



OPEN ACCESS

EDITED BY

Sébastien Briot,
UMR6004 Laboratoire des Sciences du
Numérique à Nantes (LS2N), France

REVIEWED BY

Edoardo Idà,
University of Bologna, Italy
Philippe Cardou,
Laval University, Canada
Stéphane Caro,
UMR6004 Laboratoire des Sciences du
Numérique à Nantes (LS2N), France

*CORRESPONDENCE

Adhiti Raman,
✉ adhiti@clermson.edu

RECEIVED 15 October 2022

ACCEPTED 06 April 2023

PUBLISHED 17 May 2023

CITATION

Raman A, Walker I, Krovi V and Schmid M
(2023), Cable failure tolerant control and
planning in a planar reconfigurable cable
driven parallel robot.
Front. Robot. AI 10:1070627.
doi: 10.3389/frobt.2023.1070627

COPYRIGHT

© 2023 Raman, Walker, Krovi and
Schmid. This is an open-access article
distributed under the terms of the
[Creative Commons Attribution License
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or
reproduction in other forums is
permitted, provided the original author(s)
and the copyright owner(s) are credited
and that the original publication in this
journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Cable failure tolerant control and planning in a planar reconfigurable cable driven parallel robot

Adhiti Raman^{1*}, Ian Walker², Venkat Krovi¹ and Matthias Schmid¹

¹Clemson University, Automotive Engineering, Greenville, SC, United States, ²Clemson University, Electrical/Computer Engineering, Clemson, SC, United States

The addition of geometric reconfigurability in a cable driven parallel robot (CDPR) introduces kinematic redundancies which can be exploited for manipulating structural and mechanical properties of the robot through redundancy resolution. In the event of a cable failure, a reconfigurable CDPR (rCDPR) can also realign its geometric arrangement to overcome the effects of cable failure and recover the original expected trajectory and complete the trajectory tracking task. In this paper we discuss a fault tolerant control (FTC) framework that relies on an Interactive Multiple Model (IMM) adaptive estimation filter for simultaneous fault detection and diagnosis (FDD) and task recovery. The redundancy resolution scheme for the kinematically redundant CDPR takes into account singularity avoidance, manipulability and wrench quality maximization during trajectory tracking. We further introduce a trajectory tracking methodology that enables the automatic task recovery algorithm to consistently return to the point of failure. This is particularly useful for applications where the planned trajectory is of greater importance than the goal positions, such as painting, welding or 3D printing applications. The proposed control framework is validated in simulation on a planar rCDPR with elastic cables and parameter uncertainties to introduce modeled and unmodeled dynamics in the system as it tracks a complete trajectory despite the occurrence of multiple cable failures. As cables fail one by one, the robot topology changes from an over-constrained to a fully constrained and then an under-constrained CDPR. The framework is applied with a constant-velocity kinematic feedforward controller which has the advantage of generating steady-state inputs despite dynamic oscillations during cable failures, as well as a Linear Quadratic Regulator (LQR) feedback controller to locally dampen these oscillations.

KEYWORDS

Reconfigurable CDPR (Cable Driven Parallel Robot), IMM (interacting multiple model) algorithm, LQR (linear quadratic regulator) control, FTC (fault tolerant control), FDD (fault diagnosis and detection), EKF (extended kalman filter), redundancy resolution, switching control

1 Introduction

Cable driven parallel robots (CDPRs) are lightweight mechanisms with favourable power-to-weight ratios. They can be designed and structured to be manipulable over large workspaces, which can prove useful in application domains such as painting, bricklaying, material hauling, warehousing and concrete 3D printing. Large workspace domains bring

with them a set of challenges such as increased cable sag, unmodeled cable dynamics and the increased effects of environmental factors such as unmodeled vibrations and air resistance. Heavy and dynamically-changing payloads may also cause cables to experience high tensions while possibly exciting unmodeled dynamic effects, such as swinging payloads (if the system is under-constrained) or unexpected cable collisions with surrounding objects in the workspace. Here, the addition of geometric reconfigurability is able to offer the flexibility needed to overcome some of these challenges. Geometric reconfigurability in this context is the ability of a robot to change its configuration through kinematic redundancy which is introduced by actuating the cable attachment points on the base frame of the CDPR. This robot is known as a *reconfigurable CDPR* or *rCDPR* in this work. Although most CDPRs can be manually reconfigured to accommodate different workspace reachability requirements by changing their geometric attachment points (as designed in (Gagliardini et al., 2016)), automating this process for online reconfiguration provides unique advantages that are still only minimally explored. For example, the robot can not only move to new configurations or new locations in the workspace without manual intervention, but can also gain capabilities that would otherwise not be possible (Rasheed, 2019), such as changing the quality (or performance index) of the pose at the end-effector without effecting its position or orientation (Zhou et al., 2014; Raman et al., 2020). Online reconfigurability introduces the properties of kinematic redundancy (i.e. singularity avoidance, self-motion) to a naturally actuation redundant system (i.e. the kinematically controllable degrees-of-freedom at the end-effector of a CDPR is less than the number of cables that support it). In this work, we exploit geometric reconfigurability to introduce a new control strategy for fault tolerance against cable failures.

This research is motivated by the development of a cooperative robot (cobot) assisting in concrete delivery tasks such as concrete 3D printing or human-directed concrete application (Figure 1). The cobot is expected to consist of two subsystems: (i) a cable driven parallel manipulator that controls a payload over a large workspace (called the *macro-manipulator*); and (ii) a continuum robot enclosing a concrete delivery tube that provides finely directed control through congested spaces (called the *micro-manipulator*) (Srivastava et al., 2022).

This work contributes to the development of the first subsystem by focusing on research questions involving the challenges posed by the addition of kinematic redundancies to conventional CDPRs (Raman et al., 2020; 2021; Walker et al., 2022). Online geometric reconfiguration can recover the static workspace and mobility lost through cable failures while also optimizing the quality of that mobility to a feasible extent. The basic idea is pictorially demonstrated in Figure 2. This form of fault tolerant control (FTC) may be unique to the addition of kinematic reconfigurability in CDPRs.

In our previous work (Raman et al., 2022) we demonstrated a proof-of-concept implementation of the automatic cable failure tolerant control framework that could simultaneously identify and overcome cable failures in an automatically without requiring an accessory diagnostic fault detection protocol. Failure identification, pose estimation, and cable failure tolerant control are accomplished simultaneously in this framework by relying on pose measurements from the end-effector alone. An interactive multiple model (IMM)

adaptive estimation algorithm with a set of extended Kalman filters (EKF), where each EKF corresponds to a different motion model, is applied to this measurement data and integrated into the control scheme which in turn provides input information from a set of controllers dependent on the different motion models of the failure modes.

In this research, we build upon that body of work by eliminating some design assumptions and increasing the uncertainties in model and noise measurements. The plant model is simulated to be as realistic as possible by incorporating disturbances, actuator dynamics and non-linearities arising from cable elasticity to demonstrate the real-world feasibility of the framework. In addition, we track complete trajectories despite cable failures by automatically returning to the point of failure to continue the task. The framework is updated to include two controllers (i) a feed-forward controller which has the advantage of input stability in the face of dynamic oscillations during cable failures and (ii) an LQR-based feedback controller stabilized around the desired joint inputs that is helpful in damping dynamic oscillations during a failure event and in recovering the trajectory quickly. The addition of the feedback controller eliminates the need for the resource-intensive feedforward kinematic controller to run at every timestep.

2 Related work

Fault tolerant control in robotics has a long history (Visinsky et al. (1994)). The use of kinematic redundancy to compensate for joint failures in serial robots is also well-established (Maciejewski (1990)). Early studies accounting for cable failures in CDPRs (Roberts et al. (1998)) showed that static equilibrium can be maintained (upon failure) if the robot is aligned to specific singular configurations. Bosscher and Ebert-Uphoff (2004) noted two kinds of failure modes: cable breakage due to excessive positive cable tensions or slackness due to the lack of cable tension. These failure modes could be precluded by limiting the desired wrenches at the end-effector to lie within the available wrench set Bouchard et al. (2010); Notash (2012) examined additional modes of cable failure such as stuck actuators (applying a passive restraint on the end-effector) or situations in which the actuator moves but the output is biased. However, these works are limited to kinematic models of CDPRs with ideal and inelastic cables. In a later work, Caro and Merlet (2020) outlined all possible dynamic failures for cable driven suspended robots and presented a practical passive solution to raise the end-effector load and ensure human safety in the event of cable failure. The authors noted the challenges of relying only on cable length measurements through the encoders or the loss of cable tensions to recognize that a cable failure has occurred. The authors instead recommended measuring the cable angles through IMUs affixed on the cables to detect a cable failure.

Outside of the context of cable failure, the challenges of relying on encoder and force measurements have been noted extensively in the context of CDPR control. Le Nguyen and Caverly (2021); Korayem et al. (2018); Caverly and Forbes (2016) addressed the futility of relying on forward kinematics from cable length measurements to determine end-effector pose, reasoning that encoder measurements do not necessarily correspond to the true cable length, which may be affected by unknown deformations

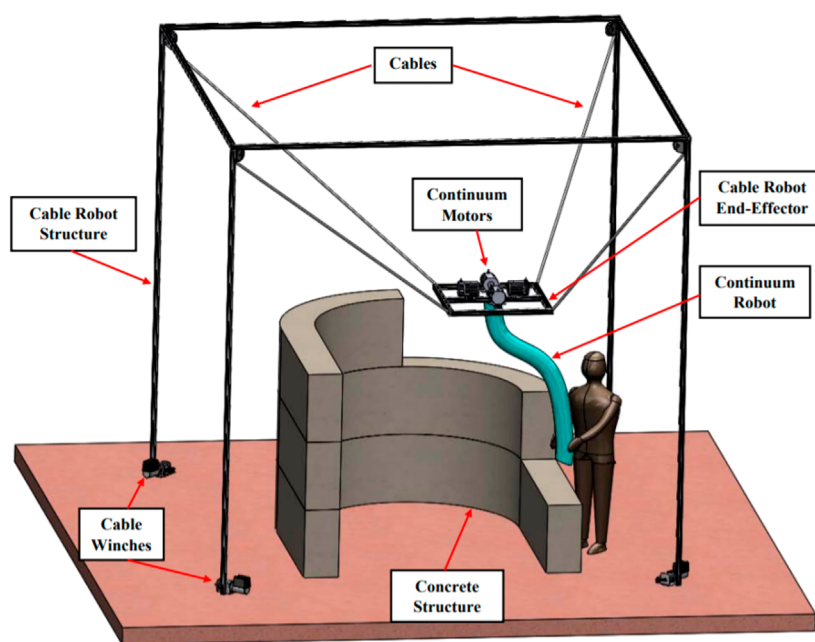


FIGURE 1
Micro + macro manipulator concept drawing for concrete 3D printing.

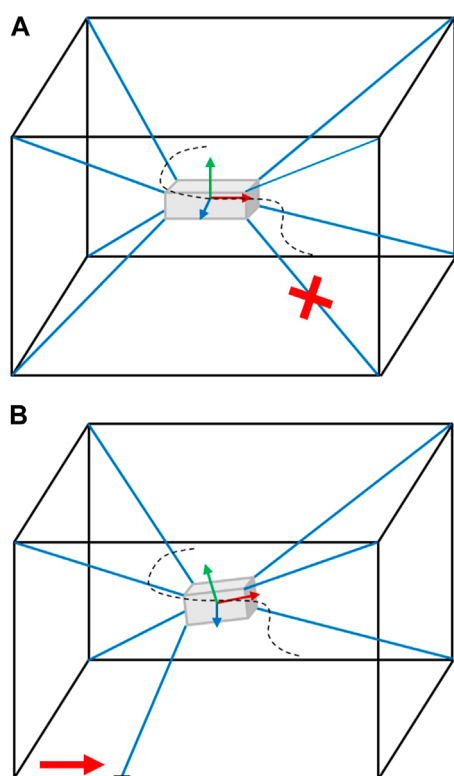


FIGURE 2
Geometric reconfigurability allows the trajectory tracking to continue. (A) A cable fails. (B) However, sliders reconfigure to allow the trajectory tracking to continue.

and stiffnesses. To mitigate this issue, pose estimation techniques based on Extended (EKF) and Unscented (UKF) Kalman Filters applied on data from payload mounted IMUs were implemented. [Korayem et al. \(2018\)](#) further incorporated filtering with linearized feedback control for CDPs, demonstrating that payload mounted IMUs can improve pose estimations in the presence of moderate-to-high dynamic effects.

