robotics

MDPI

Article

Experimental Investigation of a Cable Robot Recovery Strategy

Giovanni Boschetti *, Riccardo Minto and Alberto Trevisani

Department of Management and Engineering (DTG), University of Padova, 36100 Vicenza, Italy; riccardo.minto@phd.unipd.it (R.M.); alberto.trevisani@unipd.it (A.T.)

* Correspondence: giovanni.boschetti@unipd.it; Tel.: +39-0444-99-8748

Abstract: Developing an emergency procedure for cable-driven parallel robots is not a trivial process, since it is not possible to halt the end-effector by quickly braking the actuators as in rigid-link manipulators. For this reason, the cable robot recovery strategy is an important topic of research, and the literature provides several approaches. However, the computational efficiency of the recovery algorithm is fundamental for real-time applications. Thus, this paper presents a recovery strategy adopted in an experimental setup consisting of a three degrees-of-freedom (3-DOF) suspended cable robot controlled by an industrial PC. The presentation of the used control system lists the industrial-grade components installed, further highlighting the industrial implication of the work. Lastly, the experimental validation of the recovery strategy proves the effectiveness of the work.

Keywords: CDPR; cable failure; recovery strategy; TwinCAT; motion tracking

check for updates

Citation: Boschetti, G.; Minto, R.; Trevisani, A. Experimental Investigation of a Cable Robot Recovery Strategy. *Robotics* **2021**, *10*, 35. https://doi.org/10.3390/robotics10010035

Received: 7 January 2021
Accepted: 12 February 2021
Published: 16 February 2021

1. Introduction

Cable-driven parallel robots can be easily described as parallel robots whose rigid links are replaced by flexible cables. As parallel robots, they present several advantages with respect to serial ones because of their particular structure: thanks to their low inertial properties, the end-effector can reach high speed, ad they can manipulate high-payloads since the payload is distributed among the links [1].

One of the shortcomings faced when adopting a parallel robot is usually the limited workspace, due to the possible collisions between links and to the presence of passive joints which have mechanical limits [2]. Cable-driven parallel robots, hereafter called cable robots, provide a solution to this problem by replacing the links with cables wound around winches, leading immediately to a larger workspace thanks to the cable length. Moreover, they are relatively cheap, and their simple design allows for modular transportable and reconfigurable solutions [3]. Therefore, it is easy to understand the importance of cable robots, which are adopted not only for industrial applications [4] and entertainment [5], but even for home assistance and rehabilitation purposes [6], since they are non-invasive in comparison to other robots.

However, in contrast to rigid-link parallel robots, it is fundamental that the cable tension remains positive and bounded during the motion to avoid cable slackness or failure since cables can only exert tension. One possible solution is to use an external force acting on the robot moving platform to satisfy this condition. This is possible by adopting a redundant—also called overconstrained—structure [7]; i.e., a robot composed of m cables, with m greater than the n degrees of freedom. Otherwise, suspended parallel robots, which are considered in this work, take advantage of gravity to pull the cable. Despite their design being simpler than overconstrained robots (since they require fewer cables and actuators), they are more difficult to control.

Furthermore, the unilateral properties of the wires also lead to safety concerns. Thus, the design of an emergency stop procedure is more difficult than in rigid-link manipulators. Indeed, whereas for conventional robots it is usually possible to halt the end-effector and the robot structure by braking the actuators, this does not apply to cable robots: indeed, braking the actuators is usually not sufficient to stop the end-effector. Conversely, a sudden motor brake could generate uncontrollable motions [8]. An initial analysis of cable robot failures is presented in [9], and intuitively, the most expected failure for these robots is cable breakage. Despite following norms and guidelines that allow the reliable operation of cable robots to be ensured, faulty mechanical components or control algorithms could lead to slack or breaking cables [10]. In [11], the authors suggested doubling each cable with a passive auxiliary cable as a safety solution. Since this solution adds complexity to the system, a further proposal was to use a single auxiliary cable that raises the end-effector in case of failure by releasing a counterweight. The authors noted that the influence of the counterweight mass is not negligible, adding that it may difficult to determine a proper value. Indeed, a light counterweight may not raise the end-effector, whereas a heavier one may lead to an excessive overshoot of the $z-$position.

Another solution could be to use healthy cables to recover a lost wrench, as seen in [12]. Indeed, when a cable fails, the wrench applied by the end-effector is different from the expected one. Thus, Notash proposed two methods for recovering a lost wrench by imposing an additive tension on the remaining cables. However, this work considers that the robot remains overconstrained after failure, and thus it does not take into account any changes in the static equilibrium workspace (SEW), nor that the end-effector could lay outside it.

Indeed, the SEW—i.e., the set of end-effector poses for which static equilibrium can be achieved with positive tension in all cables [13]—changes for underconstrained parallel robots when a cable failure occurs. Thus, it is necessary to bring the robot inside the new SEW to reach a static equilibrium. Moreover, even if it is not possible to stop the robot outside the SEW, the end-effector could be controlled outside it with positive cable tensions, as seen in [14], where the authors performed a dynamic motion reaching outside the SEW. In their work, the end-effector reached the target point at zero speed while maintaining a positive cable tension, despite presenting a non-null acceleration of the end-effector. A similar result was presented in [15], where the authors presented a geometric approach to define the workspace characterized by positive cable tensions. This approach was then used to design feasible trajectories, and the work shows examples of point-to-point and periodic trajectories. These works show that it is possible to move and control the end-effector outside the SEW. Moreover, they show that a major focus in recovery strategy for cable robots should be the definition of trajectories that drive and stop the end-effector inside the after-failure SEW.

Passarini et al. [16] presented a recovery strategy based on a single elliptical trajectory that was used to steer the end-effector to a safe point inside the new SEW. Moreover, this method could be designed to avoid collisions; this is possible by keeping the platform inside a cuboid and above a predetermined height. Since the landing point was identified by minimizing the recovery time, the authors provided two approximation strategies as a replacement of the multivariable constrained optimization for real-time applications. A different approach was presented in [17], where a recovery strategy aiming to drive the end-effector to a rest position inside the new SEW was divided into three phases. The trajectory was composed of an elliptical path, used to drive the platform inside the after-failure SEW, followed by two seventh-degree polynomial trajectories to stop the robot. This approach exploits elliptical trajectories to maintain positive cable tensions outside the SEW; moreover, it avoids the oscillatory motion through the polynomial trajectories, ensuring continuity in terms of position, velocity, acceleration and jerking. Boschetti et al. [18] proposed a recovery strategy based on two different phases: a connecting path to change the direction of motion and a linear trajectory designed to drive and stop the end-effector at the chosen rest positions. Moreover, the authors exploited the Wrench Exertion Capability (WEC) performance index [19] to evaluate the maximum braking forces which could be applied to the end-effector. In contrast to the previous approach, the adoption of a linear trajectory avoids large motions, reducing the risk of collisions during the recovery strategy.

However, this approach requires solving linear programming problems to evaluate the force limits on the end-effector, which are time-consuming or must be precomputed.

Despite the literature presenting approximate methods for industrial applications, the feasibility of the proposed recovery strategies must be assessed in real time. For this reason, Boumann et al. [20] investigated the computation time and real-time suitability of the method presented in [21]. The authors adopted an alternative method based on geometrical analysis within the cable force space since the previous instance of the method was based on the resolution of an optimization problem. Tests on the algorithm computational efficiency showed that it needed about 10% of the cycle time for a 2 kHz control frequency.

