Numerical and experimental investigation on the synthesis of extended Kalman filters for cable-driven parallel robots modeled through DAEs

Jason Bettega[1] · Giovanni Boschetti[1] · Borja R. Frade[2] · Francisco González[2] ·
Giulio Piva[1] 🅭 · Dario Richiedei[1] · Alberto Trevisani[1]

Abstract

Cable-driven parallel robots are parallel robots where light-weight cables replace rigid bodies to move an end-effector. Their peculiar design allows obtaining large workspaces, high-dynamic handlings, ease of reconfigurability and, in general, low-cost architecture. Knowing the full state variables of a cable robot may be essential to implement advanced control and monitoring strategies and imposes the development of state observers. In this work a general approach to develop nonlinear state observers based on an extended Kalman filter (EKF) is proposed and validated both numerically and experimentally by referring to a cable-suspended parallel robot. The state observer is based on a system model obtained by converting a set of differential algebraic equations into ordinary differential equations through different formulations: the penalty formulation, the Udwadia–Kalaba formulation, and the Udwadia–Kalaba–Phohomsiri formulation, which have been chosen since they can handle the presence of redundant constraints as often happens in cable-driven parallel robots. In the numerical investigation, the EKF is validated simulating encoders heavily affected by quantization errors to demonstrate the filtering capabilities of EKF. In the experimental investigation, a very challenging validation is proposed: only two sensors measuring the rotations of two motors are used to estimate the actual position and velocity of the end-effector. This result cannot be achieved by sole forward kinematics and clearly proves the effectiveness of the proposed observer.

Keywords State estimation · Extended Kalman filter · Cable-driven parallel robots · Penalty formulation · Udwadia–Kalaba · Udwadia–Kalaba–Phohomsiri

1 Introduction

Cable-driven parallel robots (CDPRs) are a class of parallel robots where the end-effector is actuated by light-weight cables. Each cable is wound around a winch driven by a motor. The winch can easily provide several meters of cable by enabling large workspaces, reduced inertias, and high velocities at the end-effector. CDPRs also give the opportunity to implement deployable and reconfigurable topologies that may expand the application of robotics

---

in new environments (e.g., search and rescue operations, large buildings maintenance and construction).

Despite these advantages, the current employment of CDPRs seems hampered mainly by the consequences deriving from the unilateral nature of cables: they can only exert pulling forces. In practice, CDPRs need cable tensions to be kept always positive during motion, thus requiring accurate control schemes [1, 2]. Nonetheless, the use of CDPRs seems very promising and has already been suggested in several different operation fields, such as the already cited heavy handling and industrial manufacturing, but also: medical rehabilitation, rescue and home assistance, or sport shooting (see for example [3, 4] and the references therein). In the future, a wide use of CDPRs is expected thanks to their lightweight structure (which makes them energy efficient), modularity and reconfigurability (which makes them flexible and easy to transport), and finally, the potentially high dynamics and payload capacity (which makes them effective in a wide range of industrial applications).

The development of advanced control schemes mainly relies on the availability of models and, often, of the full state vector [1] (usually defined as positions and velocities in many control schemes for multibody systems [5]). Whenever the direct measurement of the state, or of the controlled outputs of interest, is not possible (or it is difficult to implement), state observers are required to provide estimates of such quantities. State observers (also named state estimators) are widely adopted in the field of multibody dynamics and several recent studies discuss this relevant issue [6, 7]. The need of a state observer is further stressed in the field of CDPRs by the difficulties in directly measuring the position of the suspended load in large dimensional robots, as shown in [8, 9].

The dynamics of a CDPR is nonlinear and imposes the development of nonlinear state observers, such as extended Kalman filters (EKFs), which are widely used in state estimation of nonlinear systems starting from a complete model formulated through first-order ordinary differential equations (ODEs). A widespread approach in the literature of CDPRs is writing the Newton–Euler equations of motion for the end-effector under the hypothesis that cables are stiff, massless, and straight, and often the contribution of the motor inertial properties to the overall system dynamics is neglected or just considered through approximations or through nonsystematic approaches. The use of redundant coordinates has only been rarely suggested in the literature of CDPRs, although this choice may provide some benefits that are well known to scholars working in the field of multibody dynamics, to handle some peculiarities of cable robots, such as simulating cable failures or bouncing motions (that make some kinematic constraints vanishing [10]), or to easily include motor dynamics [11].

In this paper, an EKF for CDPRs modeled through redundant coordinates is developed to estimate the state variables of a CDPR. The paper is focused on the EKF; other formulations that can be found in the literature (such as the sigma-point Kalman filters discussed in [7]) go beyond the goal of this study. The theoretical achievements presented in the paper can be applied to any CDPR topology, but for the sake of clarity, they are here applied to a simple and widespread CDPR design: a CDPR actuated by four cables in a crane-like configuration, which means that all the cables reach the end-effector from above. This cable layout classifies the robot as cable-suspended parallel robot (CSPR).

Since the models implemented in the state observer need to be in ODE representation, the conversion of DAEs into ODEs must be performed. This step can be carried out taking advantage of different formulations. In this work, three well established formulations are investigated: the penalty formulation [12], the Udwadia–Kalaba, and Udwadia–Kalaba–Phohomsiri formulations [13, 14], since all these approaches effectively handle multibody systems with redundant constraints [12, 15] as often occurs in CDPRs, where overconstrained (also denoted as overactuated) architectures are adopted to increase the

workspace [16]. Additionally, these formulations easily allow the computation of the Lagrange multipliers linked to the algebraic constraints due to the presence of cables, thus allowing for a straightforward evaluation of the cable tensions [17].

2 Dynamic modeling of a cable-driven parallel robot

2.1 Differential-algebraic equations (DAEs) modeling technique

Let us collect the s generalized coordinates of an arbitrary CDPR into the vector of redundant coordinates $\mathbf{q} \in \mathbb{R}^s$. By adopting the usual formalism of multibody system dynamics [18], the following set of DAEs of index-3 is obtained to model the system dynamics:

$$\begin{cases} \mathbf{M\ddot{q}} + \mathbf{\Phi_q}^T \boldsymbol{\lambda} = \mathbf{f}, \\ \quad \mathbf{\Phi(q)} = \mathbf{0}, \end{cases} \tag{1}$$

where $\mathbf{\Phi(q)}$ is the set of n position kinematic constraint equations, $\mathbf{M} \in \mathbb{R}^{s \times s}$ is the mass matrix, $\mathbf{f} \in \mathbb{R}^s$ is the vector of the generalized external forces, $\boldsymbol{\lambda}$ is the vector of the Lagrange multipliers, and $\mathbf{\Phi_q} = \mathbf{\Phi_q(q)} \in \mathbb{R}^{n \times s}$ is the Jacobian of the kinematic constraint equations.

In the case of CDPRs, $\mathbf{\Phi(q)}$ includes the constraint equations set by cables. By assuming that cables are perfectly stiff and taut, as reasonably done by most of the papers in the literature, and that the exit points of cables are fixed, cables behave as holonomic, scleronomic, ideal kinematic constraints. The constraint equation of the arbitrary ith cable relates the absolute coordinates of the point of the end-effector where the cable is attached (denoted as $\mathbf{p} = [x_p \ y_p \ z_p]^T$), the absolute coordinates of the cable exit point (denoted as $\mathbf{a}_i = [a_{i,x} \ a_{i,y} \ a_{i,z}]^T$), and the cable length l_i:

$$l_i = \|\mathbf{p} - \mathbf{a}_i\|. \tag{2}$$

In turn, l_i can be written as a function of the rotation θ_i of the ith motor that winds and unwinds the ith cable once the cable length corresponding to $\theta_i = 0$, denoted as $l_{0,i}$, and the winch radius r_i are known

$$l_i = l_{0,i} + r_i \theta_i. \tag{3}$$

The ith position constraint Φ_i can therefore be expressed in the following form:

$$\Phi_i = \mathbf{p}^T \mathbf{p} - 2\mathbf{p}^T \mathbf{a}_i + \mathbf{a}_i^T \mathbf{a}_i - l_{0,i}^2 - 2l_{0,i} r_i \theta_i - r_i^2 \theta_i^2 = 0. \tag{4}$$

The set of DAEs in Eq. (1) does not fit the form of dynamic models required to design state observers, which are usually formulated as discrete time, first-order ODEs. Therefore, DAE to ODE transformation is required. The literature in the field of multibody system dynamics has widely discussed the issue of converting DAEs to ODEs, especially for solving the issue of numerical simulation. In the selection of the conversion method, it should be noted that an analytical model representation is usually required in the design of EKFs, and therefore approaches based on iterations (such as, for example, the augmented Lagrangian formulation [19]) are not suitable for being used in developing EKFs, at least in their classical formulations (see, e.g., [20]). Estimation approaches that do not exploit analytical models, which have been sometimes proposed in the literature such as for example [21], are not discussed in this paper.

Additionally, the presence of redundant constraints should be often addressed in the case of CDPRs, as in the case of this paper. In the light of these requirements, Sects. 2.2 through 2.4 briefly recall some of the most widely adopted methods. Just their main equations and assumptions are mentioned; for further details, the readers could refer to the quoted papers.

The vector of the external forces $\mathbf{f}$ includes the gravity force, the control torques exerted by the motors (collected in vector $\boldsymbol{\tau}_m$), and friction forces on the motor shafts (whenever it is included in the model, as discussed in Sect. 5.4). $\mathbf{f}$ might also include the velocity dependent inertia forces. Hence, in the general case, $\mathbf{f} = \mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}})$.

2.2 DAE to ODE conversion through the projection matrix

The first approach here discussed to convert a DAE model into a minimal set of ODE consists in using the "projection matrix method" by means of the velocity projection matrix relating the redundant velocities $\dot{\mathbf{q}} \in \mathbb{R}^s$ and the independent (minimal) ones $\dot{\mathbf{z}} \in \mathbb{R}^{n_{dof}}$, here denoted as $\mathbf{T}(\mathbf{q}) \in \mathbb{R}^{s \times n_{dof}}$:

$$\dot{\mathbf{q}} = \mathbf{T}(\mathbf{q})\dot{\mathbf{z}}. \tag{5}$$

Matrix $\mathbf{T}$ can be determined via several methods proposed in the literature (see [22] and the references therein). Taking the time derivative of Eq. (5), $\ddot{\mathbf{q}} = \mathbf{T}(\mathbf{q})\ddot{\mathbf{z}} + \dot{\mathbf{T}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{z}}$ and considering that $\mathbf{T}^T \boldsymbol{\Phi}_{\mathbf{q}}^T = \mathbf{0}$ leads to the minimal set of ODEs

$$\mathbf{T}(\mathbf{q})^T \mathbf{M} \mathbf{T}(\mathbf{q}) \ddot{\mathbf{z}} = \mathbf{T}(\mathbf{q})^T \left(\mathbf{f} - \mathbf{M} \dot{\mathbf{T}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{z}} \right), \tag{6}$$

Equation (6) can be expressed in the following compact form:

$$\bar{\bar{\mathbf{M}}} \ddot{\mathbf{z}} = \bar{\bar{\mathbf{f}}}, \tag{7}$$

where $\bar{\bar{\mathbf{M}}} = \mathbf{T}(\mathbf{q})^T \mathbf{M} \mathbf{T}(\mathbf{q})$ and $\bar{\bar{\mathbf{f}}} = \mathbf{T}(\mathbf{q})^T \left(\mathbf{f} - \mathbf{M} \dot{\mathbf{T}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{z}} \right)$. An advantage of this formulation is that it provides an exact conversion of the DAEs without any tuning parameters (as will be required by the following formulations) and allows the rigorous solution of the constraint equations at the position, velocity, and acceleration levels [23]. On the other hand, in the case of design of state observers for overconstrained CDPRs, it does not retain all the actuated coordinates.

2.3 DAE to ODE conversion through the penalty formulation

To retain all the redundant coordinates in the multibody model used in the observer design, a set of nonminimal ODEs should be formulated. This approach allows directly estimating the time evolution of all the generalized redundant coordinates. A common approach is using the penalty formulation [18]. The use of the penalty formulation in designing state observers for multibody systems has been already discussed in [20, 24]; here, just the main equations are proposed to clarify the implementation of the proposed state observer and to compare this formulation with those in Sect. 2.4 (that are, in contrast, rarely used in the design of state observers for multibody systems).