Other literature on CDPR cable failures focused on dynamic failure recovery strategies or dynamic collision avoidance strategies. For instance, [Passarini et al. \(2019\)](#) studied emergency stop strategies for preventing collisions in cable suspended robots, whereas [Boumann and Bruckmann \(2022\)](#) focused on a force control strategy in the remaining cables of a cable-suspended robot to minimize the safety risk of dynamic effects during cable failures in the workspace. The authors assumed that the cable failures, when triggered, were always correctly identified and the proposed control framework was restricted to a post-failure setting.

Kinematic redundancies have also been shown to enable robust fault tolerant systems in parallel kinematic manipulators (PKMs). [Notash \(2011\)](#), for instance, investigated the effect of joint failures on the motion of kinematically redundant PKMs by analyzing the behavior of the changed kinematic model and setting task priorities to recover complete or partial motion. However, literature on the use of such redundancies for fault tolerant control (FTC) in cable driven manipulators is non-existent.

In this work we have explored a popular approach to failure detection and diagnosis (FDD) that has been utilized abundantly in aerospace applications. This FDD technique employs Multi Model Adaptive Estimation (MMAE) through the use of a bank of filters, with each filter correlated to a possible failure mode that may occur.

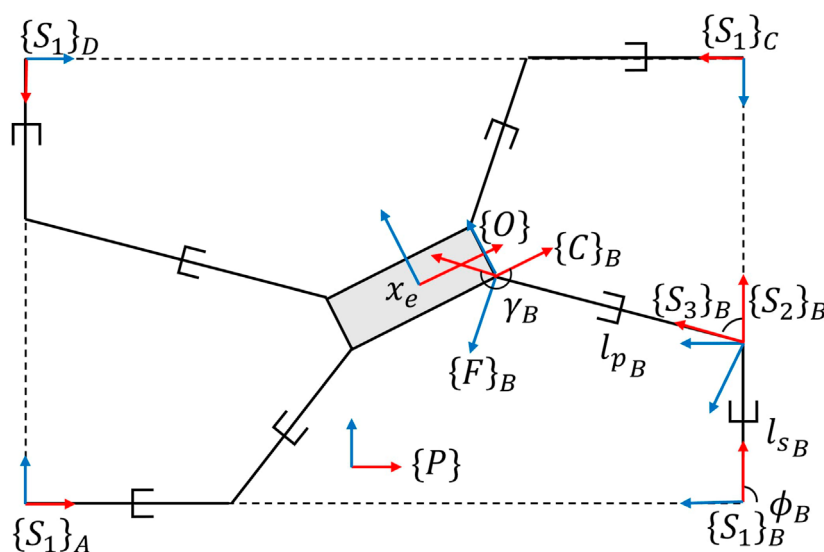


FIGURE 3
The kinematically redundant 4 cable CDPR.

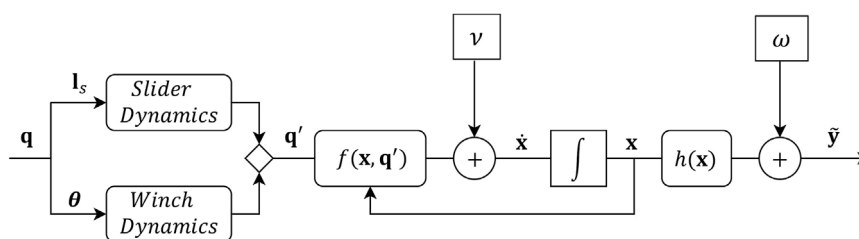


FIGURE 4
Plant model.

In this study, we utilize an Interactive MMAE or IMM for short. Within robotics, IMMs have been employed for tracking trajectories (Mazor et al. (1998)) and behaviour prediction (Gill et al. (2019)) as well as in actuator fault detection and identification (Tudoroiu and Khorasani (2005)). Many of these studies focus on FDD only and do not attempt to integrate controllers for task recovery. One of the few exceptions is the work of Hill et al. (2021), where a non-linear Model Predictive Controller is applied to recover from reaction wheel failures that are identified through an UKF-based IMM for satellite maneuvering; an approach that is similar to our proposed strategy.

In this study, the cables are modeled as elastic springs. Furthermore, we assume that the employed dynamic models for control and estimation are imperfect and do not correspond perfectly to physical reality (e.g., through the inclusion of parametric uncertainties and process noise). We also assume that encoder readings do not correspond accurately to the true position of the end-effector and constitute an unreliable source of information; a reasonable assumption to make when accounting for elasticity in cables.

The inclusion of elastic behavior in the cables presents an additional control challenge. A feedforward controller that employs

a simplified inelastic model does not account for this flexibility, causing the internal dynamics to present as undesired oscillations or vibrations at the end-effector. A feedback controller can be challenging to design when the local cable stiffness parameters (at time t_k) are unknown or uncertain and the mapping from input to end-effector state is highly non-linear. The challenges of dynamic feedback control in CDPRs are well studied in literature, but their application has largely been focused on systems with cables of high stiffness. Feedback linearization Begey et al. (2018); Picard et al. (2020), sliding mode, and H_∞ controllers have been successfully employed. Feedforward dynamic controllers like input-shaping have shown success (Baklouti et al. (2021)) in minimizing unwanted dynamics, especially in underactuated CDPRs (Idà et al. (2021)). In all these systems, the CDPR dynamics is often well known and the topology is unchanging making the system predictable and controllable. Nevertheless, model uncertainties and measurement noise can continue to present a challenge and can be overcome by the inclusion of an additional state estimation framework Kosari et al. (2013). Another approach to optimal control is the Linear-Quadratic-Gaussian (LQG) approach. In a recent work Chen et al. (2022) successfully employed an LQG with time-varying LQR gains on a CDPR and demonstrated improved performance

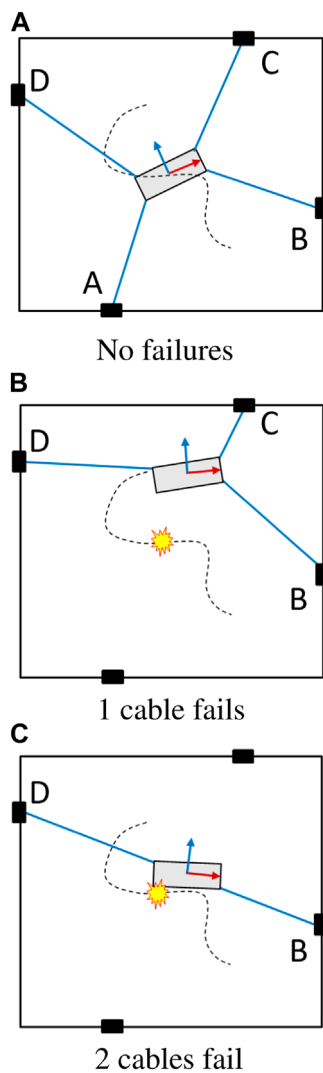


FIGURE 5
The 3 kinds of failures in a 4-PRPR CDPR (A) No failures (B) 1 cable fails (C) 2 cables fail.

compared to a baseline feedforward controller. [Korayem et al. \(2017\)](#) employed LQG alongside a feedback linearized controller to demonstrate improved tracking performance in the presence of model uncertainty and measurement noise. In this work we use LQRs for local stability by employing the state estimates from the IMM for state feedback. The steady-state control inputs that the LQRs are stabilized around are generated through kinematic feedforward controllers.

In this research, the applied IMM utilizes a parallel bank of Extended Kalman Filters, with each filter corresponding to one of the various cable failure modes. The mixed estimate from the IMM provides a corrected pose estimate together with an understanding of the current working mode. A parallel bank of feedforward controllers computes a set of possible joint inputs to the plant model (robot simulation) based on the desired trajectory information from the waypoint following algorithm. This forms the steady-state input that is modulated through a set of LQRs to generate updated inputs

to minimize dynamic oscillations. The control inputs from the LQR are mixed with the IMM weights to control the robot. This approach simultaneously detects the fault, provides a corrected estimate and updates the input such that the system can robustly cope with sudden failures.

This framework is implemented in a simulation-only setting and thus serves as a proof-of-concept implementation of this framework. The highlights of this paper are as follows: (i) an Interactive Multi Model Filter is derived and demonstrated as a robust estimator for cable failure detection and diagnosis (FDD) as well as for real-time pose estimation of the rCDPR end-effector; (ii) a waypoint following and a constant-velocity path tracking algorithm is introduced to maintain constant velocity while path tracking and to recover the trajectory from the waypoint closest to the break-point after a cable failure event; (iii) a bank of kinematic feedforward controllers generate steady-state control inputs through kinematic redundancy resolution of each rCDPR model corresponding to each failure mode; and finally (iv) and LQR-based local stabilization framework is applied on the non-linear system to minimize local oscillations at the plant and hasten the recovery.

3 System models

This section introduces the mathematical model of the robot utilized in this research. The robot is a planar kinematically redundant CDPR with a platform that is driven by 4 cables and 4 linear actuators housing the cable winch mechanism. The system model is described in [Figure 3](#). When the robot is fully functional, the system possesses four cables which are actuated by cable winches mounted on platforms on a linear actuator. The end-effector is a planar rigid body whose center is given by the origin of frame $\{O\}$. Of the three available degrees of freedom $\{x, y, \phi\}$, only two $\{x, y\}$ are actively controlled. The cables are modeled as linear springs whose spring constant is a function of the current free (untensioned) cable length. The material constant of the cable stiffness is given by k_0 , where k_0 is interpreted as the stiffness per unit reciprocal length of free cable. This indicates that longer cable lengths lead to lower stiffness in the cable and thus higher elastic deformations for the same tension. The current stiffness of the cable is therefore given by $k = \frac{k_0}{l_{p,0}}$ where $l_{p,0}$ is the current free length of the cable. The true cable length l_p is the length of the prismatic joint from $\{S_3\}_i$ to $\{F\}_i$ as seen in [Figure 3](#) such that $l_p = l_{p,0} + \Delta l_p$ where Δl_p is the cable elongation due to tension. Many approaches to modeling cable stiffness assume that the true cable length and free cable length are equal in the calculation of the current cable stiffness [Verhoeven \(2006\)](#), which helps decouple the dependence of cable stiffness on the cable tension and helps linearize the system. This assumption is not made here. The free cable length $l_{p,0}$ was determined to be closer to the encoder value given by $l_c = r\theta$ where r is the winch radius and the interdependence of cable stiffness and cable tension is maintained. The tension in the cables is represented by $\tau = k(l_p - l_c)$ where the condition $l_c < l_p$ is maintained as much as possible to prevent cable sag. The cable tension is modulated by a unit step function, such that when this constraint is violated, the cable tension becomes zero, mimicking a realistic consequence of cable sag.

The cable-winch attachment points, $\{S_2\}$, are not constant as in traditional CDPRs, but instead move along linear actuators along the

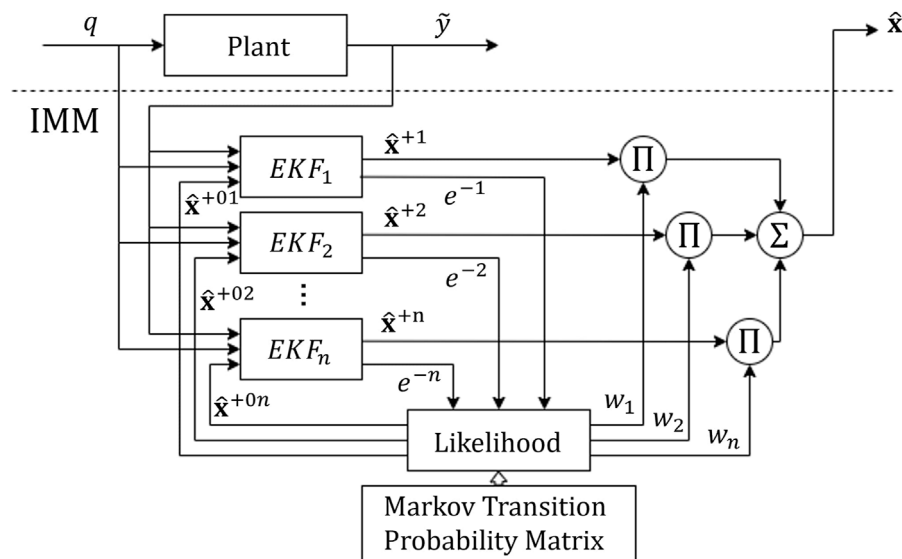


FIGURE 6
The Interactive Multiple Model (IMM) filter.

