

Time varying LQR-based optimal control of geometrically exact Reissner's beam model

Suljo Ljukovac^{1,2a}, Adnan Ibrahimbegovic^{*1,3} and Maida Cohodar-Husic⁴

¹University of Technology Compiègne-Alliance Sorbonne University, Laboratoire Roberval, Centre de Recherche Royallieu, Rue du Docteur Schweitzer, Compiègne, 60200, Hauts-de-France, France

²Faculty of Civil Engineering, University of Sarajevo,

Patriotske Lige 30, Sarajevo, 71000, Bosnia & Herzegovina

³Institut Universitaire de France, 1 Rue Descartes, Paris, 75000, France

⁴Faculty of Mechanical Engineering, University of Sarajevo,

Vilsonovo Setaliste 9, Sarajevo, 71000, Bosnia & Herzegovina

(Received November 14, 2023, Revised November 16, 2023, Accepted November 21, 2023)

Abstract. In this work, we propose combining an advanced optimal control algorithm with a geometrically exact beam model. For simplicity, the 2D Reissner beam model is chosen to represent large displacements and rotations. The difficulty pertains to the nonlinear nature of beam kinematics affecting the tangent stiffness matrix, making it non-constant, which compromises direct use of optimal control methods for linear problems. Thus, we seek to accommodate a time varying control using linear-quadratic regulator (LQR) algorithm with the proposed geometrically nonlinear beam model. We provide a detailed theoretical formulation and its numerical implementation in a variational format form. Several illustrative numerical examples are provided to confirm an excellent performance of the proposed methodology.

Keywords: geometrically exact kinematics; linear-quadratic regulator; optimal control; Reissner's beam

1. Introduction

With a quest for increasing renewable energy share, the European Community has launched the grand challenge of delivering wind-turbine installations that can provide 10 MW electric power per year, which doubles the current maximal capacity in Europe. Our main *research hypothesis* (see Fig. 1) is that such a production increase can be achieved with combined efforts of exploring technological innovation towards *larger wind-turbines with flexible blades* as flexible multibody systems in order to guarantee the turbine safety under extreme wind conditions.

The main impact of this project is to significantly improve upon the currently dominant system of wind-turbines with stiff blades developed for offshore locations in Denmark and Northern Germany, which offer near optimal conditions with steady winds. In order to further quantify

*Corresponding author, Professor, E-mail: adnan.ibrahimbegovic@utc.fr

^aPh.D. Student, E-mail: suljo.ljukovac@utc.fr

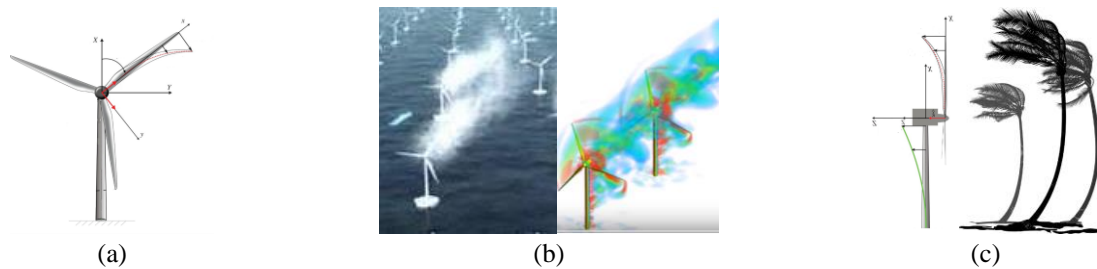


Fig. 1 Meeting 10 MW EU challenge (a) single wind-turbine with 10MW production rough estimate (Ibrahimbegovic 2018) at 10 m/s wind speeds requires 100 m long flexible blades; (b) direct problem: wind-turbine farm loads computations with reduced power for each wind-turbine due to wakes and drag forces; (c) control of large vibration of flexible wind-turbine blades

existing margins for progress for the location with variable wind speeds, we note that the Danish-German wind-turbine system, which is currently predominantly used in France, remains operational on the average only 24.7% of the time. Furthermore, even the optimal wind conditions cannot deliver the maximum production, since the blades are stopped for wind speeds of over 90 km/h (e.g., only 78.8% of full capacity was provided by storm Ciara that hit France on February 10, 2020, since the turbines at coastal location had to be switched off due to wind speeds reaching 120 km/h).

The main goal of this work is to provide the increased interval of wind speeds in that we will allow for an increase of the highest operational wind speeds and thus for better harvesting of wind resources. Such improvement requires corresponding contributions in nonlinear dynamics and, more precisely, the numerical models of large wind-turbines with flexible blades that will provide improved safety under extreme wind conditions. Also, in order to control stability and avoid damage, we need to provide the corresponding algorithms for control of the vibrations of the geometrically exact beam that is used as the basic ingredient in constructing the numerical model of flexible wind-turbine blades.

In order to ensure the desired response of the structure and suppress its vibrations, many papers and books are focused on searching for the most appropriate control algorithm. The proportional-integral-derivative (PID) controllers have been used for many years for linear problems, due to their simplicity, efficiency and reliability, PID controllers and their different modifications are still the most popular to use for control processes of flexible systems. Special attention is focused on optimization of PID controller parameters. To design an optimal analogue and discrete PID controller, an analogue and discrete time linear quadratic regulator (LQR) as suggested in (Das *et al.* 2013). Optimal tuning of PI/PID controller for time-delay processes using LQR is further proposed in (He *et al.* 2000).

Many studies have investigated the control problems of the flexible system mechanics. In addition to other analyses, (Rafiee *et al.* 2017) gave a critical review of control algorithms applied to rotating composite beams and blades. The position control of a flexible four bar mechanism has been provided by PID controller while proportional controller has been used to suppress vibration in (Trevisani *et al.* 2003). Optimal control algorithms, based on linear quadratic regulator (LQR), are very popular due of their robustness and has been used to improve overall performances of flexible structures: with friction compensation for flexible high-speed rack feeders in (Schindele *et al.* 2014); for Stewart platform with flexible legs (Chen *et al.* 2018), for flexible beam linkage

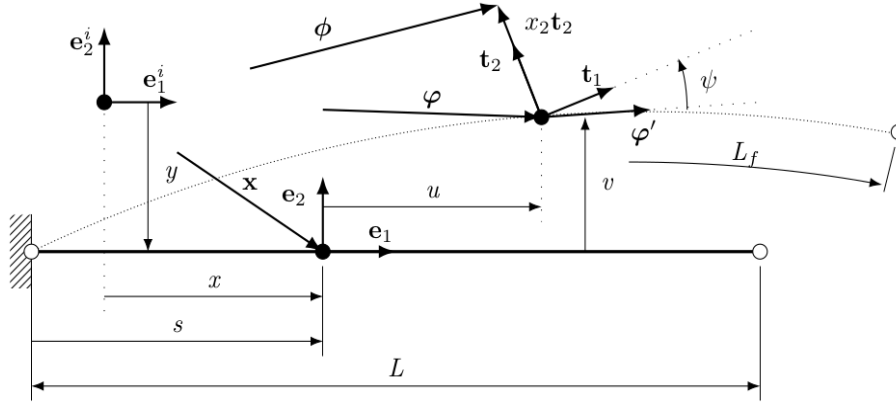


Fig. 2 Initial and deformed configuration of a geometrically exact 2D beam

(Zhou *et al.* 2001), flexible riser (Song *et al.* 2021), Timoshenko beam (Hernández *et al.* 2011).

Active vibration control of a piezoelectric beam was investigated in (Neto *et al.* 2013, Vasques *et al.* 2006), where several different popular control strategies were applied and compared: the constant gain velocity feedback (CGVF), the constant amplitude velocity feedback (CAVF), LQR and Linear Quadratic Gaussian controller (LQG) in (Vasques *et al.* 2006) and CAVF, CGVF and LQR regulator (Neto *et al.* 2013). More works on control theory can be referenced in the books of (Trelat 2005, Lewis *et al.* 2012, Glad *et al.* 2000, Kirk 2004). The main novelty of this work is to choose among numerous control algorithms the most suitable strategy for control of the vibrations of the geometrically exact Reissner beam. In this paper we still use the optimal control based on LQR algorithm but accounting for changing tangent stiffness resulting with a nonlinear form of the Kalman gain based upon Riccati's solution.

The outline of the paper is as follows. In Section 2, we briefly present the geometrically exact beam model that is used for representing the large motion of the flexible blades in nonlinear dynamics framework. In Section 3, we present the corresponding developments of optimal nonlinear control for this model with a state feedback loop. Details of numerical implementation are given in Section 4, followed by the results of numerical simulation given in Section 5. In Section 6, we state the conclusions.

2. Brief on 2D Reissner's beam model

We consider a geometrically exact 2D Reissner's beam with the points of its neutral axis parametrized by $s \in [0, L]$, see Fig. 2, where L is its length and A is the beam cross-section. Thus, the beam volume V is given as $V = L \times A$. Let the beam domain Ω can be considered as an open bounded set $V = L \times A$ with smooth boundary $\partial\Omega = \partial\Omega_u \cup \partial\Omega_\sigma$ and $\emptyset = \partial\Omega_u \cap \partial\Omega_\sigma$. The initial configuration is described by the position vector \mathbf{x} of a point on the beam neutral axis and an orthonormal basis with unit vectors \mathbf{e}_i , ($i = 1, 2$).

Thus, the tangent to the neutral axis is defined as $\mathbf{e}_1 = \mathbf{x}'$, where $\mathbf{e}_2^T \cdot \mathbf{e}_1 = 0$. The notation $(\cdot) = \partial(\cdot)/\partial s$ indicates the partial derivative with respect to $s \equiv x$. The beam deformed configuration is described by

$$\boldsymbol{\phi} = \boldsymbol{\varphi} + x_2 \mathbf{t}_2; \quad \boldsymbol{\varphi} = (x + u)\mathbf{e}_1 + (y + v)\mathbf{e}_2 \quad (1)$$

where $\boldsymbol{\phi}$, \mathbf{t}_2 and $x_2 \in A$ are the position vector of the neutral axis in the deformed configuration, the unit vector normal to \mathbf{t}_1 that lies in the cross-sectional plane and corresponding cross-sectional parameter, respectively. The unit vector \mathbf{t}_1 in the deformed configuration is orthogonal to the beam cross-section, but not necessarily tangent to the neutral axis.