Such a formulation assumes that the Lagrange multipliers are proportional to the constraint violation at the position, velocity, and acceleration levels. In its simplest form, the following definition is assumed by means of the three scalar tuning parameters α, ξ, and ω:

$$\boldsymbol{\lambda} = \alpha \left(\ddot{\boldsymbol{\Phi}} + 2\xi \omega \dot{\boldsymbol{\Phi}} + \omega^2 \boldsymbol{\Phi} \right). \tag{8}$$

The choice of their values has been carried out by following the suggestions proposed in the literature (see, e.g., [12]). Since the constrains represented through Eq. (4) are sclero-nomic, their time derivative can be expressed as

$$\dot{\mathbf{\Phi}}(\mathbf{q}) = \mathbf{\Phi_q}\dot{\mathbf{q}},$$
$$\ddot{\mathbf{\Phi}}(\mathbf{q}) = \mathbf{\Phi_q}\ddot{\mathbf{q}} + \dot{\mathbf{\Phi}}_\mathbf{q}\dot{\mathbf{q}},$$

(9)

and therefore, the following set of s ODEs is obtained to model the dynamics of the CDPR:

$$\left(\mathbf{M} + \mathbf{\Phi_q}^T \alpha \mathbf{\Phi_q}\right) \ddot{\mathbf{q}} + \mathbf{\Phi_q}^T \alpha \left(\dot{\mathbf{\Phi}}_\mathbf{q} + 2\xi\omega\mathbf{\Phi_q}\right) \dot{\mathbf{q}} + \mathbf{\Phi_q}^T \alpha\omega^2 \mathbf{\Phi} = \mathbf{f}.$$

(10)

Again, Eq. (10) can be written in the following compact form with the obvious meaning of the new variables introduced:

$$\bar{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \bar{\mathbf{k}}(\mathbf{q}) = \mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}).$$

(11)

It should be noted that this method effectively handles systems with redundant constraints or singular kinematic configurations.

2.4 DAE to ODE conversion through the Udwadia–Kalaba formulation

2.4.1 General background

A different approach to convert the DAEs into a set of ODEs retaining all the s redundant coordinates is through the methods exploiting the exact evaluation of the Lagrange multi-pliers, most of which are related to the Gauss principle of least constraint. The most famous of these methods is, probably, the Udwadia–Kalaba formulation, originally proposed in [25] for systems with nonsingular mass matrix, lately extended in [14] to avoid the use of the ma-trix pseudoinverse and in [26] to handle systems with singular mass matrix. Other similar approaches, whose equivalence with the Udwadia–Kalaba formulations has been proved, have been proposed in [27] and in [28] by providing a different way to evaluate the La-grange multipliers, which results in a slightly different final form of the ODEs of motion. These methods should theoretically lead to equal results. In practice, slightly different re-sults might be experienced for some cases that are for example related to the ill numerical conditioning of the system matrices and to the numerical methods used in computing pseu-doinverse matrices [29] or other matrices required by each method (such as the orthogonal complement required in [14]). Despite the simplicity and effectiveness of this kind of for-mulations that are widely adopted in simulating multibody systems [29], their use in the design of state observers is rarely investigated in the literature.

Basically, all these methods formulate the acceleration of the s redundant coordinates of the constrained system $\ddot{\mathbf{q}}$ as the sum of the free body (unconstrained) accelerations $\ddot{\mathbf{q}}_f = \mathbf{M}^{-1}\mathbf{f}$ and the perturbation due to the kinematic constraints $\ddot{\mathbf{q}}_c$:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}\mathbf{f} + \ddot{\mathbf{q}}_c.$$

(12)

Depending on the specific method adopted, among [14, 15, 25–28], a different evaluation of $\ddot{\mathbf{q}}_c$ is provided. In this paper, two of these formulations are evaluated in Sects. 2.4.2 and 2.4.3. Similar treatments should be adopted for the other ones.

A similar interpretation can be provided to the penalty formulation by introducing the matrix $\boldsymbol{\alpha} = \alpha \mathbf{I}_{n \times n}$ and by developing the inverse of $\left(\mathbf{M} + \boldsymbol{\Phi}_{\mathbf{q}}^T \boldsymbol{\alpha} \boldsymbol{\Phi}_{\mathbf{q}}\right)$ through the Woodbury formula for inverse matrix [30]

$$\left(\mathbf{M} + \boldsymbol{\Phi}_{\mathbf{q}}^T \boldsymbol{\alpha} \boldsymbol{\Phi}_{\mathbf{q}}\right)^{-1} = \mathbf{M}^{-1} - \mathbf{M}^{-1} \boldsymbol{\Phi}_{\mathbf{q}}^T \left(\boldsymbol{\alpha}^{-1} + \boldsymbol{\Phi}_{\mathbf{q}} \mathbf{M}^{-1} \boldsymbol{\Phi}_{\mathbf{q}}^T\right)^{-1} \boldsymbol{\Phi}_{\mathbf{q}} \mathbf{M}^{-1} \tag{13}$$

that leads to

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} \mathbf{f} + \mathbf{M}^{-1} \boldsymbol{\Phi}_{\mathbf{q}}^T \left(\boldsymbol{\alpha}^{-1} + \boldsymbol{\Phi}_{\mathbf{q}} \mathbf{M}^{-1} \boldsymbol{\Phi}_{\mathbf{q}}^T\right)^{-1} \boldsymbol{\Phi}_{\mathbf{q}} \mathbf{M}^{-1} \left(\bar{\mathbf{C}} \dot{\mathbf{q}} + \bar{\mathbf{k}}\right). \tag{14}$$

2.4.2 The original formulation

Let us now consider the original Udwadia–Kalaba formulation, which is a milestone in the field of multibody system dynamics as proved by the several papers exploiting it (see, e.g., [25, 31, 32] and the references therein). The other ones can be treated in a similar way. According to such a method, $\ddot{\mathbf{q}}_c$ is computed as follows:

$$\ddot{\mathbf{q}}_c = \mathbf{M}^{-\frac{1}{2}} \mathbf{B}^\dagger \left(\boldsymbol{\gamma} - \boldsymbol{\Phi}_{\mathbf{q}} \ddot{\mathbf{q}}_f\right), \tag{15}$$

matrix $\mathbf{B}^\dagger$ is the pseudoinverse of $\mathbf{B}$ with

$$\mathbf{B} = \boldsymbol{\Phi}_{\mathbf{q}} \mathbf{M}^{-\frac{1}{2}}, \tag{16}$$

and $\boldsymbol{\gamma}$ arises from the acceleration constraint equations

$$\boldsymbol{\gamma} = -\left(\boldsymbol{\Phi}_{\mathbf{q}} \dot{\mathbf{q}}\right)_{\mathbf{q}} \dot{\mathbf{q}} - 2 \dot{\boldsymbol{\Phi}}_{\mathbf{q}} \dot{\mathbf{q}}. \tag{17}$$

Since this formulation has been obtained from a DAE system of index-1 by writing the constraints at the acceleration level, position and speed constraints usually drift during numerical integration of the equation of motions. Therefore, the Baumgarte stabilization [33] is usually introduced by leading to the following set of ODEs to be integrated:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} \mathbf{f} + \mathbf{M}^{-\frac{1}{2}} \mathbf{B}^\dagger \left(\boldsymbol{\gamma} - 2\chi \dot{\boldsymbol{\Phi}} - \varphi^2 \boldsymbol{\Phi} - \boldsymbol{\Phi}_{\mathbf{q}} \ddot{\mathbf{q}}_f\right), \tag{18}$$

where χ and φ are Baumgarte stabilization parameters that are tuned either with some established rules [34] or with a trial-and-error procedure. The terms of the Baumgarte stabilization play a role similar to $2\xi \omega \alpha$ and $\alpha \omega^2$ used in the penalty formulation [22].

2.4.3 The Udwadia–Kalaba–Phohomsiri formulation

The Udwadia–Kalaba formulation relies on the pseudoinversion of matrix $\mathbf{B}$, which could be computationally demanding or ill-conditioned in some cases [29] the previous formulation has been extended [14] through a new formulation to compute $\ddot{\mathbf{q}}_c$ by exploiting the orthogonal complement matrix of $\mathbf{B}$ (as defined in Eq. (16)), here denoted as $\mathbf{V}$ ($\mathbf{BV} = \mathbf{0}$):

$$\ddot{\mathbf{q}}_c = \mathbf{M}^{-\frac{1}{2}} \left(\mathbf{B}^T \mathbf{B} + \mathbf{V} \mathbf{V}^T\right)^{-1} \mathbf{B}^T \left(\boldsymbol{\gamma} - \boldsymbol{\Phi}_{\mathbf{q}} \ddot{\mathbf{q}}_f\right). \tag{19}$$

Again, the Baumgarte stabilization is usually adopted, leading to the following set of ODEs:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} \mathbf{f} + \mathbf{M}^{-\frac{1}{2}} \mathbf{N} \mathbf{B}^T \left(\boldsymbol{\gamma} - 2\chi \dot{\boldsymbol{\Phi}} - \varphi^2 \boldsymbol{\Phi} - \boldsymbol{\Phi}_{\mathbf{q}} \ddot{\mathbf{q}}_f\right). \tag{20}$$

3 Development of the extended Kalman filter (EKF)

3.1 Model formulation for state estimation

The EKF is implemented in discrete time through the state space, difference equation, exploiting the discretized form of the following continuous time, first-order dynamic model

$$\dot{\mathbf{x}}(t) = \boldsymbol{f}_c(\mathbf{x}(t), \mathbf{u}(t)), \tag{21}$$

where $\mathbf{x}(t)$ is the state vector, $\mathbf{u}(t)$ is the input vector, and $\boldsymbol{f}_c$ is obtained from the dynamic models discussed in Sect. 2. Let us now assume that redundant ODEs are adopted through one of the methods in Sects. 2.3–2.4, and therefore let us introduce the state vector of the first-order model $\mathbf{x}(t) = [\dot{\mathbf{q}}^T \ \mathbf{q}^T]^T$.

This state definition is adopted for all the three model formulations investigated in this paper and it is common in several EKF designed for multibody systems [35–37]. In contrast, different model matrices are formulated, as described in the following.

If the penalty formulation is adopted, as in Eq. (11), then accelerations are written in the form

$$\ddot{\mathbf{q}} = \bar{\mathbf{M}}(\mathbf{q})^{-1}\mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}) - \bar{\mathbf{M}}(\mathbf{q})^{-1}\bar{\mathbf{k}}(\mathbf{q}) - \bar{\mathbf{M}}(\mathbf{q})^{-1}\bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}, \tag{22}$$

thus, leading to the following first-order representation of the nonlinear model:

$$\begin{bmatrix} \ddot{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} -\bar{\mathbf{M}}(\mathbf{q})^{-1}\bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} \bar{\mathbf{M}}(\mathbf{q})^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{f}}(t, \mathbf{q}, \dot{\mathbf{q}}) \\ \mathbf{0} \end{bmatrix}, \tag{23}$$

where the equivalent external forces vector $\bar{\mathbf{f}}(t, \mathbf{q}, \dot{\mathbf{q}})$ is

$$\bar{\mathbf{f}}(t, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}) - \bar{\mathbf{k}}(\mathbf{q}), \tag{24}$$

clearly, given the structure of Eq. (23), $\mathbf{u} = \begin{bmatrix} \bar{\mathbf{f}} \\ \mathbf{0} \end{bmatrix}$.

If the Udwadia–Kalaba formulation is used (together with the Baumgarte stabilization), as in Eq. (18), accelerations are written in the following form:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{I} - \mathbf{M}^{\frac{1}{2}}\mathbf{B}^{\dagger}(\varphi^2\boldsymbol{\Phi} + \boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}))\mathbf{f} + \mathbf{M}^{-1}\mathbf{M}^{\frac{1}{2}}\mathbf{B}^{\dagger}(\dot{\boldsymbol{\Phi}}_{\mathbf{q}} - 2\chi\boldsymbol{\Phi}_{\mathbf{q}})\dot{\mathbf{q}}, \tag{25}$$

thus, leading to the following first-order representation of the nonlinear model:

$$\begin{bmatrix} \ddot{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{M}^{-1}\widehat{\mathbf{C}} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} \mathbf{M}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{f}} \\ \mathbf{0} \end{bmatrix}, \tag{26}$$

where the equivalent external forces vector is

$$\widehat{\mathbf{f}} = (\mathbf{I} - \mathbf{M}^{\frac{1}{2}}\mathbf{B}^{\dagger}(\varphi^2\boldsymbol{\Phi} + \boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}))\mathbf{f} \tag{27}$$

and the velocity-dependent matrix is

$$\widehat{\mathbf{C}} = \mathbf{M}^{\frac{1}{2}}\mathbf{B}^{\dagger}(\dot{\boldsymbol{\Phi}}_{\mathbf{q}} - 2\chi\boldsymbol{\Phi}_{\mathbf{q}}). \tag{28}$$

Finally, when the Udwadia–Kalaba–Phohomsiri formulation is employed (together with the Baumgarte stabilization), as in Eq. (20), the first-order representation of the nonlinear model is expressed as

$$\begin{bmatrix} \ddot{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{M}^{-1}\check{\mathbf{C}} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} \mathbf{M}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \check{\mathbf{f}} \\ \mathbf{0} \end{bmatrix}, \tag{29}$$

where equivalent vector of external actions is composed as

$$\check{\mathbf{f}} = (\mathbf{I} - \mathbf{M}^{\frac{1}{2}}\mathbf{N}\mathbf{B}^{T}(\varphi^{2}\mathbf{\Phi} + \mathbf{\Phi_{q}}\mathbf{M}^{-1}))\mathbf{f} \tag{30}$$

and the velocity-dependent matrix is modified as

$$\check{\mathbf{C}} = \mathbf{M}^{\frac{1}{2}}\mathbf{N}\mathbf{B}^{T}(\dot{\mathbf{\Phi}}_{\mathbf{q}} - 2\chi\mathbf{\Phi_{q}}). \tag{31}$$

Since these three formulations embed the constraint equation, the models in Eqs. (23), (26), and (29) fit the form of Eq. (21) and are therefore suitable to be adopted for developing Kalman filters.