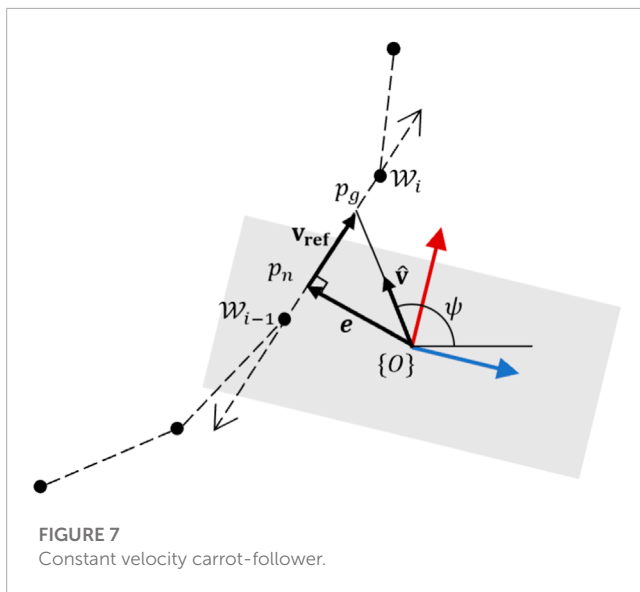


FIGURE 7
Constant velocity carot-follower.

x -axis of the base frames $\{S_1\}$. The longitudinal positions of these attachment points with respect to the base frame are given by l_s . Both the slider joints and the cable winching mechanisms are subject to actuation dynamics which inform the cable tensions and cable stiffnesses that effect the dynamics at the end-effector.

3.1 Actuator dynamics

Let the complete joint input vector from the controller (i.e. desired reference values to the plant model) be expressed as $\mathbf{q} = [\mathbf{l}_s \quad \mathbf{l}_c]$ and the realized joint vector (actual values at time t) be given by $\mathbf{q}' = [\mathbf{l}'_s \quad \mathbf{l}'_c]$. The moving slider bases are modeled as single

degree-of-freedom (DOF) second order linear systems subject to maximum and minimum velocity constraints to mimic the motion of a screw-type linear actuator. The bases are modeled with inertia and friction and a simple proportional controller as shown in Eq. 1. For every i th actuator in the vector $\mathbf{l}_s$ (dropping the subscript i to indicate each individual slider joint) we model the dynamic equation as,

$$\begin{aligned} m\ddot{l}'_s + d\dot{l}'_s &= g_{p,l_s}(l_s - l'_s) \\ \text{s.t. } \dot{l}'_{s,min} &< \dot{l}'_s < \dot{l}'_{s,max} \end{aligned} \quad (1)$$

where g_{p,l_s} is the proportional gain and the maximum velocity that a linear screw actuator can travel in is limited to a small value ($|\dot{l}'_{s,min}| = |\dot{l}'_{s,max}|$). The motor cable spooling $\mathbf{l}'_c$ is modeled as a first order lag system, i.e.

$$T_s \dot{l}'_c + l'_c = g_{p,l_c}(l_c - l'_c) \quad (2)$$

where T_s is the settling time and g_{p,l_c} is the proportional gain. In addition to the specified models and constraints, there are also limits placed on the position values of the actuator joints in order to set hard constraints defined by geometric limits.

3.2 End-effector dynamics

The cable failure is modeled as a linear drop in the cable stiffness of the failed joint, i.e., $k_i \rightarrow 0$ over a small but finite time interval Δt_f . Thus if there is no change in winch angle as the cable fails, the tension in the corresponding cable falls. The full dynamic model of the rCDPR can be derived via the Lagrangian approach as

$$M\ddot{\mathbf{x}}_e + D(\dot{\mathbf{x}}_e)\dot{\mathbf{x}}_e = \mathbf{J}_w(\mathbf{x}_e, \mathbf{l}'_s) K_q [\mathbf{l}_p(\mathbf{x}_e, \mathbf{l}'_s) - \mathbf{l}'_c] \quad (3)$$

where end-effector pose is given by $\mathbf{x}_e = [x \quad y \quad \phi]^T$ and $\mathbf{J}_w(\mathbf{x}_e, \mathbf{l}'_s)$ is the pulling map or wrench Jacobian. The prismatic length, $\mathbf{l}_p$, comes

from inverse kinematics. Both are dependent on the current pose, $\mathbf{x}_e$, and the realized geometric attachment points, $\mathbf{l}'_s$, while the joint state stiffness matrix, K_q , is a diagonal matrix with the elements $(k_1, \dots, k_4)$. M is the mass/inertia matrix whose coefficients are informed by the mass and shape of the end effector. This formulation ignores the effects of Coriolis forces. The damping coefficients of the damping matrix D are minor and expected to be a result of surface friction alone. An effect of low cable stiffness and damping constants is that oscillations induced due to system dynamics in operation or during cable failure do not dissipate easily. A feedforward controller for such a system would be intractable. The complete dynamic system is shown in [Figure 4](#).

3.2.1 First order motion model

The states of the end-effector are given by $\mathbf{x} = [\mathbf{x}_e \quad \dot{\mathbf{x}}_e]^T$. The resulting non-linear system model is given by

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}(t), \mathbf{q}'(t), t) + \boldsymbol{\omega} \\ \tilde{\mathbf{y}} &= h(\mathbf{x}(t), t) + \mathbf{v}\end{aligned}\quad (4)$$

where $\boldsymbol{\omega}$ and $\mathbf{v}$ are zero mean Gaussian vector processes accounting for model and measurement delta correlated noise. The non-linear vector functions f and h emerge as

$$\begin{aligned}f(\mathbf{x}, \mathbf{q}') &= \begin{bmatrix} \dot{\mathbf{x}}_e \\ M^{-1} J_w(\mathbf{x}_e, \mathbf{l}'_s) K_q (l_p(\mathbf{x}_e, \mathbf{l}'_s) - \mathbf{l}'_c) - M^{-1} D \dot{\mathbf{x}}_e \end{bmatrix} \\ h(\mathbf{x}) &= [\mathbf{x}_e]\end{aligned}\quad (5)$$

where the explicit dependence on time has been dropped for ease of notation. The measurement model assumes that unbiased noisy position measurements are directly available, for instance by processing and filtering information from an on-board IMU or a range-based localization system.

3.3 Model assumptions for estimation and control

The motion models employed in the Extended Kalman Filters for the IMM ([Section 3.2](#)) and the LQR ([Section 5.4](#)) use Eq. 4 and 5 as the motion model. Three motion models are employed to describe the seven working modes 4.1. For each working mode, the effect of the corresponding cable failure is incorporated in the dynamic model through the loss of stiffness in the respective cables. However, the model does not consider the true stiffness constant value from the plant model but instead employs static parameters corresponding to the different failure modes. In addition, it does not consider the model for actuator dynamics, but instead accepts a measurement of the current joint state, $\tilde{\mathbf{q}}' = \mathbf{q}' + \mathbf{v}'$, where $\mathbf{v}'$ is modeled as a zero mean Gaussian delta correlated noise. On the other hand, the LQR assumes that the joint input to the motion model is the desired joint state. This is further discussed in [Section 5.4](#).

The kinematic feedforward controller does not use the dynamic models and instead relies on the computation of the forward and inverse kinematic solutions for the redundant system. The cable stiffness model used to compute the desired cable tensions is approximated to consider the true cable length instead of the free

cable length ($k = \frac{k_0}{l_p}$). This was done to remove the direct dependence of the cable stiffness parameter on l_c which is a control variable.

These assumptions add unmodeled uncertainties to the control and estimation problem. The complexity introduced by this bears resemblance to similar challenges associated with real-world experimental implementations. In addition, only a single measurement source (at the end-effector) is considered. It must be emphasized that the simulated measurement noise is unbiased and Gaussian. The deteriorating effect of measurement bias (such as IMU drift) on most estimation approaches is well known and well explained and are not considered in this work. Sensors such as ultrasonic sensors, GPS devices or visual systems can help eliminate bias. The selected measurement noise characteristics in this work was not based on simulations of a real measurement device.

4 Estimation of failure states

4.1 Failure mode classification

Cable failure in a CDPR reduces the degrees of freedom (DOFs) or the quality of the DOF (i.e. manipulability or dexterity) available at the end-effector. We control only 2 DOFs at the end-effector for the four-cable planar CDPR model in this paper. The total working modes considered can be described by 3 motion models (i) an over-constrained CDPR with 4 cables, (ii) a fully constrained CDPR with 3 cables and (iii) and under-constrained CDPR with two cables ([Figure 5](#)).

Therefore, there are 7 total working modes (mode 1) no cables have failed, (modes 2–5) only cable A, B, C, or D fail, (mode 6) cables A and B have failed, (mode 7) cables C and D have failed. If any other failure combination occurs, significant areas of the workspace become unreachable and are considered to be complete failure modes (i.e. cannot be recovered from). If the sliders were not constrained to geometric limits and free to move like mobile robots or drones, then these additional failure modes can also be considered.

4.2 Interactive multiple model estimation

An interactive multiple model filter (IMM) is a dynamic estimator ([Blom and Bar-Shalom \(1988\)](#)) which can be used when model changes appear suddenly or gradually over time, providing means for failure detection. The process flow of an IMM is shown in [Figure 6](#).

In general, the bank of filters in an IMM can be realized by a variety of estimation approaches, i.e., Kalman Filters (KF), Extended Kalman Filters (EKF), Unscented Kalman Filters (UKF), Particle Filters (PF), et cetera. In this study, we use an EKF formulation in which the error dynamics are approximated by a first order Taylor series expansion about the estimated state, $\hat{\mathbf{x}}$, i.e.

$$F = \frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}, \quad H = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \quad (6)$$

Here, $F_{6 \times 6}$ is determined numerically by a complex step derivative approximation and $H_{3 \times 6}$ is simply given by $[I_{3 \times 3} \quad 0_{3 \times 3}]$, since we directly measure the end-effector's position. In this study, the EKF

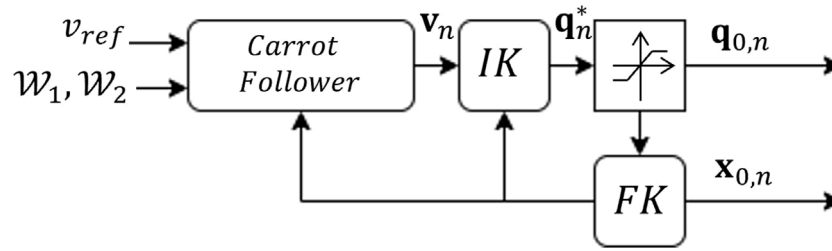


FIGURE 8
Feed-forward kinematic controller.

is applied in the discrete time domain; hence, the Euler-Maruyama discretization of the motion model yields.

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \underbrace{f(\mathbf{x}_{k-1}, \mathbf{q}'_{k-1})}_{\phi(\mathbf{x}_{k-1}, \mathbf{q}'_{k-1})} \Delta t + \boldsymbol{\omega}_k \sqrt{\Delta t} \quad (7)$$

$$\tilde{\mathbf{y}}_k = h(\mathbf{x}_k) + \mathbf{v}_k \quad (8)$$

where $\tilde{\mathbf{y}} \in \mathbb{R}^m$ is the measurement vector, $\boldsymbol{\omega}_k \sim \mathcal{N}(0, Q_k)$ denotes the process noise with covariance $Q_k \in \mathbb{R}^{n \times n}$, and $\mathbf{v}_k \sim \mathcal{N}(0, R_k)$ is the measurement noise with covariance $R_k \in \mathbb{R}^{m \times m}$. The sampling time is denoted by Δt , and k is the current time step.

Let $\mathcal{M}$ be the number of all working modes in a CDPR at the k th time step. Then, the input to the filter is the current measurement and mixed state estimate $\hat{\mathbf{x}}^{+0}$. The algorithm broadly consists of four stages (Gill (2019)).