By considering the shear flexibility of the beam, we can state that the orthogonality condition no longer holds in the deformed configuration

$$\mathbf{t}_2^T \cdot \boldsymbol{\varphi}' \neq 0; \quad \boldsymbol{\varphi}' = \left(1 + \frac{\partial u}{\partial s}\right)\mathbf{e}_1 + \frac{\partial v}{\partial s}\mathbf{e}_2 \quad (2)$$

We recall the special orthogonal transformation $SO(2)$, which is the necessary ingredient of the finite strain theory, and it is responsible for planar rotation or mapping between the basis \mathbf{e}_i to \mathbf{t}_i , ($i = 1, 2$). Such transformation is given by the rotation tensor $\boldsymbol{\Lambda}(\psi): \mathbf{e}_i \rightarrow \mathbf{t}_i$ with the components

$$\boldsymbol{\Lambda} = \mathbf{t}_i \otimes \mathbf{e}_i; \quad \Rightarrow \quad \mathbf{t}_i = \boldsymbol{\Lambda} \mathbf{e}_i; \quad \boldsymbol{\Lambda} = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \quad (3)$$

2.1 Beam kinematics and weak form of balance equations in statics

The spatial strain measure for shear flexible beams with curvature is defined as the difference between beam neutral axis tangent $\boldsymbol{\varphi}'$ and cross-section unit normal \mathbf{t}_1 (see Fig. 2), which is defined as

$$\boldsymbol{\varepsilon} = \begin{Bmatrix} \boldsymbol{\epsilon} \\ \boldsymbol{\kappa} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{\varphi}' - \mathbf{t}_1 \\ \boldsymbol{\varphi}' \mathbf{e}_3 \end{Bmatrix} \quad (4)$$

where $\boldsymbol{\varepsilon}$ represents axial, and shear strains and $\boldsymbol{\kappa}$ contains the curvature strain, respectively. The same strain measure can be formulated in the material form by using the pull-back operation (Ibrahimbegovic *et al.* 2023), along with rotation tensor $\boldsymbol{\Lambda}$ in Eq. (3)

$$\begin{aligned} \mathbf{E} &= \boldsymbol{\Lambda}^T \boldsymbol{\varepsilon}; & \mathbf{E} &= \begin{Bmatrix} \left(1 + \frac{\partial u}{\partial s}\right) \cos \psi + \frac{\partial v}{\partial s} \sin \psi - 1 \\ -\left(1 + \frac{\partial u}{\partial s}\right) \sin \psi + \frac{\partial v}{\partial s} \cos \psi \end{Bmatrix} \\ \mathbf{K} &= \boldsymbol{\kappa}; & & \end{aligned} \quad (5)$$

Once we obtain the strain measures in Eqs. (4) and (5), we can state the stress resultant forces \mathbf{N} and moment \mathbf{M} in the material description by employing the simplest constitutive law of linear elasticity

$$\begin{aligned} \mathbf{N} &= \mathbb{C}_n \mathbf{E}; & \mathbb{C}_n &= \text{diag}(EA, GA) \\ \mathbf{M} &= \mathbb{C}_m \mathbf{K}; & \mathbb{C}_m &= EI \end{aligned} \quad (6)$$

where the strains \mathbf{E} and \mathbf{K} are given in Eq. (5). The spatial force \mathbf{n} and moment \mathbf{m} resultants can be transformed by push-forward operation of the material objects in Eq. (6)

$$\begin{aligned} \mathbf{n} &= \boldsymbol{\Lambda} \mathbf{N}; & \boldsymbol{\epsilon} &= \boldsymbol{\Lambda} \mathbf{E}; \\ \mathbf{m} &= \mathbf{M}; & \boldsymbol{\kappa} &= \mathbf{K}; \end{aligned} \quad (7)$$

For clarification, the stress resultants in Eq. (6) are obtained through the integration of the Biot-type stress $\mathbf{T} \mathbf{e}_1 = \boldsymbol{\Lambda}^T \mathbf{P} \mathbf{e}_1$ (where \mathbf{P} is the first Piola-Kirchhoff stress) over the beam cross-section (e.g., see Reissner 1972) with

$$N_1 = \int_A T^{11} dA; \quad N_2 = \int_A T^{21} dA; \quad M = - \int_A x_2 T^{11} dA \quad (8)$$

where $\mathbf{N} = (N_1, N_2)^T$ and $\mathbf{M} = M \mathbf{e}_3$. The work conjugate couple of the stress resultants in Eq. (8) and strain measures in Eq. (5) can be given in an alternative form proposed by Reissner as $\mathbf{r} = (N, V, M)^T = (N_1, N_2, M)^T$ and $\boldsymbol{\Sigma} = (\Sigma, \Gamma, K)^T = (E_1, E_2, K)^T$, respectively. Thus, one can write the Biot strain measure as

$$\mathbf{H} = \begin{bmatrix} \Sigma - x_2 K & 0 \\ \Gamma & 0 \end{bmatrix} \quad (9)$$

The internal strain energy of the geometrically exact beam can be stated with respect to different work conjugate couples

$$\Pi^{int} = \frac{1}{2} \int_L (\boldsymbol{\epsilon}^T \cdot \mathbf{n} + \boldsymbol{\kappa}^T \cdot \mathbf{m}) ds = \frac{1}{2} \int_L (\mathbf{E}^T \cdot \mathbf{N} + \mathbf{K}^T \cdot \mathbf{M}) ds = \frac{1}{2} \int_L \boldsymbol{\Sigma}^T \cdot \mathbf{r} ds \quad (10)$$

By finding the first variation of the strain energy in Eq. (10) with respect to the virtual displacement and rotation, one can obtain the internal virtual work in following form

$$G^{int}(\boldsymbol{\varphi}, \psi; \delta\boldsymbol{\varphi}, \delta\psi) := \int_L \int_A \delta\mathbf{F} : \mathbf{P} dA ds = \int_L \int_A \delta\mathbf{H} : \mathbf{T} dA ds = \int_L \delta\boldsymbol{\Sigma}^T \cdot \mathbf{r} ds \quad (11)$$

where $\delta\mathbf{F}$, $\delta\mathbf{H}$ and $\delta\boldsymbol{\Sigma}$ are variations of the deformation gradient, the Biot strain tensor and the Reissner strain measure [4], respectively. These variations of the strains in Eq. (4) are obtained by push-forward (e.g. see [1]) sequence of the material strains in Eq. (5)

$$\begin{aligned} \delta\boldsymbol{\epsilon} &= \boldsymbol{\Lambda} \frac{d}{dt} [\boldsymbol{\Lambda}_t^T (\boldsymbol{\varphi}' - \mathbf{t}_{1,t})]_{t=0} = \delta\boldsymbol{\varphi}' - \delta\psi \mathbf{W} \boldsymbol{\varphi}' \\ \delta\boldsymbol{\kappa} &= \mathbf{I} \frac{d}{dt} [\mathbf{I}\psi]_{t=0} = \delta\psi' \mathbf{e}_3 \end{aligned} \quad (12)$$

2.2 Dynamics framework

To simulate the beam response in real-time control, we need to extend the Reissner beam model to the dynamics framework. The inertial velocity of the material point in Eq.(1) can be obtained by taking the first derivative with respect to time

$$\dot{\boldsymbol{\phi}} = \dot{\boldsymbol{\varphi}} + x_2 \dot{\mathbf{t}}_2; \quad \dot{\boldsymbol{\varphi}} = \dot{u} \mathbf{e}_1 + \dot{v} \mathbf{e}_2 \quad (13)$$

where the notation superposed dot indicates the time derivative $(\dot{\cdot}) = \partial(\cdot)/\partial t$. The material velocity of points located on the neutral axis is given by $\dot{\boldsymbol{\varphi}}$, while the change in direction of the unit base vectors \mathbf{t}_i corresponds to the material velocity of the rotating frame. In that sense, we first need to find the time derivative of the orthogonal tensor $\boldsymbol{\Lambda}$ as

$$\dot{\boldsymbol{\Lambda}} = \dot{\psi} \mathbf{W} \boldsymbol{\Lambda}; \quad \mathbf{W} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (14)$$

where \mathbf{W} is a skew-symmetric matrix. By using the result in Eq. (14), one can further find the time derivative of unit vectors \mathbf{t}_i in Eq. (13) as follows

$$\dot{\mathbf{t}}_i = \dot{\psi} \mathbf{W} \boldsymbol{\Lambda} \mathbf{e}_i = \dot{\psi} \mathbf{W} \mathbf{t}_i; \quad \Rightarrow \quad \begin{Bmatrix} \dot{\mathbf{t}}_1 \\ \dot{\mathbf{t}}_2 \end{Bmatrix} = \dot{\psi} \begin{Bmatrix} \mathbf{t}_2 \\ -\mathbf{t}_1 \end{Bmatrix} \quad (15)$$

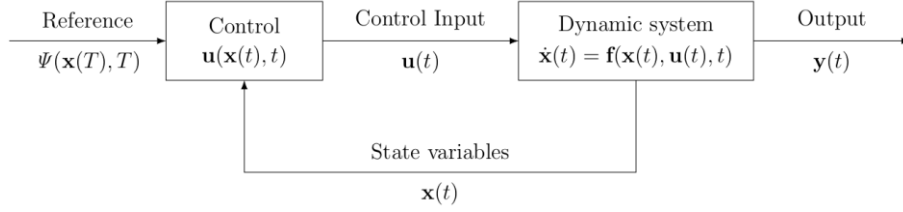


Fig. 3 Modern control sequence with state feedback (closed) loop signal

With the help of the result in Eqs. (13) and (15), the kinetic energy of the beam can be written as a quadratic form

$$K = \frac{1}{2} \int_L \int_A \rho \dot{\boldsymbol{\phi}}^T \cdot \dot{\boldsymbol{\phi}} dA ds = \frac{1}{2} \int_L \left(A_\rho \dot{\boldsymbol{\phi}}^T \cdot \dot{\boldsymbol{\phi}} + J_\rho \dot{\psi}^2 \right) ds \quad (16)$$

where $A_\rho = \int_A \rho dA$ and $J_\rho = \int_A \rho x_2^2 dA$ are mass of the beam per unit length and cross-sectional moment of inertia, respectively. The total system energy is constructed as the Hamiltonian functional (as a quadratic form of the generalized coordinates and its time derivatives), which involves the sum of kinetic and potential energy given in Eqs. (10) and (16)

$$H(\boldsymbol{\varphi}, \psi; \dot{\boldsymbol{\varphi}}, \dot{\psi}) = K(\dot{\boldsymbol{\varphi}}, \dot{\psi}) + \Pi(\boldsymbol{\varphi}, \psi) \quad (17)$$

The dynamic equilibrium can be obtained by finding the variation of the energy functional in Eq. (17) with respect to virtual displacements and rotation. Next, by applying Hamilton's principle we can state the weak form of equilibrium equations in spatial description as

$$\begin{aligned} G(\boldsymbol{\varphi}, \psi; \delta\boldsymbol{\varphi}, \delta\psi) := & \int_L \left(\delta\boldsymbol{\varphi}^T \cdot A_\rho \dot{\boldsymbol{\varphi}} + \delta\psi \cdot J_\rho \ddot{\psi} \right) ds \\ & + \int_L \left((\delta\boldsymbol{\varphi}' - \delta\psi \mathbf{W}\boldsymbol{\varphi}')^T \cdot \mathbf{n} + \delta\psi' \mathbf{e}_3^T \cdot \mathbf{m} \right) ds - G^{ext} = 0 \end{aligned} \quad (18)$$

where the static term corresponds to the one already stated in Eq. (11).