Discretization requires casting the model in the following form (k is the index of the discrete time step):

$$\begin{cases} \mathbf{x}_k = \boldsymbol{f}\,(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}), \\ \mathbf{y}_k = \boldsymbol{g}\,(\mathbf{x}_k, \mathbf{u}_k), \end{cases} \tag{32}$$

$\mathbf{y} \in \mathbb{R}^{n_y}$ is the output vector that is related to the state and input vectors through the algebraic equation $\boldsymbol{g}$. The output vector collects the sensor measurements (excluding the measurements of the input that are collected within $\mathbf{u}$) and should be carefully selected to ensure observability. Basically, a system is said to be observable if it is uniquely determined by the system model, its inputs, and its output [38]. Several discretization schemes can be adopted with different accuracy, stability, and computational effort [7, 18].

In this paper, to simplify the computational cost for boosting real time estimation, and being aware of the effect of the closed-loop correction introduced by the state observer (see Sect. 3.2) that can compensate for energy losses due to the numerical integration scheme, discretization is performed with a simplified method based on an approximation of the forward Euler scheme (with sample time Δt), as often done in control theory and proved to be effective in multibody systems as well [1, 39]. This approach is a first-order exponential discretization scheme that assumes a zero-order hold (ZOH) approximation of the system input over the time step Δt.

Within this formulation, matrix exponentials and integral of matrix exponentials appear. Since the presence of these terms can lead to high computational efforts, the state-space matrices involved in the continuous representation are approximated through a Taylor's series expansion. This choice remarkably reduces the computational burden without introducing noticeable discretization errors. Higher-order discretization methods could be used at the cost of an increase of the computational effort.

The nonlinear, discrete time model $\mathbf{x}_k = \boldsymbol{f}\,(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ is cast in the following compact form (that apparently resembles the one of a linear system):

$$\mathbf{x}_k = \mathbf{A}_{d,k-1}\mathbf{x}_{k-1} + \mathbf{B}_{d,k-1}\mathbf{u}_{k-1}. \tag{33}$$

As an explicative example, by adopting the first-order representation achieved through the penalty formulation in Eq. (23) (the extension to other two methods is straightforward)

$$\mathbf{u}_{k-1} = \begin{bmatrix} \bar{\mathbf{f}}_{k-1} \\ \mathbf{0} \end{bmatrix}, \tag{34}$$

the state-dependent matrices, due to the dependence of some submatrices on $\mathbf{q}$ and $\dot{\mathbf{q}}$ (that is omitted for clarity of representation), are defined as follows:

$$\mathbf{A}_{d,k-1} \approx \mathbf{I} + \begin{bmatrix} -\bar{\mathbf{M}}^{-1}\bar{\mathbf{C}} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}_{k-1} \Delta t, \qquad \mathbf{B}_{d,k-1} \approx \begin{bmatrix} \bar{\mathbf{M}}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}_{k-1} \Delta t. \tag{35}$$

In practice, $\mathbf{A}_{d,k-1}$ and $\mathbf{B}_{d,k-1}$ represent the discrete counterpart of the continuous time matrices of Eq. (23). The computation of $\mathbf{A}_{d,k-1}$, $\mathbf{B}_{d,k-1}$, and $\mathbf{u}_{k-1}$ is performed by firstly evaluating matrices $\bar{\mathbf{M}}$, $\bar{\mathbf{C}}$ and vector $\bar{\mathbf{f}}$ at sample k-1 through Eq. (11) exploiting $\mathbf{q}_{k-1}$, $\dot{\mathbf{q}}_{k-1}$ and secondly by employing Eqs. (34), (35).

$\bar{\mathbf{M}}$, $\bar{\mathbf{C}}$, and $\bar{\mathbf{f}}$ in Eq. (35) are, obviously, replaced by $\mathbf{M}$, $\widehat{\mathbf{C}}$, and $\widehat{\mathbf{f}}$ (see Eq. (26)) in the Udwadia–Kalaba formulation, and by $\mathbf{M}$, $\check{\mathbf{C}}$, and $\check{\mathbf{f}}$ (see Eq. (29)) in the Udwadia–Kalaba–Phohomsiri formulation.

3.2 The prediction-correction scheme

An EKF provides optimal estimates $\hat{\mathbf{x}}(t)$ of the actual state by fusing the prediction of a nominal model with a closed-loop correction inferred through the measurements retrieved from a proper set of sensors ensuring observability. The resulting closed-loop estimation is based on a prediction-correction scheme, where the correction is aimed at compensating modeling errors, including those related to energy leaks in numerical integration of the equations of motions.

The discrete time model $\boldsymbol{f}$ and the noisy input measurements $\mathbf{u}_{k-1}$ are adopted for computing the prediction (or a-priori estimation) $\hat{\mathbf{x}}_{k|k-1} = \boldsymbol{f}\left(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}\right)$, which is then corrected through the output estimation error $\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right)$, usually denoted as the innovation, with $\hat{\mathbf{y}}_{k|k-1} = \boldsymbol{g}\left(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_{k-1}\right)$ being the estimated output vector. The following recursive scheme is formulated:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{L}_{k|k}\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right). \tag{36}$$

$\mathbf{L}_{k|k} \in \mathbb{R}^{2s \times n_y}$ is a time-varying filter gain and $\mathbf{L}_{k|k}\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right)$ is the closed-loop correction in the control theory sense, forcing the estimation to track sensor measurements by compensating for noise and model uncertainty. To compute $\mathbf{L}_{k|k}$ at each time step, the EFK algorithm replaces the nonlinear model with its Jacobian matrices computed about the estimated state trajectory and uses them in the propagation of the noise covariance matrices [7].

By following the recursive scheme of the EKF, the updated covariance propagation matrix $\hat{\mathbf{P}}_{k|k-1} \in \mathbb{R}^{2s \times 2s}$ is computed by exploiting the covariance propagation matrix $\mathbf{P}_{k-1}$ of the previous time step as follows:

$$\hat{\mathbf{P}}_{k|k-1} = \mathbf{A}_{d,k-1}\mathbf{P}_{k-1}\mathbf{A}_{d,k-1}^T + \mathbf{Q}, \tag{37}$$

where $\mathbf{Q} \in \mathbb{R}^{2s \times 2s}$ is the covariance matrix of the model noise, which is in practice a tuning parameter that represents in an abstract way the amount of model uncertainty [38, 40].

$\mathbf{A}_{d,k-1}$ is the Jacobian matrix of the discrete transition model with respect to the state vector [38], and it is directly obtained in this work through an approximated and computationally efficient formulation that exploits the discretized model in Eq. (35) obtained through the first-order approximation. In practice, the computation of $\mathbf{A}_{d,k-1}$ is performed by firstly evaluating matrices $\bar{\mathbf{M}}$ and $\bar{\mathbf{C}}$ at sample k-1 through Eq. (11). It should be noted that the recent literature has often shown that approximate Jacobians are still effective in state estimation [35].

Then, the filter gain is computed as follows:

$$\mathbf{L}_{k|k} = \hat{\mathbf{P}}_{k|k-1}\mathbf{H}^T \left(\mathbf{H}\hat{\mathbf{P}}_{k|k-1}\mathbf{H}^T + \mathbf{R} \right)^{-1}, \tag{38}$$

where $\mathbf{R} \in \mathbb{R}^{n_y \times n_y}$ is the covariance matrix of measurement noise that can be treated as a further tuning parameter, and $\mathbf{H}$ is the Jacobian of $\boldsymbol{g}$.

$$\mathbf{H} = \left[\frac{\partial \boldsymbol{g}}{\partial \dot{\mathbf{q}}} \quad \frac{\partial \boldsymbol{g}}{\partial \mathbf{q}} \right] \tag{39}$$

Finally, the covariance propagation matrix is updated by setting

$$\mathbf{P}_k = \left(\mathbf{I} - \mathbf{L}_{k|k}\mathbf{H} \right) \hat{\mathbf{P}}_{k|k-1}. \tag{40}$$

4 Simulation assessment

4.1 Motivations

The first assessment of the proposed estimation scheme is done through numerical simulations. Section 5 provides experimental validation in a slightly different condition and with a different, and even more challenging, goal. The choice of firstly providing a numerical validation is justified by several motivations. First of all, in numerical simulations the reference system (whose state should be tracked by the state observer) is represented by a numerical simulator, and therefore the actual system state is exactly known; in contrast, the actual state is not available in experimentations. Therefore, a correct and fair comparison of different DAE to ODE conversion methods within the EKF can be done in a fair way [20]. Secondly, numerical simulations allow assuming the use of encoders providing a very coarse quantization of the measured positions, which is very critical in speed estimation, especially when numerical time derivation approaches are adopted [41].

4.2 Description of the test case

The CDPR studied in this work, as sketched in Fig. 1, is a cable-suspended parallel robot: a three-DOF suspended end-effector (modeled as a lumped mass $m = 3$ kg) is driven by four cables winding on winches and actuated by motors (whose equivalent moments of inertia reflected to the motor shaft are $J_{m,1}, J_{m,2}, J_{m,3}, J_{m,4}$). The system is therefore overconstrained.

Vector $\mathbf{q} = [\mathbf{p}^T \ \boldsymbol{\theta}^T]^T$ includes the absolute Cartesian positions of the end-effector $\mathbf{p} = [x_p \ y_p \ z_p]^T$ and the angular positions of the motors $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T$.
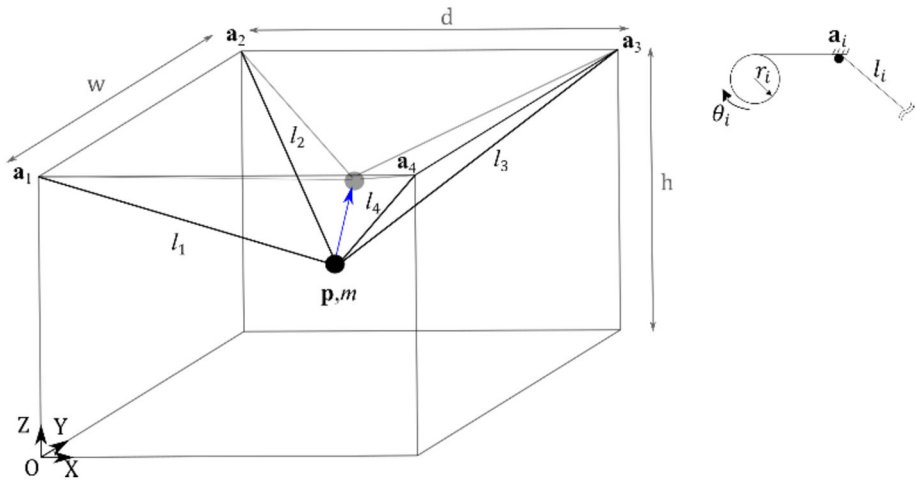
Fig. 1 Scheme of the studied CSPR

The following matrices for the model in Eq. (1) are therefore obtained:

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & J_{m,1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & J_{m,2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & J_{m,3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & J_{m,4} \end{bmatrix}, \tag{41}$$

$$\mathbf{\Phi_q} = 2 \begin{bmatrix} (\mathbf{p} - \mathbf{a}_1)^T & -r(r\theta_1 + l_{0,1}) & 0 & 0 & 0 \\ (\mathbf{p} - \mathbf{a}_2)^T & 0 & -r(r\theta_2 + l_{0,2}) & 0 & 0 \\ (\mathbf{p} - \mathbf{a}_3)^T & 0 & 0 & -r(r\theta_3 + l_{0,3}) & 0 \\ (\mathbf{p} - \mathbf{a}_4)^T & 0 & 0 & 0 & -r(r\theta_4 + l_{0,4}) \end{bmatrix}. \tag{42}$$

The frame dimensions are $1.69 \times 1.775 \times 1.89$ m (w × d × h in Fig. 1). The exit points are assumed to coincide with the upper the vertices $\mathbf{a}_i$ of the frame ($i = 1, .., 4$). The actuators have equal rotational moments of inertia $J_{m,i} = 5.12 \cdot 10^{-4}$ kg m^2 ($i = 1, .., 4$) (including both the motor rotor and the winch). The winches have equal radii $r_i = 0.036$ m ($i = 1, .., 4$). Rigid and taut cables are assumed, as it is often and reasonably assumed in the literature [8].

