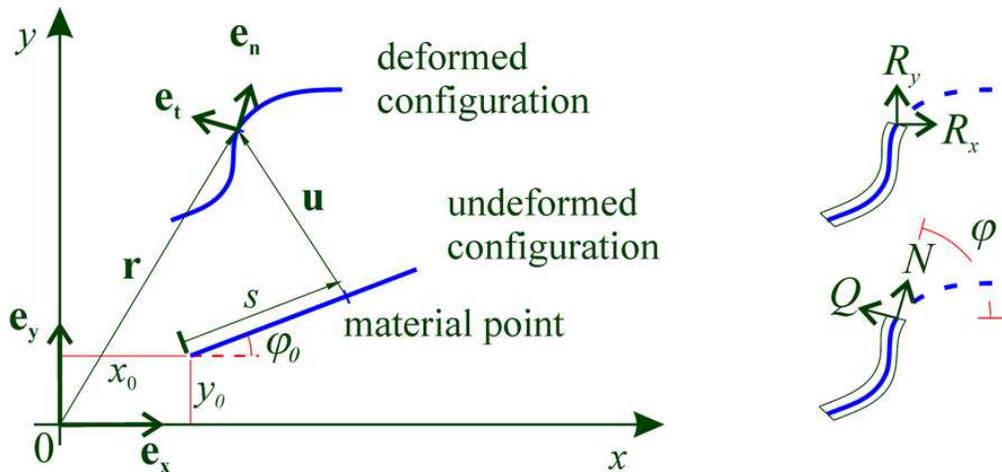
Figure 8. Rotation of the pinned end versus time after optimization (left), and tip x coordinate of the flexible robot arm compared to the rigid arm (right).

Figure 9. The initial geometry of the frame, with point masses, control points CP and design elements DE marked (left), and the optimized shape (right). Both drawings are in true scale.

Figure 10. Prescribed base motion (functional relation, left, and its plot, right).

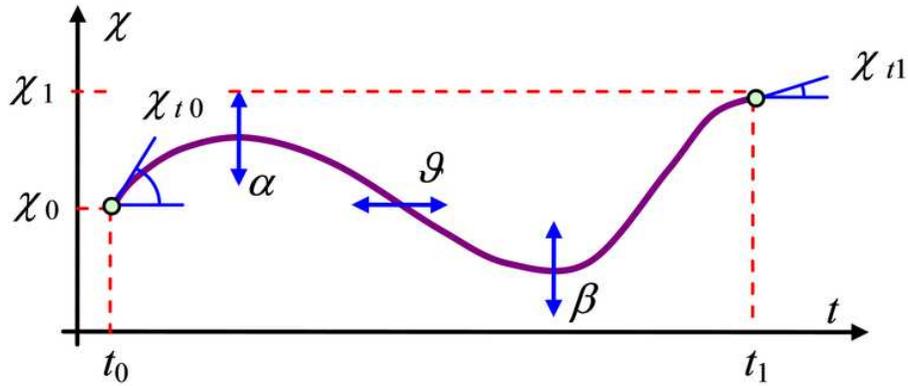
Figure 11. x coordinate of the node at the top of the frame (solid line) and at mid-height (dashed line). The solution of the first phase with strong mid-storey oscillations still remaining(left), and the final solution (right).

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60



37x17mm (600 x 600 DPI)

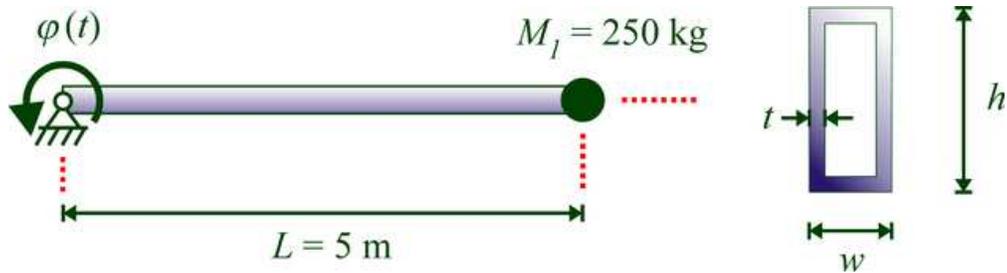
er Review Only



35x15mm (600 x 600 DPI)

er Review Only

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60



26x7mm (600 x 600 DPI)