From our review, the literature can be seen to propose several methods to solve cable failure in cable robot applications; however, very few of the approaches present an experimental validation besides simulations. Despite the literature often accompanying each recovery strategy with an equivalent fast approach, these have not been tested in a real prototype, and thus their suitability for real-world application is not ensured.

The proposed work aims to introduce a recovery strategy that is suitable for industrial applications without adopting time-consuming algorithms. This recovery strategy starts from a previously defined approach that features linear programming optimization; thus, it may not be suitable for real-time industrial applications. Indeed, an effective recovery strategy should avoid collision and provide for a fast recovery of the end-effector even when its computational efficiency is critical and in the presence of only integrated sensors, such as encoders or resolvers. Therefore, this work tests the feasibility of the proposed recovery strategy on a prototype of a three degrees-of-freedom (3-DOF) cable-suspended parallel robot; moreover, this experimental setup is composed of industrial-grade components, further highlighting the practical implication of the strategy.

This work is structured as follows: Section 2 presents the proposed control model and recovery strategy; Section 3 presents the experimental setup used for the tests; Section 4 presents the results, comparing the nominal trajectory with measurements obtained by means of a motion tracking system; and lastly, Section 5 concludes the work.

2. Control Model

This work considers a suspended cable-driven parallel robot (CDPR) with m cables, with each controlled by an actuator. The end-effector is connected to the frame by means of the m cables through the exit points B_i to the attach points P_i. We consider two reference frames: the absolute frame (O) and the local frame (P), which is placed in the center of mass of the platform, as seen in Figure 1. The length of the cables is represented by the vector $\mathbf{l}$

$$\mathbf{l} = [l_i] \tag{1}$$

where l_i is the length of the i-th cable. A generic platform presents six degrees-of-freedom (6-DOF):

$$\mathbf{d} = [x, y, z, \alpha, \beta, \gamma]^T = [\mathbf{p}, \boldsymbol{\theta}]^T \tag{2}$$

where $\mathbf{d}$ represents the position $\mathbf{p}$ and orientation $\boldsymbol{\theta}$ of the platform. In this work, the end-effector was considered as a point mass since a recovery strategy generally focuses on the end-effector position. Thus, only the vector $\mathbf{p}$, and thus the first three DOFs, have been considered. Moreover, this work studies redundant cable robots characterized by $m \geq n + 1$. Since this work takes into account a point mass end-effector, n is equal to 3. Lastly, the proposed recovery strategy applies only if the after-failure architecture is at least fully actuated and the after-failure feasible workspace is not null. This last condition can be guaranteed by at least three cables whose exit points do not lay on the same vertical plane. Therefore, this work applies to over-actuated spatial cable robots; i.e., robots with at least four cables.

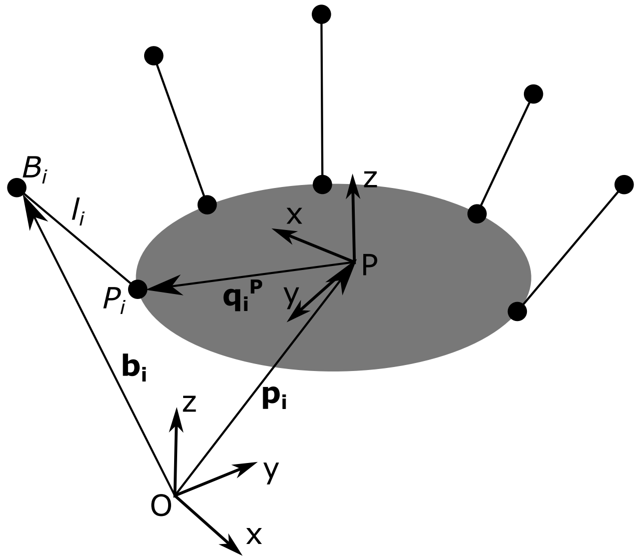


Figure 1. Example of a generic suspended cable-driven parallel robot.

2.1. Motion Control

2.1.1. Position Control

A successful recovery strategy requires that the end-effector follows the evaluated trajectory; thus, a position control algorithm should be introduced. In this paper, a PID controller on the position has been used. This work adopted a PID controller due to its simple structure, avoiding the use of more complex and time-consuming algorithms. Moreover, the results were deemed suitable, as presented in Section 4; thus, other strategies were not investigated. However, other controllers may be adopted.

Once the desired trajectory has been defined, each pose is sent to the position control algorithm. To obtain the cable length, the following inverse kinematic approach was adopted, based on the closure equation of the vectors in Figure 1. At first, we define the vector $\mathbf{q_i^P}$ as the positions of the distal attachment points on the platform in the local reference frame; i.e.,

$$\mathbf{q_i^P} = [q^P_{i,x}, q^P_{i,y}, q^P_{i,z}] \tag{3}$$

The rotation matrix $\mathbf{R_P^O}$ represents the orientation of the local reference frame in the global frame; thus, it is possible to evaluate the cable length l_i from the geometry as

$$l_i = ||\mathbf{p} + \mathbf{R_P^O}\mathbf{q_i^P} + \mathbf{b_i}|| \tag{4}$$

Since this work models the end-effector as a point mass, $\mathbf{q_i^P}$ is null, simplifying the equation.

From l_i it is possible to evaluate the actuator angular position θ_i as

$$l_i = l_{i,0} + \rho\theta_i \tag{5}$$

where $l_{i,0}$ is the distance between the actuator and the corresponding exit point B_i, which is set as constant, and ρ is the drum radius.

The position loop compares the position feedback with the actuator sensors to evaluate the error value between the actual position of the actuators and the reference position. The adoption of the inverse kinematic model allowed us to easily compare the actuator feedback, which is defined in the joint space, with the desired position of the end-effector, which is defined in the Cartesian space.

2.1.2. Feed-Forward Control

When studying cable robots, it is necessary to ensure a positive cable tension to control the end-effector. Thus, it is necessary to take into account the system dynamics, which are not completely described by the independent joint control based on position [22]. However, a control system based only on a feed-forward position control model that takes into account the cable tensions is not an optimal solution. Indeed, a similar approach was not deemed suitable, as it could not maintain positive cable tensions under disturbance, such as the end-effector oscillatory motion. Thus, in this work, we combined a PID position control algorithm, whose output was the corrective motor torque $\boldsymbol{\tau}_{PID}$, with an inverse dynamic feed-forward approach that evaluated the motor torque $\boldsymbol{\tau}_{FF}$, achieving the overall motor torque $\boldsymbol{\tau}$, as shown in Figure 2.

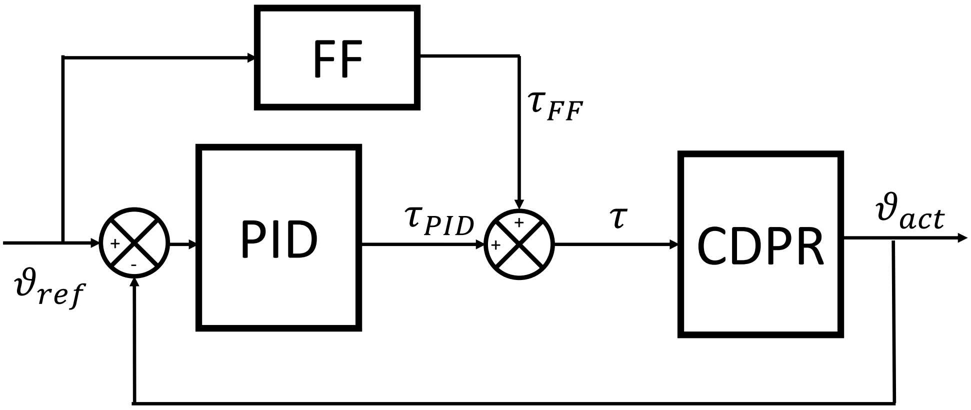


Figure 2. PID and feed-forward control scheme of the robot.