In this numerical example, the actuators are supposed to be equipped with low resolution encoders measuring θ with just 150 pulses per revolution and operating in 4x resolution. The measured angles are therefore corrupted by relevant white noise due to this coarse quantization. An estimate of torques exerted by each motor is available through the knowledge of the current provided by the motor drive, as usually supplied by commercial drivers and as required by the state observer. The reference system that produces the "actual" values of the state vector to be estimated by the state observer has been implemented through the projection matrix method, which uses a minimal coordinate representation and does not require any tuning parameter in the conversion of the DAEs to ODEs, as in contrast is required by the penalty formulation, the Udwadia–Kalaba, and Udwadia–Kalaba–Phohomsiri for-

mulation. The "actual" values of θ that are fed to the state observer are corrupted by quantization noise. Three different multibody formulations are tested by adopting the penalty, the Udwadia–Kalaba formulations for the model-based filter prediction, and the Udwadia–Kalaba–Phohomsiri formulation. In this way, the impact of different multibody formulations on the estimate accuracy is evaluated.

Besides comparing the observer outcomes with the actual state, the estimates of the end-effector position and velocities are performed through the forward kinematics and the noisy measurements provided by the encoders. As for the estimation of the motor shaft speeds, it is obtained by numerical time differentiation and by low pass filtering through a first-order filter with a 15 Hz bandwidth. This filter bandwidth has been selected, after some trial-and-errors, to remove the high frequency noise introduced by the numerical differentiation of the quantized encoder. As it is well known, increasing such a bandwidth reduces the noise filtering capabilities, while reducing it creates a phase lag in the estimated speeds, thus decreasing the stability margin if such estimates are used in feedback control loops.

Forward kinematics has been adopted since it is the usual approach adopted in cable robotics to estimate the pose and the speed of the end-effector [4, 42].

The simulated test consists of a rest-to-rest motion from point $p_i = [0.89\ 0.84\ 0.95]^T$ m to point $p_f = [1.0\ 1.0\ 1.5]^T$ m through a linear path, as shown in Fig. 1, by means of a 5th-degree polynomial law of motion.

4.3 State observer based on the penalty formulation

The EKF based on the penalty formulation relies on the dynamic model of Sect. 2.3, which requires a set of penalty parameters to avoid drifts of the algebraic constraints at the position, velocity acceleration levels. A time step $\Delta t = 1$ ms has been adopted for the numerical simulations. According to the literature [12], parameters $\alpha = 10^7$, $\xi = 1$, and $\omega = 10$ have been chosen by leading to a reduced constraints violation. Just to provide a concise evaluation of such a violation, the maximum and RMS (root mean square) values of $\mathbf{\Phi}^T \mathbf{\Phi}$ experienced in forward dynamics simulation, as suggested for example in [23, 43], are $3.8e\text{-}16$ m^4 and $2.8e\text{-}16$ m^4 for this test case, which is considered acceptable (this assumption of acceptability is corroborated by the accurate estimates sported by the EKF).

To simplify the observer tuning, the covariance matrices have been modeled as diagonal and constant matrices, although EKF can handle nonconstant values and off-diagonal terms. It has been assumed that only three encoder readings are available and, for the test case under investigation, the angular position of motors 1, 2, and 3 are fed to the EKF. The covariance matrix of the model noise is $\mathbf{Q} = 0.1 \cdot \mathbf{I}_{14 \times 14}$, the covariance matrix of the measurement noise is $\mathbf{R} = 10^{-3} \cdot \mathbf{I}_{3 \times 3}$, and the initial error covariance matrix is $\mathbf{P}_0 = 0.2 \cdot \mathbf{I}_{14 \times 14}$. The same values have been adopted also for the EKFs based on the Udwadia–Kalaba and Udwadia–Kalaba–Phohomsiri formulations. Since angular positions of the motors are part of vector $\mathbf{q}$ and encoder readings are used as measurements, the Jacobian matrix $\mathbf{H}$ employed in the EKF can be easily obtained exploiting Eq. (39), which yields to the following matrix:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{43}$$

The Cartesian coordinates of the end-effector positions (x_p, y_p, and z_p) and velocities ($\dot{x}_p$, $\dot{y}_p$, and $\dot{z}_p$) are shown in Figs. 2, 3 and 4. In each figure, a comparison is shown among the coordinates of the reference system, the estimates of the EKF based on penalty formulation (hereafter denoted as EKF-P), and the estimations obtained through forward kinematics,
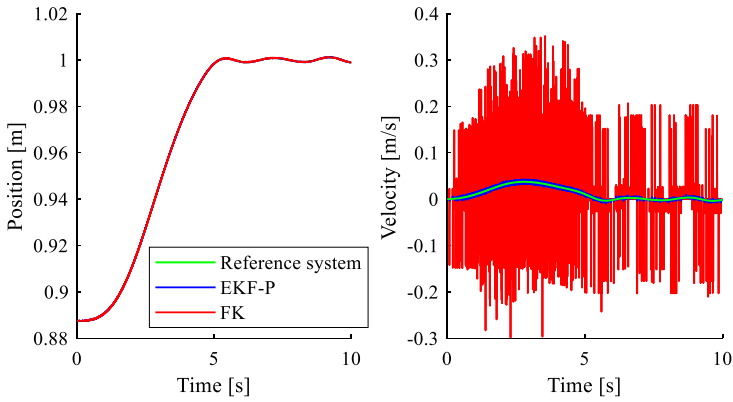
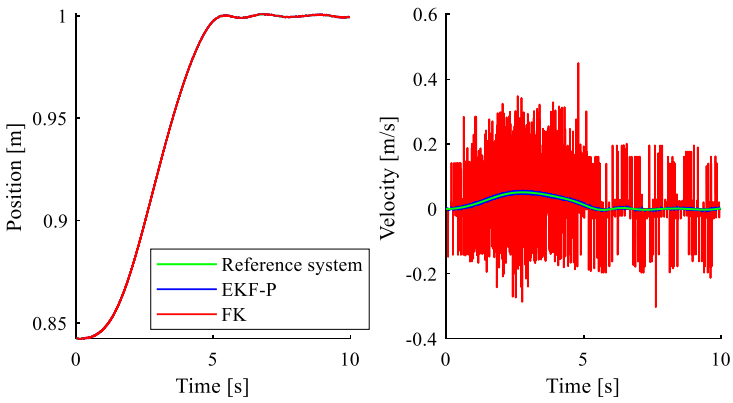**Fig. 2** Comparison of the actual and estimated x_p and $\dot{x}_p$

Fig. 3 Comparison of the actual and estimated y_p and $\dot{y}_p$
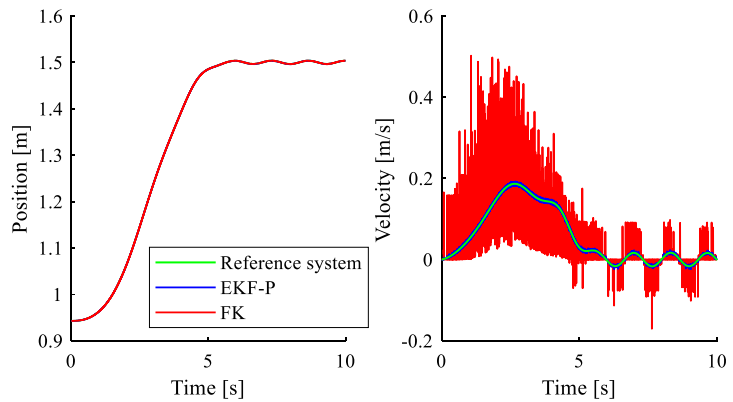

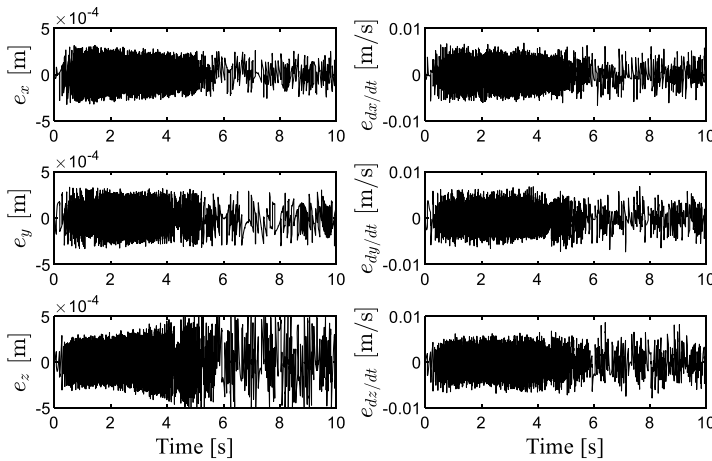**Fig. 4** Comparison of the actual and estimated z_p and $\dot{z}_p$

Fig. 5 Time-history of position and velocity estimation errors of the EKF-P

Table 1 RMS errors of position estimates against the reference system

	EKF-P	EKF-UK	EKF-UKP	FK
e_x^{RMS} [m]	1.13e-4	1.08e-4	1.05e-4	1.20e-4
e_y^{RMS} [m]	1.32e-4	1.28e-4	1.31e-4	1.26e-4
e_z^{RMS} [m]	2.04e-4	1.99e-4	1.97e-4	2.43e-4

Table 2 RMS errors of velocity estimates against the reference system

	EKF-P	EKF-UK	EKF-UKP	FK
$e_{dx/dt}^{RMS}$ [m/s]	2.13e-3	2.18e-3	2.13e-3	0.064
$e_{dy/dt}^{RMS}$ [m/s]	2.24e-3	2.29e-3	2.23e-3	0.059
$e_{dz/dt}^{RMS}$ [m/s]	2.53e-3	2.55e-3	2.52e-3	0.055

hereafter denoted as FK. The inspection of the velocity estimates reveals that the use of the EKF remarkably reduces the effect of the quantization noise on the numerical time derivatives, compared to the kinematics estimation, without introducing visible delay. A closer look to the result can be inferred from the error plots shown in Fig. 5, which are also summarized in Tables 1 and 2 through the RMS values.

4.4 State observers based on Udwadia–Kalaba and Udwadia–Kalaba–Phohomsiri formulations

The Udwadia–Kalaba and Udwadia–Kalaba–Phohomsiri formulations have been implemented as well, leading to the observers hereafter denoted as EKF-UK and EKF-UKP. The Baumgarte constraints stabilization has been adopted by setting $\chi = \varphi = 10^3$ to bound the constraint violation within an allowable threshold (that should be smaller than the accuracy required in the state estimates), while stabilizing numerical integration and making negligible the spurious dynamics introduced [34]. Again, a simulation time step $\Delta t = 1$ ms has
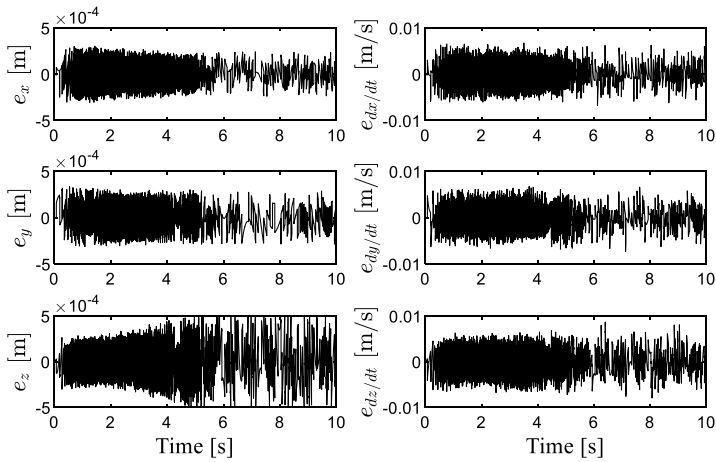
Fig. 6 Time history of position and velocity estimation error of the EKF-UK
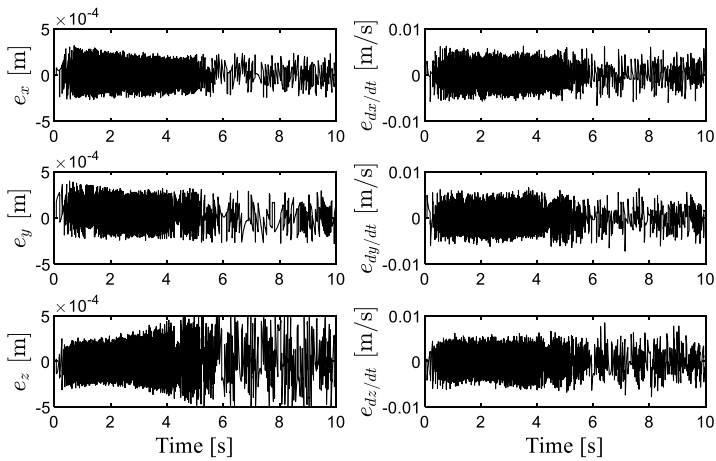
Fig. 7 Time history of position and velocity estimation error of the EKF-UKP

been adopted. The maximum and RMS values of $\mathbf{\Phi}^T\mathbf{\Phi}$ sported by the model are respectively 7.3e-15 m^4 and 1.4e-15 m^4 for both the formulations, which is, again, acceptable.

