2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Koopman Dynamic Modeling for Global and Unified Representations of Rigid Body Systems Making and Breaking Contact

Cormac O'Neill[1] and H. Harry Asada[2], *Life Fellow, IEEE*

***Abstract*—A global modeling methodology based on Koopman operator theory for the dynamics of rigid bodies that make and break contact is presented. Traditionally, robotic systems that contact with their environment are represented as a system comprised of multiple dynamic equations that are switched depending on the contact state. This switching of governing dynamics has been a challenge in both task planning and control. Here, a Koopman lifting linearization approach is presented to subsume multiple dynamics such that no explicit switching is required for examining the dynamic behaviors across diverse contact states. First, it is shown that contact/non-contact transitions are continuous at a microscopic level. This allows for the application of Koopman operator theory to the class of robotic systems that repeat contact/non-contact transitions. Second, an effective method for finding Koopman operator observables for capturing rapid changes to contact forces is presented. The method is applied to the modeling of dynamic peg insertion where a peg collides against and bounces on the chamfer of the hole. Furthermore, the method is applied to the dynamic modeling of a sliding object subject to complex friction and damping properties. Segmented dynamic equations are unified with the Koopman modeling method.**

***Index Terms*—Koopman operator, Hybrid switched systems, Contact dynamics, Friction model**

I. Introduction

ROBOT dynamics are complex. A robot's governing equations are not only intrinsically nonlinear (Coriolis and centrifugal effects, configuration-dependent link inertia, etc.) [1] but also exhibit hybrid and switched dynamics as they physically interact with their environment. When robots experience contact/non-contact transitions, non-holonomic constraints are imposed on the robot dynamics, causing switching in the dynamic structure of the system [2]. Dynamic behaviors become extrinsically hybrid and switched among multiple dynamic equations [3]. Examples of such systems include assembly robots that experience contact/non-contact transitions in guiding and mating parts [4], [5], biped robots which switch between stance and swing phases [6]–[8], and robot fingers exhibiting complex behaviors as they close and open kinematic chains [9], [10]. All of these common robotics tasks are hybrid or switched in nature, and thus their governing equations are a set of heterogeneous equations.

In such systems, state space is segmented into multiple regions. Governing equations are valid only within a specific region, or state location. Global behaviors that emerge as the system makes multiple transitions across the segmented space are difficult to analyze. The lack of a global and unified representation of segmented heterogeneous dynamics has been a major impediment in the analysis and synthesis of robotic systems as they interact with the environment.

The overarching goal of this paper is to establish a global, unified modeling methodology based on Koopman Operator theory [11]. According to Koopman Operator theory, a nonlinear dynamical system behaves linearly when represented in a high-dimensional space [3], [11]. Although the original state equation is a collection of heterogeneous equations, we can obtain a unified representation, which is linear. In the high-dimensional lifted space, there is no segmentation, no boundary, and no switching. A wealth of analysis and synthesis tools are available for such linear models [12].

There is a growing interest in applying Koopman models to robotic systems. Recent works have shown how linear models can be used to control soft and continuous robots [13], [14], multi-cable robots [15], [16], and in active learning [17]. In many of these works, the robots being modelled by Koopman are intended to perform tasks which heavily involve contact with objects and the environment. However, there has been little work exploring how best to apply Koopman linearization methods to dynamic systems with contact. Koopman theory assumes that the lifted system's dynamics can be represented in a Hilbert space of observable functions, an assumption which is not guaranteed for hybrid systems with discontinuous variations of state. While hybrid systems are a popular approach for modelling contact dynamics, they may be a poor fit for Koopman modelling. In spite of this, past work has explored the potential of using Koopman linearization for hybrid systems [18], [19]. While these works have demonstrated promising results, they do not present rigorous and general theoretical support for the applicability of the Koopman operator to hybrid systems with discontinuous variations of state. As it stands, the weak conditions for the existence of Koopman operators of discontinuous systems are limited to a short-horizon prediction [3]. Instead, we propose using a viscoelastic model of contact, which provides

[1]Cormac O'Neill is with Department of Mechanical Engineering at the Massachusetts Institute of Technology croneill@mit.edu

[2]H. Harry Asada is faculty in Department of Mechanical Engineering at the Massachusetts Institute of Technology asada@mit.edu

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

continuous dynamics amenable to Koopman theory.

In robot manipulation, contact dynamics modeling has been recognized as a critical component for robotic task planning. Contact has been found to enhance the capabilities of systems when effectively exploited [20]. Unfortunately, leveraging contact in practice poses significant challenges for the modelling and control of dynamic systems. Contact-induced non-smoothness invalidates the local accuracy of Taylor series approximations [21], which in turn hampers the application of most modern gradient-based planning techniques. Methods that are specifically designed for planning through contact often take one of two approaches: explicitly considering different contact modes as a sequence of discrete states [22], or by smoothing out the non-smoothness of contact dynamics [21]. We propose that a linear Koopman model presents an alternative approach to smooth out contact dynamics, potentially aiding with trajectory planning and control. However for such an approach to be explored, the ability for a linear Koopman model to approximate contact/non-contact dynamics must be verified.

