Dynamical System Segmentation for Information Measures in Motion

Thomas A. Berrueta, Ana Pervan, Kathleen Fitzsimons, and Todd D. Murphey

***Abstract*— Motions carry information about the underlying task being executed. Previous work in human motion analysis suggests that complex motions may result from the composition of fundamental submovements called movemes. The existence of finite structure in motion motivates information-theoretic approaches to motion analysis. We define task embodiment as the amount of task information encoded in an agent's motions. By decoding task-specific information embedded in motion, we can use task embodiment to create detailed performance assessments. We extract an alphabet of behaviors comprising a motion without *a priori* knowledge using a novel algorithm, which we call dynamical system segmentation. For a given task, we specify an optimal agent, and compute an alphabet of behaviors representative of the task. We identify these behaviors in data from agent executions, and compare their relative frequencies against that of the optimal agent using the Kullback-Leibler divergence. We validate this approach using a dataset of human subjects ($\mathbf{n = 53}$) performing a dynamic task, and under this measure find that individuals receiving assistance better embody the task. Moreover, we find that task embodiment is a better predictor of assistance than integrated mean-squared-error (MSE).**

I. INTRODUCTION

Motion signals encode information about the underlying task being executed, yet the form this information takes may vary. Typically, we represent motion using continuous real-valued signals. While this representation can provide detailed descriptions of an agent's motion, it can be cumbersome. However, based on our choice of representation, we can compress motion signals while preserving task information [1].

In [2], the authors propose that human motions are the result of the composition of a finite set of premotor signals emanating from the spinal cord. As a consequence, the neurological feasibility of motion decomposition forms the basis for action in motion primitives, also known as *movemes* [3]. Movemes are fundamental units of motion. Thus, all smooth human motions may be comprised of symbolic sequences drawn from an alphabet of movemes. Movemes motivate the application of information measures in human motion analysis, because they provide evidence of finite structure in otherwise continuous motion signals. Moreover, the existence of movemes indicates that under some choice of representation human motion can be discretized without loss of information.

In the human motion analysis literature, movemes are often characterized using causal dynamical systems [3], [4], or hybrid system identification methods, such as autoregressive models [5]. Most motor signal segmentation methods demand *prior specification* of the moveme alphabet, which limits their use in exploratory analyses where the alphabet may not be known *a priori*. Techniques in symbolic dynamic filtering can generate symbolic alphabets by creating partitions of the state-space using methods such as maximum entropy partitioning [6]. Additionally, state-space partition techniques can be applied to nonlinear transformations of the space via methods such as wavelet transforms [7]. However, these symbols are quasi-static, and cannot directly describe the dynamic nature of movemes. Motivated by movemes, we define *behaviors* as moveme analogues in general systems, and specify them using finite-dimensional nonlinear causal dynamical systems. Making a choice of representation for behaviors is very important. While there exist many data-driven function approximation methods, we choose the Koopman operator to represent behaviors [8].

We are interested in synthesizing finite sets of behaviors from agents' motions without *a priori* system knowledge, while preserving information about the underlying task. By identifying features from motion signals and constructing an alphabet of behaviors, we directly encode task-specific information into the symbolic representation. We define task embodiment as a measure of the amount of task-specific information encoded in an agent's trajectories.

The primary contributions of this paper are as follows. First, we develop a methodology for data-driven partitioning of dynamical systems. These partitions are projections onto the state-space that can be used to extract an alphabet of system behaviors, and can be represented by a graph. Second, we demonstrate that by tracking relative frequencies of behaviors we can discern relationships in human motion, such as whether an individual is receiving task assistance. We use task embodiment as an information measure that quantifies task-specifc information in an agent's motions. This formalism is important because it is agnostic to specification of the system dynamics or representation of information, which allows us to analyze motion signals generally.

We validate our information-theoretic approach by applying these methods to a dataset of human subjects $(n = 53)$ performing a dynamic task where assistance is sometimes provided, and extract an alphabet of optimal behaviors based on a synthesized exemplar agent. By tracking the relative frequencies of finite behaviors in human subjects and comparing to those of the optimal agent, we are able to quantify

This work was supported by the National Science Foundation under grant CBET-1637764, and by the National Defense Science and Engineering Graduate Fellowship program. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or of the NDSEG program.