Peer Review Only

(A)

$$f_0 = \Pi_{beam} + \Pi_{mass}$$

(B)

$$h_1 = \Pi_{beam}; \quad h_2 = \Pi_{mass},$$

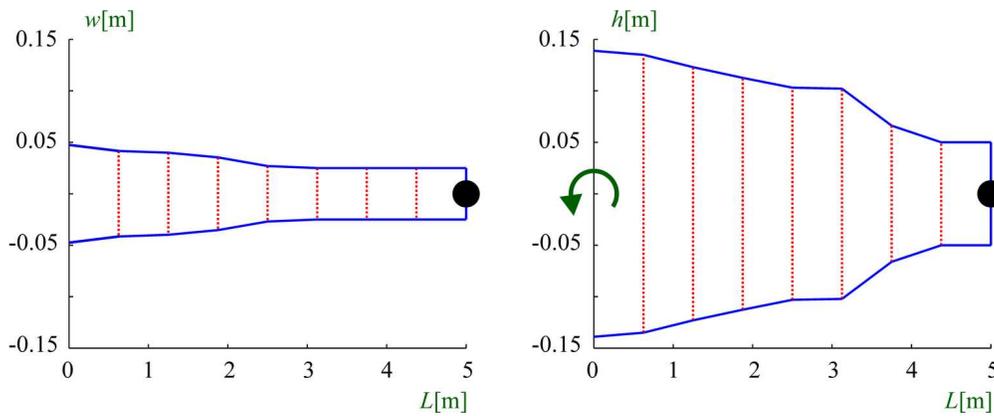
$$h_1 \leq b_{new}; \quad h_2 \leq b_{new},$$

$$f_0 = b_{new}$$

32x9mm (600 x 600 DPI)

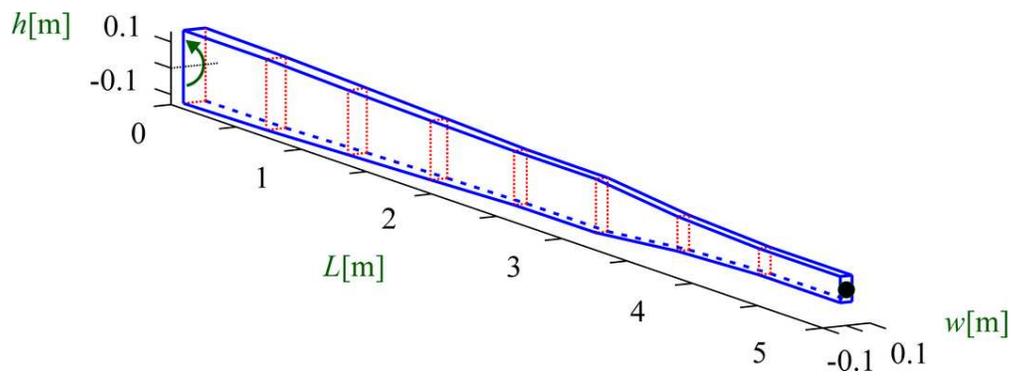
Peer Review Only

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60



65x26mm (600 x 600 DPI)

