

Model Predictive Control and Transfer Learning of Hybrid Systems Using Lifting Linearization Applied to Cable Suspension Systems*

Jerry Ng¹, H. Harry Asada¹

Abstract—Model Predictive Control (MPC) of nonlinear hybrid systems using lifting linearization underpinned by Koopman Operator is presented. Unlike standard linearization, which is valid only locally, lifting linearization provides a global linear representation of a nonlinear system in a lifted space. This allows us to obtain a unified linear model for a class of hybrid systems, which are otherwise modeled as a collection of dynamic modes that are constantly switching. This linear model created with lifting linearization is utilized to control a multi-cable robot exhibiting hybrid dynamics due to switching between taut and slack conditions of each cable. Applying MPC to the cable suspension robot, we show that even though the highly complex computation is reduced to a straightforward convex optimization, MPC can still find dexterous control actions for manipulating an object by taking into account the hybrid nature of the dynamics. A drawback of lifting linearization is that the learned model must be relearned when parameters of the system change. Here, we exploit the linearity of the model for transfer learning of a tuned model, adapted to a similar system with different parameters. Using a recursive updating method for the linear model, we demonstrate that the transferred model is able to adapt to the dynamics of the new nonlinear system and decrease prediction error over time. The resulting prediction error is comparable to that of the original model trained on its original dataset.

Index Terms—Hybrid Logical/Dynamical Planning and Verification; Optimization and Optimal Control; Transfer Learning

I. INTRODUCTION

Cable driven robotics is a significantly large field. Previous designs have ranged from very simple cranes, to more complex tensegrity robots which are composed of both tension elements and compression elements [1], [2]. There are several advantages to using cable based designs: they are lightweight parallel manipulators, capable of high acceleration and velocities, and may have very large workspaces [3]. However, the cables can only pull, not push. Thus, more complex designs may increase the number of cables, making the model more difficult to analyze, but making the system more feasible to control.

The prior art regarding controls in the field of cable driven robotics is vast. Some of these methods target specific dynamics of the system, such as antisway control [4]. This is done through a variety of means: linear control through

a standard linearization [5], feedback linearization [6], geometric control [7], and data driven controllers [8]. These works also deal with a variety of different configurations of the system: the standard single cable configuration, multiple cable configurations, and variations of those configurations involving mobile suspension points like quadrotors. In these works, it is usually assumed that the cables are in tension. However, there are several situations neglected when not considering the dynamics of the system when the cables become slack. By including this behavior, the system becomes more complex and difficult to model; the dynamics resemble a hybrid system due to the significant difference in speed of the dynamics. As the primary focus of this paper is control, we simulate a simple two-cable winch design that has no redundancies and allows for the cables to all become slack while the load is in motion.

In this work, we aim to resolve this complex control task by utilizing learned models involving linearizing a nonlinear dynamic system by augmenting the state space with additional variables; this technique is based on the Koopman Operator [9]. There are many modeling techniques that are similar, such as Carleman embedding [10]. Prior work has applied Koopman linearization to approximate a variety of nonlinear systems, including fluids [11], and hybrid pendulum systems [12]. However, it is also applicable to dynamic systems for the sake of control, specifically through MPC (model predictive control) [13]. It has been shown that MPC using the Koopman linearized model as dynamic constraints is comparable in performance and sometimes outperforms MPC using the real nonlinear system [14]. The prior work involving hybrid pendulum systems involves an autonomous systems, differing significantly from the current work, which involves a controlled system and application of MPC.

Substantial work has been done in researching methods to find the additional variables necessary to create a Koopman operator model for a nonlinear system. These works have involved deep learning [15], optimization methods [16], defined classes of functions that are usable as generalized Koopman eigenfunctions [17], or created systematic methods of finding the observable functions based on some notion of causality [18] [19]. Recently, there has also been work on tracking of time variant systems using Koopman based methods through online versions of dynamic mode decomposition [20], [21]. Other work involving online learning of a Koopman model has been done with active learning, and using recursive least squares to update the model [22]. In general, these works train a model based on a set of data and apply this model on the system that the data came

Manuscript received: September, 9, 2021; Accepted November, 8, 2021.

This paper was recommended for publication by Editor Lucia Pallottino upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by National Science Foundation Grants NSF-CMMI 1762961 and 2021625.

¹Jerry Ng and H. Harry Asada are with Department of Mechanical Engineering, Massachusetts Institute of Technology, Massachusetts, USA
jerryng@mit.edu asada@mit.edu

Digital Object Identifier (DOI): see top of this page.

from. With that in mind, a problem arises. One of the key benefits of using cable driven robotic systems is the ability to interchange the load that is carried, which causes a significant and sudden change in the dynamics of the system.

To deal with this issue, we look towards the concept of transfer learning. In machine learning, transfer learning has been consistently used to apply models that are trained for a specific task and adapted to a new task; in some cases, the model is updated. It has been used for image classification [23], feature recognition [24], defect prediction [25], and robotics [26]. The primary benefits shown from transfer learning in these works are of two parts: 1) prior models can be used for new problems with relative high success, and 2) less data needs to be collected to successfully train a model for a new task. By applying the concept of transfer learning to this problem, we seek to adapt and use a model for multiple similar systems.