3. Optimal nonlinear control with state feedback loop

In this section, we consider a development of the modern control algorithms (see Fig. 3), which we seek to use for the presented Reissner's beam model. Such a procedure includes dynamic optimization with respect to cost-to-go function (also referred to as index performance or payoff function) and imposed boundary conditions, with the goal of achieving that the system behaves in an optimal manner. The key ingredients of the optimal control problem can be formulated as follows: i) a mathematical model, which is given in Eq. (18), represented by a set of non-linear differential equations, ii) specification of the cost-to-go function, from where we derive optimal control law by finding extremum, iii) imposed boundary conditions on the states and/or controls.

We note that the optimization procedure for this kind of problem is standard, but the complexity can vary with respect to the type of control we wish to implement. Here, we are interested in delivering the optimal control for the MIMO (multiple-inputs-multiple-outputs) system based on the state variable representation, from which we derive a set of first-order

differential equations. Moreover, the dynamic optimization task involves the conditions and system variables that change in time, which is a more challenging task than controlling a system under steady-state conditions, (see e.g., Trelat 2005, Kirk 2005, Lewis 2012). We further trace the necessary steps in nonlinear control design in order to cast the optimal control law for a finite time linear quadratic regulator.

3.1 State-space form

In control theory, it is common to recast the quotations of motion in a so-called state-space form in order to simplify the analysis, resulting with a set of first-order differential equations that describe the system response in dynamics. We start by modifying the linearized form of dynamic equilibrium presented in Eq. (18), by selecting the first vector of state variables as $\mathbf{x}_1(t) = (\boldsymbol{\varphi}(t), \boldsymbol{\psi}(t))^T$

$$\mathbf{M}\ddot{\mathbf{x}}_1(t) + \mathbf{K}(t)\mathbf{x}_1(t) = \mathbf{f}^{ext}(t) + \mathbf{u}(t) \quad (19)$$

where \mathbf{M} , \mathbf{K} and \mathbf{u} are mass matrix, stiffness matrix and control variables, respectively. By further selecting another state space vector $\mathbf{x}_2(t) = \dot{\mathbf{x}}_1(t) = (\dot{\boldsymbol{\varphi}}(t), \dot{\boldsymbol{\psi}}(t))^T$, we can obtain the state-space representation

$$\begin{cases} \dot{\mathbf{x}}_1(t) = \mathbf{x}_2(t) \\ \dot{\mathbf{x}}_2(t) = \mathbf{M}^{-1}(\mathbf{K}(t)\mathbf{x}_1(t) + \mathbf{u}(t)) \\ \mathbf{x}_1(0) = \mathbf{x}_1^0 \\ \mathbf{x}_2(0) = \mathbf{x}_2^0 \end{cases} \quad (20)$$

where $\mathbf{x}_1(0)$ and $\mathbf{x}_2(0)$ are initial conditions of the state variables. With this state-space representation, which is given in Eq. (20), we have decreased the order of the differential equations from second-order to first-order. Thus, we can apply standard numerical solution procedure that are more stable and accurate. On the other side, the number of differential equations has doubled, from n to $2n$, which implies a higher computational cost. However, this increase in computational cost is often outweighed by the benefits that the state-space form offers in terms of numerical stability, analysis and control design.

A standard state-space representation can be given in matrix form, by introducing $\dot{\mathbf{x}}(t) = (\dot{\mathbf{x}}_1(t), \dot{\mathbf{x}}_2(t))^T$ in Eq. (20) as follows

$$\underbrace{\begin{Bmatrix} \dot{\mathbf{x}}_1(t) \\ \dot{\mathbf{x}}_2(t) \end{Bmatrix}}_{\dot{\mathbf{x}}(t)} = \underbrace{\begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}(t) & 0 \end{bmatrix}}_{\mathbf{A}(t)} \underbrace{\begin{Bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{Bmatrix}}_{\mathbf{x}(t)} + \underbrace{\begin{Bmatrix} 0 \\ \mathbf{M}^{-1} \end{Bmatrix}}_{\mathbf{B}} \mathbf{u}(t) \quad (21)$$

where $\mathbf{A} \in \mathbb{R}^{m \times m}$ and $\mathbf{B} \in \mathbb{R}^{m \times n}$ are state (or system) and input matrices. The expression in Eq. (21) can be rewritten in more compact form as

$$\dot{\mathbf{x}}(t) = \underbrace{\mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)}_{\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)} \equiv \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \quad (22)$$

where $\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)$ is a state function. Since we consider time-varying systems, we note that only the state matrix $\mathbf{A}(t)$ changes in time, while the input matrix \mathbf{B} remains constant. However, the following development should hold for both cases.

Table 1 The cost-to-go function can be constructed to support different problems in optimal control

1.	Minimum time	$J = \int_{t_0}^{t_f} 1 d\tau = t_f - t_0 = t^*$
2.	Minimum control energy	$J = \int_{t_0}^{t_f} \mathbf{u}^T(t) \cdot \mathbf{R}\mathbf{u}(t) dt$
3.	Terminal error	$J = \mathbf{x}^T(t_f) \cdot \mathbf{S}\mathbf{x}(t_f)$
4.	Squared tracking error	$J = \int_{t_0}^{t_f} \mathbf{x}^T(t) \cdot \mathbf{Q}\mathbf{x}(t) dt$
5.	Linear quadratic regulator (LQR)	$J(\mathbf{x}(t), t) = \mathbf{x}^T(t_f) \cdot \mathbf{Q}_f \mathbf{x}(t_f) + \int_{t_0}^{t_f} (\mathbf{x}(t) \cdot \mathbf{Q}\mathbf{x}(t) + \mathbf{u}(t) \cdot \mathbf{R}\mathbf{u}(t)) dt$

3.2 The Hamilton-Jacobi-Bellman (HJB) equation

A fundamental concept in optimal control involves various forms of cost-to-go functions which can be formulated with respect to the minimization objective, encompassing factors such as control of time, control of energy, terminal error, etc (see Table 1).

Such a cost-to-go function J , represent a quantitative measure of how well a controlled dynamic system performs over a certain time, and thus one can give a general form of the function as

$$J = \Phi(\mathbf{x}(t_f), t_f) + \int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t), t) dt \quad (23)$$

where the terms $\Phi(\cdot)$ and $\int L(\cdot) dt$ are given as a quadratic form representing terminal constraint at the final time and the sum of the cost rates, respectively. The integral function $L(\cdot)$ is also referred to as Lagrangian. Furthermore, we exploit the additive interval property of integrals to state the incremental change in the cost-to-go function for the time interval $[t, t + \Delta t]$

$$J(\mathbf{x}(t), t) = \int_t^{t+\Delta t} L(\mathbf{x}(t), \mathbf{u}(t), t) d\tau + J(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t) \quad (24)$$

where τ is a dummy variable. We note that the cost-to-go function in Eq. (24) is updated recursively, where $\mathbf{x} + \Delta\mathbf{x}$ is the state at time $t + \Delta t$ which is computed as a system response for $\mathbf{x}(t)$ and $\mathbf{u}(t)$. Among all the potential costs to go from time t to t_f , given by Eq. (24), we select only the candidates $J^*(\mathbf{x}(t), t)$ that optimize the function $J(\mathbf{x}(t), t)$ within the interval $[t + \Delta t, t_f]$. Assuming that the optimal cost $J^*(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t)$ and control law $\mathbf{u}(t + \Delta t)$ are known for all $\mathbf{x} + \Delta\mathbf{x}$, we can therefore compute optimal control law $\mathbf{u}(t)$ for all $\mathbf{x}(t)$ on the time interval $[t, t + \Delta t]$ through the corresponding minimization procedure

$$J^*(\mathbf{x}(t), t) = \min_{\mathbf{u}(\tau) \in \mathbb{R}^n} \left\{ \int_t^{t+\Delta t} L(\mathbf{x}(t), \mathbf{u}(t), t) d\tau + J^*(\mathbf{x}(t + \Delta t), t + \Delta t) \right\} \quad (25)$$

which casts the principle of optimality on $\tau \in [t, t + \Delta t]$. By finding the Taylor series expansion of the right hand side of Eq. (25) around the point $(\mathbf{x}(t), t)$, we get

$$\min_{\mathbf{u}(\tau) \in \mathbb{R}^n} \left\{ L(\mathbf{x}(t), \mathbf{u}(t), t) \Delta t + J^*(\mathbf{x}(t), t) + \frac{\partial J^*(\mathbf{x}(t), t)}{\partial t} \Delta t + \frac{\partial J^*(\mathbf{x}(t), t)}{\partial \mathbf{x}} \dot{\mathbf{x}}(t) \Delta t \right\} \quad (26)$$

Through a minor manipulation of Eq. (27) and by taking the limit as $\Delta t \rightarrow 0$, we can derive the partial differential equation that governs the optimal cost $J^*(\mathbf{x}(t), t)$

$$-\frac{\partial J^*(\mathbf{x}(t), t)}{\partial t} = \min_{\mathbf{u}(\tau) \in \mathbb{R}^n} \left\{ L(\mathbf{x}(t), \mathbf{u}(t), t) + \left(\frac{\partial J^*(\mathbf{x}(t), t)}{\partial \mathbf{x}} \right)^T \cdot \dot{\mathbf{x}}(t) \right\} \quad (27)$$

where the result is denoted as the Hamilton-Jacobi-Bellman (HJB) equation. If we denote the Lagrange multipliers as $\frac{\partial J^*(\mathbf{x}(t), t)}{\partial \mathbf{x}} = \boldsymbol{\lambda}(t)$, and the state equation as $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t)$, we can rewrite the HJB equation as follows

$$-\frac{\partial J^*(\mathbf{x}(t), t)}{\partial t} = \min_{\mathbf{u}(\tau) \in \mathbb{R}^n} \{ L(\mathbf{x}(t), \mathbf{u}(t), t) + \boldsymbol{\lambda}^T(t) \cdot \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \} \quad (28)$$

The result in Eq. (28) can be recast in a more standard form that employs the Hamiltonian $H(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t)$ in place of the Lagrangian $L(\mathbf{x}(t), \mathbf{u}(t), t)$, where the relation is given as

$$H(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t) = L(\mathbf{x}(t), \mathbf{u}(t), t) + \boldsymbol{\lambda}^T(t) \cdot \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \quad (29)$$

We can further introduce the result Eq. (29) into the HJB equation in Eq. (28), which results

$$-\frac{\partial J^*(\mathbf{x}(t), t)}{\partial t} = \min_{\mathbf{u}(\tau) \in \mathbb{R}^n} \{ H(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t) \} \quad (30)$$

The result obtained in Eq. (30) yields the optimal control law for a nonlinear system using closed-loop or state feedback, where the solution, in most cases, can only be obtained numerically.