Authors are with the Neuroscience and Robotics Laboratory (NxR) at the Department of Mechanical Engineering, Northwestern University, Evanston, IL. Email: tberrueta@u.northwestern.edu, anapervan@u.northwestern.edu, k-fitzsimons@u.northwestern.edu, t-murphey@northwestern.edu

the degree of task embodiment, and determine whether a subject received assistance. We found that task embodiment outperforms integrated MSE as a predictor of assistance.

This paper is organized as follows. In Section 2, we present an overview of dynamical system segmentation (DSS). In Sections 3 and 4, we apply DSS to a dataset of human subjects, and detect the presence of assistance by calculating each subject's task embodiment. In Section 5, we discuss conclusions and future work.

II. METHODS

An agent's state trajectories simultaneously encode information about the system dynamics and the task it executes. By examining system trajectories, one can uncover patterns in how it traverses the underlying state-space manifold. We propose DSS: a nonparametric, data-driven algorithm for creating low-dimensional, graphical representations of system behaviors by generating partitions of the state-space manifold sensitive to the underlying distribution of task information.

A. Koopman Operators

We use Koopman operators as our choice of representation for system behaviors. The Koopman operator is an infinite-dimensional linear operator capable of describing the evolution of any system [8]. Recently developed methods such as DMD and EDMD have enabled the identification of nonlinear dynamics through finite-dimensional approximations of the Koopman operator [9], [10].

The Koopman operator describes the evolution of a set of nonlinear basis functions of state, and generates a linear mapping in this function space. Given a choice of basis functions, $\Psi(x) = [\psi_1(x), ..., \psi_N(x)]^T$, and a dataset, $X = [x_0, ..., x_M]$, we want to develop a mapping $\Psi_{X'} = \Psi_X K + r(X)$ where the Koopman operator K evolves the transformed dataset $\Psi_X = [\Psi(x_0), ..., \Psi(x_{M-1})]^T$ to its next iterate $\Psi_{X'} = [\Psi(x_1), ..., \Psi(x_M)]^T$. By minimizing the residual $r(X)$ over the squared-error loss functional, we can synthesize a closed-form solution for the approximated Koopman operator

$$K = G^{\dagger} A \tag{1}$$

where † denotes the Moore-Penrose pseudoinverse and the individual matrix components are

$$G = \frac{1}{M} \sum_{k=0}^{M-1} \Psi(x_k) \Psi(x_k)^T \tag{2}$$

$$A = \frac{1}{M} \sum_{k=0}^{M-1} \Psi(x_k) \Psi(x_{k+1})^T \tag{3}$$

We obtain a matrix $K \in \mathbb{R}^{N \times N}$ that is an estimate of the system dynamics over the observed domain of the data [11]. We will use Koopman operators to describe individual behaviors expressed in data.

B. Dynamical System Segmentation

DSS characterizes all system behaviors over the state-space by synthesizing a non-redundant set of local estimates of the true system dynamics using a collection of Koopman operators. Given a dataset $X = [x_0, ..., x_M]$, and a set of basis functions $\Psi(x) \in \mathbb{R}^N$, we can apply the basis functions to the dataset X in order to generate a transformed dataset $\Psi_X = [\Psi(x_0), ..., \Psi(x_M)]^T$. We then split the transformed dataset Ψ_X into a set of W overlapping rectangular windows, and calculate a Koopman operator for each, thereby generating a set of symbols $\mathbb{K} = \{K_0, ..., K_W\}$. However, depending on the system under study, the size of the dataset, choice of window size, and overlap percentage, some of these symbols may be redundant.

We are interested in creating a minimal alphabet of Koopman operators with which to span all system behaviors. Unsupervised learning methods such as clustering algorithms that specialize in the identification of classes within datasets are well-suited for this task. By considering each $\mathbb{R}^{N \times N}$ Koopman operator as a point in $\mathbb{R}^{N^2}$ space, we can divide the set $\mathbb{K}$ into subsets using a clustering algorithm. In particular, we use Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN), which is a nonparametric, clustering algorithm that performs well in large spatial databases subject to noise [12]. The algorithm groups the operators into $B+1$ classes $\{C_0, ..., C_B\}$ using only the minimum of number of points required to make up a cluster as a parameter. We compose a set $\overline{\mathbb{K}} = \{K_0, ..., K_B\}$ of class exemplars by taking a weighted-average of all $K_i \in C_j,\ \forall j \in \{0, ..., B\}$, according to the class-membership probability $p(K_i | K_i \in C_j)$. The class-membership probability function is provided by the HDBSCAN software package [13].