We attempt to model this hybrid system behavior as a linear system using the Koopman Operator. Through modeling the system linearly in a higher dimension, we are able to embed both the incredibly fast dynamics caused from the cables going into tension and the slow dynamics of the motion of the winches and projection motion of the mass into the lifted space and then apply MPC to the system without explicitly modeling discrete dynamics that would typically make the system difficult to control [27]. This allows the controller to quickly determine the control sequence of the system, and approximate when the discrete dynamics should take place according to the optimization problem. We demonstrate this method on a simulated system with applications to cable suspended systems. Lastly, we address the idea of transfer learning; as a solution to the issue of significant changes in parameters and characteristics of a nonlinear system after creating a Koopman model, we implement an adaptive method for updating an existing Koopman model and demonstrate success across significant changes in parameters and characteristics of the nonlinear system. This would resolve the problem of needing to train a new linear predictor model whenever changing the load or the cables.

A. Outline

Section II gives a brief overview on the Koopman Operator and its uses for nonlinear systems, as well as recursive least squares and its relevance in adapting linear models. In section III, the model is introduced. Section IV motivates the use of MPC and lifted linearizations through discussion of issues when using optimal control for hybrid systems. Section V presents the results of the simulation with regards to the estimation of the linear predictor model and utilization of that model for model predictive control. Section VI showcases the transfer learning and model adaptation of a Koopman model under parameter changes. Concluding remarks are then given in section VII.

Through this paper, we demonstrate:

- 1) modeling a controlled hybrid system with lifted linearization models.

- 2) model predictive control for this hybrid system.
- 3) transfer learning and adaptation for this lifted model.

II. PRELIMINARIES

In this section, we give a brief overview of the Koopman Operator and its application to linearizing nonlinear control systems and the application of weighted recursive least squares.

A. Koopman Operator: Linearization

Consider a discrete-time dynamical system, given by

$$x_{k+1} = f(x_k) \quad (1)$$

where $x \in \mathbb{R}^n$ is the independent state variable vector representing the system, f is a nonlinear function $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$, and k is the current time step. Also consider a real-valued observable function of the state variables $\phi: \mathbb{R}^n \rightarrow \mathbb{R}$. The Koopman Operator is an infinite-dimensional linear operator acting on the observable ϕ [28]:

$$(\mathcal{K}\phi)(x) = \phi(f(x)) \quad (2)$$

where the Koopman operator $\mathcal{K}$ is linear, even though the dynamic system is nonlinear.

Although the original Koopman Operator is infinite dimensional, effective methods have been established for truncating the system order [16]–[18]. Furthermore, the original theory is for autonomous systems with no exogenous input, but has been extended to non-autonomous systems with control inputs [29].

These allow us to represent a nonlinear control system accurately in a higher dimensional space, or lifted space, as The state space is projected into a larger state space In this space, the system is represented as

$$z_{k+1} = Az_k + Bu_k \quad (3)$$

$$\hat{x}_k = Cz_k \quad (4)$$

where $u \in \mathbb{R}^m$ is input, and $z \in \mathbb{R}^o$ is high-dimensional state variables in the lifted space, $o \geq n$, $\hat{x}$ is the estimate of x and

$$z_k = \phi(x_k) = \begin{bmatrix} \phi_1(x_k) \\ \phi_2(x_k) \\ \dots \\ \phi_o(x_k) \end{bmatrix} \quad (5)$$

Matrices A and B are constant matrices with consistent dimensions.

It should be noted that, unlike standard linearization, which is valid only locally, the lifted linearization underpinned by the Koopman Operator theory is global. Another unique feature is that the system matrices A and B can be determined from data.

Given a dataset, the Koopman operator can be found through a least squares minimization

$$\min \sum ||z(x_{i+1}) - (Az(x_i) + Bu_i)||^2 \quad (6)$$

The solution then equates to

$$\begin{bmatrix} A & B \end{bmatrix} = FG^\dagger \quad (7)$$

where

$$F = [z_2, \dots, z_{k+1}] \quad (8)$$

$$G = \begin{bmatrix} z_1, \dots, z_k \\ u_1, \dots, u_k \end{bmatrix} \quad (9)$$

and $G^\dagger$ is the pseudoinverse of the G matrix.

B. Weighted Recursive Least Squares

The method used in this paper to adapt the system is a weighted recursive least squares method with no forgetting factor based on previous work [30] [31] [22]. For this method, the error vector at time step i is

$$e_i = x_i - \hat{x}_i \quad (10)$$

where

$$\hat{x}_i = Hx_{i-1} \quad (11)$$

The weighted error cost function is then

$$J = \frac{1}{2} \sum_{i=1}^t e_i^T W e_i \quad (12)$$

where W is a diagonal weight matrix that emphasizes the different elements of the error vector, and t is the number of measurements. This cost function can then substitute in (11) and (10), and then the partial derivative with respect to H and set to zero which yields

$$\frac{\partial J}{\partial H} = W \sum_{i=1}^t (-x_i x_{i-1}^T + H x_{i-1} x_{i-1}^T) = 0 \quad (13)$$

In the application to the Koopman operator, the coefficient matrix H to be updated is composed of A and B . This enables us to find that

$$R = \sum_{i=1}^t (x_{i-1} x_{i-1}^T) \quad (14)$$

$$Q = \sum_{i=1}^t (x_i x_{i-1}^T) \quad (15)$$

$$H = QR^{-1} \quad (16)$$

This extends to the time-varying situation where the coefficient H matrix is considered to be time-varying at each time step. This matrix, H_i is updated using the equation

$$H_{i+1} = H_i + e_{i+1} x_{i+1}^T W P_{i+1} \quad (17)$$

where e_i is found at each time step and $P = R^{-1}$ is recursively updated at each time step using the Matrix Inversion Lemma:

$$P_{i+1} = P_i - P_i x_{i+1} [1 + x_{i+1}^T W P_i x_{i+1}]^{-1} x_{i+1}^T W P_i \quad (18)$$

III. MODELING OF CABLE SUSPENSION SYSTEMS

In this section, we introduce the system, and the address how the system was modeled in a lifted space.