Construction of Koopman operators requires an effective set of observables with which to augment the system's state. Finding such lifting functions is even more challenging for systems with contact. Many past works have proposed and explored approaches for choosing lifting functions, and for approximating the Koopman matrix [23]. In this paper, we explore and compare a few prominent techniques for performing Koopman linearization on systems with contact. We focus on comparing the predictive capabilities of a Koopman approximation composed of: a manually chosen library of radial basis functions (RBFs) [24], a library of learned Deep Koopman Network (DKN) observables [25], and a combined model composed of a set of both learned and manually chosen observables. In particular, we evaluate the effectiveness of these approaches for predicting how systems can evolve through instances of making and breaking contact.

In summary, this work makes the following contributions:

- Koopman modeling of dynamically contacting objects using a viscoelastic contact model
- A combined DKN-RBF method for constructing a Koopman model for systems with contact
- Applications of the method to dynamic peg insertion and complex contact friction systems.

II. Preliminary

Koopman operator theory was first established in the seminal work by B. Koopman in the 1930's [11] and has been applied to the modeling and analysis of complex nonlinear engineered systems and extended to a data-driven methodology [12], [25], [26]. Consider a system with discrete-time, nonlinear dynamics $x_{t+1} = F(x_t)$. It can be shown that a higher-order representation of the system composed of a potentially infinite set of observable functions $\mathbf{g}(x_t) = [g_1(x_t), g_2(x_t), \ldots]$ in Hilbert space can be evolved linearly with the Koopman operator, $\mathcal{K}$.

$$\mathcal{K}\mathbf{g} \triangleq \mathbf{g} \circ F \rightarrow \mathcal{K}\mathbf{g}(x_k) = \mathbf{g}(x_{k+1}) \tag{1}$$

In practice, a finite approximation of the Koopman operator is used to generate a lifted linear model. We refer to such an approximation as a Koopman matrix. Commonly applied through Dynamic Mode Decomposition (DMD) and Extended Dynamic Mode Decomposition (eDMD) in the literature [24], [27], Koopman linearization was recently extended and supplemented with the Direct Encoding (DE) approach and its data-driven algorithm, Data-Driven Direct Encoding (DDE) [3], [16], [28], [29].

Selecting the observable functions $g(x_t)$ is an area of ongoing discussion within the field. Manually chosen observables have shown promise in a number of applications. The Havok Alternative View of Koopman (HAVOK) methodology [30] uses the state at previous times to augment a nonlinear system, while the Taylor Series has motivated the use of higher-order derivatives as observables [31]. A library of pre-selected nonlinear functions can be generated and used, with basis functions in a Hilbert space being potentially well-suited [24]. In past work, libraries of such functions have been used in algorithms that then select and tune the most optimal potential observables [32]. Radial basis functions are one such possible set of pre-chosen observables, with a degree of local tunability. Placing center points of RBFs at specific regions of interest, the Koopman model can represent highly nonlinear behaviors for a long time horizon [3]. For the purposes of this work, we are using Gaussian radial basis functions, which are parameterized by their center location, c, and the dilation factor ε.

$$\texttt{RBF}(x, c, \varepsilon) = \exp(\frac{-1}{\varepsilon}||x-c||_2^2) \tag{2}$$

Despite salient features, RBFs do not scale well. Machine learning techniques for identifying Koopman observables are more powerful and scale well for higher-order systems. Deep Koopman Networks (DKNs) have been widely adopted and draw upon machine learning approaches in order to learn lifting functions from data [25]. The neural networks used in Deep Koopman have the added use of providing an alternative way to generate the linear model's transformation matrix. Rather than calculating this matrix through a least-squares approach, a linear layer in the neural network can itself be used to learn and represent the time evolution of the lifted system. A particular advantage of this is to help stabilize the Koopman matrix, which often can include unstable eigenvalues when developed in a way that only considers a single time-step. This is done by introducing a multi-time step horizon into the DKN loss, which encourages the network to find weights for the linear layer that do not lead to unbounded predictions. The loss function of a DKN can also be modified so that minimizing errors in the predictions of the physical state of the system are prioritized over predictions of how the observable functions will evolve. While there are alternative approaches for stabilizing these Koopman matrices [33], we selected this technique for its simplicity and ease of implementation.

III. Koopman Modeling of Dynamically Contacting Objects

With the use of Koopman operator theory and computational methods, we aim to obtain unified, globally-linear Koopman models of dynamically contacting objects. As shown in Fig. 1, an object contacts with the environment's surfaces, bouncing back and forth through a series of collisions. The key condition for applying Koopman operator theory are that the composites of observables g_i with a self-map state transition function F are involved in a Hilbert space.

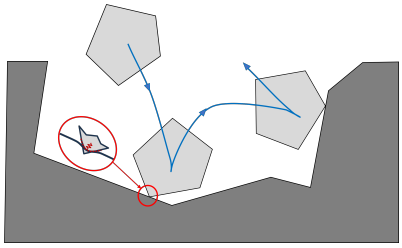


Fig. 1: An object may make and break contact multiple times during a realistic trajectory with collisions.

$$g_i \circ F \in \mathcal{H}, i = 1, 2, \cdots \tag{3}$$

If the function $F(x)$ is continuous and the observables are chosen such that the compositional function is square integrable, a Koopman operator exists. However, this condition does not hold for most hybrid systems with discontinuities. Colliding objects, however, are not necessarily discontinuous when examined at a micro-scale, according to applied mechanics literature [34], [35].