The results of the simulations are compared with the actual values, and the estimation errors are plotted in Figs. 6 and 7. The results are very similar to those provided by the EKF-P, and an effective speed noise rejection is, again, obtained. Tables 1 and 2 allow comparing the four different estimation approaches. While similar errors are obtained in terms of position, the use of any of the EKFs drastically reduces the speed RMS estimation error.

4.5 Estimation in the presence of model uncertainty

A sensitivity analysis on the developed state observers has also been carried out by simultaneously applying a 20% increase of all the entries of **M**, i.e., m and $J_{m,i}$, thus remarkably

Table 3 RMS errors of position and velocity with model mismatch

	EKF-P	EKF-UK	EKF-UKP		EKF-P	EKF-UK	EKF-UKP
e_x^{RMS} [m]	1e-4	1e-4	1e-4	$e_{dx/dt}^{RMS}$ [m/s]	1.9e-2	2e-2	2e-2
e_y^{RMS} [m]	1e-4	1e-4	1e-4	$e_{dy/dt}^{RMS}$ [m/s]	2.3e-2	2.2e-2	2.2e-2
e_z^{RMS} [m]	2e-4	2e-4	2e-4	$e_{dz/dt}^{RMS}$ [m/s]	1.1e-1	1.1e-1	1.1e-1
	ΔM-P	ΔM-UK	ΔM-UKP		ΔM-P	ΔM-UK	ΔM-UKP
e_x^{RMS} [m]	1.5e-2	1.5e-2	1.5e-2	$e_{dx/dt}^{RMS}$ [m/s]	4.7e-2	4.7e-2	4.7e-2
e_y^{RMS} [m]	2e-2	2e-2	2e-2	$e_{dy/dt}^{RMS}$ [m/s]	7e-2	7e-2	7e-2
e_z^{RMS} [m]	3.1e-1	3.1e-1	3.1e-1	$e_{dz/dt}^{RMS}$ [m/s]	7e-1	7e-1	7e-1
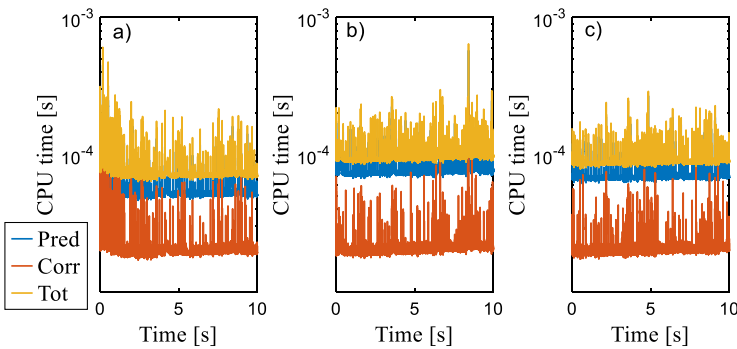
Fig. 8 CPU time relative to EKF-P a), EKF-UK b) and EKF-UKP c)

perturbing the inertial terms and gravity forces. Despite the relevant mismatch between the actual system model and the ones used in the state observer, the estimates computed by the EKFs are accurate since the maximum position errors is 2e-4 m in the vertical direction (that is the one also affected by the mismatch in the gravity force), as shown in Table 3.

It should be noted that even better results could have been obtained by performing a better tuning of **Q**, **R** matrices (to reflect the worse accuracy of the model); however, to perform a fair comparison, such values have been not changed compared to the nominal case.

To further stress the benefits of the state observers, the same perturbations have been applied to the models (developed through the three formulations) in forward dynamics, i.e., without the "closed loop" correction provided by the Kalman filter innovation. The results are shown in Table 3 and denoted as ΔM-P for the penalty formulation, ΔM-UK for the Udwadia–Kalaba formulation, ΔM-UKP for the Udwadia–Kalaba–Phohomsiri formulation. In this case, the position errors are two (in the x and y directions) and three (in the z direction) orders of magnitude greater than those sported by the EKFs.

4.6 Brief analysis of the computational effort

A simplified analysis of the computational effort of the three estimation schemes is here reported by displaying the CPU time required to perform the state estimation at each time step in Fig. 8. The total estimation time (named "Tot") is composed by the contribution of the

prediction ("Pred"), comprehensive of discretization, and prediction of the state through the model in Eq. (35), and by the contribution of the correction (named "Corr") that includes the computation of $\hat{\mathbf{P}}_{k|k-1}$, Eq. (37), of $\mathbf{L}_{k|k}$, Eq. (38), the correction of the state, Eq. (36), and the final update of $\mathbf{P}_k$, Eq. (40). Simulations have been carried out with a laptop PC with 8 GB RAM and a quad-core processor Intel Core i7-8565U. The filter implementation has been done through MatLab with a single-core programming by using the standard native functions for algebraic calculations, for example, for computing matrix inverse (through function *inv*), pseudoinverses (through function *pinv*), matrix orthogonal complement (through function *null*), for linear system solution (through the function *mldivide*).

For the case under investigation, and with the adopted implementation, shorter computational effort is required by the penalty formulation, requiring a root mean square (RMS) value of $0.8e$-4 s and a maximum value of $6.1e$-4 s. Slightly greater computational efforts are required by the EKF-UKP with $0.9e$-4 s and $2.9e$-4 s, and by the EKF-UK with $0.99e$-4 s and $6.4e$-4 s. The increase is due to the prediction phase of the filter, and hence to the way of formulating the model. In particular, the calculation of the pseudoinverse of matrix $\mathbf{B}$ for EKF-UK, and of its orthogonal complement for EKF-UKP cause such increases. All these values are, however, remarkably smaller than the sample time $\Delta t = 1$ ms.

5 Experimental results

5.1 Physical setup

The state observers have been implemented in the physical setup shown in Fig. 9 to verify the real time capability and the online state estimation of the full state of the system. The experimental setup resembles the one used in the numerical analysis, except for the use of resolvers in measuring the angular positions of the motors. These sensors, that are embedded in the motors, provide high-resolution and low-noise position signals, that enable accurate estimation of the rotational velocities, with negligible noise and lags. The model adopted is the one developed in Sect. 2 (and in the simulation analysis in Sect. 4) with 3 DOFs. Any deviation of the actual system dynamics from such a simplified model (such as, for example, the neglected cable dynamics, the lumped model approximation of the end-effector, the ideal model of the actuators) is implicitly treated as an uncertainty that the EKF should be able of compensating through the filter innovation (i.e., the feedback correction based on some sensor measurements).

The EKF has been deployed in TwinCAT which acts as a benchmark for an industrial environment. TwinCAT 3 is a software developed by Beckhoff that is suited for controlling industrial servodrivers and motors by deploying a control scheme into an industrial PLC. The main advantage of this setup is focused on the presence of TwinCAT 3, which allows the design of the control scheme, and in this case the state observer, in a MatLab/Simulink environment. In this case, the PLC is simulated by a workstation that has embedded and EtherCAT board that enables the EtherCAT communication with the servodrivers. This virtual PLC controls two Beckhoff AX5206 drivers, which in turn control two third-part rotary motors each. The clock of the PLC has been set to 500 Hz and in every sample time the system manages controlling the four servodrivers and running the EKF (i.e., $\Delta t = 2$ ms). Four independent controllers are developed for the motors, by exploiting the classical multi-loop scheme: an inner proportional-integral (PI) current controller, an intermediate PI speed controller, and an outer P position controller.

Fig. 9 Prototype of CSPR and a detail of a motor
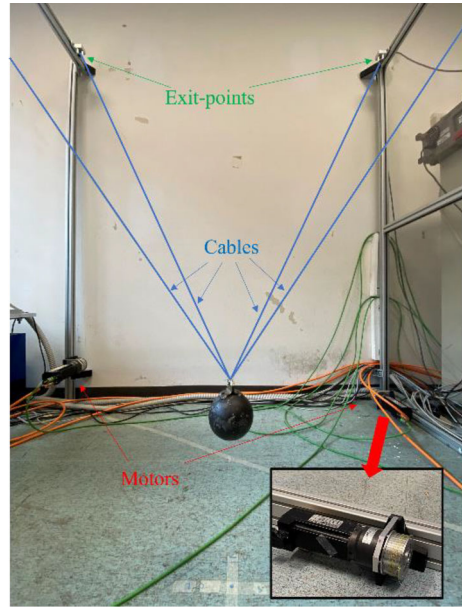
The four motors have integrated gearboxes, whose moment of inertia is collected into the motor inertia.

In both the experimental proposed tests, estimation has been done through the EKF exploiting the penalty formulation, named EKF-P, since the numerical analysis has shown that it provides the smallest computational effort among the various approaches adopted (with reference to the implementation developed in this research), while ensuring similar estimation errors. The same model of Sect. 4 is also exploited.

5.2 Solution of the forward kinematics

The solution of the forward position kinematics for redundant CDPRs usually exploits algebraic equations involving all the cable lengths (see, e.g., [44]). In this case, by exploiting the relation between the four cable lengths

$$l_1^2 + l_3^2 - \left(l_2^2 + l_4^2\right) = 0 \tag{44}$$

the following is obtained (with $\mathbf{a}_i = [a_{i,x}\ a_{i,y}\ a_{i,z}]^T$):

$$
\begin{aligned}
x_p &= \frac{l_1^2 - l_2^2 + a_{2,x}^2 - a_{1,x}^2}{2\left(a_{2,x} - a_{1,x}\right)}, \\
y_p &= \frac{l_2^2 - l_3^2 + a_{3,y}^2 - a_{2,y}^2}{2\left(a_{3,y} - a_{2,y}\right)}, \\
z_p &= a_{4,z} - \sqrt{l_4^2 - \left(x_p - a_{4,x}\right)^2 - \left(y_p - a_{4,y}\right)^2},
\end{aligned}
\tag{45}
$$

where each cable length is estimated through the sensed motor position and through Eq. (3). In the following, this scheme will be denoted as $(FK)_{1,2,3,4}$.

Actually, since the robot under investigation has three DOF, the knowledge of just three cable lengths suffices for the kinematic estimation of the end-effector position. By assuming an arbitrary triplet of cables (denoted through three indexes $c_i, c_j, c_k = [1, 2, 3, 4], c_i \neq c_j \neq c_k$), four other kinematic estimation schemes are obtained through the following analytical equations:

$$x_p = \frac{l_{c_i}^2 + l_{c_j}^2 + a_{j,x}^2}{2a_{j,x}},$$

$$y_p = \frac{l_{c_i}^2 - l_{c_k}^2 + a_{k,y}^2}{2a_{k,y}}, \tag{46}$$

$$z_p = a_{j,z} - \sqrt{l_{c_i}^2 - x_p^2 - y_p^2}.$$

The availability of four triplets leads to four schemes, henceforth denoted as $(FK)_{1,2,3}$, $(FK)_{1,2,4}$, $(FK)_{1,3,4}$, $(FK)_{2,3,4}$. Theoretically, these five kinematic models should lead to identical results. In practice, calibration and measurement errors create mismatches between their outcomes, and these differences change along the motion (both because of the unpredictable measurement error and of the position dependent sensitivity of calibration errors). The calibration of CDPR is a relevant and still-open issue [42, 45] since this kind of robots are affected by uncertainty or errors on the evaluation of initial end-effector pose and on the parameters of the kinematic constraint equations.