1. *Interaction and Mixing*: The weights and the estimates from the last cycle are mixed as per their associated Markov transition probabilities given by the Markov transition matrix, A (also sometimes called the Markov switching matrix). The predicted probability for the filter to end up in mode j in each cycle, given that it was in the mode i during the previous cycle is given by

$$w_k^{(ij)} = \frac{1}{\tilde{c}_k^j} w_k^{(i)} a_{ij} \quad (9)$$

$$\tilde{c}_k^j = \sum_{i=1}^{\mathcal{M}} w_k^{(i)} a_{ij}$$

with $\tilde{c}_k^j$ being the normalization factor, and where a_{ij} is the Markov transition probability from mode i to mode j . As the true transition probabilities are unknown, the matrix A is a symmetric positive definite matrix of scalars that are tuned experimentally. The selected values have a significant impact on the response time and sensitivity in responding to cable failures. High values at the diagonals may cause the IMM to be sluggish whereas evenly distributed values along the whole matrix may cause the system to be overly sensitive to change. The mixed initial state estimate at the start of the current time step is provided by

$$\hat{\mathbf{x}}_k^{+0j} = \sum_{i=1}^{\mathcal{M}} w_k^{(ij)} \hat{\mathbf{x}}_k^{+i}$$

$$P_k^{+0j} = \sum_{i=1}^{\mathcal{M}} w_k^{(ij)} \left[\left(\hat{\mathbf{x}}_k^{+i} - \hat{\mathbf{x}}_k^{+0j} \right) \left(\hat{\mathbf{x}}_k^{+i} - \hat{\mathbf{x}}_k^{+0j} \right)^T + P_k^{+i} \right] \quad (10)$$

2. *Model Specific Filtering*: The mixed initial estimates are fed into the EKF and processed in two steps, i.e.

• Propagation:

$$\hat{\mathbf{x}}_k^{-j} = \hat{\mathbf{x}}_k^{+0j} + \mathbf{f}(\hat{\mathbf{x}}_k^{+0j}, \mathbf{q}'_k) \Delta t$$

$$P_k^{-j} = \Phi_k^j P_k^{+0j} \Phi_k^{jT} + Q_k \quad (11)$$

$$\Phi_k^j = I + F_k^j \Delta t$$

• And update:

$$K_k^{(j)} = P_k^{-j} H_k^{jT} \left[H_k^j P_k^{-j} H_k^{jT} + R_k \right]^{-1}$$

$$\hat{\mathbf{x}}_k^{+j} = \hat{\mathbf{x}}_k^{-j} + K_k \left[\tilde{\mathbf{y}}_k - h(\hat{\mathbf{x}}_k^{-j}) \right] \quad (12)$$

$$P_k^{+j} = [I - K_k H_k] P_k^{-j}$$

Here, '+' denotes the *a posteriori* estimate whereas '-' denotes the *a priori* quantity before the update. The likelihood of a measurement is then given by

$$p(\tilde{\mathbf{y}}_k | \hat{\mathbf{x}}_k^{-j}) = \frac{1}{[\det(2\pi E_k^{-j})]^{1/2}} \cdot \exp \left[-\frac{1}{2} \mathbf{e}_k^{-jT} (E_k^{-j})^{-1} \mathbf{e}_k^{-j} \right] \quad (13)$$

with $E_k^{-j} = H_k^j P_k^{-j} H_k^{jT} + R_k$

$$\mathbf{e}_k^{-j} = \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k^{-j}$$

and where $\mathbf{e}_k^{-j}$ is the estimation error.

3. *Model Probability Update*: Now, the model probabilities are updated via the likelihood with subsequent normalization, i.e.

$$w_k^j = w_{k-1}^j p(\tilde{\mathbf{y}}_k | \hat{\mathbf{x}}_k^{-j})$$

$$w_k^j \leftarrow \frac{w_k^j}{\sum_{i=1}^{\mathcal{M}} w_k^i} \quad (14)$$

4. *Combination*: Finally, the updated estimate is given by

$$\hat{\mathbf{x}}_k^{+} = \sum_{j=1}^{\mathcal{M}} w_k^j \hat{\mathbf{x}}_k^{+j}$$

$$P_k^{+} = \sum_{j=1}^{\mathcal{M}} w_k^j \left[\left(\hat{\mathbf{x}}_k^{+j} - \hat{\mathbf{x}}_k^{+} \right) \left(\hat{\mathbf{x}}_k^{+j} - \hat{\mathbf{x}}_k^{+} \right)^T + P_k^{+j} \right] \quad (15)$$

The weight vector, $\mathbf{w}_k$, displays the importance associated with each model in the bank. For example, if there are no cable failures, the IMM will determine the first model (mode 1) to have the largest weight. This vector also informs the balance of joint inputs to the controller as described in the next section.

5 Control framework

The objectives of the control framework are to not only stably manipulate the end-effector for trajectory tracking, but to do so with constant velocity and reliability of trajectory recovery upon cable failure. An earlier effort [Raman et al. \(2022\)](#) deployed a feedforward PD based control law for task space trajectory error minimization, but this was considered insufficient for addressing some finer requirements. For example, in 3D printing tasks the end-effector is required to (i) return to the point of failure before continuing, (ii) maintain constant velocity during trajectory tracking and, (iii) minimize dynamic oscillations at the end-effector after cable failure.

While designing a path following controller for a CDPR, the end-effector can generally be modelled as a holonomic robot. However, in most CDPRs, the ability to control the orientations of the end-effector is severely limited by geometric constraints. Reconfigurability can be an advantage here, but since our system model limits the motion of this attachment point to a straight line with finite dimensions, the orientations are only moderately more controllable than a traditional CDPR. Thus, we drop these from consideration for control and seek to only align the desired 2-D end-effector velocity vector with the tangent of the required trajectory.

In this section we present waypoint following algorithm integrated with a velocity level path following controller to recover the trajectory from the break-point (i.e. the precise location on the trajectory where the cable fails). This framework allows the recovery algorithm to be completely reliable even in the event of false positives by the estimator. In addition, this section also details the challenges of cascaded control and redundancy resolution in both the static space and task space while manipulating a kinematically- and actuation-redundant parallel manipulator.

5.1 Path following

We first discretize and parameterize the reference trajectory into a set of constant arc-length waypoints $\mathcal{W}_s$.

$$\mathcal{W}_s = \{\mathbf{p}_i = (x_i, y_i) \quad i \in 1 \rightarrow N_i\} \quad (16)$$

where N_i is the total number of waypoints. The constant arc-length waypoints are numerically determined from the path equation. This re-parameterization ensures that regardless of the chosen trajectory equation the distance between any two waypoints is never too large. Waypoints function as an ordered set of markers without time parameterization that can be dropped from the waypoint map as the robot traverses within a threshold of it. The robot always moves toward the next waypoint in the map regardless of cable failure or any other unpredictable disturbance at the end-effector that may temporarily navigate the robot away from the desired trajectory.

Next, we use a constant-velocity carot following controller to traverse straight line intersections between any given pair of waypoints.

The shortest distance from the current position of the robot end-effector origin, shown in [Figure 7](#) by frame $\{O\}$, intersects with the line segment at the point p_n and forms a normal vector $\mathbf{e}$ whose magnitude forms the cross-track error. If the projection of the robot frame origin $\{O\}$ lies outside of the line segment, the shortest distance is simply the distance to the closest end-point. Therefore, the cross-track error vector is determined by first projecting the robot origin on the line segment $\mathbf{p}$ through a dot product and capping the magnitude of the projection $\|\mathbf{p}\| \in [0, l_{\mathcal{W},i}]$ where $l_{\mathcal{W},i}$ is the length of the current line segment. If $\mathbf{o}_{\mathcal{W}}$ is the vector between frame $\{O\}$ and $\mathcal{W}_i$, then the cross-track error vector is always given by $\mathbf{e} = \mathbf{o}_{\mathcal{W}} + \|\mathbf{p}\|\hat{\mathbf{p}}$. The reference velocity is a constant and is given by v_{ref} . Thus the reference velocity vector which lies along the current line segment is $\mathbf{v}_{ref} = v_{ref}\hat{\mathbf{p}}$. The desired direction of the velocity vector is given by $\mathbf{v}_d = \mathbf{e} + \mathbf{v}_{ref}$. The final desired velocity vector has its velocity always capped at v_{ref} and is therefore given by $\mathbf{v} = v_{ref}\hat{\mathbf{v}}_d$.

As the cross-track error approaches zero, the desired velocity vector and the reference velocity vector are more closely aligned. The joint states required to achieve this velocity are computed through redundancy resolution-based inverse kinematics. When the end-effector reaches within a certain threshold of the next waypoint, the waypoint map $\mathcal{W}_s$ updates by dropping the first value from the list. This strategy allows the robot to always move towards the second waypoint in $\mathcal{W}_s$, while the first waypoint serves as a marker for the connecting line segment to be tracked. If the robot snaps or moves off track, it still moves towards the second waypoint ensuring that the trajectory we want to track is traversed approximately from where it deviates. The choice of arc length chosen above determines the maximum size of the missed line segment.

5.2 Inverse kinematics and redundancy resolution

For a given end-effector pose, there are infinitely many solutions in the joint space of kinematically redundant CDPR. Considering that this system has two kinds of redundancies, actuation and kinematic redundancy, the solutions for determining the optimal joint states can be quite complex. In this study, a feedforward kinematic controller was applied and as a result, the redundancy resolution schemes and joint determination use kinematic models.

5.2.1 Infeasibility avoidance

Singularity avoidance is an important part of the kinematic redundancy resolution control that determines the joint states. With CDPRs, it is additionally of importance to avoid infeasibilities in the wrench space. i.e., the configurations of the robot should always maintain positive cable tensions. Recent approaches use maximization of the volume of the wrench polytope [Bosscher et al. \(2006\)](#) (or the Available Wrench Set, AWS) as the cost function for ensuring this feasibility [Rasheed \(2019\)](#); [Xiong et al. \(2022\)](#). The available wrench polytope is a description of the wrench or manipulability ellipse that, unlike Yoshikawa's measure of manipulability [Yoshikawa \(1985\)](#), provides a more accurate and scaled representation of the true wrench space. This is especially

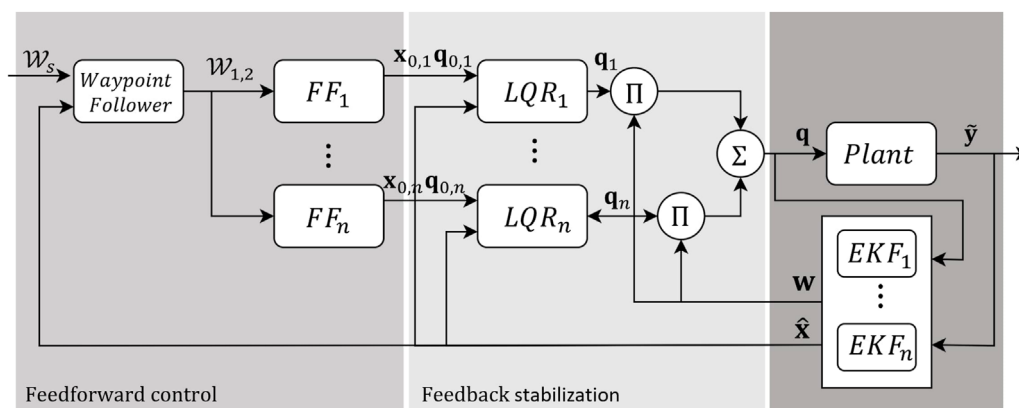


FIGURE 9
LQR stabilization and input mixing.

TABLE 1 Dynamic Model Parameters.

Parameters	Notation	Value	Units
End-effector dimensions	$b \times h$	0.16×0.08	$m \times m$
Base Span	l_b	0.56	m
Mass	m	0.5	kg
Cable Stiffness Proportionality Constant	k_0	5	N
Surface friction coefficient	μ	0.1	-
Gravitational constant	g	9.81	m/s^2
Minimum Cable Tension	τ_i	0.5	N
Time period for cable failure	Δt_f	0.1	s
Process noise covariance	Q	$\begin{bmatrix} 10^{-10} I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 10^{-3} I_{3 \times 3} \end{bmatrix}$	$\{m, rad, s\}$
Measurement noise covariance	R	$10^{-5} I_{3 \times 3}$	$\{m, rad\}$

useful for CDPRs, where negative joint forces cannot be realized. However, the calculation of the AWS is computationally expensive and can still result in solutions that settle within local minima. For this reason, we choose to look at the traditional manipulability and wrench feasibility indices together by selecting the pareto-optimal solution maximizing both.

Yoshikawa's measure of manipulability quantifies the twist and wrench capabilities of the end-effector. In 3 dimensions, the manipulability for translational and rotational motion is considered independently, and this holds true for 2 dimensions as well. Since we do not wish to control the yaw motion of the end-effector, we only consider the eigenvalues $\sigma \in \{\sigma_x, \sigma_y\}$ of the twist Jacobian $J_w^T(x_e, I_s)$ corresponding to x and y . The measure of manipulability is then given by

$$\lambda = \frac{\sigma_{min}}{\sigma_{max}}, \quad 0 \leq \lambda \leq 1 \quad (17)$$

For an n -DOF CDPR with m cables, the wrench quality index κ indicates the quality of wrench feasibility of the system. For a fully constrained cable robot it is simply the ratio of the smallest and

largest tensions required to maintain a pose [Pott and Bruckmann \(2013\)](#). For a redundantly constrained CDPR, it is the largest value that this ratio can take. In simple terms, the more evenly distributed the tensions in the CDPR are to maintain the pose, the better its wrench quality index will be. This index is calculated as follows.