Friction is another complex factor significantly affecting the collision mechanisms. Dry friction, in particular, is a discontinuous phenomenon at the macroscale. Detailed studies in tribology show that continuous models derived from micro-scale contact mechanisms can nevertheless represent complex friction accurately [36].

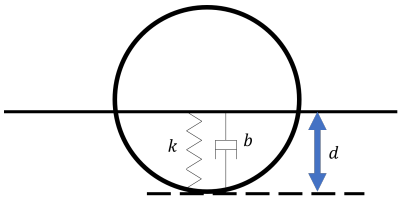


Fig. 2: In the viscoelastic model, contact is represented by the force exerted from a spring and damper connecting the two contacting bodies. These forces are only exerted when the two rigid bodies are penetrating one another.

Here we use viscoelastic models of contact by introducing a spring and damper between the two colliding bodies [37]. Initially inspired by considering the collisions of spherical particles, these models have shown promising alignment with experimental data in spite of their simplicity [37], [38]. Viscoelastic models allow for rigid bodies to inter-penetrate, and then use the penetration distance to compress the spring-damper system and exert a normal force, as depicted in Fig. 2. While this penetration is clearly non-physical, viscoelastic models can nevertheless offer a number of benefits: the objects involved in contact can be modelled as rigid, even though compliance may play a crucial role in physically accurate collisions; and the dynamics are continuous and do not involve sudden jumps in state space, as is the case for hybrid models with reset maps. This continuity allows us to readily apply Koopman operator theory.

While viscoelastic models are continuous, they nevertheless come with drawbacks. Accurately representing physical contact dynamics often requires very stiff springs, which can lead to stiff ODEs when evaluating the dynamics.

An added challenge emerges when using these models to generate data for data-driven Koopman methods, such as DKNs. Usually, contact between objects is kept very brief, leading to only a small fraction of the data highlighting the impact of the collision dynamics. As was shown in [28], Koopman models can be vulnerable to favoring approximations that are more accurate in regions with denser training data, at the expense of performance in regions with sparser measurements. This could lead to issues for systems in which brief instances of contact have significant effects over longer time horizons. This is an especially significant problem during instances of bouncing, where a brief contact causes a rapid change in state.

IV. Combined DKN and RBF Observables

As an alternative approach to using learned or manually chosen functions for Koopman observables, we developed a combined DKN-RBF approach for Koopman linearization (Fig. 3). In this approach both learned observables and locally tuned RBFs are used to lift a nonlinear system. The RBFs are intended to better model the stiff, and highly local, dynamics during instances of contact, while the learned neural net observables offer broader coverage of the smoother dynamics away from switching contact surfaces.

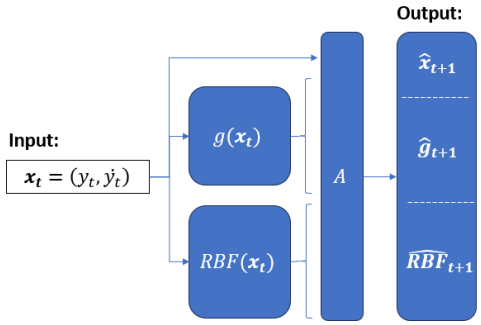


Fig. 3: We learn a set of Koopman observables that are aware of the RBFs. Combining these sets of observables together leads to improved performance when modelling complex systems with contact.

When the system's state is located close to one of the RBFs' centers, it has a relatively greater impact on the

evolution of the lifted dynamics. In contrast, the same RBF would have a smaller impact on the dynamics when the system is operating in a more distant region of state space. While RBFs have been used to great effect in Koopman linearizations, it is prohibitive to cover a larger and higher-dimensional domain with narrow RBFs. Doing so involves placing many functions, which can rapidly cause the lifted dimension to increase. Observables learned through a DKN are far more efficient at covering larger, and more benign, dynamic regions. While we might expect a suitably large neural network within a DKN, as a universal approximator [39], to replicate or exceed the performance of RBFs, doing so may involve significant quantities of both data and training time.

By using both RBFs and learned observables to lift our system, we can gain some of the benefits of both. In systems with contact, we are able to densely pack RBFs in the small region where contact dynamics dominate, thereby leaving the DKN to learn the more benign dynamics away from switching contact surfaces. A simple approach for implementing this method would be to learn observables from a DKN, and then concatenate them with the RBFs prior to calculating the linear transformation matrix. However, this runs the risk of the learned observables providing redundant, or even conflicting, information that is already afforded from the RBFs. Instead, we first define the RBFs with centers that are close to areas of state space with contact. The outputs from these RBFs are then provided to the linear time-evolution layer in the DKN during training, thereby encouraging the learned observables to augment the information provided by the RBFs without redundancy.

V. Dynamic Peg Insertion

A. Task and Data Generation

Peg insertion is a classical manipulation problem of practical and scientific importance [4]. The celebrated Remote Center Compliance (RCC) hand can guide a misaligned peg towards the center of a hole by contact with the chamfer [40]. See Fig. 4. This peg guidance and insertion process is highly dynamic, but prior work treated it as a quasi-static process except for [41]. As the peg slides off at the bottom of the chamfer, point C in the figure, the static force balance conditions are no longer held and, thereby, the peg accelerates, i.e. the process is dynamic.Furthermore, the peg collides with the chamfer surface and may bounce on it, which may lead to a failure of peg insertion. Understanding such dynamic interactions between a peg and a chamfer is thus important.