3.3 Finite time linear quadratic regulator (LQR) algorithm

Here we present the advanced control algorithm, denoted as linear quadratic regulator (LQR), which optimizes both the control effort (minimal control energy) and the performance of the system with respect to time. In order to determine the optimal control law $\mathbf{u}^*(t)$, we first define the cost-to-go function $J(\mathbf{x}(t), t)$ as a quadratic form (see Table 1)

$$J(\mathbf{x}(t), t) = \frac{1}{2} \mathbf{x}^T(t_f) \cdot \mathbf{Q}_f \mathbf{x}(t_f) + \frac{1}{2} \int_{t_0}^{t_f} (\mathbf{x}(t) \cdot \mathbf{Q} \mathbf{x}(t) + \mathbf{u}(t) \cdot \mathbf{R} \mathbf{u}(t)) dt \quad (31)$$

where $\mathbf{Q} = \mathbf{Q}^T > 0$, \mathbf{Q}_f and $\mathbf{R} = \mathbf{R}^T > 0$ are matrices related to the penalty of state error, terminal error and control effort. According to Eq. (29) we can define the Hamiltonian as follows

$$H(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), t) = \frac{1}{2} (\mathbf{x}^T(t) \cdot \mathbf{Q} \mathbf{x}(t) + \mathbf{u}^T(t) \mathbf{R} \cdot \mathbf{u}(t)) + \boldsymbol{\lambda}^T(t) \cdot (\mathbf{A}(t) \mathbf{x}(t) + \mathbf{B} \mathbf{u}(t)) \quad (32)$$

To perform optimization of function in Eq. (31), we employ the HJB principle in Eq. (30), where the minimization task reads

$$-\frac{\partial J^*(\mathbf{x}(t), t)}{\partial t} = \min_{\mathbf{u}(\tau) \in \mathbb{R}^n} \left\{ \frac{1}{2} (\mathbf{x}^T \cdot \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \cdot \mathbf{u}) + \boldsymbol{\lambda}^T \cdot (\mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u}) \right\} \quad (33)$$

Now, we can obtain stationary condition, state and co-state equations by finding the first derivative of the Hamiltonian H with respect to control variable \mathbf{u} , state \mathbf{x} and Lagrange multipliers $\boldsymbol{\lambda}$, respectively

$$\begin{aligned} 0 &= \left(\frac{\partial H}{\partial \mathbf{u}} \right) = \mathbf{R} \mathbf{u} + \mathbf{B}^T \boldsymbol{\lambda}; & \Rightarrow & \quad \mathbf{u}^*(\boldsymbol{\lambda}(t), t) = -\mathbf{R}^{-1} \mathbf{B}^T \boldsymbol{\lambda} \\ -\dot{\boldsymbol{\lambda}} &= \left(\frac{\partial H}{\partial \mathbf{x}} \right) = \mathbf{Q} \mathbf{x} + \mathbf{A}^T \boldsymbol{\lambda} \\ \dot{\mathbf{x}} &= \left(\frac{\partial H}{\partial \boldsymbol{\lambda}} \right) = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u}^* \end{aligned} \quad (34)$$

where we obtain the optimal control law $\mathbf{u}^*(\boldsymbol{\lambda}(t), t)$ which minimizes the functional in Eq. (33). The result in Eq. (34) can be rewritten in matrix form as

$$\begin{Bmatrix} \dot{\mathbf{x}} \\ \dot{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{Bmatrix} \quad (35)$$

To provide the closed-loop or state feedback control, we a priori need to redefine the optimal control $\mathbf{u}^*(\boldsymbol{\lambda}(t), t)$ in terms of the state variable $\mathbf{x}(t)$. If we assume that the optimal cost-to-go function has a following solution form $J^*(\mathbf{x}(t), t) = \frac{1}{2}\mathbf{x}^T \cdot \mathbf{S}(t)\mathbf{x}$, where $\mathbf{S}(t) = \mathbf{S}^T(t) > 0$, then the Lagrange multipliers and time variation of the cost-to-go function can be given as

$$\frac{\partial J^*}{\partial \mathbf{x}} = \mathbf{S}(t)\mathbf{x} \equiv \boldsymbol{\lambda}; \quad \frac{\partial J^*}{\partial t} = \frac{1}{2}\mathbf{x}^T \cdot \dot{\mathbf{S}}(t)\mathbf{x}; \quad (36)$$

Now, we can rewrite the optimal control law $\mathbf{u}^*(\mathbf{x}, t)$ in Eq. (34) as

$$\mathbf{u}^*(\mathbf{x}, t) = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t)\mathbf{x} \quad (37)$$

where $-\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t)$ is denoted as Kalman gain. By combining the results in Eqs. (35), (36) and (37), one can cast the Riccati differential equation in matrix form as follows

$$-\dot{\mathbf{S}}(t) = \mathbf{Q} - \mathbf{S}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t) + \mathbf{S}(t)\mathbf{A} + \mathbf{A}^T\mathbf{S}(t) \quad (38)$$

where the final state condition is given $\mathbf{S}(t_f) = \mathbf{Q}_f$. For clarification, the solution procedure of the LQR problem reduces to numerical integration of the Riccati differential equation Eq. (38) backwards in time. Once the solution $\mathbf{S}(t)$ is obtained, we can compute the optimal feedback gain $\mathbf{u}^*(t)$ at the time t (see Eq. (37)), and then update the state equation

$$\dot{\mathbf{x}}(t) = (\mathbf{A}(t) - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t))\mathbf{x}(t) \quad (39)$$

Thus, we can proceed with the computation of the system response forward in time for the given control signal in Eq. (37).

4. Numerical implementation

In this section, we present the numerical implementation of the 2D beam with nonlinear kinematics. Since the linearization of the weak form was elaborated in [2,3], we will omit a detailed explanation. Furthermore, we show the essential steps involved in the development of discrete computation algorithm of optimal control law.

4.1 Discrete approximation for geometrically exact beam with 2-node element

Here, we briefly discuss the choice of the shape functions. The piece-wise linear shape functions are chosen for discretization of the displacement and rotation fields that correspond to a 2-node element. We consider a domain $\Omega \in \mathbb{R}^2$ divided into a finite number of elements with length Ω^e . It holds that $\Omega = \cup_{\Omega^e \in I} \Omega^e$, where I is a set with all elements. To construct the real and virtual fields, we use the same isoparametric shape functions defined as follows

$$\begin{aligned} \varphi^h(\xi) &= \sum_{a=1}^2 N_a^e(\xi)\varphi_a^e; & N_a^e(\xi) &= \frac{1}{2}(1 + \xi_a\xi), & \xi_a &= \begin{cases} -1, & a = 1 \\ 1, & a = 2 \end{cases}, & \xi &\in [-1, 1] \end{aligned} \quad (40)$$

where $N_a^e(\xi)$ is the linear piece-wise shape function. The corresponding first derivatives of displacement and rotation fields, which are necessary ingredient to compute the strains, can easily be computed with

$$\begin{aligned}\varphi'^h(\xi) &= \frac{d\varphi^h(\xi)}{d\xi} = \sum_{a=1}^2 B_a^e(\xi)\varphi_a^e; & B_a^e(\xi) &= \frac{1}{j(\xi)} \frac{dN_a^e(\xi)}{d\xi} = \frac{(-1)^a}{L^e} \\ \psi'^h(\xi) &= \frac{d\psi^h(\xi)}{d\xi} = \sum_{a=1}^2 B_a^e(\xi)\psi_a^e;\end{aligned}\quad (44)$$

where the transformation to isoparametric coordinates employs the Jacobian, which is given by $j(\xi) = \frac{d\xi}{ds} = \frac{L^e}{2}$. The same choice of shape functions can be used for discretization of real and virtual fields of velocity and acceleration.

4.2 Linearized weak form of dynamic equilibrium

Such a highly nonlinear dynamic problem we need to linearize, by finding the directional derivative with respect to incremental change in displacements and rotation. Taking into account that equilibrium is attained for the previous time step, we can write the linearized internal work of weak equilibrium equations in Eq. (18) as

$$\begin{aligned}Lin[G(\varphi, \psi)] &= \frac{d}{dt}[G(\varphi + t\Delta\varphi, \psi + t\Delta\psi)]_{t=0} \\ &= \int_L \left(\underbrace{\delta\varepsilon^T \cdot \Lambda C_n \Lambda^T \Delta\varepsilon + \delta\kappa^T \cdot \Lambda C_m \Lambda^T \Delta\kappa}_{\text{material}} + \underbrace{\Delta\delta\varepsilon^T \cdot \mathbf{n} + \Delta\delta\kappa^T \cdot \mathbf{m}}_{\text{geometric}} \right) ds \\ &\quad + \int_L \underbrace{(\delta\varphi^T \cdot A_\rho \Delta\ddot{\varphi} + \delta\psi J_\rho \Delta\ddot{\psi})}_{\text{inertia}} ds\end{aligned}\quad (44)$$

where $\Delta\varphi$ and $\Delta\psi$ are incremental displacements and rotation, respectively. It can be noticed that in Eq. (46) we get material \mathbf{K}_m and geometric \mathbf{K}_g part of the tangent stiffness as a product of consistent linearization, such that $\mathbf{K} = \mathbf{K}_m + \mathbf{K}_g$. Furthermore, we can write explicit form of the material part of internal virtual work

$$\begin{Bmatrix} \delta\varphi \\ \delta\psi \end{Bmatrix}^T \cdot \mathbf{K}_m \begin{Bmatrix} \Delta\varphi \\ \Delta\psi \end{Bmatrix} = \int_L \begin{Bmatrix} \delta\varphi' \\ \delta\psi \\ \delta\psi' \end{Bmatrix}^T \cdot \begin{bmatrix} \mathbf{I} & 0 \\ -\mathbf{W}\varphi' & 0 \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Lambda C_n \Lambda^T & 0 \\ 0 & C_m \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 \\ -\mathbf{W}\varphi' & 0 \\ 0 & \mathbf{I} \end{bmatrix}^T \begin{Bmatrix} \Delta\varphi' \\ \Delta\psi \\ \Delta\psi' \end{Bmatrix} ds \quad (45)$$

as well as the linearized geometric part of internal virtual work \mathbf{K}_g

$$\begin{Bmatrix} \delta\varphi \\ \delta\psi \end{Bmatrix}^T \cdot \mathbf{K}_g \begin{Bmatrix} \Delta\varphi \\ \Delta\psi \end{Bmatrix} = \int_L \begin{Bmatrix} \delta\varphi' \\ \delta\psi \end{Bmatrix}^T \cdot \begin{bmatrix} 0 & \mathbf{W}\mathbf{n} \\ -\mathbf{W}\mathbf{n}^T & -\mathbf{n}^T \cdot \varphi' \end{bmatrix} \begin{Bmatrix} \Delta\varphi' \\ \Delta\psi \end{Bmatrix} ds \quad (46)$$