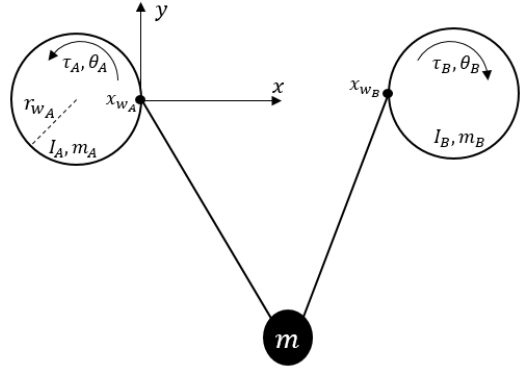


Fig. 1. Diagram of the hybrid system involving two winches, two cables, and a suspended mass. The winches are each driven directly by a motor. The point at which the cable departs from the winch is maintained as a fixed point for simplicity. We refer to the cable on the left as cable A and the cable on the right as cable B.

A. Modeling as a Hybrid System

The simulated system is based on a real robotic system from a prior work which utilized three winches and a very similar design [32]. The simulated system is a simplified version of this robotic system and a diagram for this system that can be found in Figure 1.

The object suspended by the cable system is treated as a point with mass m . Because the system is constrained to two dimensions, the state vector is defined as:

$$\mathbf{x} = [x \ y \ \dot{x} \ \dot{y} \ L_A \ L_B \ \dot{L}_A \ \dot{L}_B]^T$$

where L_A and L_B are *unstretched* cable lengths and $\dot{L}_A$ and $\dot{L}_B$ are their time derivatives.

The unstretched cable lengths and their velocities are functions of the winch rotation θ_i and $\dot{\theta}_i$ respectively, where i corresponds to the cable identifier, A or B. The unstretched cable length is defined as

$$L_i = L_0 + r_{w_i} \theta_i$$

and the unstretched cable length velocity is the time derivative of the above function and is also a state variable.

Considering this, there are four state variables associated with the winches and cables, and four state variables associated with the suspended mass.

The winch positions are fixed in place, but allowed to rotate. The winch rotation dynamics are written as

$$I_i \ddot{\theta}_i = u_i - \tau_{w_i}$$

where τ_{w_i} is the torque due to the cable when in tension, and u_i is the input torque from the motor attached to the winch. This torque is defined as

$$\tau_{w_i} = r_{w_i} \mathbf{n}_{w_i} \times T_i \mathbf{n}_i$$

where r_{w_i} is the radius of the winch, and $\mathbf{n}_i$ corresponds to the unit vector in the direction of the departure point of the cable from the center of the winch.

This system's different dynamic modes can be represented visually by Fig. 2. By modeling the system as a hybrid

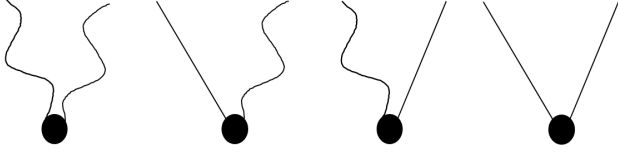


Fig. 2. The different dynamic modes of the system that are determined by whether the cables are in tension. From left to right: both cables are in tension, cable A is in tension and cable B is slack, cable A is slack and cable B is in tension, both cables are slack.

system, we have two options: 1) choose to ignore the dynamics of the system when the cable suddenly goes in tension and causes the mass to “bounce”, and instead model it as a discrete reset map, or 2) use the guard function to map the dynamic transition from slack to tension and tension to slack for each cable. For example, when not modeling the system with a discrete “bounce” the system has dynamic modes defined by

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = \begin{cases} \begin{bmatrix} 0 \\ mg \end{bmatrix} & \text{if } d_A \leq 0 \wedge d_B \leq 0 \\ T_A \mathbf{n}_A + \begin{bmatrix} 0 \\ mg \end{bmatrix} & \text{if } d_A > 0 \wedge d_B \leq 0 \\ T_B \mathbf{n}_B + \begin{bmatrix} 0 \\ mg \end{bmatrix} & \text{if } d_A \leq 0 \wedge d_B > 0 \\ T_A \mathbf{n}_A + T_B \mathbf{n}_B + \begin{bmatrix} 0 \\ mg \end{bmatrix} & \text{otherwise} \end{cases} \quad (19)$$

and where d_i is the elongation length of cable i subtracted by its unstretched length based on the positioning of the mass relative to the respective winch.

B. Linearization in Lifted Space

As opposed to modeling the system as a hybrid system as mentioned above, we can utilize the Koopman Operator methodology to linearize the system similar to prior work [12]. Instead of having the several conditions defining the different dynamic modes, we use a data driven method to create a higher dimensional linear system that attempts to embed the dynamic modes and reset maps. The dataset used to estimate the Koopman operator is created by feeding a pseudorandom binary signal to each of the winch inputs beginning from five randomized initial conditions that fulfill one of four categories of situations as shown in Fig. 2. The observable functions composing the Koopman operator are a set of radial basis functions that cover the dynamic range of the dataset, and also the horizontal and vertical components of the tension for the cables.