As the dynamic analysis in [41], we consider a peg interacting with a chamfer within a vertical plane. The peg has 6 state variables $(x, y, \theta, \dot{x}, \dot{y}, \dot{\theta})$, where x, y are the coordinates of the peg's center of mass and θ is the tilting angle of the peg (Fig.4). The chamfer angle is 50°. We represented contact with a viscoelastic model but also included friction along the chamfer edge as $F = \mu N$.

To mimic how data may be gathered for such a system in practice, data were generated by simulating the peg dynamics

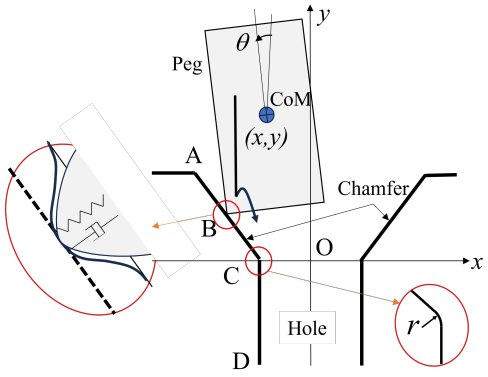


Fig. 4: Diagram of the peg and chamfer problem. When the corner of the peg penetrates the chamfer, a viscoelastic model of contact is applied. Close to the chamfer corner, a slight rounding is applied to prevent sudden changes in the direction of the normal force during contact. The peg is assumed to start with an initial pose close to vertical, $\theta = \pm 3°$

subject to the viscoelastic contact force. The governing equations were switched depending on the contact state. In this work we considered a) projectile motion without contact, b) contact between the peg's corner and the chamfer surface AC, and c) contact between the peg's corner and the bottom of the chamfer C, which is round, as shown in Fig. 4. The peg is assumed to make only these contacts, which is reasonable for small tilt angles (θ) of the peg. From these contact dynamics of the peg, the depth of insertion of the first two-point contact is determined, which is the key metric for predicting whether the peg is successfully inserted or not [40], [41]. The force applied by the viscoelastic contact model is then determined by the perpendicular distance from the peg's contacting corner to the nearest edge of the chamfer/hole.

Trajectories were computed from a set of initial conditions at varying heights above the chamfer and at a range of initial angles close to vertical (±3°). The resulting data can be viewed in Fig 5. This is intended to replicate the collection of data from a physical peg being dropped repeatedly on the chamfer, with its trajectory recorded each time.

B. Model Training

The training of the combined RBF/DKN model was performed using a loss function similar to that of the standard DKN [26]. The main difference is that the combined RBF/DKN loss function includes a loss term for the RBF observable estimates.

$$L = \sum_{t=1}^{k} \gamma_d^{t-1} (\gamma_y \text{MSE}(\hat{\mathbf{y}}_\mathbf{t}, \mathbf{y}_\mathbf{t}) + \gamma_g \text{MSE}(\hat{\mathbf{g}}_\mathbf{t}, \mathbf{g}_\mathbf{t}) + \gamma_{RBF} \text{MSE}(\mathbf{R}\hat{\mathbf{B}}\mathbf{F}_\mathbf{t}, \mathbf{RBF}_\mathbf{t})) \tag{4}$$

where γ_y, γ_g and γ_{RBF} are weights associated to output $\hat{\mathbf{y}}_\mathbf{t}$, observables $\hat{\mathbf{g}}_\mathbf{t}$, and RBFs $\mathbf{R}\hat{\mathbf{B}}\mathbf{F}_\mathbf{t}$, respectively, γ_d is time decay, and the *MSE* function refers to the mean-squared error given by

$$MSE(\mathbf{p}, \mathbf{q}) = \frac{1}{n} \sum_{i=1}^{n} (p_i - q_i)^2 \tag{5}$$

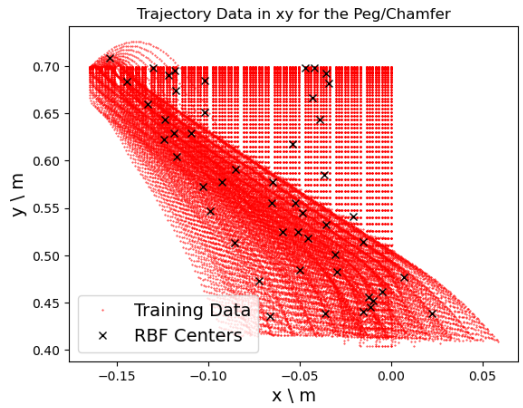


Fig. 5: Sample points from the training data trajectories used to generate the peg/chamfer models are shown in red. The center locations of the RBFs used in the combined model are also shown and were positioned by performing k-Means clustering on the data. The peg angle, θ, was excluded from the plot for readability but was still used to locate the RBF centers.

where $\mathbf{p} = [p_1, p_2, \ldots, p_n]$ and $\mathbf{q} = [q_1, q_2, \ldots, q_n]$. The RBFs are treated as another set of observables and are assigned their own weight in the loss function, γ_{RBF}, during training. A key difference between the completely learned observables, g, and the RBFs is that while the ground-truth measurement of the observables, $\mathbf{g_t}$, varies during the training process as the weights of the observable functions themselves are updated, the output of the radial basis functions for a given state remains constant. In contrast to the learned observables, updating the network weights does not change the output of a fixed RBF. This comes with the theoretical benefit of a smoother training process afforded by the RBF term of the loss function, at least when compared to that of the learned observables which are constantly being updated. Table I shows the parameter values used for training the neural network.