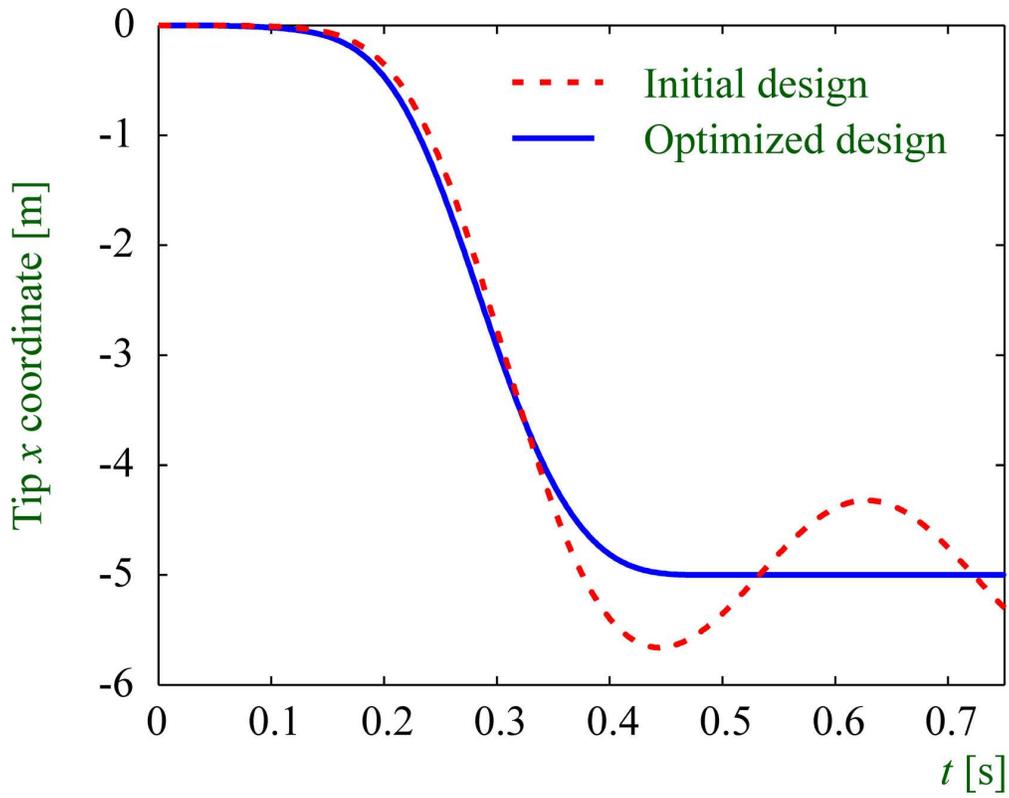
er Review Only



44x16mm (600 x 600 DPI)