These radial basis functions are of the form

$$\phi = e^{-\alpha(x_j - x_{j_a})^2} \quad (20)$$

where α is a parameter that was tuned for different state variables that tunes the shape of the function and x_j is the j th state variable, and x_{j_a} is the center of the radial basis

function. To model this system, 150 radial basis functions are used for x and y position, 6 radial basis functions are used for x and y velocities, 10 radial basis functions are used for L_A and L_B , and 4 functions which are the horizontal and vertical components for the tension in each cable.

A dynamic mode decomposition (DMD) model is created based on the same dataset [33]. This DMD model truncates the dynamic modes to a rank that contains 99% of the information based on the singular value decomposition.

IV. MODEL PREDICTIVE CONTROL

The cable suspension system is a complex hybrid system as shown in its original dynamical model. Once a cable goes slack, the mass freely drops, followed by an impact at the instant when the cable becomes taut. It tends to bounce back, unless the velocity in the direction of the cable is zero. It should be noted that, once the cable goes slack, the winch of the cable is essentially disconnected and becomes unable to influence the motion of the mass. Thus, the winch loses controllability. To maintain controllability, the robot controller must reduce the impact and extend the controllable duration by keeping the cable taut. To find such an intelligent, skillful control action, the robot must be able to predict the dynamic behavior, in particular, the consequence of impact and bouncing. Here, we consider Model Predictive Control (MPC) for realizing such skillful actions. With MPC we can expect that the robot can find an optimal control sequence, which would minimize a potential impact and retain the controllability for better manipulating the mass.

A. Standard MPC Formulation

For the model predictive control section, we solve an optimization problem of the form

$$\min V = \phi(x_N(t)) + \sum_{i=0}^{N-1} \ell(x_i(t), u_i(t)) d\tau \quad (21)$$

$$s.t. \quad x_{i+1}(t) = x_i(t) + f(x_i(t), u_i(t)) d\tau \quad (22)$$

$$x_0(t) = x_0 \quad (23)$$

$$C(x_i(t), u_i(t)) \leq 0 \quad (24)$$

where $d\tau = T/N$, T is the time horizon, and N is the number of time steps. We use an objective function of the form

$$\ell(x, u) = (x - x_d)^T Q (x - x_d) + u^T R u \quad (25)$$

$$\phi(x) = (x - x_d)^T P (x - x_d) \quad (26)$$

where ϕ is the end stage cost. The parameters chosen for the cost function are included in the Appendix.

The challenge in implementing the above MPC is due to the dynamic constraints on the problem. Because the system has hybrid dynamics, it must not only optimize the control input, but also optimize the switching times between dynamic modes. Though work has been done in creating a framework to use MPC for hybrid systems [34], the framework requires denoting a set of terminal times. In essence, the problem with implementing MPC directly onto a hybrid system

is that optimal sequence of modes may not be known *a priori* and leaving the solver of the optimization program to discover this sequence is computationally expensive and can sometimes be intractable.

B. MPC Formulation Using Lifting Linearization

With this in mind, creating a linear model where the system is no longer modeled as a hybrid system with guards, modes or reset maps is very attractive. The dynamic constraints can be replaced with a linear time invariant model, causing the problem to become a linear MPC problem. Linear MPC is known to be convex, and simple to solve. We no longer need to solve for a sequence of modes, nor incorporate any additional constraints on the time steps for each mode. This does require having an accurate linear model for the system throughout the time horizon of the problem, which is feasible because of the Koopman operator.

V. CABLE MANIPULATION BASED ON MPC IN LIFTED SPACE

In this section the MPC formulation using lifting linearization is implemented for a multi-cable robot system. We create a realistic model of tension using experimental data, demonstrate the accuracy of the linearized system in comparison to the full nonlinear system, and apply the lifted linear model to MPC for driving the system to specific reference states from randomized initial conditions.

A. Tension Modeling

The tension in the cables was assumed to be elastic and modeled through taking experimental data of the cables used in the real robotic system. The model does not include any damping component; this assures that no causality problem arises from lifting the space, since anti-causal observables pertain to damping elements alone [19]. This experimental data is shown in Fig. 3 and the parameters found from this experimental data is shown in Table I.

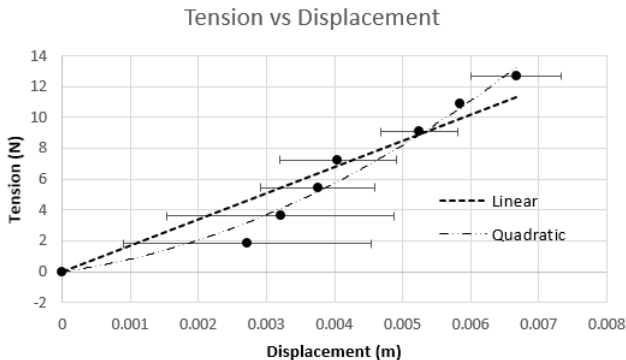


Fig. 3. Tension-Displacement data gathered from the motivating robotic system. The data is fit to polynomial models. Models of higher than 2nd order were deemed unreasonable as they gave the possibility of negative tension. The models were also required to intersect with zero tension at zero displacement.