The linearized inertia term is given as

$$\begin{Bmatrix} \delta\varphi \\ \delta\psi \end{Bmatrix}^T \cdot \mathbf{M} \begin{Bmatrix} \Delta\ddot{\varphi} \\ \Delta\ddot{\psi} \end{Bmatrix} = \int_L \begin{Bmatrix} \delta\varphi \\ \delta\psi \end{Bmatrix}^T \cdot \begin{bmatrix} A_\rho \mathbf{I} & 0 \\ 0 & J_\rho \end{bmatrix} \begin{Bmatrix} \Delta\ddot{\varphi} \\ \Delta\ddot{\psi} \end{Bmatrix} ds \quad (47)$$

thus, the linearized equilibrium equations can be written in a simplified form

$$\mathbf{M} \begin{Bmatrix} \Delta\ddot{\varphi} \\ \Delta\ddot{\psi} \end{Bmatrix} + \mathbf{K} \begin{Bmatrix} \Delta\varphi \\ \Delta\psi \end{Bmatrix} = - \underbrace{(\mathbf{f}^{ext} - \mathbf{f}^{int})}_{\text{residual} \approx 0} \quad (48)$$

4.3 Discrete time computation algorithm of optimal control

In this section, we provide the solution procedure for the LQR problem in a time discrete form. The state differential equation in Eq. (34) can be numerically integrated by explicit Euler forward scheme, where the state variables at time step $n + 1$ are computed

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{x}_n + \Delta t \cdot (\mathbf{A}_n \mathbf{x}_n + \mathbf{B}_n \mathbf{u}_n) \\ &= \underbrace{(\mathbf{I} + \Delta \mathbf{A}_n)}_{\bar{\mathbf{A}}} \mathbf{x}_n + \underbrace{\Delta t \mathbf{B}_n}_{\bar{\mathbf{B}}} \mathbf{u}_n \\ &= \bar{\mathbf{A}} \mathbf{x}_n + \bar{\mathbf{B}} \mathbf{u}_n \end{aligned} \quad (49)$$

With the help of the expression in Eq. (24), we can rewrite the cost-to-go function in discrete form for arbitrary time instant $\tau \in [0, n + 1]$ as

$$J_\tau(\mathbf{x}_\tau, \tau) = \min_{\mathbf{u}} \left\{ \frac{1}{2} \sum_{k=\tau}^n (\mathbf{x}_k^T \cdot \mathbf{Q} \mathbf{x}_k + \mathbf{u}_k^T \cdot \mathbf{R} \mathbf{u}_k) + J_{n+1}(\mathbf{x}_{n+1}, n + 1) \right\} \quad (50)$$

where the computation of $\tau < n + 1$ is performed recursively. For clarification, if we consider two adjacent time instantes n and $n + 1$, the cost-to-go function is known for $n + 1$ and it has a form

$$J_{n+1}(\mathbf{x}_{n+1}, n + 1) = \frac{1}{2} \mathbf{x}_{n+1}^T \cdot \mathbf{S}_{n+1} \mathbf{x}_{n+1} \quad (51)$$

while the cost-to-go function at time n needs to be computed backwards in time. By using result in Eqs. (51) and (53), one can express the cost-to-go function at time n with respect to variables in the same time instant as follows

$$\begin{aligned} J_n(\mathbf{x}_n, n) &= \frac{1}{2} (\mathbf{x}_n^T \cdot \mathbf{Q} \mathbf{x}_n + \mathbf{u}_n^T \cdot \mathbf{R} \mathbf{u}_n + \mathbf{x}_{n+1}^T \cdot \mathbf{S}_{n+1} \mathbf{x}_{n+1}) \\ &= \frac{1}{2} (\mathbf{x}_n^T \cdot \mathbf{Q} \mathbf{x}_n + \mathbf{u}_n^T \cdot \mathbf{R} \mathbf{u}_n + (\bar{\mathbf{A}} \mathbf{x}_n + \bar{\mathbf{B}} \mathbf{x}_n)^T \cdot \mathbf{S}_{n+1} (\bar{\mathbf{A}} \mathbf{x}_n + \bar{\mathbf{B}} \mathbf{x}_n)) \end{aligned} \quad (52)$$

According to the expression Eq. (37) the optimal control law in discrete form becomes

$$\begin{aligned} 0 = \frac{\partial J_n}{\partial \mathbf{u}_n} &= \mathbf{R} \mathbf{u}_n + \bar{\mathbf{B}}^T \mathbf{S}_{n+1} (\bar{\mathbf{A}} \mathbf{x}_n + \bar{\mathbf{B}} \mathbf{x}_n) \Rightarrow \mathbf{u}_n^* = - \underbrace{(\mathbf{R} + \bar{\mathbf{B}}^T \mathbf{S}_{n+1} \bar{\mathbf{B}})^{-1} \bar{\mathbf{B}}^T \mathbf{S}_{n+1} \bar{\mathbf{A}} \mathbf{x}_n}_{\bar{\mathbf{K}}_n} \\ &= -\bar{\mathbf{K}}_n \mathbf{x}_n \end{aligned} \quad (53)$$

where $\bar{\mathbf{K}}_n$ is Kalman gain. Now, the optimal cost-to-go function from time n to $n + 1$ can be expressed in terms of Kalman gain as follows

$$J_n^*(\mathbf{x}_n, n) = \frac{1}{2} \mathbf{x}_n^T \cdot \left(\mathbf{Q} + \bar{\mathbf{K}}_n^T \mathbf{R} \bar{\mathbf{K}}_n + (\bar{\mathbf{A}} - \bar{\mathbf{B}} \bar{\mathbf{K}}_n)^T \mathbf{S}_{n+1} (\bar{\mathbf{A}} - \bar{\mathbf{B}} \bar{\mathbf{K}}_n) \right) \mathbf{x}_n \quad (54)$$

By combining results in Eqs. (51) and (54) one can obtain the Riccati differential equation (see Eq. (38)) in discrete form

$$\mathbf{S}_n = \bar{\mathbf{Q}} + \bar{\mathbf{A}}^T \mathbf{S}_{n+1} \bar{\mathbf{A}} - \bar{\mathbf{A}}^T \mathbf{S}_{n+1} \bar{\mathbf{B}} \left(\bar{\mathbf{R}} + \bar{\mathbf{B}}^T \mathbf{S}_{n+1} \bar{\mathbf{B}} \right)^{-1} \bar{\mathbf{B}}^T \mathbf{S}_{n+1} \bar{\mathbf{A}} \quad (55)$$

where the solution is computed recursively.

5. Numerical examples

In this section, we present several numerical examples that illustrate performance of the

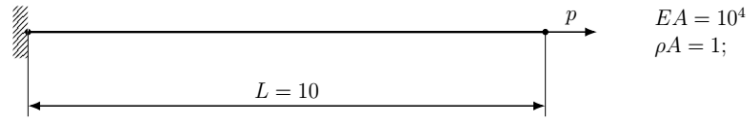


Fig. 4 Input geometry and material data

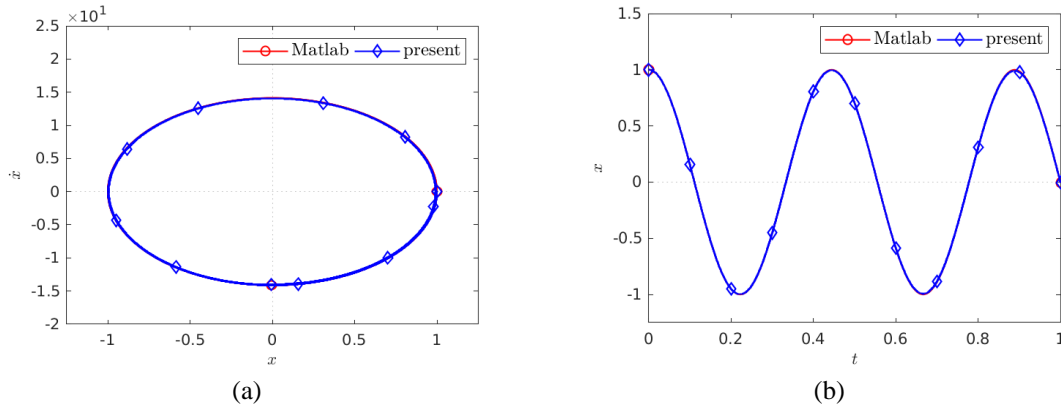


Fig. 5 Validation of the present model with Matlab functions (a) state-space response; (b) time history of displacement

proposed optimal control approach and its capabilities. The code development and computations are performed by using the research version of FEAP software, developed by Prof. R.L. Taylor at UC Berkeley (Taylor 2014).

5.1 Linear case of uniaxial vibration of Reissner's beam

In this example, we firstly consider a simple 1D problem to verify the control algorithm for linear response of Reissner's beam with an uniaxial tension load, see Fig. 4. Hence, we can verify the element control algorithm for the case where both stiffness and mass matrix are constant in time. For such a linear problem we can verify the computed response against the Matlab solvers LQR and discrete algebraic Riccati equations (*idare*). The beam of length $L = 10$ is clamped at the left end, and submitted at the right end to imposed initial displacement of magnitude $p = 1$. The rest of input data geometry and material properties are given in Fig. 4. We employed a single finite element model in order to provide the simplest possible illustration that can be compared against the spring-mass system. We note that by refining the mesh with (many) more beam elements, one has to control high frequency modes which can pollute the results and impair convergence.