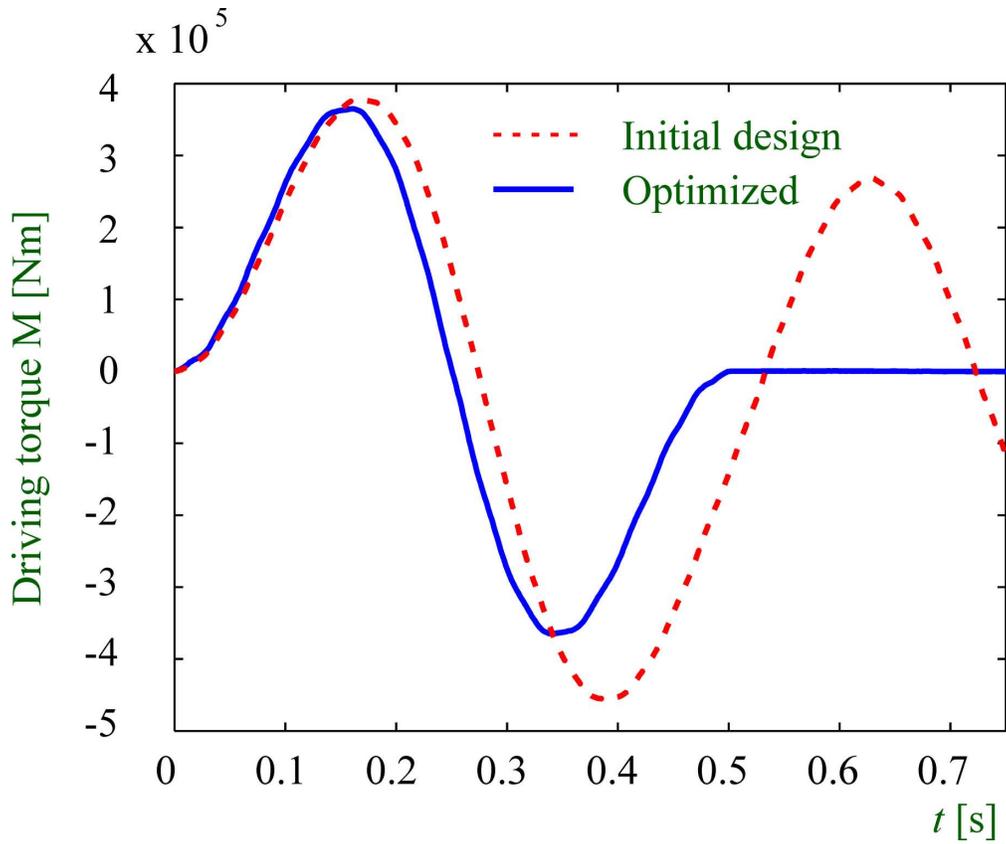
Peer Review Only

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60



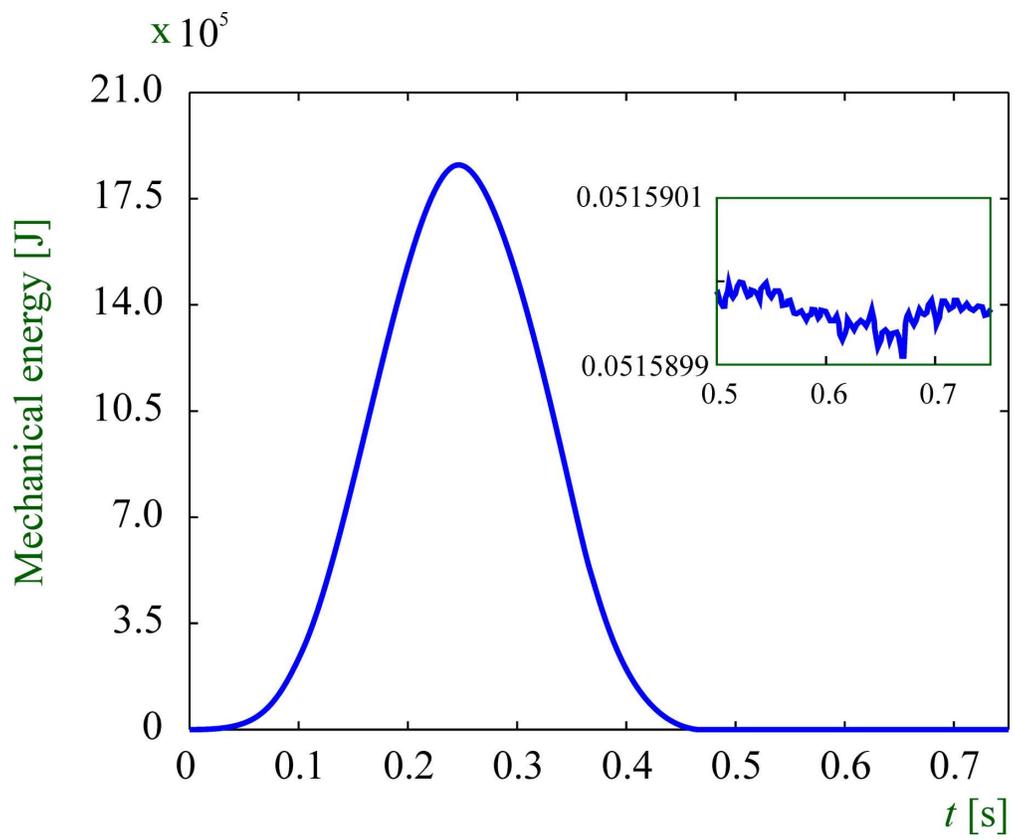
71x56mm (600 x 600 DPI)