Polynomial models were considered, and only two reasonable models were found, the linear and quadratic model.

TABLE I
TENSION MODELS

Model	R^2	Equation
Linear	0.8914	$F = 1699x$
Quadratic	0.9673	$F = 206015x^2 + 613.99x$

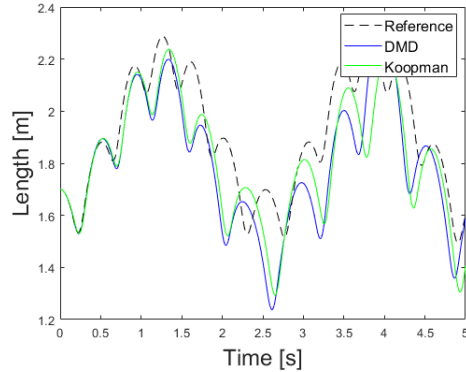


Fig. 4. Example of the estimation provided from the Koopman linear system and the DMD system for a state variable; in this plot it is the length of cable A.

For the simulations, the linear model was chosen though it had a lower R^2 value, as it emphasized the hybrid dynamic nature of the system more than the quadratic model which was continuous and differentiable through the entire domain of the system, including “negative” displacements where the tension remains zero. The quadratic model is instead used as a possible change to the underlying nonlinear system for transfer learning.

B. Model Training

Both Fig. 4 and 5 were produced for trajectories where the control law applied was a proportional controller on cable length. In Fig. 4, an estimation of the linearized system

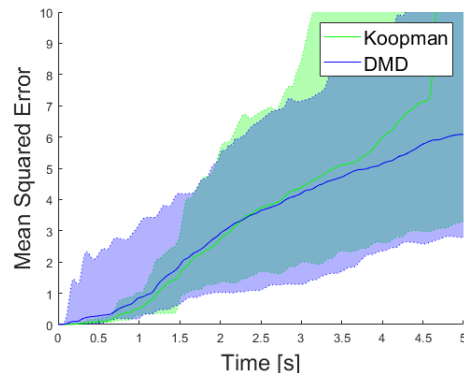


Fig. 5. A plot of MSE over time for the linear predictor models. Each predictor begins with the correct initial state and attempts to predict the next time steps. The plot demonstrates that for small amounts of time, the Koopman model is much more accurate than the DMD model, but after enough time has passed, both models approach the same level of accuracy. The shaded regions represent the variation in mean squared error over twenty five different initial conditions.

compared to the real system for one of the state variables is shown. In Fig. 5, a plot of the MSE (mean squared error) for the eight state variables is shown as a function of the length of the prediction time. As the prediction time increases, the mean squared error tends to increase. In the numerical results presented, we find that the system can be reasonably estimated using the Koopman linear system and by the DMD system, though less so, which is expected. The truncation of the system also appears to cause a significant increase in variation of the prediction error for the system, however, it is within reason for small time horizons to still be used for model predictive control.

C. Model Predictive Control

For all implementations of MPC, the time horizon was chosen to be one hundred time steps, or one second, and was selected based on the accuracy of prediction over time shown in Fig. 5. The weight matrices are defined in the Appendix.

In Fig. 6, the resultant trajectory simulated from model predictive control that used the Koopman linearized model as the dynamic constraints for the optimization problems is juxtaposed with the control input that was generated for cable A. Initially, the mass was placed 3 meters vertically above the goal position with both cables being slack. With MPC+Koopman, the mass trajectory smoothly converged to the goal position, as shown by the red line in the vertical position plot. For comparison, a naive PD control that controls the individual cable length without predicting the mass behavior is plotted by blue lines. Note that the mass bounced when the two cables became taut consecutively at $t = 1.1 \sim 1.7$ sec, as indicated by notches in the plots. This resulted in a pronounced bouncing motion of the mass, leading to the slow conversion of the naive cable length control. In contrast, MPC+Koopman could smoothly move the mass towards the goal. Interestingly, the MPC control input did not attempt to lengthen the cable rapidly, unlike the naive PD control, as seen in section A indicated by the orange circle. Instead, the MPC control allowed the mass to pull on the cable, and dampened the motion of the mass. These behaviors cannot be created unless the robot can predict the nonlinear hybrid nature of the system dynamics.

As stated previously, the dynamic constraints are all that are necessary to generate a solution for this MPC problem, and no additional constraints or equations were set on time. The MPC result is successful in that it does rapidly decrease the SSE with respect to the reference trajectory as shown in Fig. 7; the slight differences between using the Koopman linear system as dynamic constraints and the DMD system as linear constraints is expected given prior work.