The tension of each cable can be evaluated by considering the wrench $\mathbf{w}$; i.e., the forces applied to the end-effector by the cables ($\mathbf{w}_c$) and by other external forces ($\mathbf{w}_e$). In other words, the overall wrench can be defined as follows:

$$\mathbf{w} = \mathbf{w}_c + \mathbf{w}_e = m\mathbf{a} \tag{6}$$

The relation between the cable wrench $\mathbf{w}_e$ and the cable tension $\mathbf{t}$ is given by the structure matrix $\mathbf{S}$, which is defined as follows:

$$\mathbf{S} = [\mathbf{u}_1, \ldots, \mathbf{u}_m] \tag{7}$$

where $\mathbf{u}_i$ are the unit vectors oriented from the end-effector towards the corresponding exit point B_i. The structure matrix allows the following equation to be defined:

$$\mathbf{w}_c = \mathbf{St} \tag{8}$$

An interesting approach to take into account the overall wrench (i.e., cable wrench and external wrench) with respect to cable tension $\mathbf{t}$ is given by the wrench matrix $\mathbf{W}$. This matrix is composed of matrix $\mathbf{S}$ with the vector $\mathbf{w_e}$ added as the column m+1:

$$\mathbf{W} \ = [\mathbf{S}, \mathbf{w}_e] \tag{9}$$

In this way, the overall wrench exerted on the end-effector can be defined as follows:

$$\mathbf{w} = \mathbf{St} + \mathbf{w}_e = \mathbf{W}\{\mathbf{t}^T, 1\}^T \tag{10}$$

However, in this application, the external wrench considers only the force of gravity. Thus, the first part of Equation (10) can be rewritten as follows:

$$\mathbf{w} - \mathbf{w}_e = \mathbf{St} = m\mathbf{a} - m\mathbf{g} \tag{11}$$

Since the before-failure cable robot architecture is overconstrained—i.e., redundant—**S** is an nxm rectangular matrix, and it is therefore not possible to evaluate its inversion; however, it is possible to evaluate **t** by adopting the Moore–Penrose generalized inverse $\mathbf{S}^+$:

$$\mathbf{t} = \mathbf{S}^+(\mathbf{w} - \mathbf{w}_e) \tag{12}$$

Conversely, in the case of a fully actuated cable robot (i.e., $m = n$), **S** is a square matrix, and therefore its inverse matrix can be computed.

Once the proper cable tensions have been computed, the required torques $\boldsymbol{\tau}_{FF}$ that have to be exerted by the motors can be defined as follows:

$$\boldsymbol{\tau}_{FF} = \mathbf{I}\ddot{\boldsymbol{\theta}} + \mathbf{R}\mathbf{t} \tag{13}$$

where **I** is the motor inertia matrix, $\ddot{\boldsymbol{\theta}}$ is the motor acceleration given by the inverse kinematics and **R** contains the pulley radii in its diagonal. Nonlinear terms have been neglected.

2.2. Recovery Strategy

Among the recovery strategies proposed by the literature, this work will focus on the one presented in [23]. The goal of this strategy consists of driving the end-effector towards a safe point inside the residual SEW. Indeed, as the cable breaks, the end-effector could lay outside the static workspace, and so a pose, called SP, must be accordingly defined inside the residual SEW. Previous works have suggested several factors that influence the SP position, such as the robot pose at failure or the presence of obstacles in the workspace. In this work, the main focus was to keep the end-effector above a suitable height, defined to avoid a collision, whereas the x, y coordinates of the SP were defined as the barycenter of the area obtained by the intersection of the residual SEW with the horizontal plane defined at the chosen height.

Once the SP is defined, the strategy aims to drive the end-effector towards this point. However, to generate a feasible linear trajectory, the end-effector velocity should be null or already oriented towards the SP. Therefore, before starting the linear path, it is necessary to change the direction of motion with an initial phase called a connecting path.

A mobile reference frame placed at the center of mass of the end-effector is then defined. This frame is used to readily identify the velocity components that must be reduced to a suitably small value. The mobile $x-$axis is aligned with the vector joining the end-effector with the SP, hereafter called **d**. The $y-$axis is instead aligned with the undesired velocity component. The velocity of the end-effector expressed on the mobile reference frame $\mathbf{v}'$ is therefore

$$\mathbf{v}' = \begin{bmatrix} v'_x \\ v'_y \\ 0 \end{bmatrix} \tag{14}$$

where the third component is always null due to the axis definition and v'_y is always positive, whereas v'_x depends on the initial motion, as seen in Figure 3.

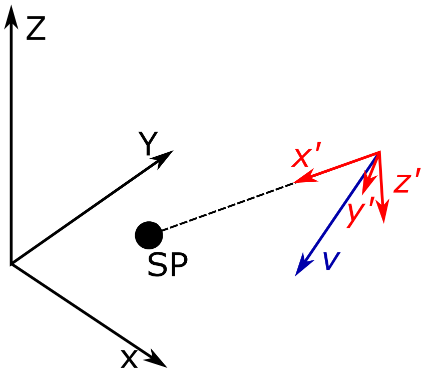


Figure 3. Three-dimensional view of the velocity components in the mobile reference system.

The corresponding braking force used to minimize the v'_y component while keeping the $z-$component null is defined as

$$\mathbf{f}' = \begin{bmatrix} f'_x \\ f'_y \\ f'_z \end{bmatrix} \text{ with } \begin{cases} f'_x > 0 \\ f'_y < 0 \\ f'_z = 0 \end{cases} \tag{15}$$

From Equation (11), it is possible to define the cable tensions **t** required to achieve this cable force as

$$\mathbf{f}' = \mathbf{R} \cdot \mathbf{f} = \mathbf{R} \cdot \mathbf{W} \cdot \mathbf{t} = \begin{bmatrix} \boldsymbol{w}^T_{Rx} \\ \boldsymbol{w}^T_{Ry} \\ \boldsymbol{w}^T_{Rz} \end{bmatrix} \mathbf{t_w} \tag{16}$$

where **R** is the rotation matrix between the mobile reference frame and the global frame, $\boldsymbol{w}^T_{Rx}$ represents the total force applied along the $x-$axis, and therefore **d**, whereas $\boldsymbol{w}^T_{Ry}$ and $\boldsymbol{w}^T_{Rz}$ represent the total force applied along the other two axes: $y-$ and $z-$, respectively.

The maximum tension t_w required to change the velocity direction can then be evaluated. This can be evaluated by solving a linear optimization problem or by iterative algorithms based on a fast formulation of the wrench [24], which may suitable when the computational efficiency of the recovery algorithm is not a critical issue.

However, despite previous approaches having been proven to be effective, this work aims to provide a recovery strategy that drives the end-effector to the SP and that is feasible for different industrial real-time applications. Thus, these algorithms cannot be expensive in terms of computational time. On the other hand, a simplified recovery strategy composed of only a linear path towards the SP may present discontinuities in the cable tensions, which could lead to further cable failures if the end-effector speed is high enough. Therefore, it is necessary to reduce the end-effector speed before starting the linear path planning.