These mismatches between different models do not allow the end-effector pose to be exactly estimated with the forward kinematics. It is, therefore, more reasonable to assume a band where it is expected that the actual position should lie by adopting a statistical analysis of the outcomes of the five methods. The uncertainty band has been defined through the standard deviations coming from the five methods in each direction ($\sigma^x, \sigma^y, \sigma^z$), which have been computed by averaging the standard deviation of each forward kinematics scheme, which has been as usual defined with respect to the average values (computed at each time step k). A representative motion covering the workspace of interest for the two test cases discussed in Sects. 5.5 and 5.6 has been performed to estimate such standard deviations, leading to the following values: $\sigma^x = 9.5e\text{-}4$ m, $\sigma^y = 9.5e\text{-}4$ m, $\sigma^z = 1.5e\text{-}3$ m. By choosing the desired percentage of covered population to be equal to 95%, the lower and upper bounds have been set to $\pm 2\sigma$.

5.3 Goals of state estimation

The availability of high-resolution, low-noise resolvers allows the speed estimation to be easily and effectively performed with a simple Euler derivative, without the need of filtering the signal. This feature of the setup makes it different from the numerical test of Sect. 4, and therefore there is no need for the EKF to speed estimation. Another ambitious goal is therefore pursued in this experimental application of the EKF: the estimation of the end-effector pose, and velocity as well, by means of just two measurements of motor rotations, i.e., with just the availability of the lengths of two cables. Clearly, this set of measurements does not allow for using the kinematic models that require at least three measurements of the motor angular position: by means of two sensor readings, it is possible to retrieve just two cable lengths leading to infinite solutions of the kinematic forward problem that belong to the circular arc spanned by such two cables with known lengths. Figure 10 shows an example of such an indeterminacy by sketching the possible positions of the end-effector in a sample

Fig. 10 Possible solutions with just two resolver readings
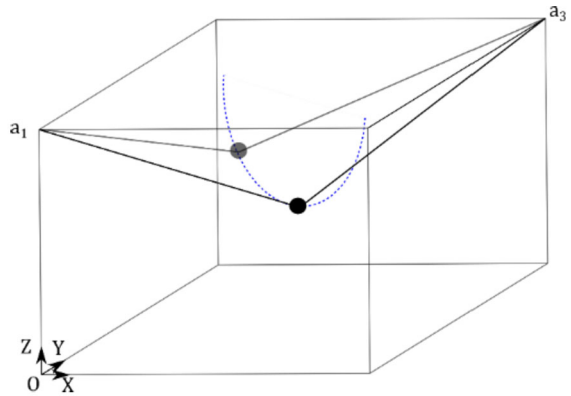
case of the measurement of motor 1 and 3 rotations. In such a figure, the configurations with the end-effector lying above the exit points of the cables are excluded, since they are clearly unfeasible for the CSPR; nonetheless, the kinematic equations would include these configurations as well.

The state observer can overcome this indeterminacy by fusing the measurements of the two resolvers with the information of the control torques driving the motor. Indeed, two resolver measurements, and in the presence of gravity, make the system observable.

Matrices $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$ have been slightly modified compared with the simulative analysis proposed in Sect. 4 to improve estimation, given the different uncertainty on the model (with respect of the actual system dynamics) and of the sensors that are actually employed in the setup. The following values have been tuned: $\mathbf{Q} = diag(10^{-10}, 10^{-10}, 10^{-10}, 10^{-5}, 10^{-5},$ $10^{-5}, 10^{-5}, 10^{-5}, 10^{-5}, 10^{-5}, 10^{-5}, 10^{-10}, 10^{-5}, 10^{-10})$, $\mathbf{P}_0 = diag(10^{-6}, 10^{-6}, 10^{-6}, 10^{-6},$ $10^{-6}, 10^{-6}, 10^{-6}, 10^{-15}, 10^{-15}, 10^{-15}, 10^{-5}, 10^{-5}, 10^{-5}, 10^{-5})$, and $\mathbf{R} = diag(10^{-3}, 10^{-3})$.

The results proposed in Sects. 5.5 and 5.6 have been obtained by assuming that just the angular positions of motors 1 and 3 are available, together with all the motor torques. The observer will estimate position and velocity of the remaining motors (2 and 4) and of the end-effector. Since only two encoder readings are used in the EKF, $\mathbf{H}$ takes the following form:

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{47}$$

It is interesting to note that, thanks to the low computational effort, two EKFs can run simultaneously in the developed software architecture. This feature could be favorably exploited in fault detection schemes that often use more state observer running in parallel.

5.4 Friction

The experimental identification of the model parameters revealed that the actuators, made by motor with integrated gearboxes, have relevant friction, also exhibiting hysteretic behaviors. In particular the "average" Coulomb friction for each motor is 0.18 Nm, with local variation due to the motor cogging torque ranging from 0.15 to 0.21 Nm. These values cannot be neglected if compared with the torques required for compensating gravity and inertial terms.

The issue of friction should be carefully accounted for since the presence of hysteretic nonlinear terms severely threats the accuracy of the state observer. As a first option, friction forces could be included in the state observer through accurate models, as those often dis-
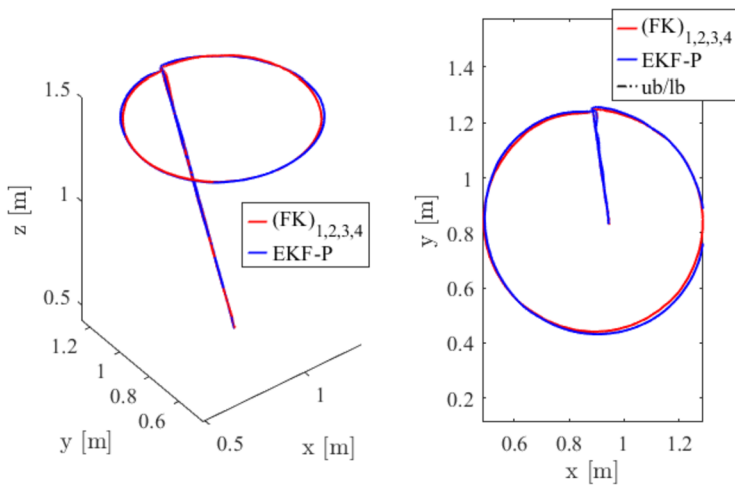
Fig. 11 Three-dimensional and xy-plane view of the end-effector's trajectory

cussed in [46]. This choice would set some critical issues since friction is a discontinuous phenomenon that is difficult to model and simulate, often leading to stiff system dynamics [47]. Including it in the model used in the prediction makes the development of a simplified discrete time difficult and it does not allow the straightforward calculation of the matrix Jacobians for computing $\mathbf{L}_{k|k}$. On the other hand, the identification of the parameters required for these models is not simple, also considering that, in the presence of electric motors, friction should consider ripple and cogging terms as well [48]. A different strategy is instead adopted in this paper to remarkably simplify the implementation of the state observer. The dynamic model of the CDPR just includes viscous friction, mainly due to the electric motors. Therefore, the torque of the *ith* motor fed to the observer has been estimated as follows (index i is omitted for clarity):

$$\tau(t) = k_t i(t) - \hat{\tau}_{fr}(t), \tag{48}$$

where k_t is the motor torque constant, retrieved from the motor datasheet, i is the motor armature current that is provided by the current control loop implemented in the motor drive (although it can be also replaced with the commanded one computed by the controller due to the high bandwidth of current controllers), and $\hat{\tau}_{fr}(t)$ is the estimated friction torque, computed through a simplified Coulomb model.

Improvement of this implementation, by assuming more accurate models for friction estimation, will be object of future investigations; nonetheless, despite the simplicity of this approach and the relevance of nonlinear friction in the system, the experimental results will prove its effectiveness.

5.5 1st test: circular trajectory

The first test performed is a circular path of the end-effector: it presents a radius of 0.4 m and having a duration of 4 s, as shown in Fig. 11. Two rest-to-rest motions along a line are performed at the beginning and the end of the test, from the rest pose (corresponding to the "home" position) to the initial position of the path *vice versa* at the end. Figures 12 and 13
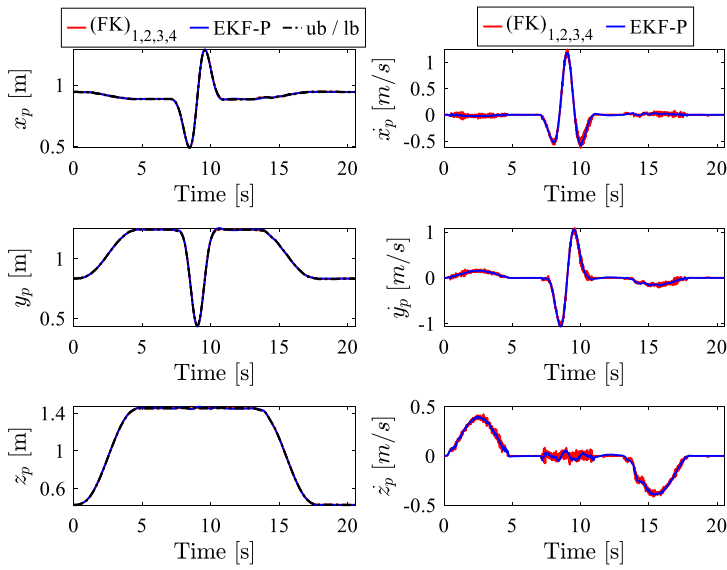
Fig. 12 End-effector Cartesian positions and velocities
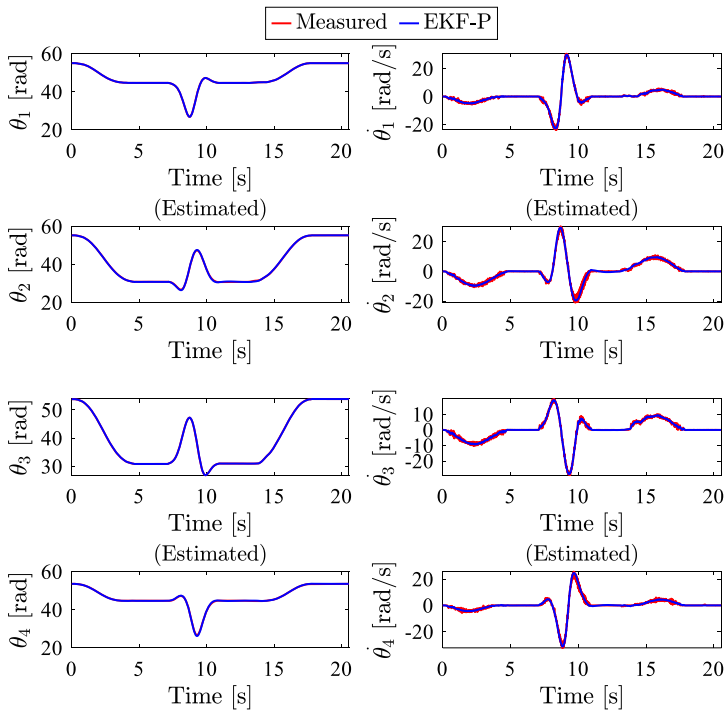
Fig. 13 Motor angular positions and velocities
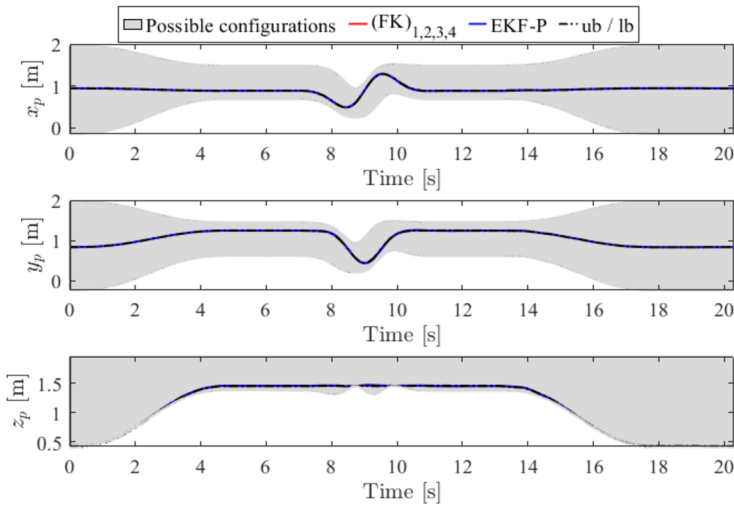
Fig. 14 End-effector's Cartesian coordinates and possible pose estimated through kinematics and two re-solver measurements

show the full state of the multibody system at the configuration and velocity levels compared with the estimations provided by the kinematics involving all the four resolver measurements (and denoted as $(FK)_{1,2,3,4}$). Figure 14 depicts through the grey area the possible positions that can be estimated through the forward kinematics exploiting just two angular measurements and by assuming a CSPR configuration (i.e., the possible configurations that can be achieved by fixing two cable lengths and imposing that the end-effector cannot over-take the vertical upper limit defined by the exit point positions $\mathbf{a}_i$). A closer view is provided by Fig. 15 that details the estimation error in the three directions. The maximum and RMS estimation errors, respectively, that have been obtained, if compared with the kinematic estimation with three or four angular measurements, are: for e_x 8.2e-3 m and 1.8e-3 m; for e_y 12.5e-3 m and 2.9e-3 m; for e_z 11.9e-3 m and 1.7e-3 m. According to the uncertainty in the forward kinematics and in the bands defined through the standard deviations, estimation error is detected when the estimated pose goes beyond such intervals.