Although we have created a minimal alphabet $\overline{\mathbb{K}}$ of system behaviors, it is of interest to project these behaviors onto the state-space manifold from this abstract operator space. We label all points in the transformed dataset Ψ_X with a label $l \in \{0, ..., B\}$ according to the class label of the Koopman operator each point was used to generate. Then, we train a support vector machine (SVM) classifier, $\Phi(\Psi(x))$ to project the class labels onto the state-space manifold, thereby generating partitions of the state-space [14].

Figure 1 depicts a cart-pendulum system used for an example application of DSS in Figs. 2(a) & 2(b), where

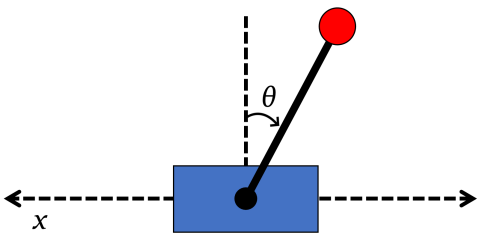
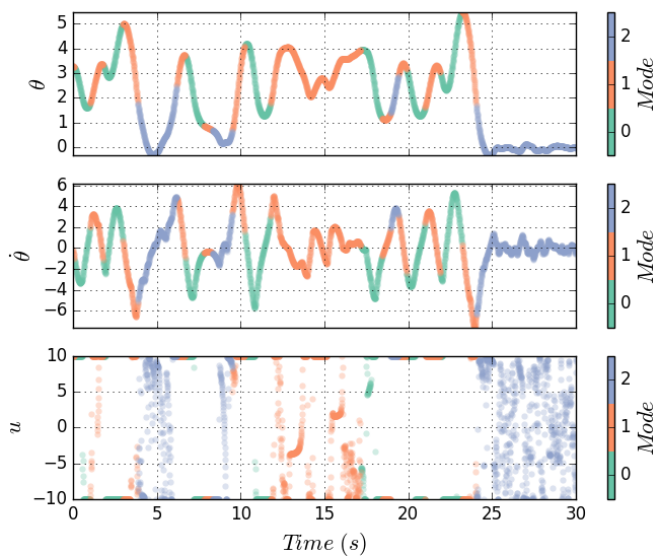
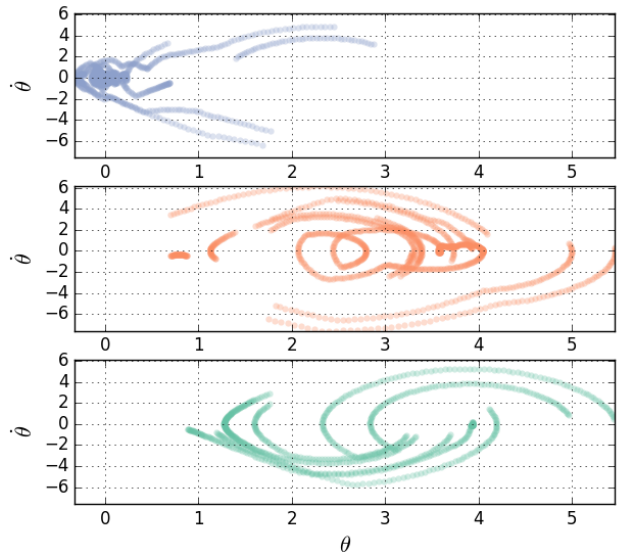


Fig. 1. Cart-pendulum system actuated about the x-axis with the unstable equilibrium defined at $\theta = 0$.



(a) Segmentation of $(\theta, \dot{\theta}, u)$ trajectories of an optimal control solution to the cart-pendulum inversion problem shown in time-domain.



(b) Segmentation of the $(\theta, \dot{\theta})$ phase portrait of an optimal control solution to the cart-pendulum inversion problem.