Preview Only



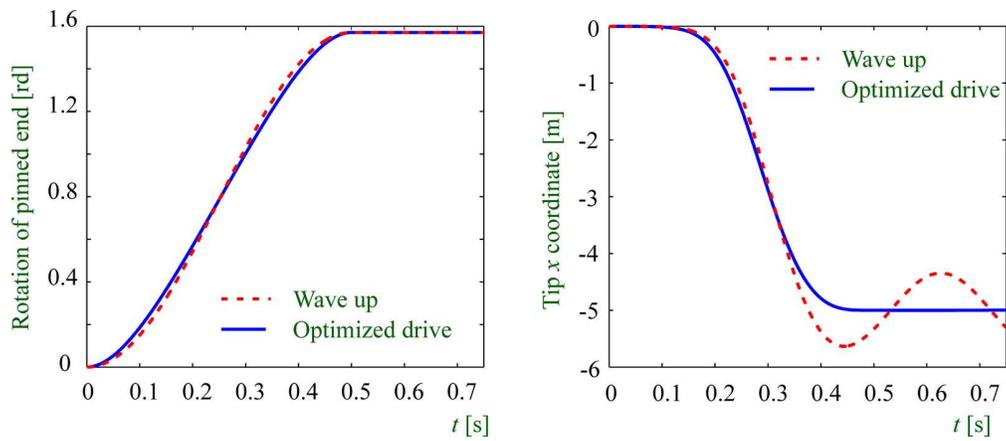
77x64mm (600 x 600 DPI)

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60



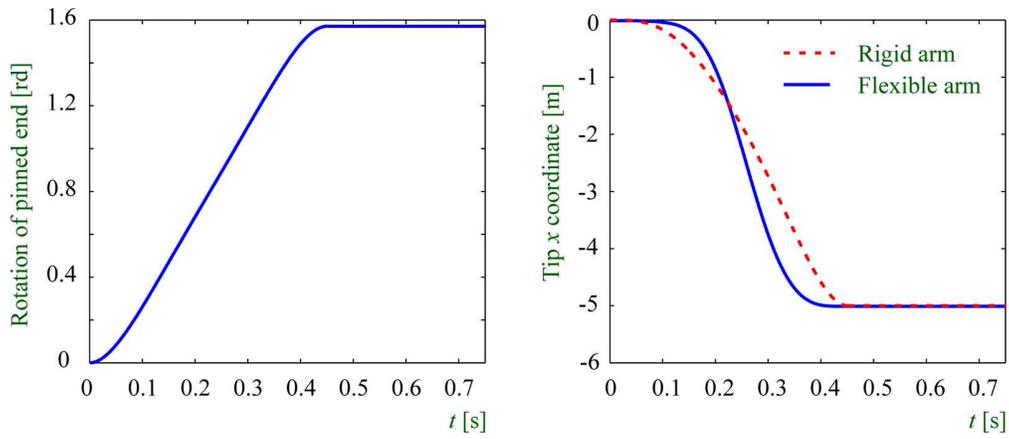
77x63mm (600 x 600 DPI)