A motion planning that is able to ensure that the z position of the end-effector remains above a certain minimum value is needed; thus, it is necessary that at least some cables sustain the platform weight. In this work, the cables adjacent to the broken one were chosen to maintain the end-effector at a proper height.

Therefore, the complete approach is divided into two phases:

- As the cable i fails, the cables $i-1$ and $i+1$ react by sustaining the platform weight; at the same time, the other remaining cables move the platform towards the SEW by keeping a constant tension **t** and thus lowering their length **l**. This allows for a feasible trajectory, since it is performed with constant **t**;
- When the end-effector speed is low enough, a linear trajectory towards the SP is planned, as mentioned above, thus stopping the end-effector where desired.

Despite the end-effector position and velocity not being known beforehand, it is however possible to define the acceleration limits and therefore plan a linear trajectory towards the SP. This is possible by exploiting the Wrench Exertion Capability (WEC) index [25] andevaluating the maximum and minimum breaking force achievable along the direction of **d**; thus, the acceleration is limited during the linear trajectory.

By referring to the previously defined mobile reference frame—i.e., with the $x-$axis aligned with **d**,

$$\mathbf{W_R} = \mathbf{R} \cdot \mathbf{W} = \begin{bmatrix} \boldsymbol{w}^T_{Rx} \\ \boldsymbol{w}^T_{Ry} \\ \boldsymbol{w}^T_{Rz} \end{bmatrix} \tag{17}$$

the WEC formulation in [19] requires a linear programming problem to be solved to evaluate the maximum exertable force $w_{f,d}$ along $\mathbf{d}$:

$$maximize : (w_{f,d} = \boldsymbol{w}_x^T\mathbf{t})$$
$$s.t. \begin{cases} \boldsymbol{w}_y^T\mathbf{t} = 0 \\ \boldsymbol{w}_z^T\mathbf{t} = 0 \\ 0 \leq \mathbf{t_{min}} \leq \mathbf{t} \leq \mathbf{t_{max}} \end{cases} \quad (18)$$

Due to the local nature of the WEC index, the approach requires the linear programming problem to be solved for a certain number of points, decided by compromising between the desired accuracy and the computational efficiency of the algorithm. Similar to the previous phase, it is possible to adopt the fast WEC formulation to evaluate the maximum breaking force $w_{f,d}$ and therefore the acceleration limits a_{lim} as

$$a_{lim} = \frac{w_{f,d}}{m} \quad (19)$$

It should be noted that this formulation does not consider the actuator torques; thus, it is necessary to add an additional step to verify that the torques corresponding to the evaluated acceleration limits satisfy the torque constraints.

Once a_{lim} has been defined, it is possible to plan the trajectory along $\mathbf{d}$. This work adopts a fifth-degree polynomial motion law since it is easy to implement and evaluate even in the case of real-time applications. On the other hand, it is only possible to constrain the acceleration on the end-points; therefore, it is necessary to later verify that the evaluated motion law respects the acceleration constraints [24].

The adopted motion law is defined on the basis of the following constraints:

$$\begin{cases} x(0) = x_0 \\ v(0) = v_0 \\ a(0) = a_0 \\ x(T) = 0 \\ v(T) = 0 \\ a(T) = 0 \end{cases} \quad (20)$$

where T is the total motion time and x_0 and v_0 are the position and the velocity, respectively, after the precedent phase. The value of the starting acceleration a_0 is set equal to the maximum possible acceleration at half of the distance, approximating the polynomial motion law to a triangular motion profile in the first design phase.

Since the computed acceleration, and therefore the motion law, is evaluated in the mobile reference frame, it is necessary to express it in the spatial reference frame, allowing the evaluation of the inverse kinematics and therefore the required motor torques:

$$\ddot{\mathbf{p}} = \mathbf{R}^T\ddot{\mathbf{p}}' \quad (21)$$

where $\ddot{\mathbf{p}}$ is the acceleration on the absolute reference system and $\ddot{\mathbf{p}}'$ is the acceleration in the mobile reference frame.

3. Experimental Setup

This section describes the system used to address the recovery strategy problem; i.e., a cable suspended parallel robot (CSPR) setup and a motion tracking system. The presented setup is composed of industrial-grade components, which were adopted to prove that the proposed method could be applied in real industrial applications.

3.1. Control System

To control the cable robot, a servo system, as represented in Figure 4 and composed by the followings, is required:

- a *controller*, an industrial PC;
- *servo drive*, Beckhoff AX5206;
- *motor*, B&R 8LVB23ee004LjFn00 servo gear motor;
- *sensor*, the resolver equipped in the actuator to observe its position.

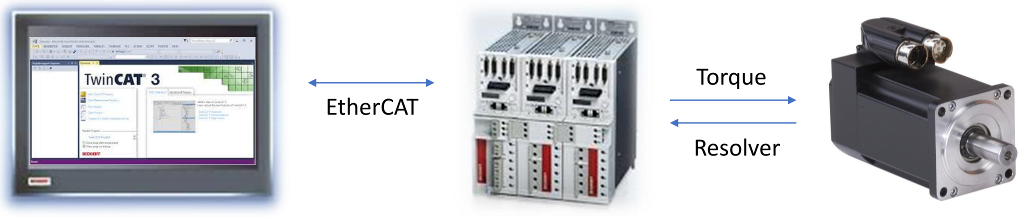


Figure 4. Servo system adopted in the presented work.

Usually, the control unit adopted in automated systems consists of a Programmable Logic Controller (PLC) [26] due to their reliability, reduced costs and guaranteed execution time. In this work, soft-PLCs have been considered; i.e., a PC as the hardware support platform and software to achieve the main functionality of traditional (hard) PLCs [27]. Indeed, this technology is currently preferable since it guarantees several benefits in comparison to hard PLC, such as open architecture, network communication capability and better data processing, as required by modern industrial systems. Moreover, a soft PLC leads to improved performance in comparison to traditional systems, achieving a faster response time [26].

In this work, the software TwinCAT 3 [28] was installed on a Windows 10 PC in kernel mode to turn it into a real-time controller. Since TwinCAT 3 is integrated into Microsoft Visual Studio, it is possible to program automation objects (modules) both using IEC 61131-3 and C/C++ programming languages. Moreover, by using the MATLAB/Simulink module, it is possible to import MATLAB Simulink projects inside TwinCAT. This approach has also been adopted in this work to import the control model previously presented since it provides several advantages. First of all, the adoption of Simulink makes it possible to use a higher-level software environment for the design of control systems directly based on calculation and control theory. Moreover, it also increases the system flexibility, since it is possible to simulate the recovery strategy in the MATLAB environment and subsequently validate it in the experimental setup. Lastly, despite MATLAB being a high-level programming language, TwinCAT is able to maintain a certain computational speed by building the Simulink block diagram in C++ code, thus making it suitable for real-time applications.

To communicate with the servo drives, TwinCAT provides for a built-in EtherCAT interface [29]. An EtherCAT connection differs from a traditional Ethernet connection as it handles the Ethernet data frame in an "on the fly" way; i.e., the standard packets are not interpreted at each node but based on a master–slave network where each slave only verifies and eventually inserts the required data in the frame. Thus, TwinCAT generates an EtherCAT master device, making use of a suitable physical Ethernet controller (e.g., Intel PRO100) from the industrial PC. The master controls the actuators by sending the proper message to the servo-drive based on the torques computed by the Simulink trajectory planner.

To achieve high-precision motion control, real-time control systems are needed; thus, the developed high-level motion control system requires low-level control systems focused on the actuator status, allowing a data flow within 250 µs and 1 ms.