We performed numerical computation by using the Euler backward time integration scheme. To avoid amplitude decay, yet called numerical damping, due to insufficient scheme accuracy, we use a very small time step of $\Delta t = 0.0001s$ with 10^4 steps for 1s. First, we verify the proposed element accuracy in free vibration without control against the system response to initial states of state-space model in Matlab, obtained by calling the functions (`sys=ss(A,B,C,D)`) and (`[[y,t]=initial(sys,x0)`). The results of these computations are given in Fig. 5., where the results are almost indistinguishable.

Next, we seek to verify the computational response of the system against two Matlab solution

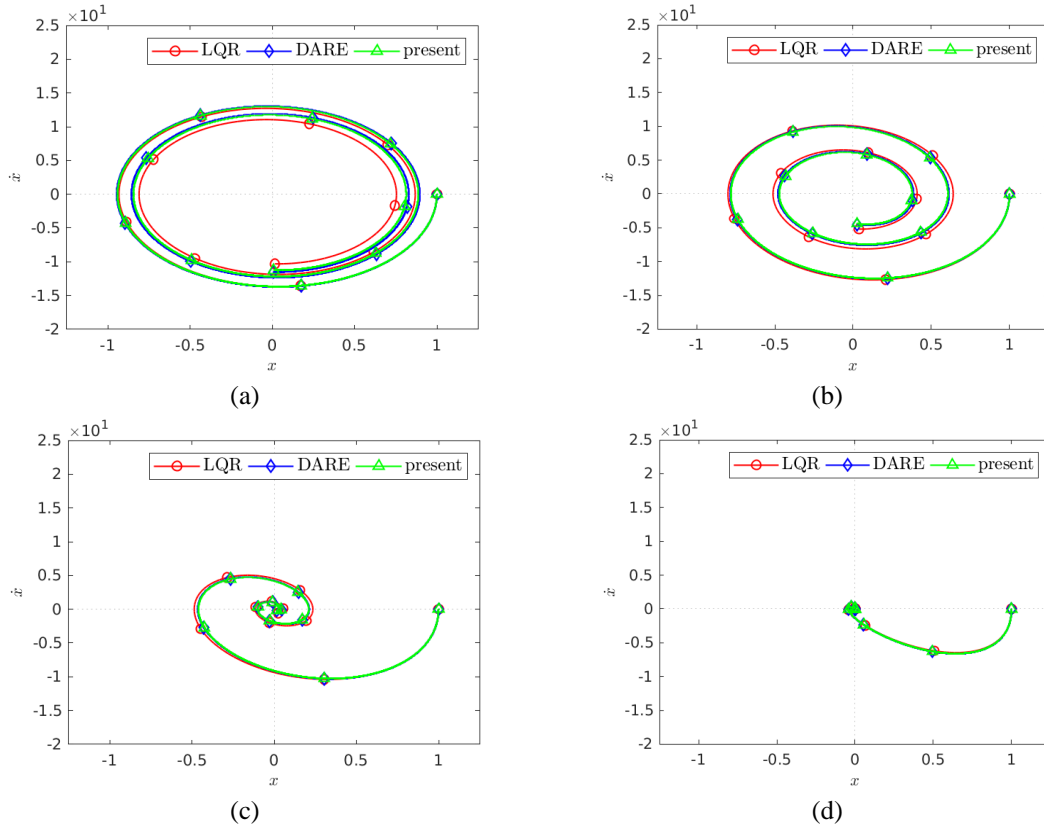


Fig. 6 State-space response (a) $Q = 10$; (b) $Q = 10^2$; (c) $Q = 10^3$, (d) $Q = 10^4$

algorithms for algebraic Riccati's equation lqr ($[K,S,P]=lqr(A,B,Q,R,N)$) and differential algebraic Riccati's equations $idare$ ($[X,L,G]=idare(A,B,Q,R)$). These algorithms provide the steady state solution and the time discrete solution of differential Riccati's equations. The computations are performed for different values of penalty state error: $Q = qI, q = (0,10,10^2,10^3,10^4)$ and $R = rI, r = 1$. In Fig. 6., we present the computer state space diagrams for several values of matrix Q .

It can be seen from the figures that with the increase of the penalty value q , the amplitudes of oscillation of the displacement and velocity decrease. In both cases, when $q = 0$, the amplitudes of oscillation of the displacement and velocity remain the same (undamped free vibrations), and at $q = 10^4$, the oscillatory character of the time change for the displacement and velocity almost completely disappears. The oscillation amplitude of the displacement decreases from the initial value of 1 to zero at a time slightly greater than 0.2 s. The velocity changes from a zero initial value and reaches a value of around -7, and at a time of 0.3 s it drops to zero. In the case of a control law that is equal to zero at $q = 0$, with an increase in the penalty value q , the control law takes on an oscillatory character and the oscillation amplitude increases with an increase of q . For value of $q = 10^4$, the oscillatory character of the control signal disappears, but there is an overshoot that is more than $6 \cdot 10^2$ at peak time of 0.09 s. Moreover, the output results match very well by comparing those obtained via $idare$ and the present model. On the other hand the lqr with constant control law does not provide the same output, unless we increase the penalty state error

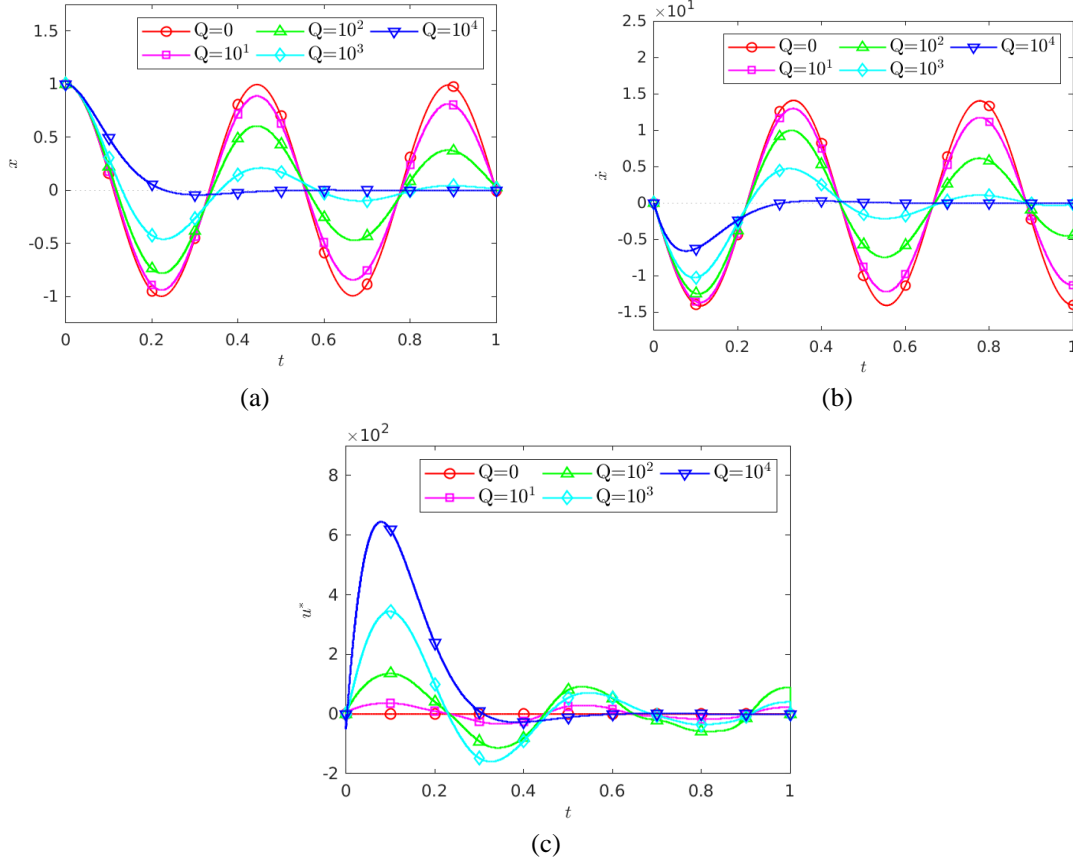


Fig. 7 Output data-the time history of (a) displacement; (b) velocity, (c) control signal

(compare Fig. 6(a) and 6(d)). The small value of the penalty state error is insufficient to achieve the optimal control goal by placing the structure in the final position within the given time.

In Fig. 7, we show the comparison of computed results for time evolution of displacement, velocity and control with the proposed model for different penalty values. We see in Fig. 7(a) that increasing value of penalty state error leads to decreasing amplitude (much analogous to increased damping) for both displacement (Fig. 7(a)) and velocity (Fig. 7(b)), whereas it results in increased values of optimal control (Fig. 7(c)).

To verify the solution algorithm of the Riccati differential equations in matrix form we present the Frobenius norm of the solution with respect again to different penalty values of Q , see Fig. 8. It can be noticed that results are practically the same in the case of *idare* and the presented model. We note that for the steady state case, the Riccati solution remains constant in agreement with Eq. (38), which has influence on convergence rate (Fig. 6(a)).