TABLE I: Key parameters for the peg/chamfer's combined DKN/RBF model.

Parameter	Value
Learned Observables	200
RBF Observables	50
γ_g	0.1
γ_y	1
γ_{RBF}	0.1
Training Steps (k)	20
Time Decay (γ_d)	0.95

The model accuracy depends on the hyperparameters of RBFs, in particular, the locations of the center point c. To place a limited number of RBFs effectively, the RBF centers were calculated by performing k-Means clustering on the data. This approach is motivated by the observation that the peg tends to 'stick' to the chamfer after contact. As such, more instances involving contact will be present in the data than instances with the peg in a ballistic trajectory. By performing a clustering operation, we would then expect the centers to preferentially be located close to the contact manifold. The results of this method can be seen in Fig. 5, which shows the sample points from the trajectories making up the training data as red dots, and the centers of the RBF observables as black crosses.

C. Results

Fig. 6 shows an example trajectory with the bottom-left corner of the peg colliding against the chamfer. The combined DKN/RBF Koopman model is able to predict the peg's bouncing off of the chamfer after the initial collision. The estimated trajectory in red shows a good agreement with the ground truth trajectory in black computed by the full nonlinear equations, with a gradual accumulation of error as the prediction length increases.

Fig.7 shows the prediction error comparison among the three Koopman models: the one using DKN neural networks only, the one with RBFs only, and the one with the combined DKN-RBF. The mean error for each of the models across the full validation trajectory dataset is plotted, along with a corresponding 95% confidence interval. While the DKN outperformed a model composed entirely of RBFs, the combined linear model composed of a combination of the two performed the best, as is indicated in Figs. 7. We posit that this is due to the challenge of learning useful observables across the multi-dimensional contact manifold. This manifold represents all the possible configurations of the peg in which it initiates contact with the chamfer.

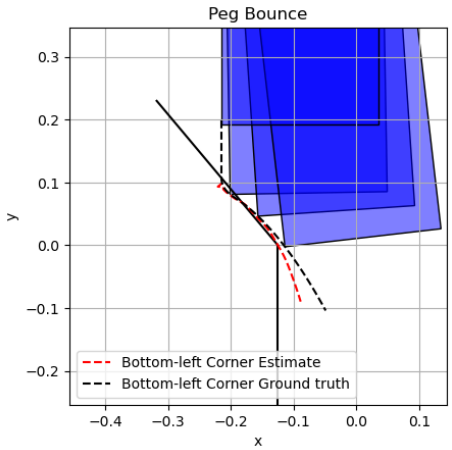


Fig. 6: Example trajectory of the peg with linear predictions.

Comparing the performance of the combined RBF/DKN model with similarly sized lifted models using just RBF or DKN observables, we can see that the combined model leads to more accurate predictions. This indicates that the act of adding in carefully placed RBFs as observables can allow for greater performance when producing a lifted linear model for dynamic systems with contact. In a way, the addition of RBFs allows for the designer to add domain-specific knowledge of the system - that is, the presence of contact - to better initialize the DKN prior to training.

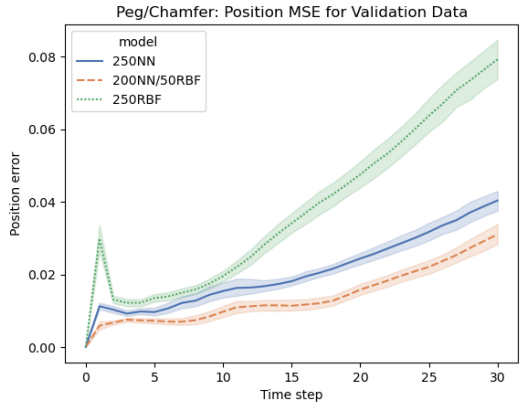


Fig. 7: Performance of the linear prediction for the peg across the full set of validation trajectories.

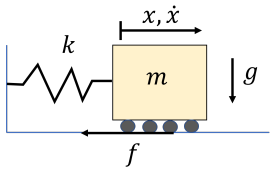


Fig. 8: Diagram of the sliding block model. The block experiences a force from a linear spring connected to the origin and friction due to contact with the ground.