For this reason, a servo drive is required to control the motor power, and in this work, the Beckhoff AX5206 Digital Compact Servo Drive with two-channels was adopted. This series of servo drives was deemed suitable for real-time application, with control loops in torque, speed and position with cycle times of 62.5 µs, 125 µs, and 250 µs, respectively. Furthermore, the AX5206 servo-drive presents two channels with different current intensities in the range from 1.5 A and 6 A per channel; since these are two separated channels with adaptable rated current, it is possible to control two different motors with a single servo drive, simplifying the control system of the cable robot. Further data of this servo drive are presented in Table 1.

Table 1. Characteristics of the adopted servo drive.

Drive Property	Value	Unit
Rated supply voltage	3 × 100 . . . 480	V (AC)
DC-link voltage (max)	875	V
Rated output current (channel)	6	A
Peak output current (channel)	13	A

The actuator should be designed in accordance with the servo drive system and the required task; thus, by means of static analysis, the actuator can be chosen in a rough design phase. Indeed, each motor should compensate the cable tension T:

$$T = \frac{m_e g}{4 sin(\alpha)} \tag{22}$$

where α is the angle between the wire and the horizontal plane; moreover, an angle between the cable and the vertical plane in the range of 0° and 45°, which is allowed by the rotating pulleys, should be considered. Thus, the worst scenario—i.e., an angle of 45°—should be considered for the static analysis, which means $sin(\alpha) = \sqrt{2}/2$. However, the considered recovery strategy requires that only two actuators sustain all the platform weight. Thus, the necessary torque is

$$\tau = \rho \frac{m_e g}{\sqrt{2}} \tag{23}$$

where ρ is the drum radius.

The characteristics of the adopted actuators are presented in Table 2. Moreover, the considered actuators are equipped with resolvers to monitor both the actuator position and velocity. The velocity data allow the implementation of the three terms of the PID control algorithm without deriving the position information; this is a great advantage since it is not necessary to define a filter required to define a filter for the position data, which is necessary for the derivate.

The presented system represents the setup of typical industrial applications fairly closely, since it has been designed with industrial-grade components. Moreover, the proposed control system does not require additional sensors and exploits those embedded in the actuators.

Table 2. Characteristics of the adopted actuators.

Actuator Property	**Value**	**Unit**
Nominal speed n_n	3000	rpm
Number of pole pairs	4	-
Nominal torque τ_n	1.3	Nm
Nominal current i_n	5.8	A
Stall torque τ_0	1.35	Nm
Stall current i_0	6	A
Maximum torque τ_{max}	4	Nm
Maximum current i_{max}	20.7	A
Maximum speed n_{max}	6600	rpm
Torque constant k_t	0.23	Nm/A
Voltage constant k_e	13.61	1000/rpm
Moment of inertia J	5.12×10^{-4}	kgm^2
Braking torque τ_{br}	2.2	Nm
Gear ratio I	4	-
Drum radius ρ	36	mm
Static friction torque	0.2	Nm
Friction coefficient b	0.0015	-

3.2. Robot Architecture

A prototype 3-DOF CSPR with four cables was developed in the Robotics and Automation Laboratory at the Department of Management and Engineering of the University of Padova, as seen in Figure 5.

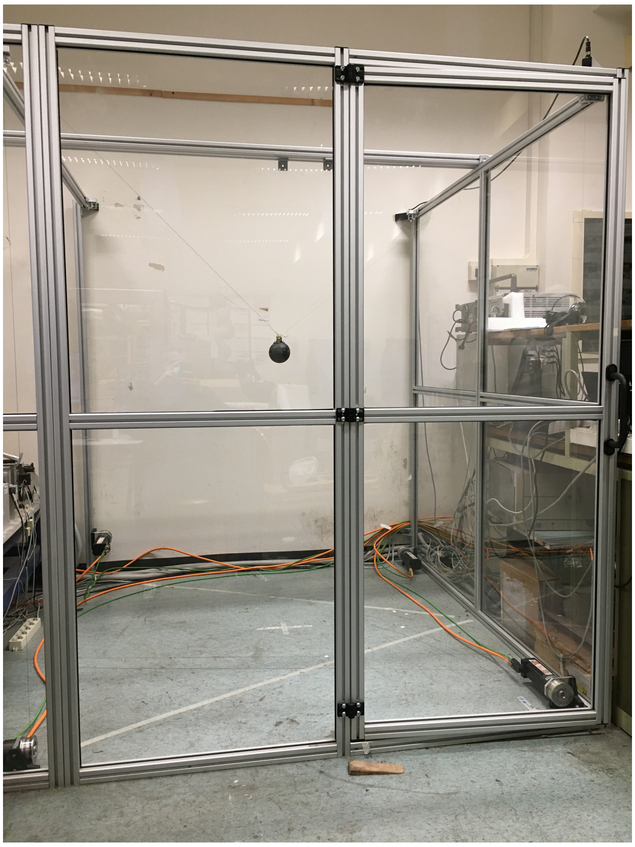


Figure 5. Prototype cable suspended parallel robot (CSPR) used during the tests.

The size of the robotic workcell, along with the mass m_e and radius r_e of the end-effector, are presented in Table 3. As previously stated, when developing a cable robot recovery strategy, it is usually possible to assume the end-effector as a point mass. Such an approach allows both the kinematics and the dynamics of the cable robot to be simplified without lack of generality; in fact, avoiding end-effector collision is the main goal of the proposed strategy. In the proposed setup, a steel-ball with a ring to attach the four cables was used. Despite this leading to a difference in the position of the cable attach point and the center of mass of the end-effector, this solution was considered as an acceptable compromise between the model hypothesis and the technological constraints.

Table 3. Size of the robot workcell and of the end-effector.

Parameter	Value	Unit
a	1775	mm
b	1690	mm
h	1950	mm
m_e	2.941	kg
r_e	90	mm

Each cable was wound around a winch fixed on the motor axis, with each motor placed on the lower vertices of the workcell; the cables exited from the upper vertices through movable pulleys.

3.3. Motion Tracking System

To verify the feasibility of the proposed control system and the recovery strategy, the embedded resolvers were not considered sufficient. Thus, a motion capture system based on markers was adopted: Vicon motion capture [30]. This choice was considered reasonable since it was already adopted in the literature [31,32] and has even been applied to verify the motion of a 6-DOF CDPR [33]; thus, we believe that its use in this context is suitable.

In greater detail, in this work, we adopted four Vicon Vantage 5 cameras. Technical data of the adopted camera are presented in Table 4. The displacement of the four cameras is represented in Figure 6 with respect to the robotic workcell (white dotted lines). To detect the spatial information of the end-effector—i.e., 3D data relative to the coordinate frame of the measurement system—four passive markers were properly displaced on the end-effector. The sampling frequency of the motion capture was set to 400 Hz.