5.2 Slightly nonlinear case: Transverse vibration of Reissner's beam

In this example, we consider a slightly nonlinear case of the Reissner's beam kinematics, where we impose small initial perturbation $p = 0.1$ in the transverse direction to the beam axis. Input

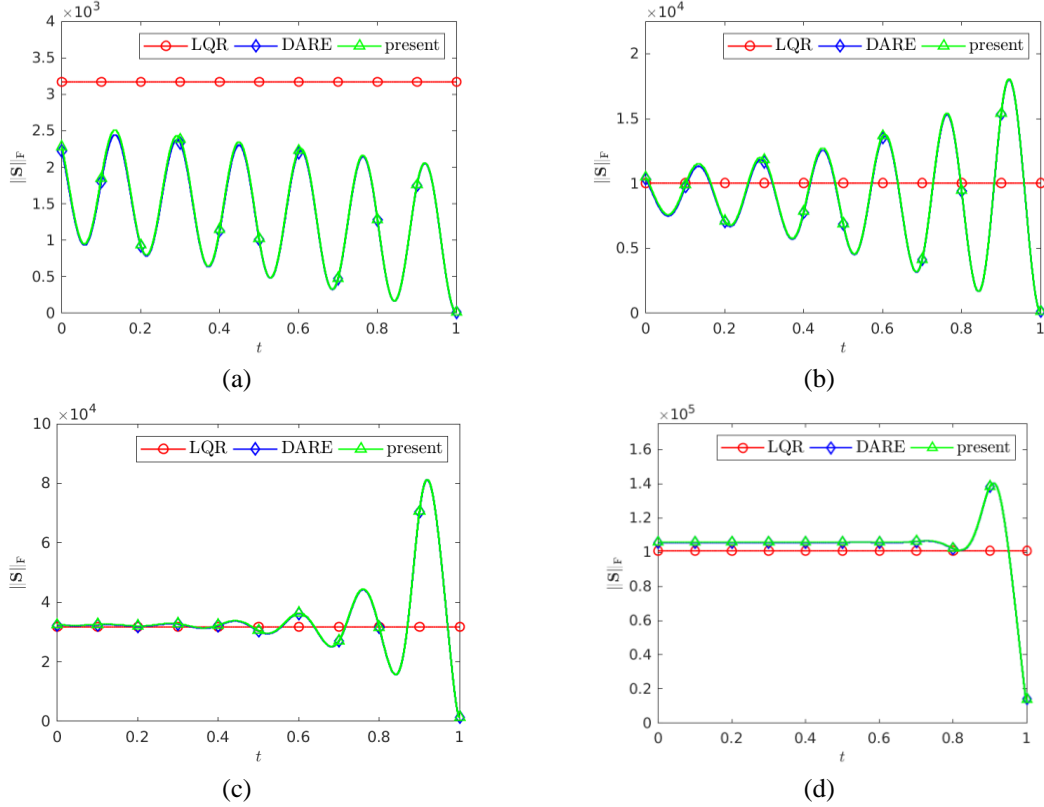


Fig. 8 The Frobenius norm for different penalty error terms (a) $q = 10$; (b) $q = 10^2$; (c) $q = 10^3$, (d) $q = 10^4$

data are the same as in the previous example, but with additional flexural stiffness $EI = 10^2$. We perform computations for 2, 4 and 8 finite elements in order to check the stability of computations due to presence of high frequency modes. Here we impose the control algorithm only in transverse direction, and we apply such control for all the elements with following penalty state error $\mathbf{Q} = q\mathbf{I}$, $q = 10^3$ and penalty control $\mathbf{R} = r\mathbf{I}$, $r = 1$. The simulations are performed with time step of $\Delta t = 0.0001s$ for total time of 5s. The output results are given in Fig. 9, for state space response and time evolutions of free end displacement, velocity and control variable.

Here we show the ability to establish the control of the system that evolves over the time, where the stiffness matrix is nearly constant in time, since the small vibration amplitudes remain close to the linear domain of small displacements and rotations. We observe that by increasing the number of elements the FE model contains more high frequency modes which makes the system more difficult to control (see Fig. 9(a)). Nonetheless, in this example the control algorithm for all discretizations successfully damped the vibrations.

5.3 Nonlinear case: Large displacements and rotations of cantilever beam

This example is chosen from (Ibrahimbegovic *et al.* 2004) controlled for quasi-static case. Here we present the computation for nonlinear dynamics that is made to transform the cantilever beam

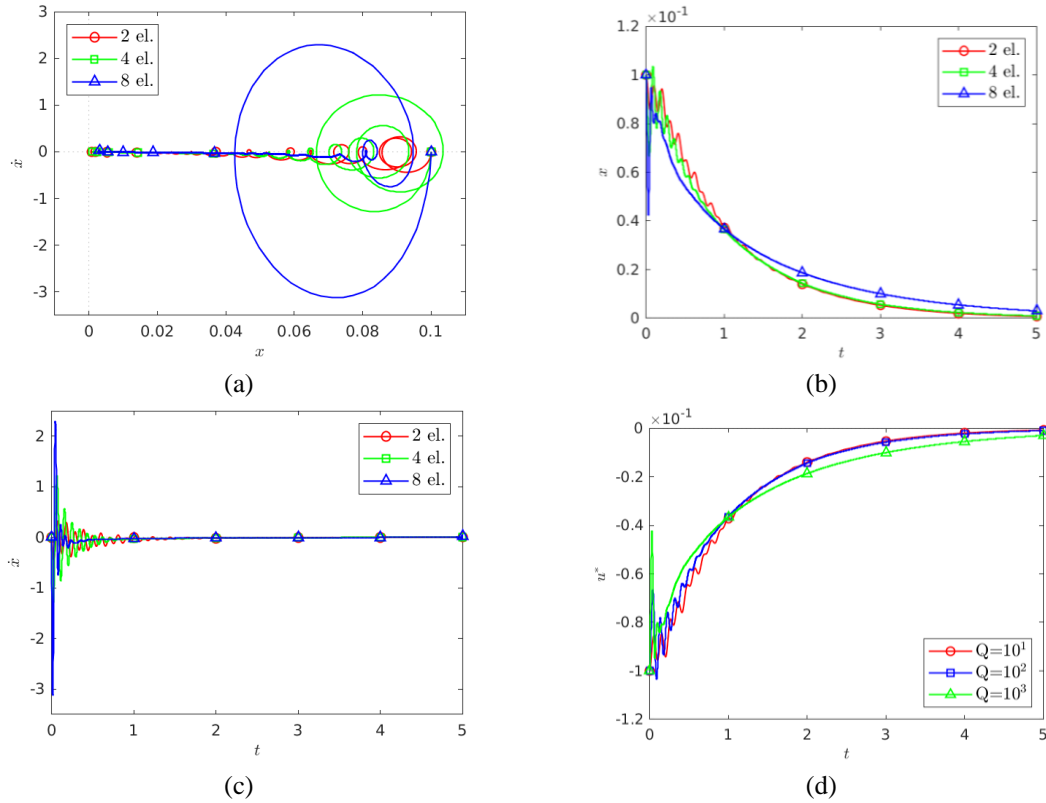


Fig. 9 Output data (a) state-space response, (b) the time history of y -displacement, (c) time history of y -velocity, (d) time history of control law

in the vertical position closest to the letter T . We present here the LQR algorithm that drives the system to non-zero setpoint, where the end states are $x = 0, y = 15.356$ and $\psi = \pi$. To simulate dynamic response we adopted an additional parameter that pertains to mass density $\rho = 1$. The geometry and the rest of material properties are given in (Ibrahimbegovic *et al.* 2004).

We simulate a simplified model with few finite elements in order to avoid activation of high frequency modes. Also, we set on the control algorithm at the node 3 (see Fig. 10) for all three degrees of freedom. The adopted values of penalty terms of state error $\mathbf{Q} = q\mathbf{I}, q = 10^3$ and control error $\mathbf{R} = r\mathbf{I}, r = 0.1$. The computation is performed with time step $\Delta t = 0.001$ s for total time of $t = 4$ s. The output configurations and its y -displacement field at 6 times instants are given in Fig. 10. The state-space diagram and the state variables time dependency are presented in Fig. 11.

The simulation results show that the control algorithm managed to drive the system to non-zero setpoint in the domain of large displacements and rotations with optimal control policy. Additionally,

one can observe in Fig. 11(a). that system converges to given values of displacement where u and v are components of displacements in x and y direction, and ψ is angle with respect to initial configuration.

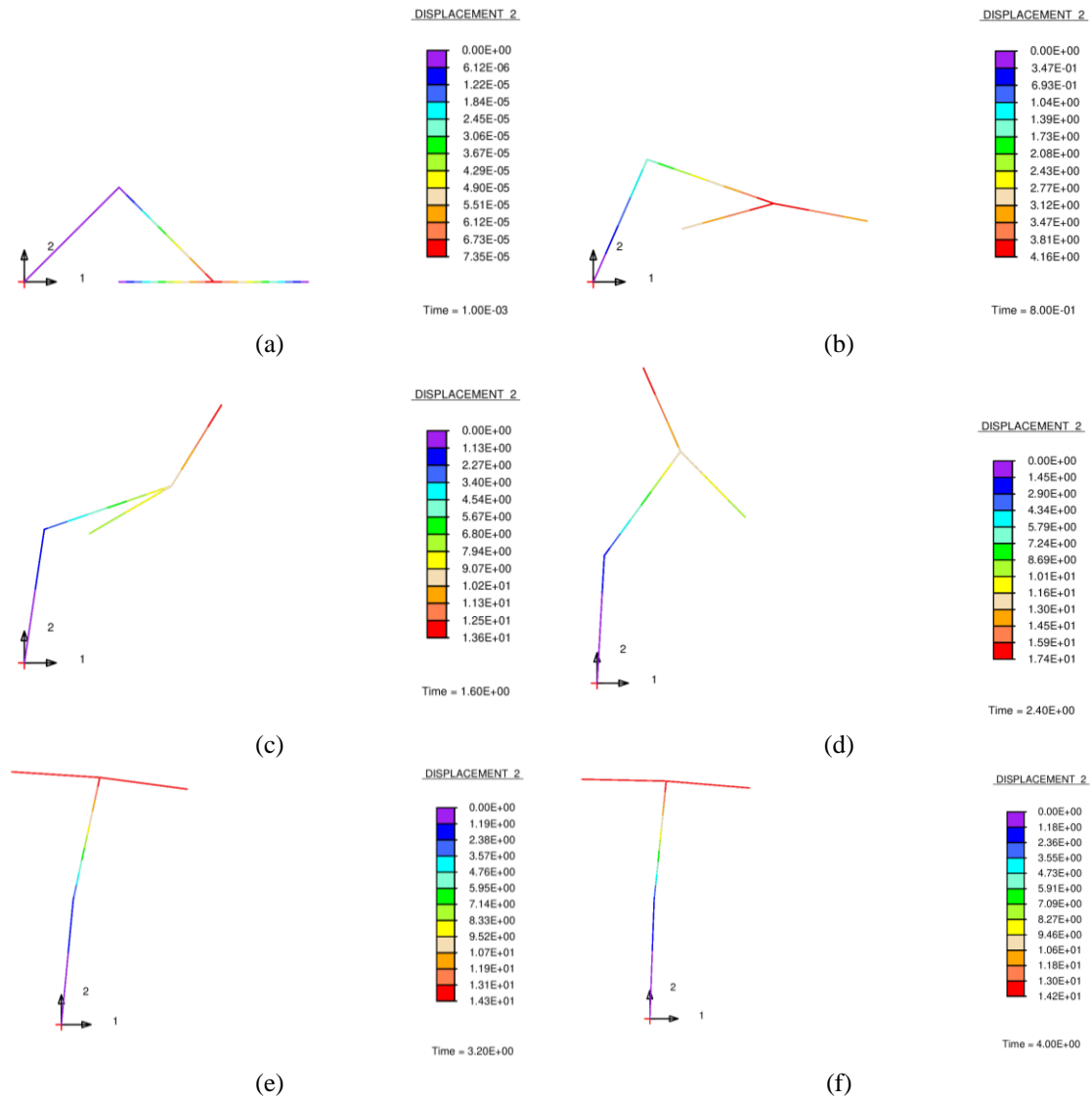


Fig. 10 Displacement field in y direction for following time instants (a) $t = 0$ s, (b) $t = 0.8$ s, (c) $t = 1.6$ s, (d) $t = 2.4$ s, (e) $t = 3.2$ s, (f) $t = 4$ s

6. Conclusions

We have explored the efficiency of the time varying advanced control algorithm applied to the geometrically exact 2D beam model. We chose the linear-quadratic regulator (LQR) algorithm to drive the system into the desired state, where the optimal control strategy considers minimization task of the cost-to-go functional with respect to the state error and control effort. Due to the nonlinear nature of kinematics of the Reissner's beam model, the beam tangent stiffness is constantly changing. Yet, the proposed optimization task reduces to the computation of the differential Riccati's equation in each time step.