VI. Modelling a Sliding Block with Friction

A. Nonlinear Dynamics

Friction plays a crucial role in many tasks involving contact, such as manipulation and locomotion. While a simple model of friction is present in the peg/chamfer model discussed in section V, we further investigate whether the proposed Koopman model can represent complex nonlinearities of friction. Consider a sliding block experiencing friction while sliding along the ground(Fig. 8). A linear spring connects the block to the origin, imparting an additional force that may or may not be sufficient to overcome friction. The state of the block is represented by its horizontal position, x, and velocity, $\dot{x}$. Additional parameters include the mass, m, and spring stiffness, k. The friction model (eq. (6)) used in this work is a modified Stribeck friction model taken from [42], with definitions for the variables in table II. A plot showing how the friction force varies with the velocity of the block can be seen in Fig. 9.

$$f = \sqrt{2e}(F_{brk} - F_c)\frac{\dot{x}}{v_{st}}\exp\left(-\left(\frac{\dot{x}}{v_{st}}\right)^2\right) + F_c \tanh\frac{\dot{x}}{v_c} + f_v\dot{x} \quad (6)$$

Note that this friction model based on tribology is smooth and continuous, including at $\dot{x} = 0$. Much like with the viscoelastic model for contact, this means that Koopman operator theory can be readily applied.

We compare the performance of two linear models: one composed of 225 RBF observables and another with 1000 DKN observables. In both cases, the states were appended as additional observables.

TABLE II: Parameters of the Stribeck friction model

Variable	Parameter
f	friction force
F_{brk}	breakaway friction
F_c	Coulomb friction
v_{st}	Stribeck velocity theshold
f_v	viscous friction coefficient
v_c	Coulomb velocity threshold

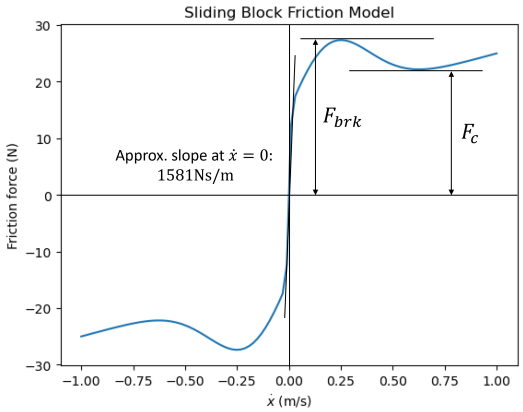


Fig. 9: The friction model (eq. (6)) used in the sliding block example is highly nonlinear, but ultimately smooth and continuous.

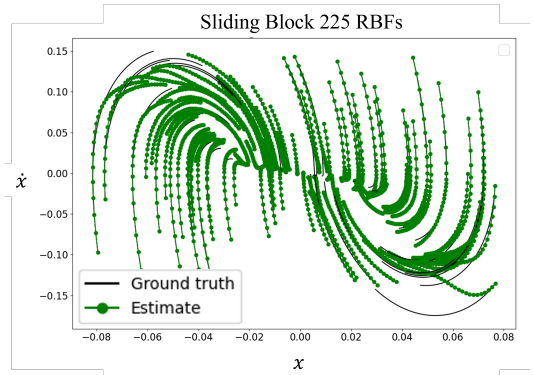
B. Results

Both the RBF and DKN based linear models show promising results in Figs. 10a and 10b respectively, although the RBF model performs especially well. The estimated trajectories for the sliding block only show slight deviations from the nonlinear ground truths. This makes a convincing case for Koopman linear models being able to reasonably approximate dry friction models, despite the macroscopic discontinuities between static and dynamic friction.

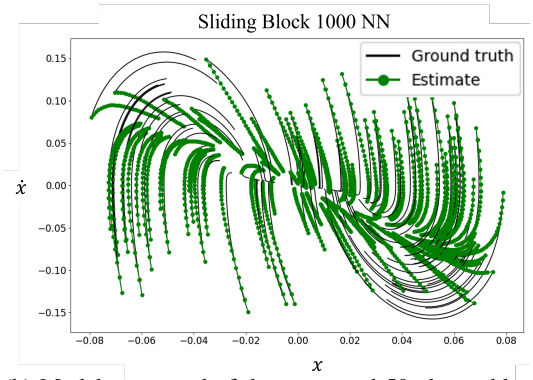
VII. Discussion

We have demonstrated that Koopman theory can generate useful linear approximations for dynamic systems with contact. Excitingly, we have shown that linear Koopman models can predict the evolution of these systems through multiple instances of making and breaking contact. While these results are promising, there are limitations. Chief among them is that the Koopman models are inherently approximations of the underlying system and so exhibit an error that - in general - grows over time. This limits the utility of predictions from these models to relatively short time horizons. An example would be model predictive control, which has been combined with Koopman models in the past [14], [15], [43]. The faster computation time and the convex-nature of optimization afforded by a linear model are significant advantages of the Koopman approach over the traditional counterpart. Future work could explore whether Koopman MPC or alternative control frameworks can be successfully applied to systems with contact.

One of the consequences of performing a Koopman linearization on systems with contact, is the potential for force-from-a-distance to be observed in the approximated model. This phenomenon can be seen in the case of contact between



(a) Model composed of the states and 225 RBF observables.



(b) Model composed of the states and 50 observables learned from a DKN neural network.

Fig. 10: Results for predicting trajectories of the sliding block with different models.

a peg and chamfer in Fig. 6. The predicted trajectory of the peg begins to diverge from the true trajectory prior to physical contact occurring with the chamfer. Force-from-a-distance effects are commonly observed while modelling contact, including both reinforcement learning and analytic, model-based approaches. Moreover, while the phenomenon is non-physical and inaccurate it can provide useful information to gradient-based planners on the potential for future instances of contact [21], [44].