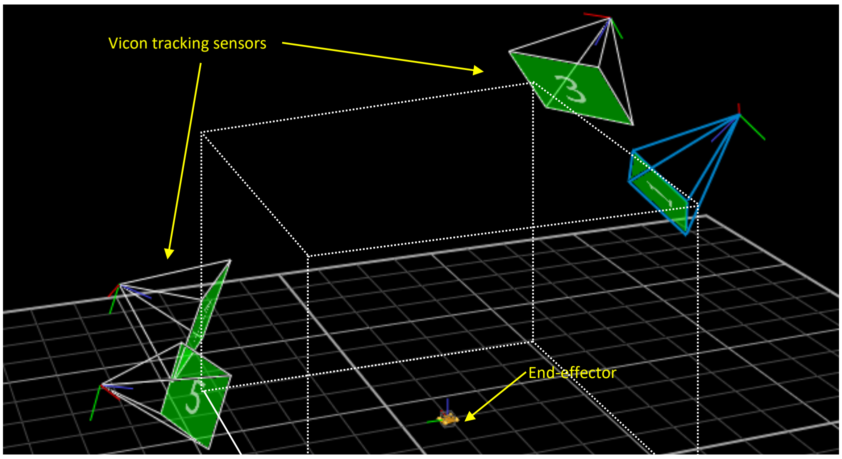


Figure 6. Vicon Tracker display showing the four cameras and markers position with respect to the robot workcell. FoV: field of view.

Table 4. Technical data of the adopted cameras, Vicon Vantage V5.

Parameter	Value	Unit
Resolution	5	MP
Maximum frame rate at full resolution	420 (5 MP)	fps
Maximum frame rate	2000	fps
Horizontal FoV	47–66	°
Vertical FoV	41–57	°

4. Experimental Results

The following section describes the performed experiments: at first, tests on a circular reference trajectory were carried out to verify the reliability of the motion tracking system, and then tests were performed with the proposed recovery strategy.

4.1. Circular Trajectory

A circular trajectory was planned for the tests. Its radius was 500 mm and its center was positioned in the center of the workspace; i.e., at coordinates $\mathbf{c} = [0, 0, 0.750]$ m. The absolute reference frame was positioned in the center of the workspace floor. The center of the circle was set as the starting and final point of the trajectory. For this reason, two connecting paths were defined, preceding and following the circular trajectory, respectively. Lastly, the motion time was set to be equal to 11 s.

The Cartesian trajectory, obtained from the resolvers' data, is compared with the Vicon data in Figure 7, where the continuous line represents the trajectory obtained from the direct kinematics algorithm (resolvers' data), and the dashed line represents the data measured by the motion tracking system.

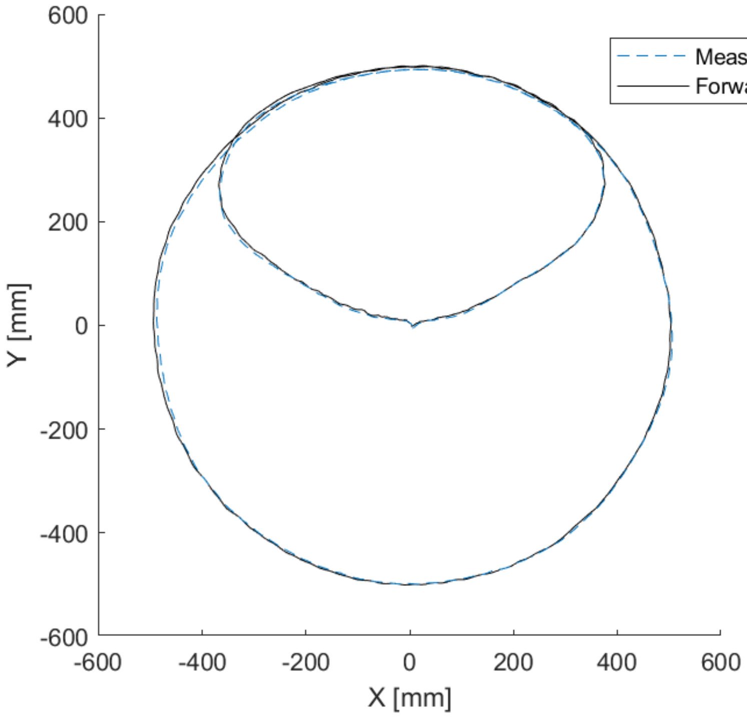


Figure 7. Spatial trajectory of the end-effector: comparison between the reference motion (actuator position) and the measurements from the motion tracking system.

These results show a close matching between the two trajectories.

Figure 8 shows the measured torque of each actuator during this motion. It is possible to notice that the motors continuously pulled the cables to keep them taut. A very small unwinding torque was measured on each actuator when the end-effector was far from the corresponding cable exit point (e.g. motor 3 presented a torque of 0.04 Nm at 5.5 s). Such a torque was imparted by the feed-forward to promptly compensate for the motor inertia.

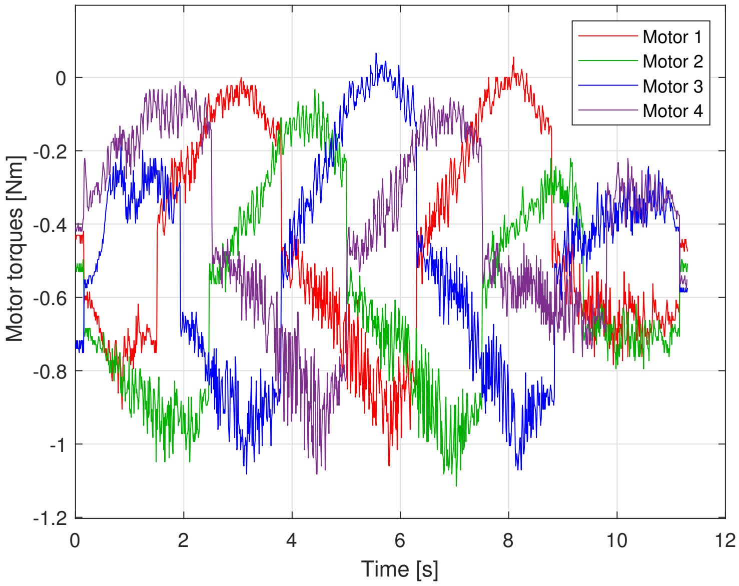


Figure 8. Motor torques during the planned trajectory.

In order to guarantee that the accuracy achieved in Figure 7 was not restricted to this scenario, a more complex trajectory was implemented throughout the workspace. In Figure 9, the proper motion composed of three squared trajectories connected by vertical motions is depicted.