- Let J_w be the wrench Jacobian for an n -DOF CDPR with the dimensions $n \times m$ where the number of cables, $m \geq n + 2$. Most poses in this workspace can be achieved with only $n + 1$ cables, so many feasible solutions of tensions may involve setting tensions in $m - n - 1$ cables to zero and lead to a wrench quality index of zero. However, a better tension set can be determined for the same pose.
- Assume a matrix $\mathcal{J}_i$ whose columns are a subset of the columns of J_w such that $\mathcal{J}_i$ is J_w with a set of $m - n - 1$ columns dropped. Thus for m columns in J_w there will be ${}^m C_{n+1}$ combinations of $\mathcal{J}_i$.
- Let $n_i = \text{null}(\mathcal{J}_i)$. If every element in n_i has the same sign, then this vector contains the minimum set of cables that

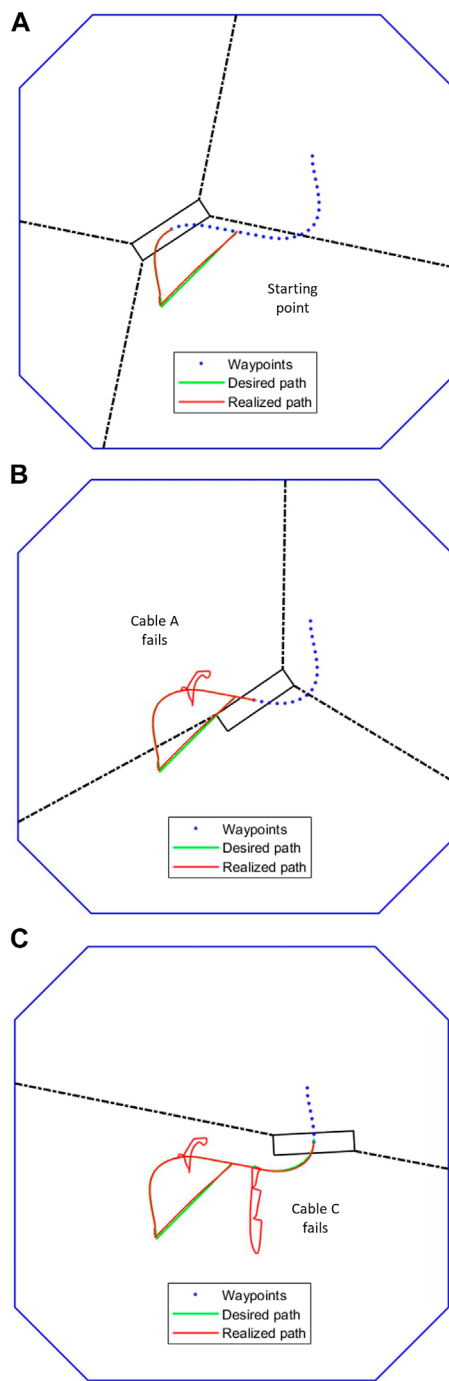


FIGURE 10
Trajectory tracking and recovery after cable failure (A) No failures yet (B) After 1 cable fails (C) After 2 cables fail.

must be tensioned in order to maintain a pose (the tension in the remaining cables is zero). Expand the vector n_i such that zeros are inserted into the vector for every dropped column. For example, if $n = 2$ and $m = 4$, and $\mathcal{J}_1$ is a reduced matrix of J_w with the first column dropped. Then $n_1 = [0 \ n_{1,1} \ n_{1,2} \ n_{1,3}]^T$

- Assemble all such vectors n_i whose elements contain the same sign and eliminate the negative signs $n_i = -n_i$ iff $n_{i,j} < 0 \forall j$.
- Sum all the assembled n_i that meet these criteria into a single vector z of the dimensions $m \times 1$.
- Drop all the elements in z that are zero. If the number of remaining elements are less than $n + 1$ then the wrench quality index, $\kappa = 0$. If κ is 0, then the system is not wrench feasible.
- If the number of elements is greater than or equal to $n + 1$, then

$$\kappa = \frac{\min(|z|)}{\max(|z|)} \quad (18)$$

For a fully constrained CDPR, the null space of the wrench Jacobian in one dimensional, so Eq. 18 is directly applied on the null vector (if the elements all contain the same sign. Otherwise $\kappa = 0$).

Cable failure can cause a sudden change in the instantaneous static workspace of the robot. The following waypoint on the desired trajectory can often lie outside of this new workspace. The sliders form the kinematically redundant joints that drive the robot away from singularities where necessary, but sometimes if the desired velocities at the end-effector are large enough or at an unusual angle, the sliders may be unable to move fast enough to compensate and the controller inadvertently pushes the robot into a singularity. To prevent this, we constrain the magnitude of the maximum allowable velocity with dependency on κ ,

$$\mathbf{v} = v_{\max} \sqrt{\kappa} \hat{\mathbf{v}} \quad (19)$$

where $\hat{\mathbf{v}}$ is the unit velocity direction vector. This is only utilized when the robot kinematics have placed the robot away from the trajectory, such as during the initialization. In normal operation for trajectory tracking the tracking velocities are slow enough that this is not a major concern.

5.2.2 Redundancy resolution

This redundancy resolution scheme is applied on kinematic equations to determine an optimal position level joint solution, $\mathbf{q}^* = [\mathbf{l}_s^* \ \mathbf{l}_c^*]$. These inputs may not be dynamically feasible, nor is the feedback being actively measured, so they are instead locally thresholded to a velocity-limited value $\mathbf{q}_0 = [\mathbf{l}_{s,0} \ \mathbf{l}_{c,0}]$. The optimal joint slider positions $\mathbf{l}_s^*$ are determined from an objective function that seeks to maximize the manipulability and wrench quality at the end-effector. This aids in avoiding singularities as the sliders travel. The joint sliders are velocity-limited through a velocity thresholding function (Eq. 20). For the i th model (subscript i is dropped),

$$\begin{aligned} \mathbf{l}_s^* &= \max_{\mathbf{l}_s}(\kappa, \lambda) \\ \mathbf{l}_{s,0}^{k+1} &= \mathbf{l}_{s,0}^k + \Delta \mathbf{l}_{s,0} \\ \Delta \mathbf{l}_{s,0} &= \min\left(\max\left(\mathbf{l}_s^* - \mathbf{l}_{s,0}^k, v_{s,\max} \Delta t\right), v_{s,\min} \Delta t\right) \end{aligned} \quad (20)$$

where κ and λ are the wrench quality index and manipulability measure respectively. The slider velocity is given by v_s . Feasible slider positions are required to calculate optimal cable winch positions for a given time-step. Ideally, slider positions and cable winching should be controlled in alternate timesteps in order to utilize slider measurements to calculate the desired cable winch position. However, by thresholding the slider velocities to $|v_{s,\min}| = |v_{s,\max}|$ such that the control actions $\mathbf{l}_{s,0}$ lie within the linear region

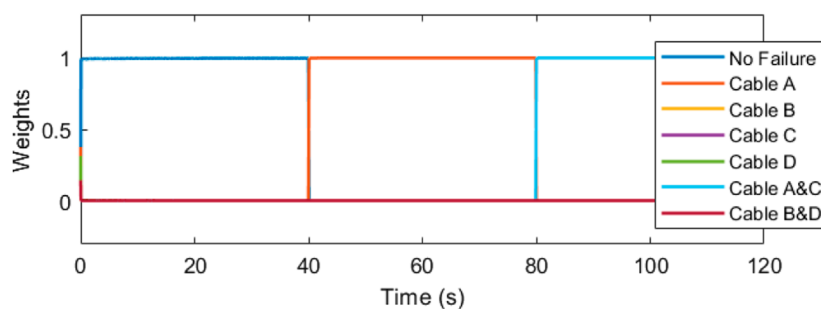


FIGURE 11

The weights denote the dominance of the corresponding cable failure case.

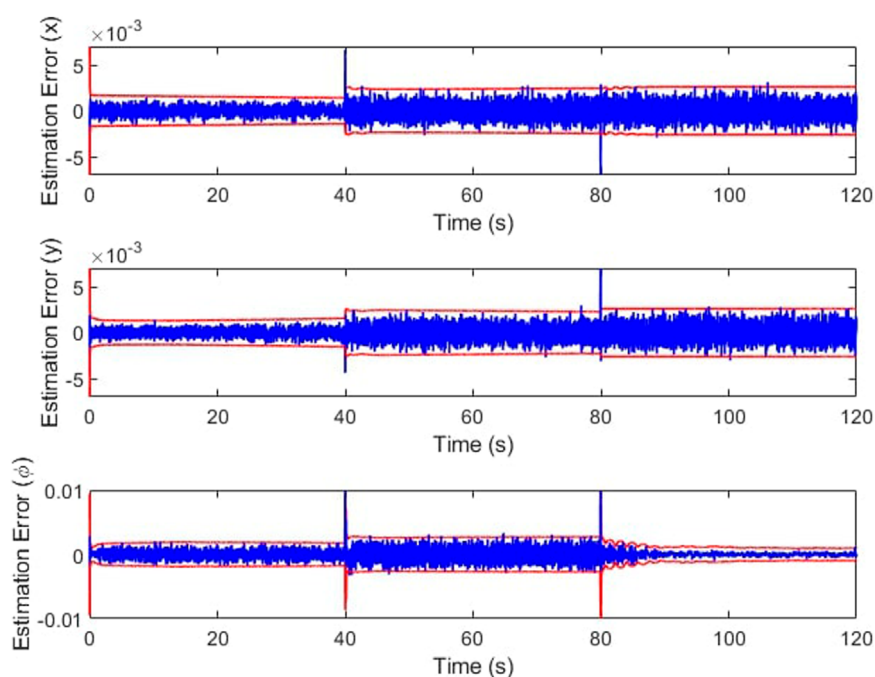


FIGURE 12

Estimation error over time.

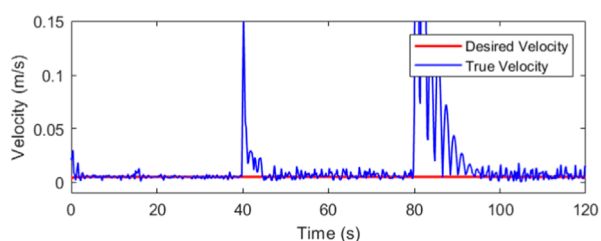


FIGURE 13

Desired and actual velocities at the end-effector.

of the slider dynamics allows us to approximate control actions for sliders and cable winches within the same time-step. Any inaccuracies can be subsumed within model disturbances. The LQR

feedback controller allows for enough leeway to compensate for these inaccuracies.

For the final motion model, the DOF at the end-effector reduces to one and the manipulability ellipse reduces to a line, making singularity avoidance a non-issue. Wrench feasibility in a single dimension is also trivially maintained. For this case, the objective function for the cables manipulates the end-effector along the single axis in the body frame, while the slider position depends on minimizing the error in a skewed axis (which is parallel to the direction of slider motion in the robot frame). Even for CDRPs with higher DOFs, l_s can be made to depend on a multi-objective function that simultaneously manipulates previously-uncontrollable DOFs while maximizing workspace and pose quality.

Cable failure can sometimes lead to slack in the remaining cables. While this property is not explicitly modeled in the system kinematics, its destabilizing effect can clearly be identified if the

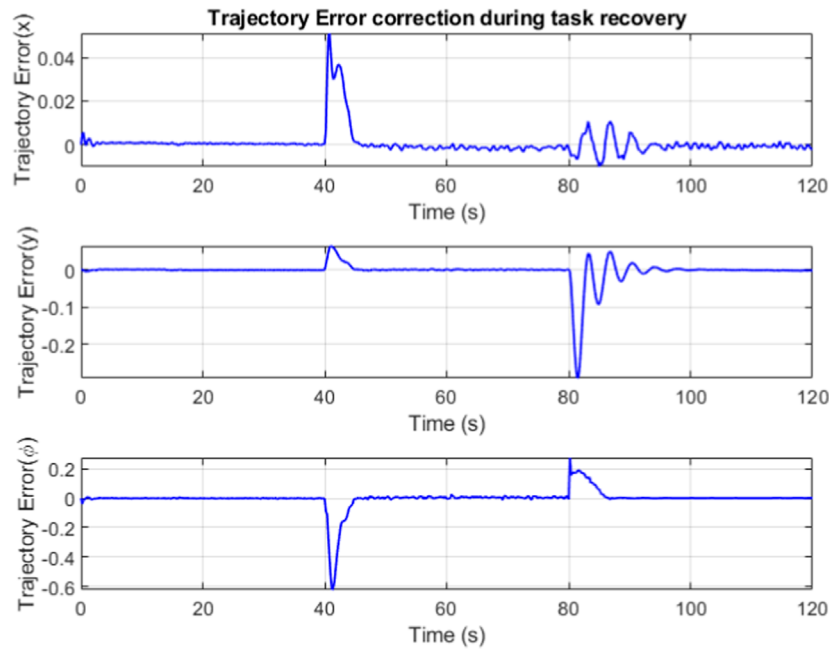


FIGURE 14
Trajectory error over time.