Using a Koopman model to approximate systems with contact can be interpreted as finding a linear approximation for a stiff differential equation. The practical implementation of the examples presented in this paper saw us producing data by numerically solving the contact dynamics for the peg/chamfer and sliding block systems. Using the viscoelastic contact and continuous friction models causes the dynamics to be represented by relatively stiff ODEs, requiring the integrator to take smaller time steps during instances of contact or sharp changes in friction. While this represents a high up-front computational cost, once the linear Koopman model is produced the resulting discrete model of the system has a fixed time step. This raises the prospect of a Koopman linearization providing an alternative approach for dealing with stiff systems, although future work must explore this observation further.

VIII. Open Questions

Does the data-driven nature of the linearization introduce biases towards regions with denser sample points?

Data-driven implementations of the Koopman operator can suffer from sampling biases [28], and the implementations in this work are no exception. Ng and Asada presented tackeld this issue with data-driven direct encoding, which weights samples based upon the density of additional nearby data points. This work has been extended to work more efficiently in higher dimensional systems [16]. This approach might be especially useful in the case of modelling contact dynamics, since brief contacts may not be as well covered by collected data. While efforts were made to apply direct encoding to modelling contact, the resulting models were unstable. Future work will explore how the direct encoding methodology can be modified to encourage the discovery of stable linear models.

Are there any guarantees on the performance of the model?

While Koopman theory offers the potential for a lifted linear model to perfectly mimic a nonlinear system [45], there is no guarantee that a finite-dimensional approximation of the true Koopman operator exists for an arbitrary system. As a result, this work does not offer any guarantee on the accuracy of the approximated dynamics. This is an advantage retained by a point-wise Taylor approximation, which is at least accurate when sufficiently close to the point at which the linearization is being performed. Ongoing work in the field has sought to impose some bounds on the modelling errors from a Koopman approximation, but they are beyond the scope of this paper [46].

Could functions other than RBFs provide benefits to the combined model?

Radial basis functions were chosen based on their prior use in Koopman models [24]. Their center locations and dilation factors allow them to be associated with meaningful measures in the original state space, allowing for them to be positioned in a way that would better linearize localized contact dynamics. As a result, this paper focused on implementing a model that specifically utilized RBFs alongside learned observables. This is not a strict restriction however, and alternative functions, such as polynomials, could be included in combined models.

IX. Conclusions

This paper introduced using Koopman linearization as an alternative for modeling robotic systems that make and break contact with the environment. The Koopman operator is used not only for linearizing nonlinear dynamics but also for obtaining a global and unified model subsuming and aggregating segmented dynamics due to contact / non-contact switching. This opens up new possibilities for solving challenging robotics problems. In future, the effectiveness of these linear Koopman predictions for task planning through contact and control will be explored and compared with existing techniques. By considering a globally relevant linearization, it is hoped that a Koopman model would allow for

linear methods to be applied to challenging robotic tasks that require agents to plan through multiple instances of making-and-breaking contact.

REFERENCES

[1] H. Asada and J.-J. Slotine, *Robot analysis and control*. John Wiley & Sons, 1991.

[2] J. E. Colgate and N. Hogan, "Robust control of dynamically interacting systems," *International journal of Control*, vol. 48, no. 1, pp. 65–88, 1988.

[3] H. H. Asada, "Global, unified representation of heterogenous robot dynamics using composition operators: A koopman direct encoding method," *IEEE/ASME Transactions on Mechatronics*, 2023.

[4] J. L. Nevins and D. E. Whitney, "Computer-controlled assembly," *Scientific American*, vol. 238, no. 2, pp. 62–75, 1978.

[5] B. J. McCarragher and H. Asada, "The discrete event control of robotic assembly tasks," 1995.

[6] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.

[7] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE transactions on automatic control*, vol. 48, no. 1, pp. 42–56, 2003.

[8] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle, "Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 876–891, 2014.

[9] R. M. Murray, Z. Li, and S. Shankar Sastry, "A mathmematical introduction to robotic manipulation," 1994.

[10] A. Billard and D. Kragic, "Trends and challenges in robot manipulation," *Science*, vol. 364, no. 6446, p. eaat8414, 2019.

[11] B. Koopman, "Hamiltonian systems and transformation in hilbert space," *Proceedings of National Academy of Science (PNAS)*, vol. 17, pp. 315–318, 1931.

[12] I. Mezić, "Analysis of fluid flows via spectral properties of the koopman operator," *Annual Review of Fluid Mechanics*, vol. 45, pp. 357–378, 2013.

[13] D. Bruder, X. Fu, R. B. Gillespie, C. D. Remy, and R. Vasudevan, "Koopman-based control of a soft continuum manipulator under variable loading conditions," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6852–6859, 2021.

[14] ——, "Data-driven control of soft robots using koopman operator theory," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 948–961, 2021.

[15] J. Ng and H. H. Asada, "Model predictive control and transfer learning of hybrid systems using lifting linearization applied to cable suspension systems," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 682–689, 2022.

[16] I. Nozawa, E. Kamienski, C. O'Neill, and H. H. Asada, "A monte carlo approach to koopman direct encoding and its application to the learning of neural-network observables," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2264–2271, 2024.

[17] I. Abraham and T. D. Murphey, "Active learning of dynamics for data-driven control using koopman operators," *IEEE Transactions on Robotics*, vol. 35, no. 5, pp. 1071–1083, 2019.