All these results show that accurate estimation is achieved through just two angular measurements, thus overcoming the indeterminacy of kinematics estimation.

Finally, Fig. 16 shows the torques retrieved by the servodrive current loop, by means of the nominal torque constant. It is interesting to note that such signals are quite noisy, and therefore the EKF should be able to filter these high-frequency components. Additionally, even if the initial and final end-effector positions are the same, the measured torques are quite different because of the hysteretic behavior of the Coulomb friction. Finally, keeping in mind that Coulomb friction can be up to 0.21 Nm, it is evident that it plays a very significant role in the overall exerted torques.

5.6 2nd test: spiral trajectory

The second trajectory has been chosen because in CDPRs the cable tensions are strongly dependent on the height of the end-effector with respect to the exit point positions. This test accounts for an ascending spiral, Fig. 17, which evaluates a wide range of motor torques. Two rest-to-rest motions along a line are performed at the beginning and at the end of the
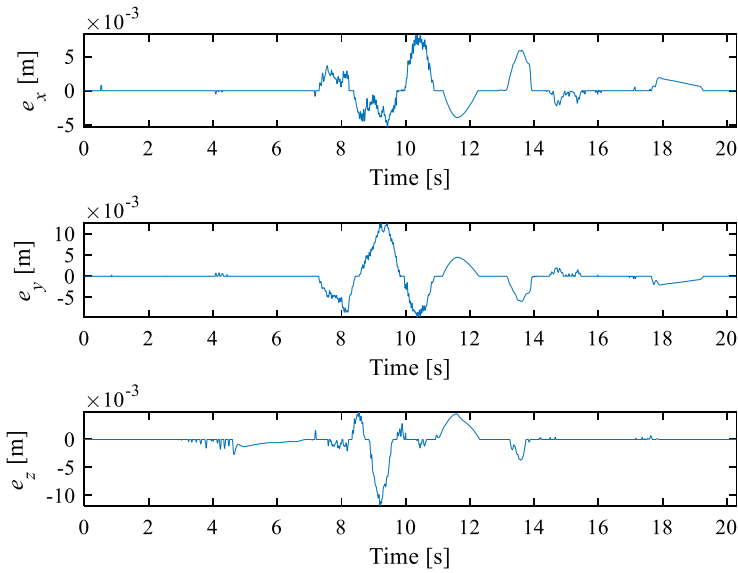
Fig. 15 Time evolution of the estimation errors
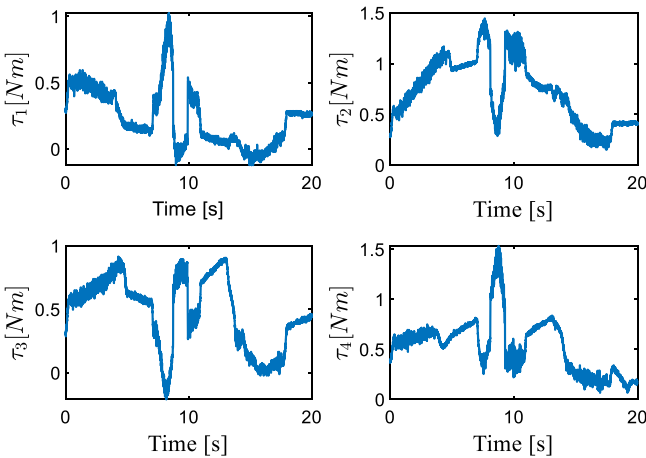

Fig. 16 Motor torques

test, from the rest pose (corresponding to the home position) to the initial position of the path *vice versa* at the end.

Figures 18 and 19 show the full state of the multibody system at the configuration and velocity levels and compared with the estimations provided by the kinematics involving all the four resolver measurements (and denoted as $(FK)_{1,2,3,4}$). Figure 20 depicts through the grey area the possible positions that can be estimated through the forward kinematics exploiting just two angular measurements. The estimation errors in the three directions are shown in Fig. 21. The maximum and RMS estimation errors, respectively, that have been obtained, if compared with the kinematic estimation with three or four angular measurements are: for e_x
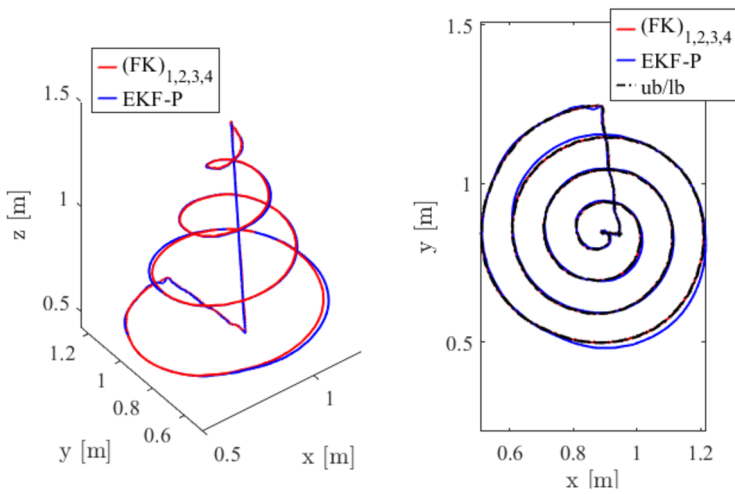
Fig. 17 Three-dimensional and xy-plane view of the end-effector's trajectory
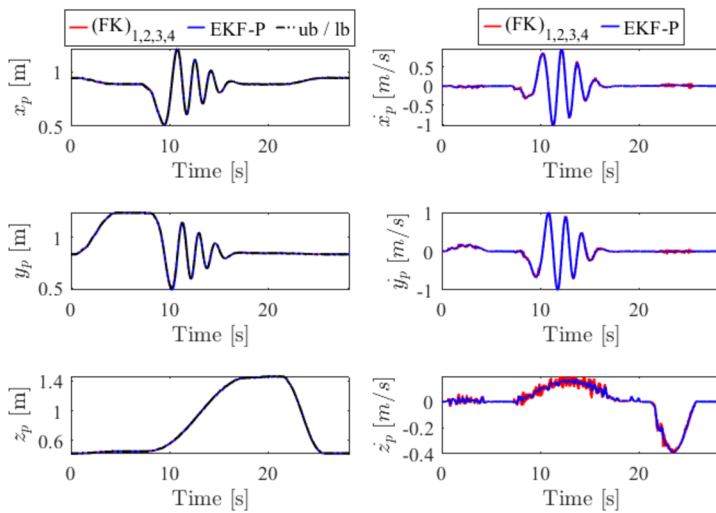
Fig. 18 End-effector's Cartesian positions and velocities

$6e$-3 m and $1.9e$-3 m; for e_y $16.8e$-3 m and $3.2e$-3 m; for e_z $2.6e$-3 m and $4e$-4 m. All these results show that accurate estimation is achieved through just two angular measurements, thus overcoming the indeterminacy of kinematics estimation.

6 Conclusions

This work addresses the development of nonlinear state observers for CDPRs. The theory developed can be applied to any CDPR topology and merges the advantages provided by EKFs and redundant coordinate dynamic modeling. To identify the observer design best
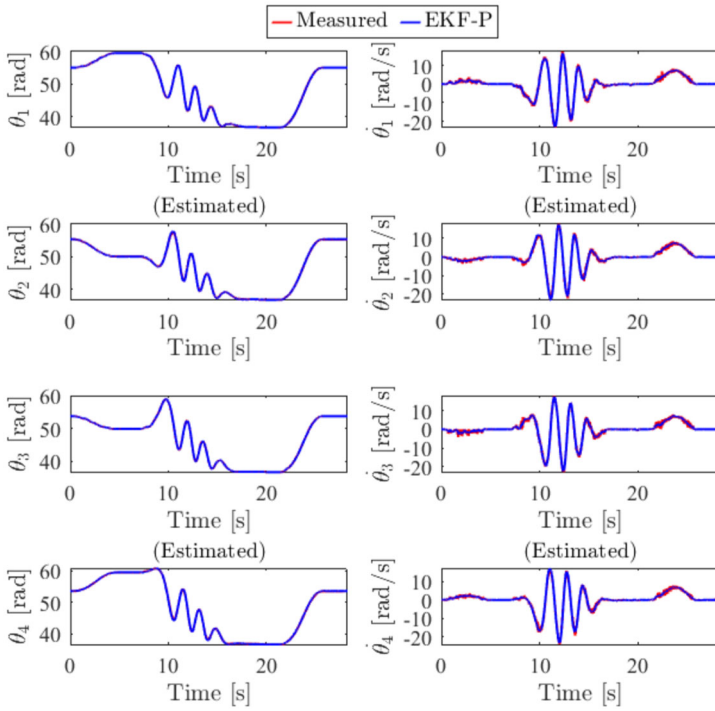
Fig. 19 Motor angular positions and velocities
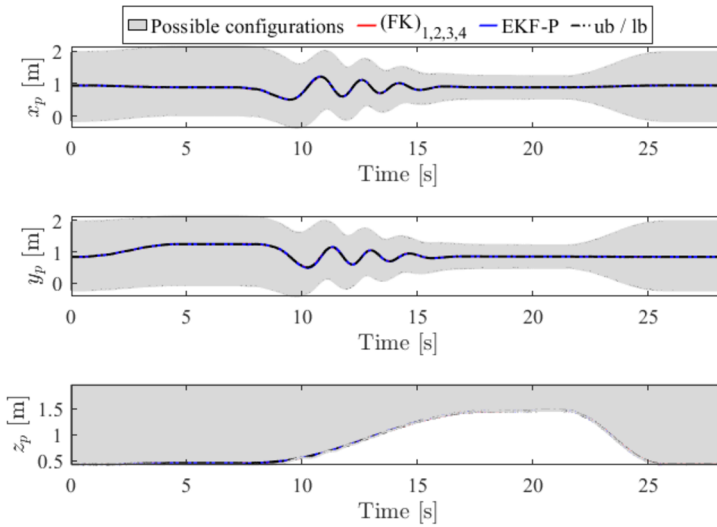
Fig. 20 End-effector's Cartesian coordinates and possible pose estimated through kinematics and two resolver measurements
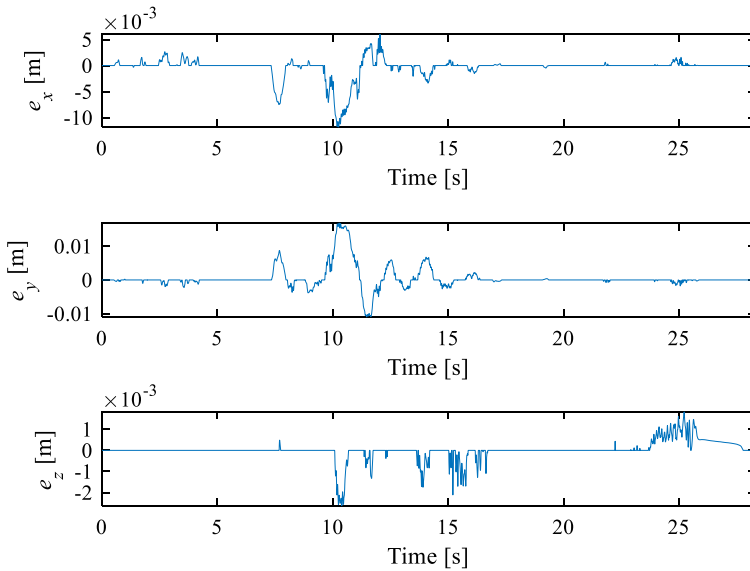
Fig. 21 Time evolution of the estimation errors

fitting CDPRs, the synthesis of three alternative formulations of EKFs is first discussed and compared through numerical simulations, which allow knowing the actual state. Three sound and well-known strategies to obtain ODEs from DAEs in multibody models are adopted, and a dedicated numerical investigation is carried out: the penalty formulation, the Udwadia–Kalaba formulation, and the Udwadia–Kalaba–Phohomsiri formulation. The results achieved prove that very similar results can be obtained if both the model (through some parameters adopted to enforce constraints) and the EKF (through the model and sensor covariances) are properly tuned. The analysis of the computational effort shows that penalty formulation requires slightly smaller CPU time, although two other formulations require CPU time that is remarkably smaller than the sample time adopted.