Fig. 2. Example of dynamical system segmentation applied to an optimal model predictive control solution to the cart-pendulum inversion problem, specified by the goal state of $(\theta, x, \dot{\theta}, \dot{x}) = (0,0,0,0)$. Despite the fact that the behavioral modes are not determined *a priori* we can still have some principled understanding of what the modes are. The identified behaviors encode negative velocities, positive velocities, and low velocity motion in modes 0, 1 and 2 respectively.

we segment an optimal control solution to the pendulum inversion task. Often the partitions generated by dynamical system segmentation may be intuitively related to the examined task. In Fig. 2(a), one can see that modes 0 and 1 represent trajectories with negative and positive velocities respectively, while mode 2 represents lower velocity motion and stabilization. Since the state-space trajectories used to train the model encode task-specific information, the behavioral modes do as well. Once a dynamical system has been segmented, the SVM's partitions of the state-space are set, and new data points will be classified according to which partition they fall into. Figure 3 shows a cross-section of the partitioned state-space manifold of the optimal controller solution to the cart-pendulum inversion shown in Fig. 2(a).

C. Graphical Representation

The product of DSS is best represented by a graph. We can define a graph $\mathbb{G} = (\overline{\mathbb{K}}, \mathbb{E})$ where the node set $\overline{\mathbb{K}}$ contains the exemplar Koopman operators synthesized from the clustering procedure. The set of edges $\mathbb{E}$ is determined by directly observing the sequences of class labels in the dataset, and tracking all unique transitions. Figure 4 illustrates how DSS relates to the resulting graph. Each node in the graph represents a distinct dynamical system over its respective partition of the state-space manifold. By traversing the graph symbolically from one node to another, traversal of the state manifold is implied. The DSS algorithm is summarized in Algorithm 1.

The graph itself encodes task-specific information embedded in the state trajectories of the training dataset. In particular, the graph's state distribution is an information-rich object that can be used for data analysis purposes. Given

Algorithm 1 Dynamical System Segmentation (DSS)

Input: Dataset $X = [x_0, ..., x_M]$, basis functions $\{\Psi(x) | \Psi(x) \in \mathbb{R}^N\}$, window size S_w, overlap % P_{ov}, minimum # of points required to form a cluster N_c

Procedure:

1: Transform the X dataset into $\Psi_X = [\Psi(x_0), ..., \Psi(x_M)]^T$ using the selected basis functions
2: Split Ψ_X into W windows of size S_w overlapping by P_{ov}
3: Calculate a Koopman operator for each window, generating the set $\mathbb{K} = \{K_0, ..., K_W\}$
4: Construct a feature array $\mathbb{K}_{flat}$ by flattening all $K_i \in \mathbb{R}^{N \times N}$ in $\mathbb{K}$ into points in $\mathbb{R}^{N^2}$ and appending them
5: Cluster using HDBSCAN($\mathbb{K}_{flat}$, N_c), and label all K_i's from one of $B+1$ discerned classes $\{C_0, ..., C_B\}$
6: Construct a set $\overline{\mathbb{K}} = \{K_0, ..., K_B\}$ of class exemplars by taking a weighted-average of all $K_i \in C_j$, $\forall j \in \{0, ..., B\}$, according to the membership probability $p(K_i | K_i \in C_j)$
7: Label all points in Ψ_X with the label $l \in \{0, ..., B\}$ of the Koopman operator they were used to generate
8: Train an SVM, $\Phi(\Psi(x))$, for projecting class labels directly onto the state-space
9: Construct a set of unique transitions $\mathbb{E}$ by tracking all sequential labels in the dataset
10: Consider $\overline{\mathbb{K}}$ as a set of nodes, and $\mathbb{E}$ as a set of edges, and construct a graph $\mathbb{G} = (\overline{\mathbb{K}}, \mathbb{E})$