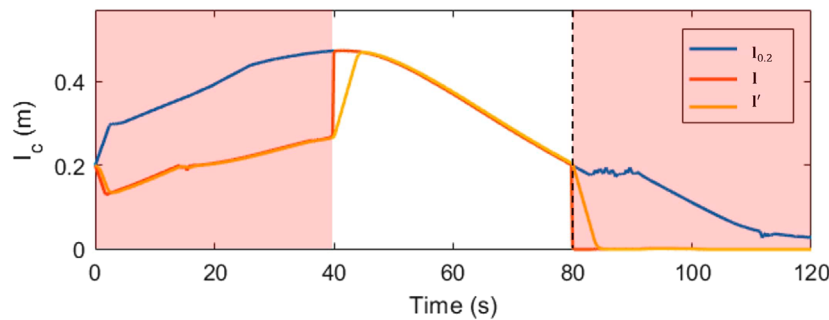


FIGURE 15
Desired, true and measured slider positions over time.

true winch position l_c is larger than the true cable length. This can be prevented by setting a minimum pretension τ_{min} and utilizing the static redundancy to extract the desired tension in the cables. Given the desired end-effector position and $\mathbf{l}_{s,0}^{k+1}$, we determine the minimum positive tensions required to maintain this pose.

$$\boldsymbol{\tau}^* = \min_{\boldsymbol{\tau}} \boldsymbol{\tau}^T \boldsymbol{\tau}, \exists J_w(\mathbf{x}_e^{k+1}, \mathbf{l}_{s,0}^{k+1}) \boldsymbol{\tau} = 0 \quad (21)$$

The optimal winch positions, $\mathbf{l}_c^*$, are then determined in a straightforward fashion by considering the cable stiffness model,

$$\mathbf{l}_c^{*,k+1} = \frac{K_q \mathbf{l}_p^{k+1}}{(\boldsymbol{\tau}^* + K_q)} \quad (22)$$

where $\mathbf{l}_p$ is the prismatic length or true (steady-state) length of the cables, determined from inverse kinematics. The joint angle velocities are not thresholded, so $\mathbf{l}_{c,0} = \mathbf{l}_c^*$.

5.3 Forward kinematics

Given the desired joint state $\mathbf{q}_{0,n}$ of the n th working mode, the desired steady state end-effector pose $\mathbf{x}_{0,n}$ can be determined through forward kinematics which is implemented by minimizing the potential energy of the system.

$$\mathbf{x}_{0,n} = \operatorname{argmin}_{\mathbf{x}_{0,n}} U = \frac{1}{2} (\mathbf{l}_p - \mathbf{l}_{c_{0,n}})^T K_q (\mathbf{l}_p - \mathbf{l}_{c_{0,n}}) \quad (23)$$

$$\text{s.t. } \mathbf{l}_p - \mathbf{l}_{c_{0,n}} > 0 \quad (24)$$

where $\mathbf{l}_p$ is true cable length calculated for each value of $\mathbf{x}_{0,n}$. The cable tensions given by $K_q(\mathbf{l}_p - \mathbf{l}_{c_{0,n}})$ must always be positive. If this condition is not met, it indicates that the initial joint state provided was not feasible which is generally a result of one or more cables being slack. Fortunately, in the feedforward control setting considered in this work, the desired winch position

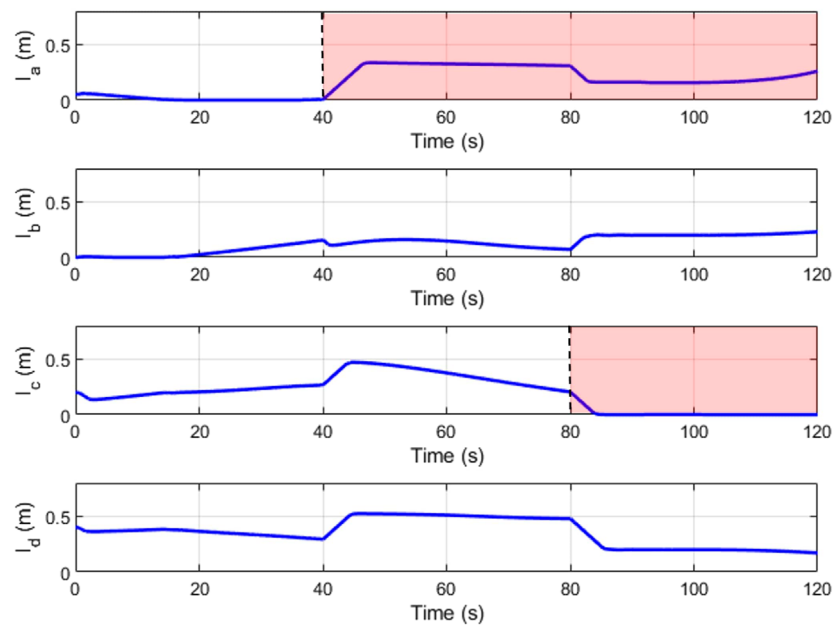


FIGURE 16

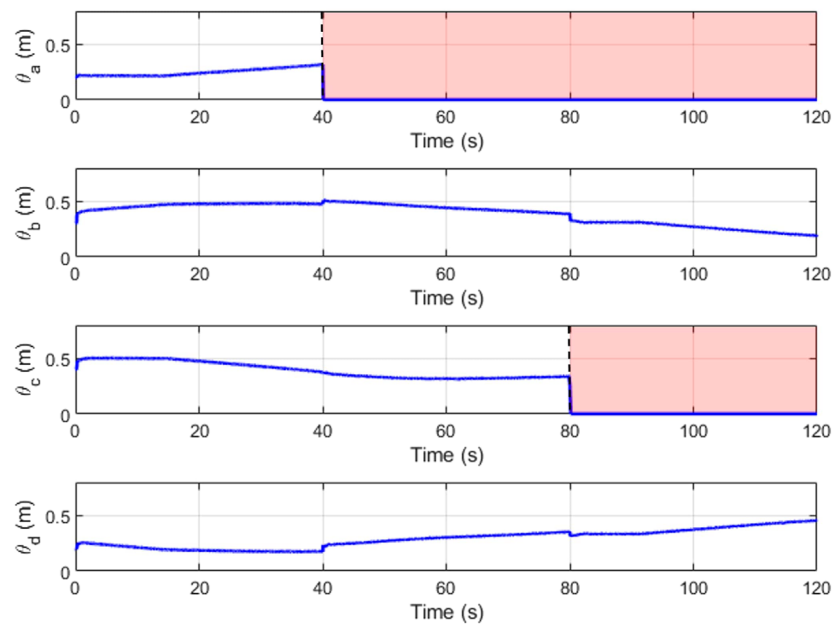
Slider positions (l_s) over time.

FIGURE 17

Winch positions (l_c) over time.

$l_{c_{0,n}}$ employed in the calculation of the forward kinematics is always less than the prismatic cable length l_p . This calculation is not influenced by the true input l_c that is sent to the plant. Thus, although cable slackness can exist in the plant model, it is not encountered during the calculation of the forward kinematics.

The task recovery algorithm has a parallel bank of trajectory tracking controllers, each corresponding to a working mode. Each working mode, n , runs an independent kinematic controller that propagates the desired joint inputs in time such that it is the only working mode. The controllers use simple kinematic models for the different redundant CDPR motion models.

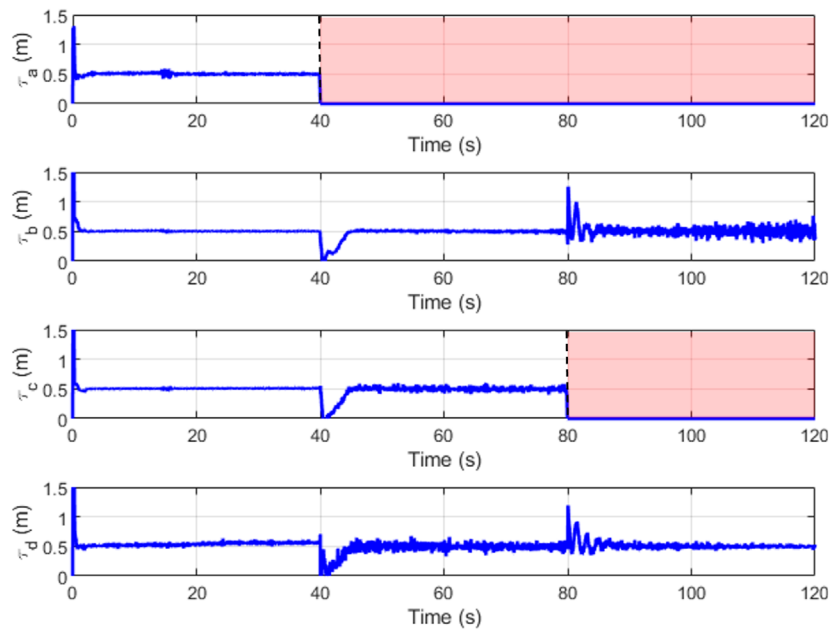


FIGURE 18
True cable tensions (τ) over time.

The control loop accepts the current joint inputs, $\mathbf{q}$ to the plant and determines the steady-state robot pose through forward kinematics for the n th model (Figure 8). The waypoint tracking algorithm accepts the current state estimate, $\hat{\mathbf{x}}$ to update the waypoint and reference trajectory information if necessary. Through the carrot following algorithm, the desired task space state is determined and the redundancy resolution scheme is applied to determine the desired joint state, $\mathbf{q}_{0,n}$. This is propagated through forward kinematics again to determine the steady-state end-effector pose, $\mathbf{x}_{0,n}$.

5.4 LQR stabilization

Although the full system is highly non-linear, local stabilization (vibration dampening) is performed utilizing a Linear Quadratic Regulator. The LQR assumes the system model in Eq. 5 and ignores the actuator dynamics. This does not cause unwelcome destabilizing effects as the update frequency of the LQR is low compared to the dynamic joint controllers. The system model is linearized around the desired steady-state operating point $(\mathbf{x}_{0,n}, \mathbf{q}_{0,n})$ that results from a combination of forward and inverse kinematics in the feedforward controller described in Section 5.2. The local system dynamics then arise as

$$\begin{aligned} \delta \mathbf{x}_{n,k+1} &= A_n \delta \mathbf{x}_{n,k} + B_n \delta \mathbf{q}_{n,k} \\ A_n &= \left. \frac{\partial f_n(\mathbf{x}, \mathbf{q})}{\partial \mathbf{x}} \right|_{(\mathbf{x}_{0,n}, \mathbf{q}_{0,n})}, \quad B_n = \left. \frac{\partial f_n(\mathbf{x}, \mathbf{q})}{\partial \mathbf{q}} \right|_{(\mathbf{x}_{0,n}, \mathbf{q}_{0,n})} \end{aligned} \quad (25)$$

where k is the time-step index within the LQR feedback control loop. Please note that the desired steady-state operation point $(\mathbf{x}_{0,n}, \mathbf{q}_{0,n})$ is also time-varying, but at a much slower update rate than the

LQR loop. The slow update rate of the feedforward controller that provides the desired steady-state operating point is useful here as the redundancy resolution algorithm is the most computationally expensive piece of the framework. Therefore, a zero-order hold assumption without loss of generality is implied in Eq. 25. The dynamic correction $\delta \mathbf{q}_{n,k}$ for the kinematically optimal joint input $\mathbf{q}_{n,k}$ is now determined via a quadratic cost function, i.e.