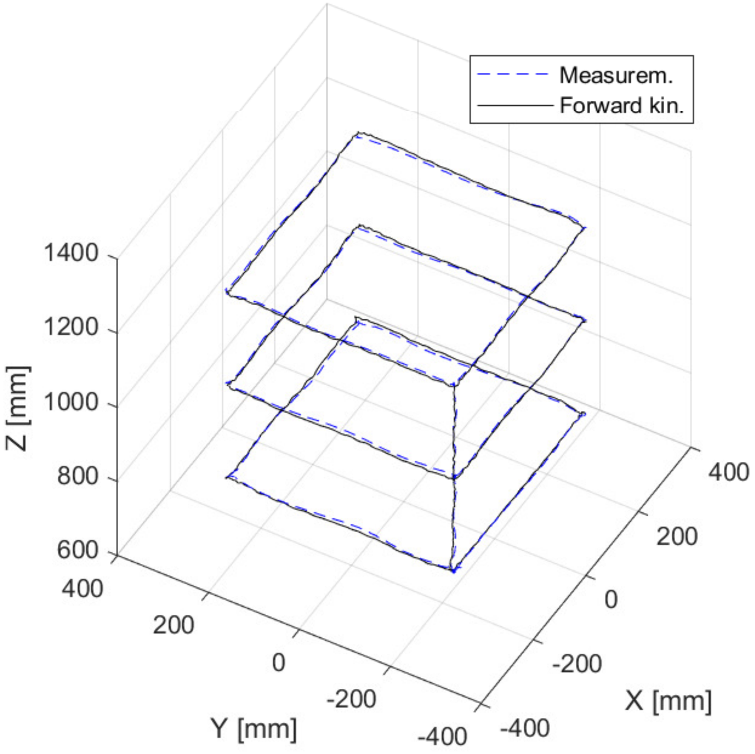
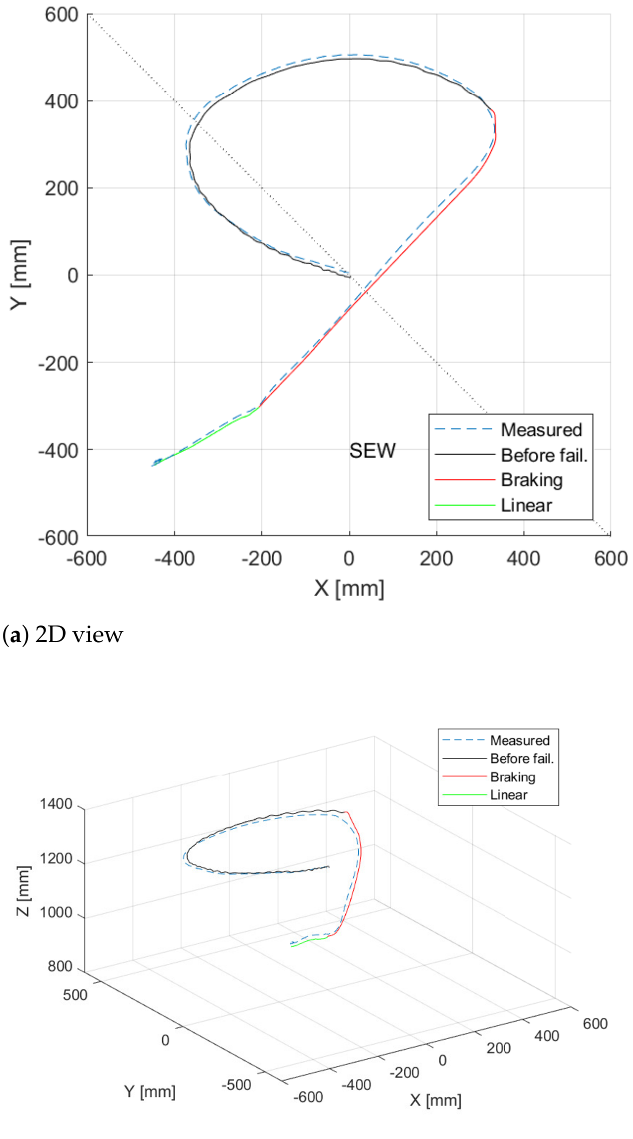


Figure 9. Comparison between the reference motion (actuator position) and the measurements from the motion tracking system using a more complex trajectory.

The novel comparison confirms the close matching between the two trajectories into the workspace through which the failure test passed.

4.2. Recovery Strategy

After the initial testing phase, the end-effector trajectory was compared with the Vicon data for the recovery strategy, as seen in Figure 10. As in the previous figure, the dashed line represents the measurement data and the continuous lines are the end-effector trajectory divided into three different phases. Lastly, the dotted line in the 2D view (Figure 10a) identifies the residual SEW.



(**a**) 2D view

(**b**) 3D view

Figure 10. Spatial trajectory of the end-effector during the recovery strategy: the black line is the trajectory before cable failure, while the red and green lines represent the two phases of the recovery strategy.

The aim of the proposed work was to provide a recovery strategy that is feasible for real-time applications. Our method was designed to be independent of the sensors employed for the cable failure detection. Thus, in the proposed test, the cable failure event was simulated by quickly unfolding the cable to make it slack and cause it to be unable to pull the end-effector.

As previously described, the recovery approach was divided into two phases: a braking phase while moving towards the new SEW, and a linear trajectory, respectively (the black line is the circular pre-failure trajectory). The first phase, in red, shows how the end-effector moved towards the new SEW. Then, the linear trajectory drove the end-effector towards the designed SP, whose coordinates were [−0.45, −0.42, 1.2] m.

Figure 11 represents the trend of the actuator torques during the cable failure test. Two vertical dashed lines are depicted to better identify the two phases of the recovery strategy.

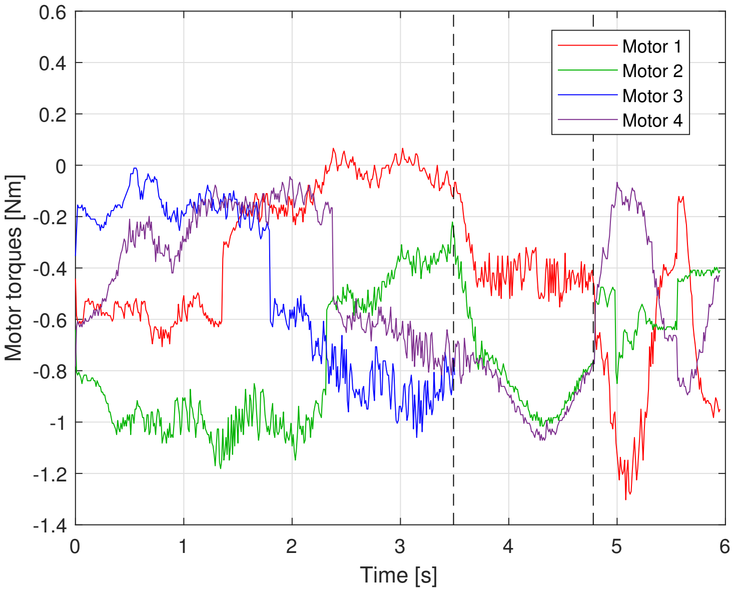


Figure 11. Motor torques during the recovery strategy for motor 1 (blue), motor 2 (red), motor 3 (green) and motor 4 (violet).

In the first part of the plot, the torque behavior corresponded to the movement of the end-effector actuated by the four cables. The first vertical dashed line in the figure shows the time of failure of cable 3 (t = 3.5 s). Then, the first phase of the recovery strategy began. An increase in the motor torques of the cables closest to the broken one can be observed. Indeed, they were required to sustain the end-effector and attract it to the SEW. Meanwhile, a proper constant torque was applied by motor 1, not to help the other cables to attract the end-effector into the SEW but also to prevent cable 1 from becoming slack. Conversely, the torque of the broken cable motor was not plotted because it is not meaningful. The final phase of the recovery strategy began after the second vertical dashed line. The measured torques corresponded to the linear trajectory planned to make the end-effector reach the SP. The values of motor torques exerted during the recovery guaranteed that the cable tensions remained positive and hence that the cables were taut.

5. Conclusions

This work presents a cable robot recovery strategy that is suitable for control systems even when the computational efficiency of the algorithm is critical. Indeed, other approaches based on more complex algorithms may not be performed in such constrained scenarios, and alternative solutions must be developed. The proposed strategy was applied in an experimental environment composed of industrial-grade components, further highlighting the practical implications of the study. The recovery strategy has been tested by measuring the end-effector trajectory using a marker-based motion capture system. The results show a close matching between the motion capture data and the input data. Moreover, the actuator torques remained negative during the entirety of the recovery process, corresponding to positive cable tensions under proper conditions.

Future works will mainly focus on the methods to detect cable failures. Firstly, the use of proper sensors to detect cable breaking will be investigated; moreover, the design and the development of sensorless algorithms will be evaluated.