Return: Graphical model $\mathbb{G}$, and trained SVM $\Phi(\Psi(x))$

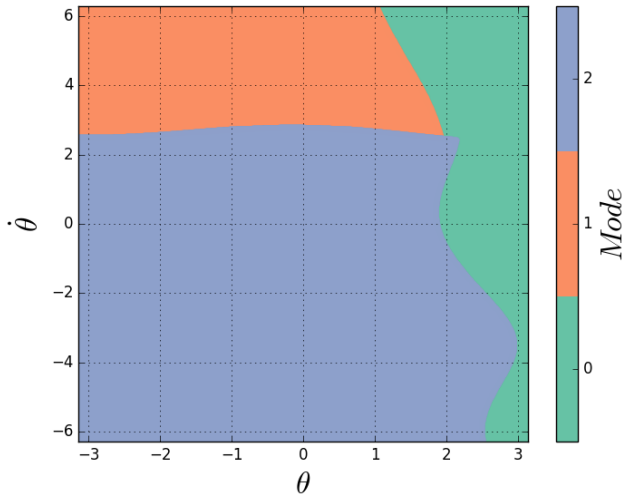


Fig. 3. State-space manifold partitions generated by an SVM trained on an optimal controller's solution for a cart-pendulum system. DSS identified 3 modes, and the SVM partitions are shown at the $(x,\dot{x}) = (1,-1)$ cross-section of the manifold.

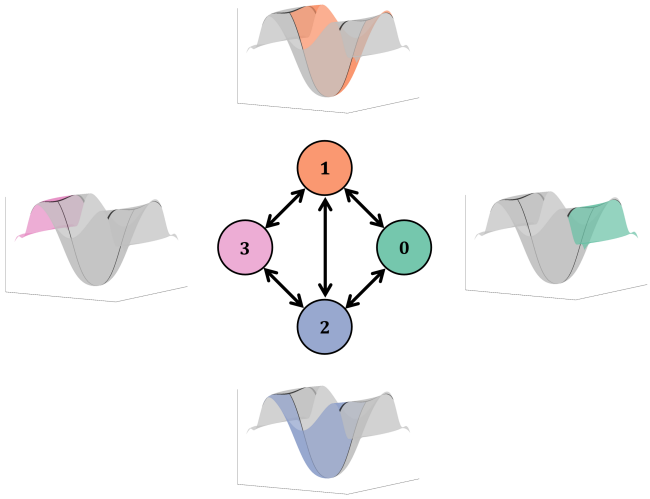


Fig. 4. **Output of the Dynamical System Segmentation algorithm:** each node in the graph is a distinct dynamical system that governs its partition of the state-space manifold generated by the SVM $\Phi(\Psi(x))$.

an optimal agent's graph $\mathbb{G}_{opt}$ constructed with DSS, we can use the trained SVM classifier $\Phi_{opt}(\Psi(x))$ to identify behaviors from the optimal agent in data from other agents. By tracking the relative frequencies of behaviors B_k from the optimal agent in another agent's trajectories, we can calculate a distribution $q(B_k)$, and directly compare it to $\mathbb{G}_{opt}$'s optimal state distribution $p(B_k)$ using the Kullback-Leibler divergence (D_{KL}) [15]

$$D_{KL}(p(B_k)||q(B_k)) = -\sum_k p(B_k) log\left(\frac{q(B_k)}{p(B_k)}\right) \qquad (4)$$

The state distributions encode coarse-grained information about the task, and their comparison can be used for performance assessment. Since an optimal agent's distribution is the most representative state distribution of a task, $D_{KL}(p(B_k)||q(B_k))$ represents the amount of task information embedded in an agent's motions, which we refer to as *task embodiment*.

III. EXPERIMENTS

The proposed assessment of task embodiment was applied to data collected from human subjects performing a cart-pendulum inversion task[1]. Data was collected using the NACT-3D—an admittance-controlled haptic robot, similar to that described in [16] and [17]. We synthesize a dataset representative of an optimal user using an optimal controller. Data from the expert is segmented by applying the DSS algorithm proposed in Section 2 in order to generate a graphical model $\mathbb{G}_{opt}$, and a set of optimal behaviors to track. $\mathbb{G}_{opt}$'s state distribution is then used as a reference to compare against the human subjects, and assess their task embodiment.

[1]The authors utilized de-identified data from a study approved by the Northwestern Institutional Review Board.