VI. TRANSFER LEARNING AND MODEL ADAPTATION

For adapting the model, the weight matrix from Eq. 12 was manually tuned to emphasize the errors of the state variables as this tended to yield faster adaptation during experiments than otherwise. The control inputs were disregarded because as it is not necessary to predict the control input at the future time steps. The actual weight matrix can be found in the

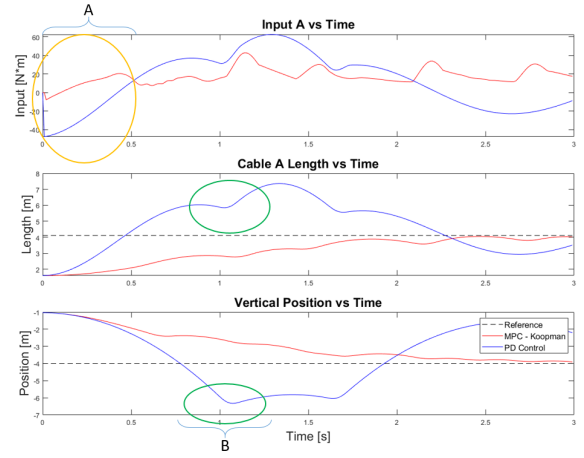


Fig. 6. A juxtaposition of the input used for cable A and the trajectory of the length of cable A over time when using model predictive control with the Koopman linearization as the dynamic constraints. This is compared to the trajectory found from using a PD controller.

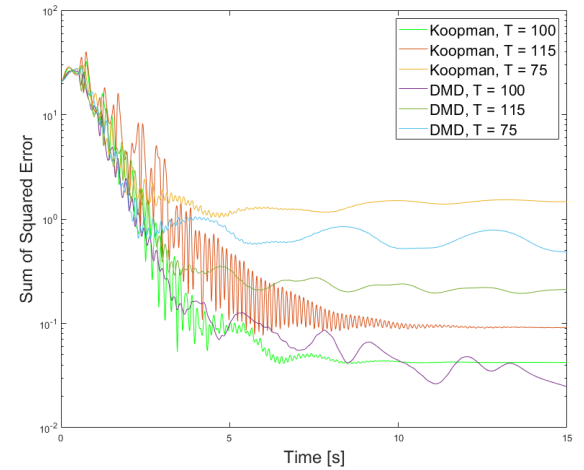


Fig. 7. SSE (Sum of Squared Error) over the trajectory comparing an MPC scheme using the DMD model and the Koopman model as dynamic constraints. The number indicated in the legend denotes the prediction time horizon in time steps for the model. It was found that after approximately 100 time steps that MPC began performing poorly, likely due to inaccuracy of the linear model.

Appendix. The adaptation process is visually interpreted in Fig. 8.

To demonstrate the ability for the system to adapt using recursive least squares, we isolate the situation to a scenario that does not utilize model predictive control. Instead, the system uses a naive proportional controller for each winch that attempts to maintain a constant length for each cable. Results of this method in terms of mean squared error (MSE) are shown in Fig. 9, where all estimators are given the correct initial state and attempt to predict the next one hundred time steps, or one second. They then reinitialize at each time step at the correct initial condition and repeat the prediction process, comparing the estimated trajectory to the real trajectory. In the first experiment, the mass of the

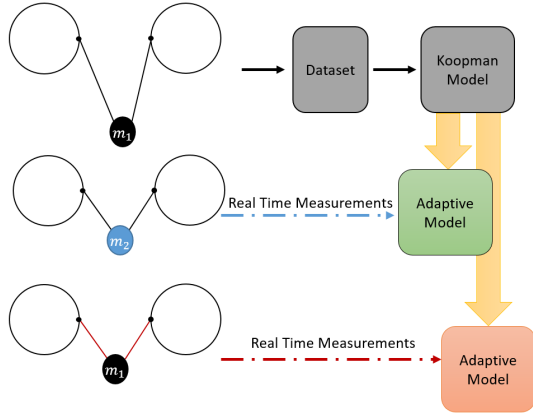


Fig. 8. Visual representation of the experiments run in the transfer learning and model adaptation experiment.

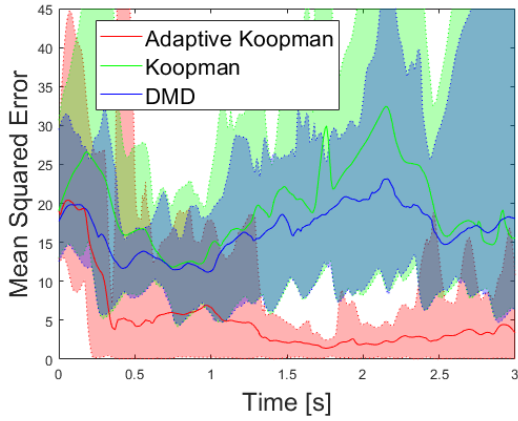


Fig. 9. Comparison between different estimators for tracking a nonlinear system that the models were not trained on. All estimators were trained on a cable system with a 10 kg mass load, and attempted to predict a cable system with a 5 kg mass. The shaded regions indicate the range between minimum and maximum of the MSE across 25 trajectories for the specified estimator. The plotted line is the average MSE for the estimator across all trials.

system is reduced from 10 kg to 5 kg, and 25 different initial conditions with the same distribution as the initial conditions used for training the system.

The second experiment has similar structure to the first, but instead involved significantly changing the underlying nonlinear system. For that, the tension model of the cables was changed from the linear model to the quadratic model mentioned in Table I. Notably, the prediction error is significantly higher with this change in the nonlinear system when compared to changing the mass of the system. The prediction errors as a function of time step can be seen in Fig. 10, for a single trajectory of the system.