View Only



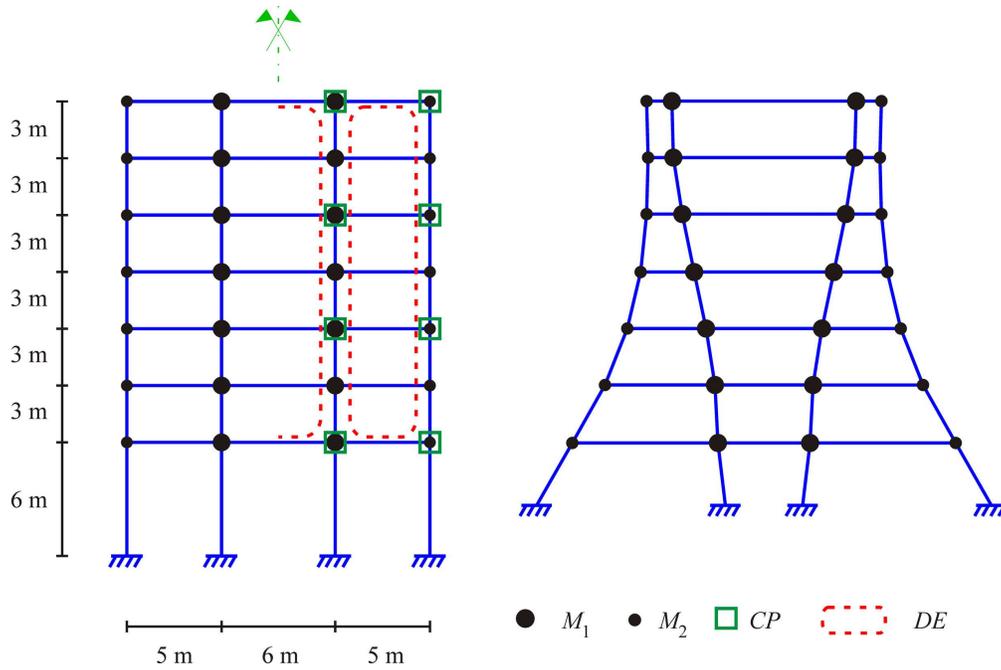
68x29mm (600 x 600 DPI)

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60



68x29mm (600 x 600 DPI)

Peer Review Only



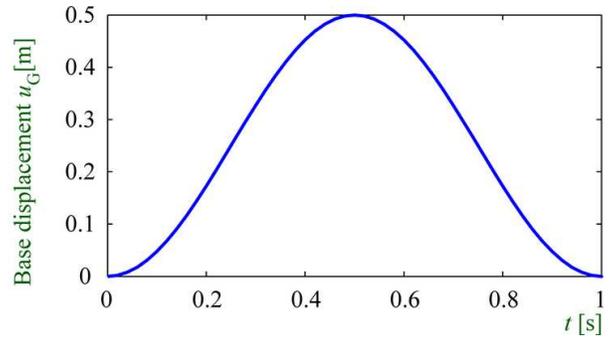
104x68mm (600 x 600 DPI)

view Only

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

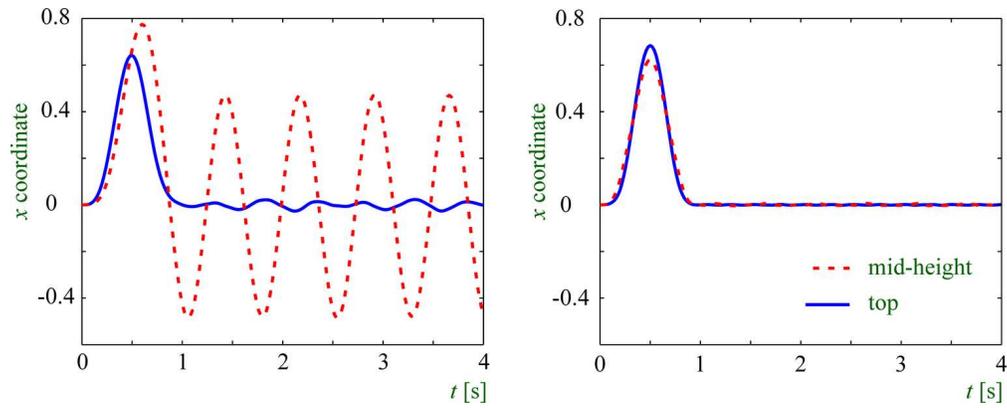
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

$$u_G(t) = \begin{cases} \frac{1}{4} \left(1 - \cos\left(\frac{\pi t}{t_1}\right) \right); & 0 \leq t \leq t_2 \\ 0; & t > t_2 \end{cases}$$



52x17mm (600 x 600 DPI)

Peer Review Only



64x25mm (600 x 600 DPI)

Peer Review Only