Author Contributions: Conceptualization, G.B., R.M. and A.T.; investigation G.B., R.M. and A.T.; methodology, G.B., R.M. and A.T.; writing, G.B., R.M. and A.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: Adrian Stefan Chirciu and Matteo Peruzzo are gratefully acknowledged for their help in the experimental tests.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Maya, M.; Castillo, E.; Lomelí, A.; González-Galván, E.; Cárdenas, A. Workspace and Payload-Capacity of a New Reconfigurable Delta Parallel Robot. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 56. [CrossRef]
2. Taghirad, H. *Parallel Robots: Mechanics and Control*; CRC Press: Boca Raton, FL, USA, 2013.
3. Trevisani, A. Planning of dynamically feasible trajectories for translational, planar, and underconstrained cable-driven robots. *J. Syst. Sci. Complex.* **2013**, *26*, 695–717. [CrossRef]
4. Pott, A.; Mütherich, H.; Kraus, W.; Schmidt, V.; Miermeister, P.; Verl, A. IPAnema: A family of Cable-Driven Parallel Robots for Industrial Applications. In *Cable-Driven Parallel Robots*; Bruckmann, T., Pott, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 119–134.
5. Chesher, C. Robots and the moving camera in cinema, television and digital media. *Lect. Notes Comput. Sci.* **2016**, *9549*, 98–106.
6. Rosati, G.; Gallina, P.; Masiero, S. Design, Implementation and Clinical Tests of a Wire-Based Robot for Neurorehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2007**, *15*, 560–569. [CrossRef] [PubMed]
7. Pham, C.B.; Yeo, S.H.; Yang, G.; Kurbanhusen, M.S.; Chen, I.M. Force-closure workspace analysis of cable-driven parallel mechanisms. *Mech. Mach. Theory* **2006**, *41*, 53–69. [CrossRef]
8. Boschetti, G.; Carbone, G.; Passarini, C. Cable Failure Operation Strategy for a Rehabilitation Cable-Driven Robot. *Robotics* **2019**, *8*, 17. [CrossRef]
9. Roberts, R.; Graham, T.; Lippitt, T. On the inverse kinematics, statics, and fault tolerance of cable-suspended robots. *J. Robot. Syst.* **1998**, *15*, 581–597. [CrossRef]
10. Boumann, R.; Bruckmann, T. Development of Emergency Strategies for Cable-Driven Parallel Robots after a Cable Break. In *International Conference on Cable-Driven Parallel Robots*; Springer: Cham, Switzerland, 2019; pp. 269–280.
11. Caro, S.; Merlet, J.P. Failure Analysis of a Collaborative 4-1 Cable-Driven Parallel Robot. In *New Trends in Mechanism and Machine Science*; Pisla, D., Corves, B., Vaida, C., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 440–447.
12. Notash, L. Wrench Recovery for Wire-Actuated Parallel Manipulators. In *Romansy 19 – Robot Design, Dynamics and Control*; Padois, V., Bidaud, P., Khatib, O., Eds.; Springer: Vienna, Austria, 2013; pp. 201–208.
13. Riechel, A.; Ebert-Uphoff, I. Force-Feasible Workspace analysis for underconstrained, point-mass cable robots. In Proceedings of the IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, 26 April–1 May 2004; pp. 4956–4962.
14. Mottola, G.; Gosselin, C.; Carricato, M. Dynamically feasible motions of a class of purely-translational cable-suspended parallel robots. *Mech. Mach. Theory* **2019**, *132*, 193–206. [CrossRef]
15. Zhang, N.; Shang, W. Dynamic trajectory planning of a 3-DOF under-constrained cable-driven parallel robot. *Mech. Mach. Theory* **2016**, *98*, 21–35. [CrossRef]
16. Passarini, C.; Zanotto, D.; Boschetti, G. Dynamic trajectory planning for failure recovery in cable-suspended camera systems. *J. Mech. Robot.* **2019**, *11*. [CrossRef]
17. Berti, A.; Gouttefarde, M.; Carricato, M. Dynamic Recovery of Cable-Suspended Parallel Robots After a Cable Failure. In *Advances in Robot Kinematics 2016*; Lenarčič, J., Merlet, J.P., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 331–339.
18. Boschetti, G.; Passarini, C.; Trevisani, A. A recovery strategy for cable driven robots in case of cable failure. *Int. J. Mech. Control* **2017**, *18*, 41–48.
19. Boschetti, G.; Trevisani, A. Cable robot performance evaluation by Wrench exertion capability. *Robotics* **2018**, *7*. [CrossRef]
20. Boumann, R.; Bruckmann, T. Real-Time Cable Force Calculation beyond the Wrench-Feasible Workspace. *Robotics* **2020**, *9*, 41. [CrossRef]
21. Boumann, R.; Bruckmann, T. Computationally Efficient Cable Force Calculation Outside the Wrench-Feasible Workspace. *Mech. Mach. Sci.* **2020**, *78*, 177–188.
22. Hsia, T.C.; Lasky, T.A.; Guo, Z.Y. Robust independent robot joint control: design and experimentation. In Proceedings of the 1988 IEEE International Conference on Robotics and Automation, Philadelphia, PA, USA, 24–29 April 1988; pp. 1329–1334. [CrossRef]
23. Boschetti, G.; Passarini, C.; Trevisani, A. A Recovery Approach for Spatial Cable Robots After Cable Failure. In *Advances in Italian Mechanism Science*; Carbone, G., Gasparetto, A., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 457–464.

24. Boschetti, G.; Minto, R.; Trevisani, A. Improving a Cable Robot Recovery Strategy by Actuator Dynamics. *Appl. Sci.* **2020**, *10*, 7362. [CrossRef]
25. Boschetti, G.; Passarini, C.; Trevisani, A.; Zanotto, D. A fast algorithm for wrench exertion capability computation. In *Cable-Driven Parallel Robots*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 292–303.
26. García, S.; Parejo, A.; Gómez, A.A.; Molina, F.J.; Larios, D.F.; León, C. Training Equipment for Automatic Control Systems and Industrial Automation subjects in Engineering Degrees. In Proceedings of the 2020 XIV Technologies Applied to Electronics Teaching Conference (TAEE), Porto, Portugal, 8–10 July 2020; pp. 1–6. [CrossRef]
27. Liang, Q.; Li, L. The study of soft PLC running system. *Procedia Eng.* **2011**, *15*, 1234–1238. [CrossRef]
28. Beckhoff Automation GmbH & Co. KG. Available online: https://www.beckhoff.com/en-en/products/automation/twincat/ (accessed on 11 December 2020).
29. Beckhoff Automation GmbH & Co. KG. Available online: https://www.beckhoff.com/english.asp?start/default.htm (accessed on 11 December 2020).
30. Vicon Motion Systems Ltd. Available online: https://www.vicon.com/ (accessed on 11 December 2020).
31. Baek, J.; Park, S.; Cho, B.; Lee, M. Position tracking system using single RGB-D Camera for evaluation of multi-rotor UAV control and self-localization. In Proceedings of the 2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Busan, Korea, 7–11 July 2015; pp. 1283–1288. [CrossRef]
32. Wang, J.; Garratt, M.; Anavatti, S.; Lin, S. Real-time visual odometry for autonomous MAV navigation using RGB-D camera. In Proceedings of the 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, China, 6–9 December 2015; pp. 1353–1358. [CrossRef]
33. Chellal, R.; Cuvillon, L.; Laroche, E. A kinematic vision-based position control of a 6-DoF cable-driven parallel robot. *Mech. Mach. Sci.* **2015**, *32*, 213–225. [CrossRef]