$$\delta \mathbf{q}_{n,k} = \operatorname{argmin}_{\mathbf{q}_n} \sum_{k=1}^{\infty} \delta \mathbf{x}_{n,k}^T \mathcal{Q} \delta \mathbf{x}_{n,k} + \delta \mathbf{q}_{n,k}^T \mathcal{R} \delta \mathbf{q}_{n,k} \quad (26)$$

subject to the dynamics in Eq. 25 and where $\mathcal{Q}$ and $\mathcal{R}$ are positive semi-definite and positive definite weighting matrices, respectively. The tuning of these terms depends on the desired overall impact of the corresponding states on the cost function as well as limits on the control effort. Here, the structure for $\mathcal{Q}$ and $\mathcal{R}$ has been chosen as diagonal. Furthermore, the diagonal terms for the joint inputs ($\mathcal{R}$) are weighted equally. For $\mathcal{Q}$, the entries corresponding to the velocities in x and y outweigh the other values, as it is desired to minimize velocity offsets dominantly. The resulting controller is then given by

$$\mathbf{q}_{n,k} = \mathbf{q}_{0,n} - \mathcal{K}_n \delta \hat{\mathbf{x}}_{n,k} = \mathbf{q}_{0,n} - \mathcal{K}_n (\hat{\mathbf{x}}_k - \mathbf{x}_{0,n}) \quad (27)$$

where $\mathcal{K}_n$ is the LQR gain resulting for the n th model while the estimated state $\hat{\mathbf{x}}_k$ from the IMM framework has been employed as the feedback variable for the controller. When tuned well, the LQR stabilization not only mitigates dynamic disturbances due to the elasticity of the cables during normal operation, but also dampens significant oscillations from cable breakage quickly. An additional beneficial effect from the LQR inclusion is observed, as even slackness after failure is more efficiently eliminated. Here, the kinematic solution provides a new equilibrium point, and the LQR

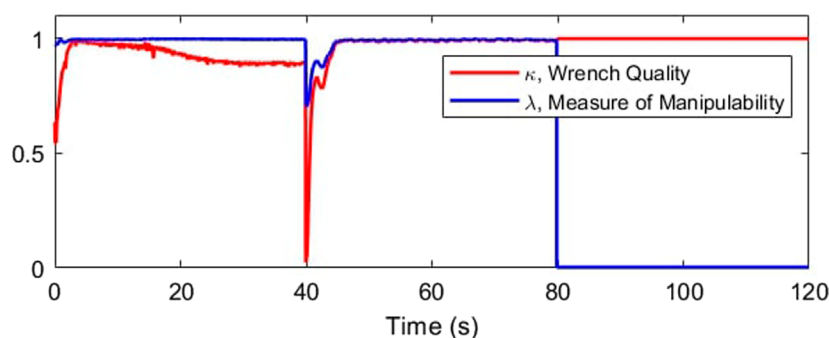


FIGURE 19
Maximizing workspace quality and manipulability.

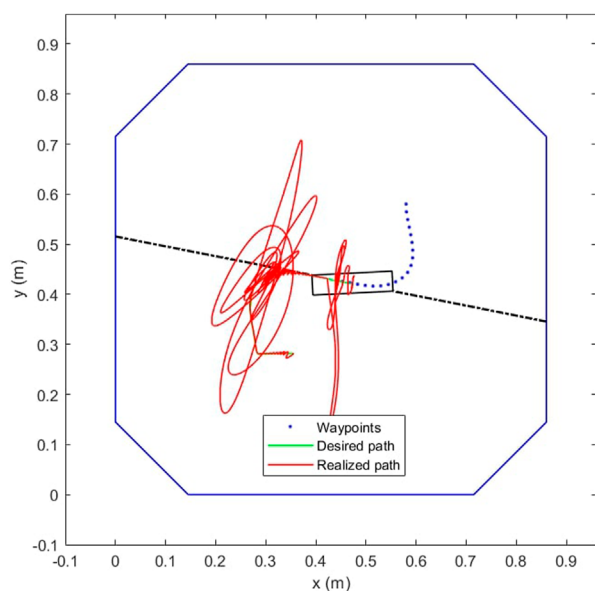


FIGURE 20
Trajectory tracking and recovery without LQR for oscillation damping.

inherently rushes the system to arrive at this new stable kinematic solution, thus significantly aiding in maintaining positive cable tensions. For the over-constrained (4-cable) and fully constrained (3-cable) rCDPR models, only $\theta_{0,n}$ is subject to LQR stabilization as this is enough to dampen perturbations in both x - and y -directions. For the under-constrained (2-cable) model, the redundant joints $\mathbf{l}_{0,n}$ are also utilized to dampen disturbance components orthogonal to the cable direction.

5.5 Input mixing

The bank of controllers independently determine the next steady-state input for each failure mode. The bank of LQRs generate optimal joint inputs that are mixed with the weighting matrix to the plant model. The final mixed input gives precedence to the

dominant working mode identified by the IMM. Due to this mixing, the final mixed input from this stage will always be subject to some perturbation. However, this perturbation is considered a reasonable trade-off to accommodate a solution that is robust and reliable during cable failure conditions.

The mixed joint input is applied to the dynamic model (plant) and the measurements of the end-effector state inform the state estimation and weight vector through the IMM. The complete algorithmic framework is described in [Figure 9](#).

6 Implementation and results

6.1 Simulation framework

The plant model described in [Section 3](#) is perturbed with a process and measurement noise to mimic unmodeled dynamics. The cables are modeled as linear springs (where stiffness is a function of free cable length) with no mass and no sag, while all other model parameters are constants. Cable failure is modeled as a drop in cable stiffness to zero over a finite time. If the stiffness is zero, the cable can have no tension and thus has no impact on the platform dynamics. In addition, if the dynamic input to the system causes a situation where $l_c > l_p$, the tension in the dynamic equation is set to zero to mimic cable sag. The plant dynamics and estimation run at a frequency of 1,000 Hz while the closed loop LQR feedback is run at 50 Hz. The feedforward kinematic model, which is the most computationally-intensive system, is run at 2 Hz. The entire framework is modeled in MATLAB. The model parameters are given in [Table 1](#).

6.2 Task recovery

[Figure 10](#) demonstrates the application of the complete task recovery algorithm. The red trajectory shows the true path of the center of the end-effector while the green trajectory shows the desired path. The blue dots are the predefined waypoints. The failure of the first cable causes the end-effector to jump away from its current path, but the algorithm immediately responds by rearranging the remaining three winches and pulling the end-effector back. This breakaway from the trajectory and subsequent

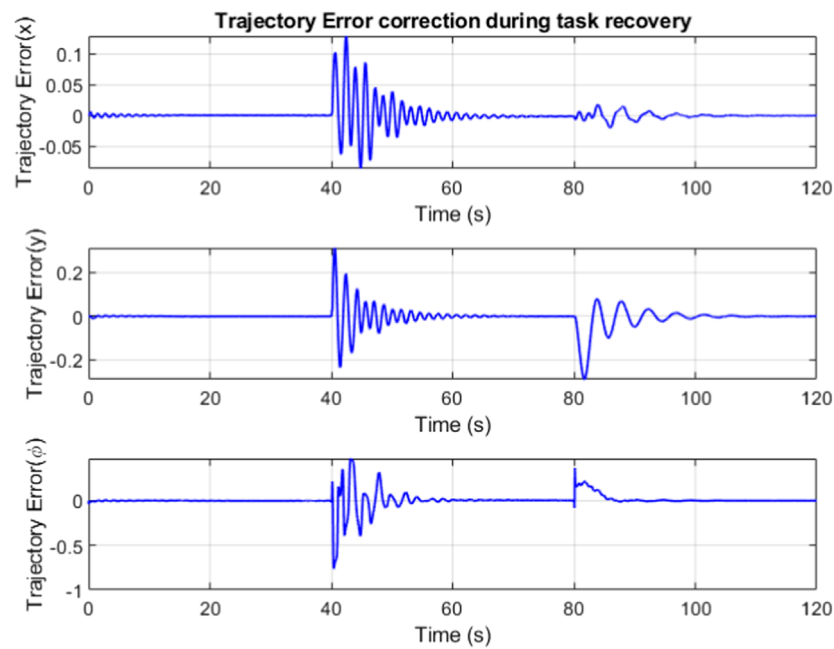


FIGURE 21
Trajectory error for the case without LQR for oscillation damping.

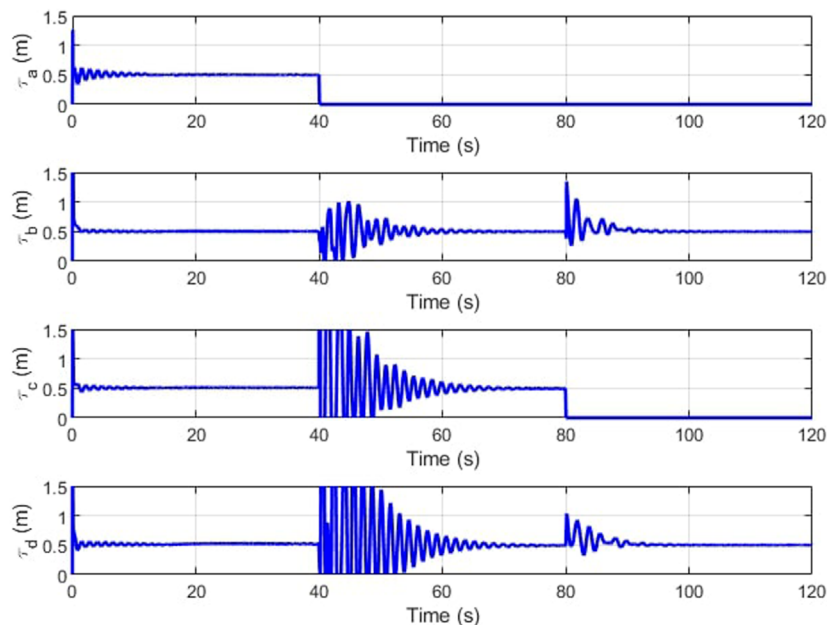


FIGURE 22
Cable tension for the case without LQR for oscillation damping.

recovery with three cables is shown by the first kink in the red line part way through the trajectory in [Figure 10B](#). The failure of the second cable and subsequent task recovery is seen further down the trajectory in [Figure 10C](#).

The IMM correctly identifies the failure, mixing and switching the inputs as needed to automatically recover the trajectory tracking.

In [Figure 11](#), the weights vector correctly assigns the largest values to the current working mode. At the forty second mark, we can see that the weight vector has correctly identified the cable A failure, followed by the correct identification of the cable C failure at the 10 s mark. The estimation error plots are in [Figure 12](#).

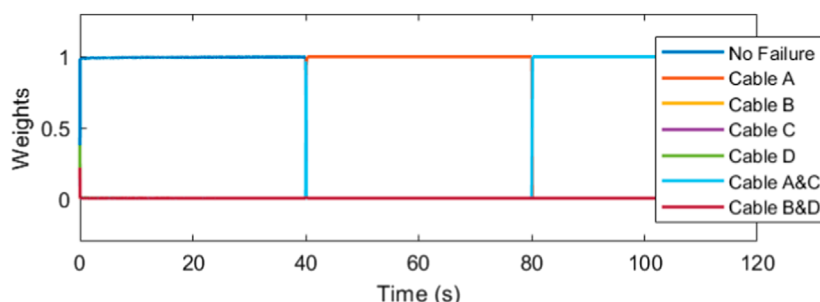


FIGURE 23
Model identification is unaffected.

The robot is maintained at constant velocity throughout its trajectory (Figure 13). The desired velocity during trajectory tracking is 0.005 m/s or 5 mm/s which is a reasonable speed for the dimensions of the workspace. During recovery, the end-effector velocity is not a constraint, so the controller, mainly through the efforts of the LQR, attempts to move the end-effector as fast as dynamically feasible.

The trajectory error is minimized by the combined efforts of feedforward controller and the LQR. The banded regions in Figure 14 represent sections where the robot is in recovery mode, attempting to minimize the trajectory error while compensating for the changes in slider and winch position.

The robot does not enter singularities in the feedforward controller. Figure 15 shows the plots of a single slider position (slider C) for the second motion model. In the example trajectory shown in Figure 10 this model is weighted highest between the 40 s and 80 s timestamps. Thus the mixed output from the LQR, I begins to weight heavily in the favour of the control input $I_{0,2}$ from model 2, while the true slider position I' begins to catch up. When cable C breaks at the 80th second, this model begins to fall out of favor and the slider position deviates from this value.

Figures 16, 17 depict the change in slider position and position angle of winch respectively. The shaded regions represent the failed state of the respective cable. The values of the slider and winch positions in these regions are perturbed by the combined attempts of the optimization to continue modulating this actuator and the weighted input mixing but has no effect on the end-effector as the corresponding cable is broken. This is seen clearly in the plots observing the true tension in the cables in Figure 18. The position values are not automatically settled to zero to account for cases where the cable failure may have been misidentified by the estimator.