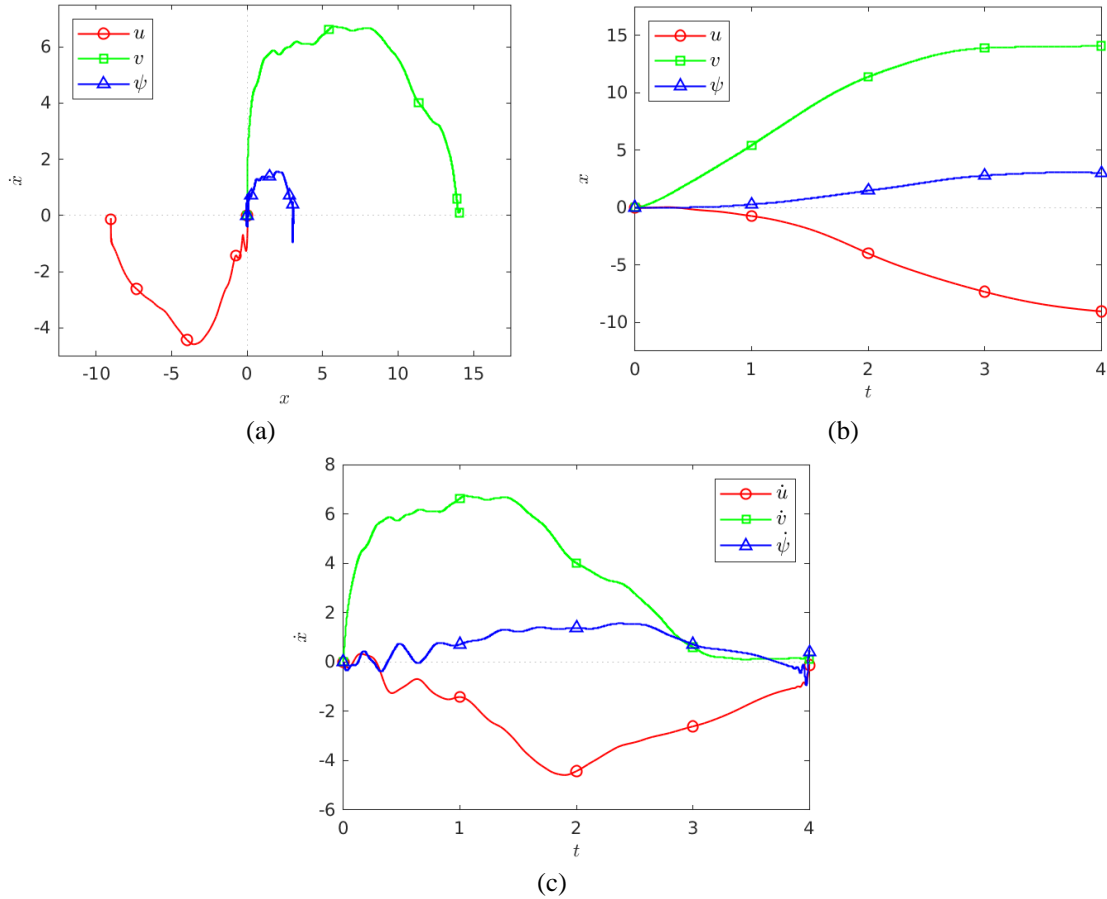


Fig. 11 Output data (a) state-space response, (b) displacement vs. time, (c) velocity vs. time

The performance of the proposed model is simulated on several numerical examples. The first of them has verified that the method applies with no difficulty to a simple 1D linear case, which is confirmed by comparing the output data with computed results of widely known algorithms *lqr* and *idare* in Matlab. The second example deals with the finite element model with a slightly nonlinear case, where one can notice that the system provided stability by converging to the initial zero setpoint. In the final example we computed the response of a nonlinear system in the domain of large displacements and rotations setting, where we report that due to occurrence of high frequency modes the system is difficult to control. Therefore, here the penalty coefficients must be selected carefully.

Acknowledgments

This work was supported by the French Research Agency, ANR - Project MS3C and by IUF-Institut Universitaire de France (for AI) and a scholarship from the French Ministry MEAE (for SLj). All the support is gratefully acknowledged.

References

- Chen, G., Rui, X., Abbas, L.K., Wang, G., Yang, F. and Zhu, W. (2018), "A novel method for the dynamic modelling of Stewart parallel mechanism", *Mech. Mach. Theory*, **126**, 397-412. <https://doi.org/10.1016/j.mechmachtheory.2018.04.024>.
- Das, S., Pan, I., Halder, K., Das, S. and Gupta, A. (2013), "LQR based improved discrete PID controller design via optimum selection of weighting matrices using fractional order integral performance index", *Appl. Math. Model.*, **37**, 4253-4268. <https://doi.org/10.1016/j.apm.2012.09.022>.
- Glad, T. and Ljung, L. (2000), *Control Theory*, CRC Press, London, UK.
- He, J.B., Wang, Q.G. and Lee, T.H. (2000), "PI/PID controller tuning via LQR approach", *Chem. Eng. Sci.*, **55**, 2429-2439. [https://doi.org/10.1016/S0009-2509\(99\)00512-6](https://doi.org/10.1016/S0009-2509(99)00512-6).
- Hernández, E., Kalise, D. and Otárola, E. (2011), "A locking-free scheme for the LQR control of a Timoshenko beam", *J. Comput. Appl. Math.*, **235**, 1383-1393. <https://doi.org/10.1016/j.cam.2010.08.025>.
- Ibrahimbegovic, A. and Boujelben, A. (2018), "Long-term simulation of wind turbine structure for distributed loading describing long-term wind loads for preliminary design", *Couple. Syst. Mech.*, **7**(2), 233-254. <https://doi.org/10.12989/csm.2018.7.2.233>.
- Ibrahimbegovic, A. and Mejia-Nava, R.A. (2023), *Structural Engineering*, Springer Cham, Switzerland.
- Ibrahimbegovic, A., Frey, F. and Kozar, I. (1995), "Computational aspects of vector-like parametrization of three-dimensional finite rotations", *Int. J. Numer. Meth. Eng.*, **38**, 3653-3673. <https://doi.org/10.1002/nme.1620382107>.
- Ibrahimbegovic, A., Knopf-Lenoir, C.A. Kučerová, A. and Villon, P. (2004), "Optimal design and optimal control of structures undergoing finite rotations and elastic deformations", *Int. J. Numer. Meth. Eng.*, **61**(14), 2428-2460. <https://doi.org/10.1002/nme.1150>.
- Ibrahimbegovic, A., Mejia-Nava, R.A. and Ljukovac, S. (2023), "Reduced model for fracture of geometrically exact planar beam: Non-local variational formulation, ED-FEM approximation and operator split solution", *Int. J. Numer. Meth. Eng.*, e7369. <https://doi.org/10.1002/nme.7369>.
- Kirk, D.E. (2004), *Optimal Control Theory: An Introduction*, Dover Publications, New York, USA.
- Lewis, F.L., Vrabie, D.L. and Syrmos, V.L. (2012), *Optimal Control*, John Wiley & Sons, New Jersey, USA.
- Marsden, J.E. and Hughes, T.J.R. (1983) *Mathematical Foundations of Elasticity*, Prentice-Hall, Englewood Cliffs, USA.
- Neto, M.A., Ambrósio, J.A.C., Roseiro, L.M., Amaro, A. and Vasques, C.M.A. (2013), "Active vibration control of spatial flexible multibody systems", *Multib. Syst. Dyn.*, **30**, 13-35. <https://doi.org/10.1007/s11044-013-9341-3>.
- Rafiee, M., Nitzsche, F. and Labrosse, M. (2017), "Dynamics, vibration and control of rotating composite beams and blades: A critical review", *Thin Wall. Struct.*, **119**, 795-819. <https://doi.org/10.1016/j.tws.2017.06.018>.
- Reissner, E. (1972), "On one-dimensional finite-strain beam theory: The plane problem", *J. Appl. Math. Phys.*, (ZAMP), **23**, 795-804. <https://doi.org/10.1007/BF01602645>.
- Schindele, D. and Aschemann, H. (2014), "Adaptive LQR-Control design and friction compensation for flexible high-speed rack feeders", *J. Comput. Nonlin. Dyn.*, **9**, 011011. <https://doi.org/10.1115/1.4025351>.
- Song, J., Chen, W., Guo, S. and Yan, D. (2021), "LQR control on multimode vortex-induced vibration of flexible riser undergoing shear flow", *Marine Struct.*, **79**, 103047. <https://doi.org/10.1016/j.marstruc.2021.103047>.
- Taylor, R.L. (2014), "Feap-finite element analysis program", University of California, Berkeley, USA.
- Trélat, E. (2005), *Contrôle Optimal: Théorie & Applications*, Vuibert, France.
- Trevisani, A. (2003), "Feedback control of flexible four-bar linkages: A numerical and experimental investigation", *J. Sound Vib.*, **268**, 947-970. [https://doi.org/10.1016/S0022-460X\(03\)00376-6](https://doi.org/10.1016/S0022-460X(03)00376-6).
- Vasques, C.M.A. and Dias Rodrigues, J. (2006), "Active vibration control of smart piezoelectric beams: Comparison of classical and optimal feedback control strategies", *Comput. Struct.*, **84**, 1402-1414.

<https://doi.org/10.1016/j.compstruc.2006.01.026>.

Zhou, P., Wang, F.Y., Chen, W. and Lever, P. (2001), "Optimal construction and control of flexible manipulators: a case study based on LQR output feedback", *Mechatron.*, **11**, 59-77.
[https://doi.org/10.1016/S0957-4158\(00\)00004-0](https://doi.org/10.1016/S0957-4158(00)00004-0).

CC