When changing the tension model as in Fig. 10, the MSE falls significantly, but not to the same range as before. This is somewhat expected as the observable functions used for the system may not be as informative for this new tension model. The chosen observable functions are static, which may be less useful depending on how the dynamics of the

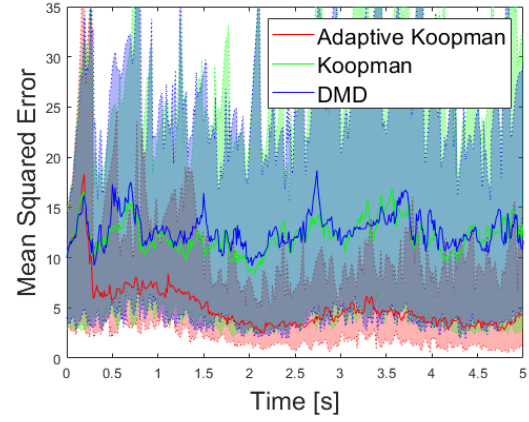


Fig. 10. Another comparison between different estimators, predicting a system that they were not trained on. In this case, the system was changed to have a different tension model, changing from quadratic to linear. Again, the shaded regions indicate the range between minimum and maximum of the MSE across 25 trajectories for the specified estimator. The plotted line is the average MSE for the estimator across all trials.

nonlinear system change. Implementing adaptable hyperparameters into these observable functions may lead to an alternative adaptive method. Despite this issue, the accuracy is significantly high considering how little training and data were required for this model and not alternative observable functions when transferring from another model. In that regard, the experiments are successful in demonstrating the application of transfer learning for Koopman models, not only for parameter changes but also for changes in the underlying nonlinear dynamics; the original systems had over ten thousand data points to be trained from, whereas the transferred system only had hundreds of additional data points. This fact warrants additional investigation into the bounds of which a Koopman model can be transferred, the number and range of measurements required for reasonable accuracy for the transferred model, and the possibility of convergence. After this analysis is completed, there may be significant benefit in using this transfer learning model in combination with MPC.

VII. CONCLUSION

In this paper, we presented a novel approach to using optimal control for hybrid systems using the Koopman operator and applied it to an elastic cable suspension system. This allowed us to use model predictive control without the difficulty of determining controllable hybrid sequences nor switching times as the hybrid dynamics were encoded into a linear time invariant system. The model predictive control input was also analyzed on its behavior, showing that it demonstrated proper model predictive behavior despite using a linear model to represent an inherently hybrid dynamic system. In addition, we presented a method to adapt the linearized model not only to parameter changes but also functional changes in the underlying nonlinear system, through recursive updates of the model based on the error of the prediction at each time step. As a future work, analysis of

the adaptive properties in relation to transfer learning and its robustness will be necessary. Further investigation may also be beneficial in adapting the observable functions chosen to accommodate the use of adaptation.

APPENDIX

A. Model Predictive Control

The weight matrices used in the experiments are $P = 0$, Q is of the form

$$Q = \begin{bmatrix} Q_x & 0 \\ 0 & 0 \end{bmatrix} \quad (27)$$

where $Q_x \in \mathbb{R}^{n \times n}$ and is diagonal; the diagonal elements of Q_x are $\{6, 0, 0.3, 0.3, 1, 1, 0.5, 0.5\}$. R is a diagonal matrix with values $\{0.01, 0.01\}$.

No inequality constraints are used in our implementation of the optimization problem.

B. Weighted Recursive Least Squares

For the updating of the Koopman system in both cases, the weight matrix $W \in \mathbb{R}^{180 \times 180}$ is a diagonal matrix where the first eight diagonal elements are $\{0.01, 0.01, 0.01, 0.01, 0.001, 0.001, 0.001, 0.001\}$ and the rest are 10^{-5} . The error covariance matrix P is the identity matrix with the same dimensionality as W .