A minimum value of pre-tension in the cables is maintained at 0.5 N as a good practice to eliminate the chances of encountering cable slackness. This tension is seen to be maintained throughout the run with the minimum required tensions in the remaining cables to propagate the end-effector along the trajectory. This demonstrates that the tension minimization scheme is working well. When a cable snaps, there are brief moments of cable slackness while the tensions re-distribute as the sliders reposition themselves and the winches

wind within their designed dynamics. These dynamics do not affect the steady-state input from the feedforward controller and thus the LQR bears the burden of recovery.

The redundancy resolution optimization for the slider positions maximizes the manipulability and wrench quality. In Figure 19 we see these properties hold high values in the first two-thirds of the plot. These are the regions where the optimization-determined slider positions are actively applied. This optimization is no longer relevant for the two cable cases, where the slider position instead functions to replace the lost degree of freedom.

6.3 Comparison study

For completeness, this section compares the results of the previous section with a feedforward controller applied directly on the system without the LQR. The update rate of the feedforward controller is 50 Hz and all other parameters are kept the same. The trajectory results are presented in Figure 20.

The trajectory error plot in Figure 21 indicates that the oscillations do eventually dampen and the kinematic controller is generally successful after this due to the slow velocities at the end-effector.

The cable breakage causes the end-effector to swing causing periodic slackness in the cable tensions (Figure 22).

The estimation depends solely on the end-effector measurements, so it is unaffected by the controller dynamics as seen in Figure 23.

7 Discussions

In this work we create an end-to-end control and estimation framework to simultaneously accomplish cable failure diagnosis and detection and cable failure tolerant control through the utilization of an interactive multi-model adaptive estimation algorithm and a bank of controllers to provide the mixed inputs. This framework is applied on a planar reconfigurable cable driven parallel robot (rCDPR); a CDPR with the ability to geometrically reconfigure its cable winch attachment points through the utilization of linear

sliders. We are able to create a system that is (i) automatic: it does not require the faulty actuators to register a fault during the event of a cable failure; (ii) robust; although it requires only measurement data from the end-effector to estimate the correct failure state, this data can be quite noisy and the models utilized in the filters do not need to be perfect (iii) reliable; when the correct failure state is weighed higher, the corresponding inputs associated with this state have a higher impact on the inputs to the system which improves the overall estimate. However, if an incorrect state is weighed high, the corresponding inputs will create dynamics that do not match the expected system profile which allows the correct state to eventually be identified. In addition, the inclusion of the waypoint following algorithm ensures that the robot always reconnects with the trajectory at the approximate location where it deviated. The utilization of a feedforward controller over a dynamic feedback control helps improve the overall stability of the system. In addition, since the feedforward framework does not require the current state of the system, it can also be designed to be an offline controller that can help improve computation times and realize a real-time control system. The LQR feedback controller ensures not only that the robot dynamics due to elasticity during normal operation are minimized but also attempts to reconfigure the system as quickly as possible when a failure mode is estimated. Although the input mixing at this stage can cause some perturbation at the end-effector, this effect is considered a reasonable trade-off for the overall advantages of dynamic stabilization that the controller offers.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

MS provided the conceptualization and the theoretical framework for the research presented in this work and supervised

its development. AR modelled the system, performed the coding, analyses and development of the complete control framework. VK and IW served as advisors to AR, supervised the alignment of the presented work with the larger context of the underlying NSF project and contributed equally towards the funding of this research. All authors provided frequent intellectual feedback and approved the paper for publication.

Funding

This work is supported in part by the U.S. National Science Foundation under NRI grant 1924721.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/frobt.2023.1070627/full#supplementary-material>

References

- Baklouti, S., Courteille, E., Lemoine, P., and Caro, S. (2021). Input shaping for feed-forward control of cable-driven parallel robots. *J. Dyn. Syst. Meas. Control* 143. doi:10.1115/1.4048354
- Begey, J., Cuvillon, L., Lesellier, M., Gouttefarde, M., and Gangloff, J. (2018). Dynamic control of parallel robots driven by flexible cables and actuated by position-controlled winches. *IEEE Trans. Robotics* 35, 286–293. doi:10.1109/tro.2018.2875415
- Blom, H. A., and Bar-Shalom, Y. (1988). The interacting multiple model algorithm for systems with markovian switching coefficients. *IEEE Trans. Automatic Control* 33, 780–783. doi:10.1109/9.1299
- Bosscher, P., and Ebert-Uphoff, I. (2004). Wrench-based analysis of cable-driven robots. In *IEEE international conference on robotics and automation, 2004. Proceedings. ICRA'04. 2004 (IEEE)*, 5, 4950–4955.
- Bosscher, P., Riechel, A. T., and Ebert-Uphoff, I. (2006). Wrench-feasible workspace generation for cable-driven robots. *IEEE Trans. Robotics* 22, 890–902. doi:10.1109/tro.2006.878967
- Bouchard, S., Gosselin, C., and Moore, B. (2010). *On the ability of a cable-driven robot to generate a prescribed set of wrenches*.
- Boumann, R., and Bruckmann, T. (2022). Simulation and model-based verification of an emergency strategy for cable failure in cable robots. *Actuators (MDPI)* 11, 56. doi:10.3390/act11020056
- Caro, S., and Merlet, J.-P. (2020). "Failure analysis of a collaborative 4-1 cable-driven parallel robot," in *European conference on mechanism science* (Springer), 440–447.
- Caverly, R. J., and Forbes, J. R. (2016). State estimator design for a single degree of freedom cable-actuated system. *J. Frankl. Inst.* 353, 4845–4869. doi:10.1016/j.jfranklin.2016.08.015
- Chen, G., Hutchinson, S., and Dellaert, F. (2022). *Locally optimal estimation and control of cable driven parallel robots using time varying linear quadratic Gaussian control*. arXiv preprint arXiv:2208.00916.
- Gagliardini, L., Caro, S., Gouttefarde, M., and Girin, A. (2016). Discrete reconfiguration planning for cable-driven parallel robots. *Mech. Mach. Theory* 100, 313–337. doi:10.1016/j.mechmachtheory.2016.02.014
- Gill, J. S., Pisu, P., Krovi, V. N., and Schmid, M. J. (2019). 59155. American Society of Mechanical Engineers (ASME), V002T25A004. Behavior identification and prediction for a probabilistic risk framework *Dyn. Syst. Control Conf.*

- Gill, J. S. (2019). *Probabilistic framework for behavior characterization of traffic participants enabling long term prediction*. Ph.D. thesis, Clemson University.
- Hill, E., Gadsden, S. A., and Biglarbegian, M. (2021). Explicit nonlinear mpc for fault tolerance using interacting multiple models. *IEEE Trans. Aerosp. Electron. Syst.* 57, 2784–2794. doi:10.1109/taes.2021.3065089
- Idà, E., Briot, S., and Carricato, M. (2021). Natural oscillations of underactuated cable-driven parallel robots. *IEEE Access* 9, 71660–71672. doi:10.1109/access.2021.3071014
- Korayem, M. H., Yousefzadeh, M., and Beyranvand, B. (2017). Dynamics and control of a 6-dof cable-driven parallel robot with visco-elastic cables in presence of measurement noise. *J. Intelligent Robotic Syst.* 88, 73–95. doi:10.1007/s10846-017-0546-1
- Korayem, M., Yousefzadeh, M., and Kian, S. (2018). Precise end-effector pose estimation in spatial cable-driven parallel robots with elastic cables using a data fusion method. *Measurement* 130, 177–190. doi:10.1016/j.measurement.2018.08.009
- Kosari, S. N., Ramadurai, S., Chizeck, H. J., and Hannaford, B. (2013)., 55935. American Society of Mechanical Engineers (ASME), V06AT07A077. Control and tension estimation of a cable driven mechanism under different tensions. *Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*
- Le Nguyen, V., and Caverly, R. J. (2021). Cable-driven parallel robot pose estimation using extended kalman filtering with inertial payload measurements. *IEEE Robotics Automation Lett.* 6, 3615–3622. doi:10.1109/lra.2021.3064502
- Maciejewski, A. A. (1990). Fault tolerant properties of kinematically redundant manipulators. *IEEE Int. Conf. Robotics Automation*, 638–642.
- Mazor, E., Averbuch, A., Bar-Shalom, Y., and Dayan, J. (1998). Interacting multiple model methods in target tracking: A survey. *IEEE Trans. Aerosp. Electron. Syst.* 34, 103–123. doi:10.1109/7.640267
- Notash, L. (2012). Failure recovery for wrench capability of wire-actuated parallel manipulators. *Robotica* 30, 941–950. doi:10.1017/s0263574711001160
- Notash, L. (2011). Motion recovery after joint failure in parallel manipulators. *Trans. Can. Soc. Mech. Eng.* 35, 559–571. doi:10.1139/tcsme-2011-0034
- Passarini, C., Zanutto, D., and Boschetti, G. (2019). Dynamic trajectory planning for failure recovery in cable-suspended camera systems. *J. Mech. Robotics* 11. doi:10.1115/1.4041942
- Picard, E., Tahoumi, E., Plestan, F., Caro, S., and Claveau, F. (2020). A new control scheme of cable-driven parallel robot balancing between sliding mode and linear feedback. *IFAC-PapersOnLine* 53, 9936–9943. doi:10.1016/j.ifacol.2020.12.2708
- Pott, A., and Bruckmann, T. (2013). *Cable-driven parallel robots*. Springer.
- Raman, A., Schmid, M., and Krovi, V. N. (2021). “Wrench analysis of kinematically redundant planar cdprrs,” in *International conference on cable-driven parallel robots* (Springer), 90–104.
- Raman, A., Schmid, M., and Krovi, V. (2020)., 83990. American Society of Mechanical Engineers (ASME), V010T10A054. Stiffness modulation for a planar mobile cable-driven parallel manipulators via structural reconfiguration. *Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*
- Raman, A., Walker, I., Krovi, V., and Schmid, M. (2022). A failure identification and recovery framework for a planar reconfigurable cable driven parallel robot. *IFAC-PapersOnLine* 55, 369–375. doi:10.1016/j.ifacol.2022.11.211
- Rasheed, T. (2019). *Collaborative mobile cable-driven parallel robots*. Ph.D. Thesis, école centrale de Nantes
- Roberts, R. G., Graham, T., and Lippitt, T. (1998). On the inverse kinematics, statics, and fault tolerance of cable-suspended robots. *J. Robotic Syst.* 15, 581–597. doi:10.1002/(sici)1097-4563(199810)15:10<581::aid-rob4>3.0.co;2-p
- Srivastava, M., Ammons, J., Peerzada, A., Krovi, V., Rangaraju, P., and Walker, I. D. (2022). “3d printing of concrete with a continuum robot hose using variable curvature kinematics,” in *IEEE international Conference on Robotics and automation (ICRA) (IEEE)*.
- Tudoroiu, N., and Khorasani, K. (2005). “Fault detection and diagnosis for satellite’s attitude control system (acs) using an interactive multiple model (imm) approach,” in *Proceedings of 2005 IEEE conference on control applications, 2005. CCA 2005 (IEEE)*, 1287–1292.
- Verhoeven, R. (2006). *Analysis of the workspace of tendon based Stewart platforms*. Ph.D. thesis, Duisburg, Essen. University of Duisburg-Essen.
- Visinsky, M. L., Cavallaro, J. R., and Walker, I. D. (1994). Robotic fault detection and fault tolerance: A survey. *Reliab. Eng. Syst. Saf.* 46, 139–158. doi:10.1016/0951-8320(94)90132-5
- Walker, I. D., Krovi, V. N., Peerzada, A. B., Raman, A., Rangaraju, P., Schmid, M. J., et al. (2022). 3d concrete printing with macro-micro robots. In *Proceedings of the future technologies conference (FTC) 2022*, 3 (Springer), 493–498.
- Xiong, H., Cao, H., Zeng, W., Huang, J., Diao, X., Lu, W., et al. (2022). Real-time reconfiguration planning for the dynamic control of reconfigurable cable-driven parallel robots. *J. Mech. Robotics* 14, 061008. doi:10.1115/1.4055255
- Yoshikawa, T. (1985). Manipulability of robotic mechanisms. *Int. J. Robotics Res.* 4, 3–9. doi:10.1177/027836498500400201
- Zhou, X., Jun, S.-k., and Krovi, V. (2014). “Stiffness modulation exploiting configuration redundancy in mobile cable robots,” in *2014 IEEE international conference on robotics and automation (ICRA) (IEEE)*, 5934–5939.