[18] N. Govindarajan, H. Arbabi, L. V. Blargian, T. Matchen, E. Tegling, and I. Mezic, "An operator-theoretic viewpoint to non-smooth dynamical systems: Koopman analysis of a hybrid pendulum." Institute of Electrical and Electronics Engineers Inc., 12 2016, pp. 6477–6484.

[19] C. Bakker, A. Bhattacharya, S. Chatterjee, C. J. Perkins, and M. R. Oster, "Learning koopman representations for hybrid systems," 6 2020. [Online]. Available: http://arxiv.org/abs/2006.12427

[20] N. Chavan-Dafle and A. Rodriguez, "Sampling-based planning of in-hand manipulation with external pushes," in *Robotics Research: The 18th International Symposium ISRR*. Springer, 2020, pp. 523–539.

[21] T. Pang, H. J. T. Suh, L. Yang, and R. Tedrake, "Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4691–4711, 2023.

[22] B. J. McCarragher and H. H. Asada, "The discrete event modeling and trajectory planning of robotic assembly tasks," *Journal of Dynamic Systems Measurement and Control-transactions of The Asme*, vol. 117, pp. 394–400, 1995. [Online]. Available: https://api.semanticscholar.org/CorpusID:62072279

[23] C. Bakker, A. Bhattacharya, S. Chatterjee, C. J. Perkins, and M. R. Oster, "The koopman operator: Capabilities and recent advances," in *2020 Resilience Week (RWS)*, 2020, pp. 34–40.

[24] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, pp. 1307–1346, 2015.

[25] E. Yeung, S. Kundu, and N. Hodas, "Learning deep neural network representations for koopman operators of nonlinear dynamical systems," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 4832–4839.

[26] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature communications*, vol. 9, no. 1, pp. 1–10, 2018.

[27] P. J. SCHMID, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, p. 5–28, 2010.

[28] J. Ng and H. H. Asada, "Learned lifted linearization applied to unstable dynamic systems enabled by koopman direct encoding," *IEEE Control Systems Letters*, vol. 7, pp. 1153–1158, 2023.

[29] ——, "Data-driven encoding: A new numerical method for computation of the koopman operator," *IEEE Robotics and Automation Letters*, vol. 8, no. 7, pp. 3940–3947, 2023.

[30] S. L. Brunton, B. W. Brunton, J. L. Proctor, E. Kaiser, and J. N. Kutz, "Chaos as an intermittently forced linear system," *Nature communications*, vol. 8, no. 1, p. 19, 2017.

[31] G. Mamakoukas, M. L. Castaño, X. Tan, and T. D. Murphey, "Derivative-based koopman operators for real-time control of robotic systems," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 2173–2192, 2021.

[32] E. Lew, A. Hekal, K. Potomkin, N. Kochdumper, B. Hencey, S. Bak, and S. Bogomolov, "Autokoopman: A toolbox for automated system identification via koopman operator linearization," in *International Symposium on Automated Technology for Verification and Analysis*. Springer, 2023, pp. 237–250.

[33] G. Mamakoukas, I. Abraham, and T. D. Murphey, "Learning stable models for prediction and control," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 2255–2275, 2023.

[34] R. M. Brach, "Rigid body collisions," 1989.

[35] V. Bhatt and J. Koechling, "Three-dimensional frictional rigid-body impact," 1995.

[36] M. Nosonovsky and B. Bhushan, "Multiscale friction mechanisms and hierarchical surfaces in nano-and bio-tribology," *Materials Science and Engineering: R: Reports*, vol. 58, no. 3-5, pp. 162–193, 2007.

[37] H. Kruggel-Emden, E. Simsek, S. Rickelt, S. Wirtz, and V. Scherer, "Review and extension of normal force models for the discrete element method," *Powder Technology*, vol. 171, no. 3, pp. 157–173, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0032591006004360

[38] C. Thornton, S. Cummins, and P. Cleary, "An investigation of the comparative behaviour of alternative contact force models during elastic collisions," *Powder Technology*, vol. 210, no. 3, pp. 189–197, July 2011.

[39] S. Sonoda and N. Murata, "Neural network with unbounded activation functions is universal approximator," *Applied and Computational Harmonic Analysis*, vol. 43, no. 2, pp. 233–268, 2017.

[40] D. E. Whitney *et al.*, "Quasi-static assembly of compliantly supported rigid parts," *Journal of Dynamic Systems, Measurement, and Control*, vol. 104, no. 1, pp. 65–77, 1982.

[41] H. Asada and Y. Kakumoto, "The dynamic analysis and design of a high-speed insertion hand using the generalized centroid and virtual mass," 1990.

[42] The MathWorks, Inc. Translational friction. [Online]. Available: https://www.mathworks.com/help/simscape/ref/translationalfriction.html

[43] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S000510981830133X

[44] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 69–81, 2014.

[45] S. L. Brunton, M. Budisic, E. Kaiser, and J. N. Kutz, "Modern koopman theory for dynamical systems," *SIAM Review*, vol. 64, no. 2, pp. 229–340, 2022.

[46] C. Bakker, T. Ramachandran, and W. S. Rosenthal, "Learning bounded koopman observables: Results on stability, continuity, and controllability," *arXiv: Dynamical Systems*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:216914572