REFERENCES

- [1] C. Li and C. D. Rahn, "Design of Continuous Backbone, Cable-Driven Robots," *Journal of Mechanical Design*, vol. 124, no. 2, pp. 265–271, 05 2002. [Online]. Available: <https://doi.org/10.1115/1.1447546>
- [2] L.-H. Chen, K. Kim, E. Tang, K. Li, R. House, E. L. Zhu, K. Fountain, A. M. Agogino, A. Agogino, V. Sunspiral, and E. Jung, "Soft Spherical Tensegrity Robot Design Using Rod-Centered Actuation and Control," *Journal of Mechanisms and Robotics*, vol. 9, no. 2, 03 2017, 025001. [Online]. Available: <https://doi.org/10.1115/1.4036014>
- [3] C. Gosselin, P. Cardou, T. Bruckmann, and A. Pott, *Cable-Driven Parallel Robots: Proceedings of the Third International Conference on Cable-Driven Parallel Robots*, ser. Mechanisms and Machine Science. Cham: Springer International Publishing AG, 2017, vol. 53.
- [4] S. Lee and H. Son, "Antisway control of a multirotor with cable-suspended payload," *IEEE Transactions on Control Systems Technology*, pp. 1–9, 2020.
- [5] R. Vrabel, "Design of the state feedback-based feed-forward controller asymptotically stabilizing the overhead crane at the desired end position," *Journal of Applied Nonlinear Dynamics*, 03 2019.
- [6] M. Korayem, H. Tourajzadeh, and M. Bamdad, "Dynamic load carrying capacity of flexible cable suspended robot: Robust feedback linearization control approach," *Journal of Intelligent & Robotic Systems*, vol. 60, pp. 341–363, 2010.
- [7] T. Lee, "Geometric control of quadrotor uavs transporting a cable-suspended rigid body," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 1, pp. 255–264, 2018.
- [8] J. Lin and G.-T. Liao, "A modularized cable-suspended robot: Implementation and oscillation suppression control," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 230, no. 9, pp. 1030–1043, 2016. [Online]. Available: <https://doi.org/10.1177/0959651816662157>
- [9] A. Mauroy and J. Goncalves, "Koopman-based lifting techniques for nonlinear systems identification," *arXiv e-prints*, p. arXiv:1709.02003, Sep 2017.
- [10] R. F. S. Andrade, "Carleman embedding and lyapunov exponents," *Journal of Mathematical Physics*, vol. 23, no. 12, pp. 2271–2275, 1982. [Online]. Available: <https://doi.org/10.1063/1.525319>
- [11] I. Mezić, "Analysis of fluid flows via spectral properties of the koopman operator," *Annual Review of Fluid Mechanics*, vol. 45, no. 1, pp. 357–378, 2013. [Online]. Available: <https://doi.org/10.1146/annurev-fluid-011212-140652>
- [12] N. Govindarajan, H. Arbabi, L. van Bargaian, T. Matchen, E. Tegling, and I. Mezić, "An operator-theoretic viewpoint to non-smooth dynamical systems: Koopman analysis of a hybrid pendulum," 2016.
- [13] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *arXiv e-prints*, p. arXiv:1611.03537, Nov 2016.
- [14] Y. Igarashi, M. Yamakita, J. Ng, and H. Asada, "Mpc performance for nonlinear systems using several linearization models," *2020 American Control Conference*, vol. 35, pp. 718–730, July 2020.
- [15] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature Communications*, vol. 9, no. 1, Nov 2018. [Online]. Available: <http://dx.doi.org/10.1038/s41467-018-07210-0>
- [16] M. Korda and I. Mezić, "Optimal construction of koopman eigenfunctions for prediction and control," 2020.
- [17] C. A. Johnson and E. Yeung, "A class of logistic functions for approximating state-inclusive koopman operators," in *2018 Annual American Control Conference (ACC)*, 2018, pp. 4803–4810.
- [18] G. Manjunath and A. de Clercq, "Universal set of observables for the koopman operator through causal embedding," 2021.
- [19] N. S. Selby, F. E. Sotiropoulos, and H. H. Asada, "Physics-based causal lifting linearization of nonlinear control systems underpinned by the koopman operator," p. arXiv:2108.10980, 2021.
- [20] H. Zhang, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "Online dynamic mode decomposition for time-varying systems," 2017.
- [21] M. Alfatawi and V. Srivastava, "An incremental approach to online dynamic mode decomposition for time-varying systems with applications to eeg data modeling," 2020.
- [22] I. Abraham and T. D. Murphey, "Active learning of dynamics for data-driven control using koopman operators," *CoRR*, vol. abs/1906.05194, 2019. [Online]. Available: <http://arxiv.org/abs/1906.05194>
- [23] A. J. Quattoni, "Transfer learning algorithms for image classification," 2009.
- [24] Y. Gao and K. M. Mosalam, "Deep transfer learning for image-based structural damage recognition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 748–768, 2018. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/mice.12363>
- [25] Y. Ma, G. Luo, X. Zeng, and A. Chen, "Transfer learning for cross-company software defect prediction," *Information and Software Technology*, vol. 54, no. 3, pp. 248–256, 2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950584911001996>
- [26] N. McDonald, R. Davis, L. Loomis, and J. Kopra, "Aspects of hyperdimensional computing for robotics: transfer learning, cloning, extraneous sensors, and network topology," in *Disruptive Technologies in Information Sciences V*, M. Blowers, R. D. Hall, and V. R. Dasari, Eds., vol. 11751, International Society for Optics and Photonics. SPIE, 2021, pp. 21 – 33. [Online]. Available: <https://doi.org/10.1117/12.2585772>
- [27] P. Manon, C. Valentin-Roubinet, and G. Gilles, "Optimal control of hybrid dynamical systems: application in process engineering," *Control Engineering Practice*, vol. 10, no. 2, pp. 133–149, 2002. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S096706610100123X>
- [28] B. O. Koopman, "Hamiltonian systems and transformation in hilbert space," *Proceedings of the National Academy of Sciences*, vol. 17, no. 5, pp. 315–318, 1931. [Online]. Available: <https://www.pnas.org/content/17/5/315>
- [29] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz, "Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control," *PLOS ONE*, vol. 11, no. 2, pp. 1–19, 02 2016. [Online]. Available: <https://doi.org/10.1371/journal.pone.0150171>
- [30] C. Wang and T. Tang, "Recursive least squares estimation algorithm applied to a class of linear-in-parameters output error moving average systems," *Applied Mathematics Letters*, vol. 29, pp. 36–41, 2014. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0893965913002966>
- [31] T. Lai and C.-Z. Wei, "Extended least squares and their applications to adaptive control and prediction in linear systems," *IEEE Transactions on Automatic Control*, vol. 31, no. 10, pp. 898–906, 1986.
- [32] R. Hoffman and H. H. Asada, "Precision assembly of heavy objects suspended with multiple cables from a crane," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6876–6883, 2020.

- [33] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of fluid mechanics*, vol. 656, pp. 5–28, 2010.
- [34] B. Altın, P. Ojaghi, and R. G. Sanfelice, "A model predictive control framework for hybrid dynamical systems," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 128–133, 2018, 6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2405896318326582>