An experimental investigation is then carried out to prove the effectiveness of the observer in the very challenging effort to the real time estimation of the full state of a point-mass cable suspended robot by relying on the knowledge of the angular positions of two out of four winches. The results achieved in terms of predicted position and velocity of the end-effector are extremely satisfactory and show how EKF can be used to perform sensor fusion by exploiting sensor measurements and a dynamic model. The use of industrial grade components for implementing the observer (industrial PLC and programming environment) and of the actuators that impose coping with relevant friction provides an additional prove of the applicability of the strategy to today's CDPRs.

ECS_00000043). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

Declarations

Competing Interests The authors declare no conflict of interest.

References

1. Bettega, J., Piva, G., Richiedei, D., Trevisani, A.: Model predictive control for path tracking in cable driven parallel robots with flexible cables: collocated vs. noncollocated control. Multibody Syst. Dyn. **58**, 47–81 (2023). https://doi.org/10.1007/s11044-023-09881-0
2. Bettega, J., Richiedei, D., Trevisani, A.: Using pose-dependent model predictive control for path tracking with bounded tensions in a 3-DOF spatial cable suspended parallel robot. Mach. **10**, 453 (2022). https://doi.org/10.3390/machines10060453
3. Trevisani, A.: Planning of dynamically feasible trajectories for translational, planar, and underconstrained cable-driven robots. J. Syst. Sci. Complex. **26**, 695–717 (2013). https://doi.org/10.1007/s11424-013-3175-1
4. Boschetti, G., Minto, R., Trevisani, A.: Experimental investigation of a cable robot recovery strategy. Robotics **10**, 1–18 (2021). https://doi.org/10.3390/robotics10010035
5. Caracciolo, R., Richiedei, D., Trevisani, A.: Experimental validation of a model-based robust controller for multi-body mechanisms with flexible links. Multibody Syst. Dyn. **20**, 129–145 (2008). https://doi.org/10.1007/s11044-008-9113-7
6. Palomba, I., Richiedei, D., Trevisani, A.: Kinematic state estimation for rigid-link multibody systems by means of nonlinear constraint equations. Multibody Syst. Dyn. **40**, 1–22 (2017). https://doi.org/10.1007/s11044-016-9515-x
7. Pastorino, R., Richiedei, D., Cuadrado, J., Trevisani, A.: State estimation using multibody models and non-linear Kalman filters. Int. J. Non-Linear Mech. **53**, 83–90 (2013). https://doi.org/10.1016/j.ijnonlinmec.2013.01.016
8. Abbasnejad, G., Carricato, M.: Direct geometrico-static problem of underconstrained cable-driven parallel robots with n cables. IEEE Trans. Robot. **31**, 468–478 (2015). https://doi.org/10.1109/TRO.2015.2393173
9. Mishra, U.A., Caro, S.: Forward kinematics for suspended under-actuated cable-driven parallel robots with elastic cables: a neural network approach. J. Mech. Robot. **14**, 041008 (2022). https://doi.org/10.1115/1.4054407
10. Gutierrez, I., Collado, J.: Bouncing behaviour in the Kapitsa pendulum. In: 2016 IEEE 55th Conference on Decision and Control, CDC 2016 (2016)
11. Heyden, T., Woernle, C.: Dynamics and flatness-based control of a kinematically undetermined cable suspension manipulator. Multibody Syst. Dyn. **16**, 155–177 (2006). https://doi.org/10.1007/s11044-006-9023-5
12. González, F., Kövecses, J.: Use of penalty formulations in dynamic simulation and analysis of redundantly constrained multibody systems. Multibody Syst. Dyn. **29**, 57–76 (2013). https://doi.org/10.1007/s11044-012-9322-y
13. Udwadia, F.E., Kalaba, R.E.: A new perspective on constrained motion. Proc. R. Soc. A, Math. Phys. Eng. Sci. **439**, 407–410 (1992)
14. Udwadia, F.E., Kalaba, R.E., Phohomsiri, P.: Mechanical systems with nonideal constraints: explicit equations without the use of generalized inverses. J. Appl. Mech. Trans. ASME **71**, 615–621 (2004). https://doi.org/10.1115/1.1767844

15. Yang, L., Xue, S., Yao, W.: Application of Gauss principle of least constraint in multibody systems with redundant constraints. Proc. Inst. Mech. Eng., Part K, J. Multi-Body Dyn. **235**, 150–163 (2020). https://doi.org/10.1177/1464419320975301

16. Mattioni, V., Idà, E., Carricato, M.: Force-distribution sensitivity to cable-tension errors: a preliminary investigation. In: Cable-Driven Parallel Robots. Mechanisms and Machine Science (2021)

17. Londi, F., Pennestrì, E., Valentini, P.P., Vita, L.: Control and virtual reality simulation of tendon driven mechanisms. Multibody Syst. Dyn. **12**, 133–145 (2004). https://doi.org/10.1023/B:MUBO.0000044419.83366.a9

18. de Jalon, J.G., Bayo, E.: Kinematic and Dynamic Simulation of Multibody Systems (1994)

19. Bayo, E., Ledesma, R.: Augmented Lagrangian and mass-orthogonal projection methods for constrained multibody dynamics. Nonlinear Dyn. **9**, 113–130 (1996). https://doi.org/10.1007/BF01833296

20. Pastorino, R., Richiedei, D., Cuadrado, J., Trevisani, A.: State estimation using multibody models and non-linear Kalman filters. Int. J. Non-Linear Mech. **53**, 83–90 (2013). https://doi.org/10.1016/J.IJNONLINMEC.2013.01.016

21. Cuesta, C., Luque, P., Mántaras, D.A.: State estimation applied to non-explicit multibody models. Nonlinear Dyn. **86**, 1673–1686 (2016). https://doi.org/10.1007/s11071-016-2985-9

22. González, F., Masarati, P., Cuadrado, J., Naya, M.A.: Assessment of linearization approaches for multibody dynamics formulations. J. Comput. Nonlinear Dyn. **12**, 041009 (2017). https://doi.org/10.1115/1.4035410

23. Marques, F., Souto, A.P., Flores, P.: On the constraints violation in forward dynamics of multibody systems. Multibody Syst. Dyn. **39**, 385–419 (2017). https://doi.org/10.1007/s11044-016-9530-y

24. Cuadrado, J., Dopico, D., Barreiro, A., Delgado, E.: Real-time state observers based on multibody models and the extended Kalman filter. J. Mech. Sci. Technol. **23**, 894–900 (2009). https://doi.org/10.1007/s12206-009-0308-5

25. de Falco, D., Pennestrì, E., Vita, L.: The Udwadia-Kalaba formulation: a report on its numerical efficiency in multibody dynamics simulations and on its teaching effectiveness. In: Multibody Dynamics, ECCOMAS Thematic Conference, pp. 21–24 (2005)

26. Udwadia, F.E., Phohomsiri, P.: Explicit equations of motion for constrained mechanical systems with singular mass matrices and applications to multi-body dynamics. Proc. R. Soc. A, Math. Phys. Eng. Sci. **462**, 2097–2117 (2006). https://doi.org/10.1098/rspa.2006.1662

27. Liu, C.Q., Huston, R.L.: Another form of equations of motion for constrained multibody systems. Nonlinear Dyn. **51**, 465–475 (2008). https://doi.org/10.1007/s11071-007-9225-2

28. Fumagalli, A., Masarati, P.: Efficient application of Gauss' principle to generic mechanical systems. Proc. Inst. Mech. Eng., Part K, J. Multi-Body Dyn. **223**, 121–131 (2009)

29. de Falco, D., Pennestrì, E., Vita, L.: Investigation of the influence of pseudoinverse matrix calculations on multibody dynamics simulations by means of the Udwadia-Kalaba formulation. J. Aerosp. Eng. **22**, 365–372 (2009)

30. Golub, G.H., Van Loan, C.F.: Matrix Computations. JHU Press (2013)

31. Pappalardo, C.M., Guida, D.: A comparative study of the principal methods for the analytical formulation and the numerical solution of the equations of motion of rigid multibody systems. Arch. Appl. Mech. **88**, 2153–2177 (2018). https://doi.org/10.1007/s00419-018-1441-3

32. Pappalardo, C.M., Guida, D.: On the computational methods for solving the differential-algebraic equations of motion of multibody systems. Mach. **6**, 20 (2018). https://doi.org/10.3390/machines6020020

33. Baumgarte, J.: Stabilization of constraints and integrals of motion in dynamical systems. Comput. Methods Appl. Mech. Eng. **1**, 1–16 (1972). https://doi.org/10.1016/0045-7825(72)90018-7

34. Flores, P., MacHado, M., Seabra, E., Tavares Da Silva, M.: A parametric study on the Baumgarte stabilization method for forward dynamics of constrained multibody systems. J. Comput. Nonlinear Dyn. **6**, 011019 (2011). https://doi.org/10.1115/1.4002338

35. Sanjurjo, E., Blanco, J.L., Torres, J.L., Naya, M.A.: Testing the efficiency and accuracy of multibody-based state observers. In: Proceedings of the ECCOMAS Thematic Conference on Multibody Dynamics 2015, Multibody Dynamics 2015 (2015)

36. Naya, M.Á., Sanjurjo, E., Rodríguez, A.J., Cuadrado, J.: Kalman filters based on multibody models: linking simulation and real world. A comprehensive review. Multibody Syst. Dyn. **58**, 479–521 (2023)

37. Simon, D., Chia, T.L.: Kalman filtering with state equality constraints. IEEE Trans. Aerosp. Electron. Syst. **38**, 128–136 (2002). https://doi.org/10.1109/7.993234

38. Grewal, M.S., Andrews, A.P.: Kalman Filtering: Theory and Practice Using MATLAB, 3rd edn. (2008)

39. Ros, J., Yoldi, R., Plaza, A., Ángeles, J.: Exponential integration schemes in multibody dynamics. In: Proceedings of the Second Joint International Conference on Multibody System Dynamics, Stuttgart, Germany (2012)

40. Kalman, R.E.: A new approach to linear filtering and prediction problems. J. Fluids Eng. Trans. ASME **82**, 35–45 (1960). https://doi.org/10.1115/1.3662552

41. Listmann, K.D., Zhao, Z.: A comparison of methods for higher-order numerical differentiation. In: 2013 European Control Conference, ECC 2013 (2013)

42. Khalilpour, S.A., Bourbour, A., Khorrambakht, R., Kariminasab, S., Taghirad, H.D.: Forward kinematics resolution of a deployable cable robot. In: 5th RSI International Conference on Robotics and Mechatronics, IcRoM 2017, pp. 27–32 (2018)

43. Pappalardo, C.M., Guida, D.: On the use of two-dimensional Euler parameters for the dynamic simulation of planar rigid multibody systems. Arch. Appl. Mech. **87**, 1647–1665 (2017). https://doi.org/10.1007/s00419-017-1279-0

44. Shiang, W.-J., Cannon, D., Gorman, J.: Optimal force distribution applied to a robotic crane with flexible cables. In: Proceedings - IEEE International Conference on Robotics and Automation, pp. 1948–1954 (2000)

45. Idá, E., Merlet, J.-P., Carricato, M.: Automatic self-calibration of suspended under-actuated cable-driven parallel robot using incremental measurements. In: Cable-Driven Parallel Robots (2019)

46. Marques, F., Flores, P., Pimenta Claro, J.C., Lankarani, H.M.: A survey and comparison of several friction force models for dynamic analysis of multibody mechanical systems. Nonlinear Dyn. **86**, 1407–1443 (2016). https://doi.org/10.1007/s11071-016-2999-3

47. Do, N.B., Ferri, A.A., Bauchau, O.A.: Efficient simulation of a dynamic system with LuGre friction. J. Comput. Nonlinear Dyn. **2**, 281–289 (2007). https://doi.org/10.1115/1.2754304

48. Rojas, S., Pérez, M.A., Rodríguez, J., Zelaya, H.: Torque ripple modeling of a permanent magnet synchronous motor. In: 2010 IEEE International Conference on Industrial Technology, pp. 433–438. IEEE (2010)

Authors and Affiliations

Jason Bettega[1] · Giovanni Boschetti[1] · Borja R. Frade[2] · Francisco González[2] · Giulio Piva[1] · Dario Richiedei[1] · Alberto Trevisani[1]

✉ G. Piva
giulio.piva@phd.unipd.it

J. Bettega
jason.bettega@phd.unipd.it

G. Boschetti
giovanni.boschetti@unipd.it

B.R. Frade
borja.rfrade@udc.es

F. González
f.gonzalez@udc.es

D. Richiedei
dario.richiedei@unipd.it

A. Trevisani
alberto.trevisani@unipd.it

[1] Department of Management and Engineering, University of Padova, Stradella S. Nicola 3, 36100, Vicenza, Italy

[2] Laboratorio de Ingeniería Mecánica, University of A Coruña, Rúa Mendizábal s/n, 15403, Ferrol, Spain