A. Human Subjects Dataset

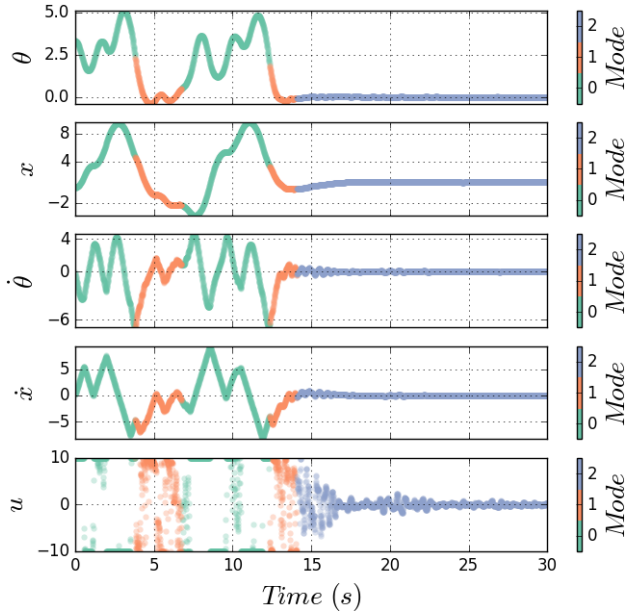
A filter-based assistance algorithm proposed in [18] for pure noise inputs, and adapted for user input in [19] and [20] was applied to a virtual cart-pendulum inversion task on the NACT-3D. The assistance physically filters the users inputs—accelerations in this case—such that their actions are always in the direction of an optimal control policy calculated in real time. All subjects were instructed to attempt to invert a virtual cart-pendulum with the goal of spending as much time as possible in the unstable equilibrium during a thirty second trial, where the cart-pendulum states were sampled at 60Hz. Subjects repeated this task for 30 trials in each of two sessions. Forty subjects completed this task with assistance in one session and without assistance in the other session. The order in which the subjects received assistance was counterbalanced to account for learning effects. An additional thirteen subjects were placed in a control group which completed both sessions without assistance.

B. Training Dataset

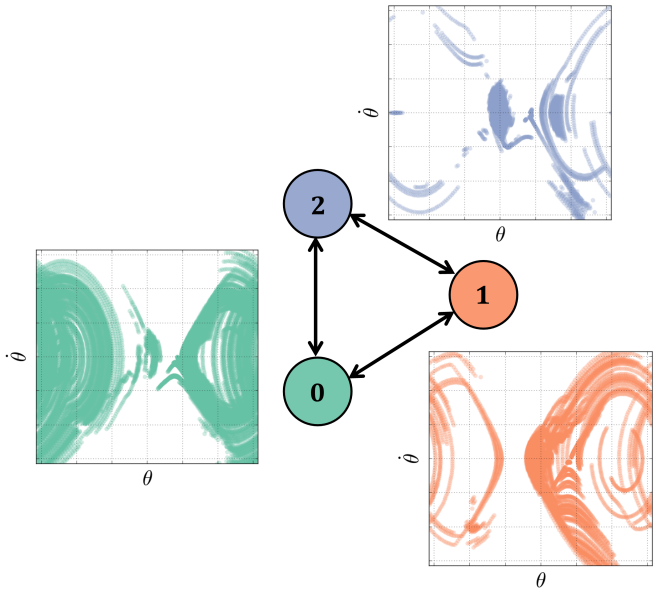
To assess task embodiment using our dynamical system segmentation technique, we synthesized an optimal baseline to compare subjects against. We generated optimal control solutions to the pendulum inversion problem using Sequential Action Control (SAC) [21], a receding-horizon model predictive optimal controller for nonlinear and nonsmooth systems, over a randomized set of initial conditions. The controller's objective was $(\theta, x, \dot{\theta}, \dot{x}) = (0,1,0,0)$, with linear quadratic cost parameters of $Q = diag([200, 80, 0.01, 0.2])$. Thirty optimal control trials of thirty seconds each were generated so as to mirror the amount of data collected from human subjects.

C. Optimal Graph

We apply the DSS algorithm to the synthesized trials to generate an optimal graphical model. The choice of



(a) Time-domain segmentation of a selected optimal control solution of the pendulum inversion task. Mode 0 corresponds to energy pumping and swing-up, mode 1 corresponds to energy removal and slow-down, and mode 2 correponds to stabilization.



(b) Graph $\mathbb{G}_{opt}$ resulting from the the dynamical system segmentation of a dataset of 30 optimal control solutions to the cart-pendulum inversion task. The set of segmented behaviors are shown projected onto the system's phase portrait over the domain $\{(\theta,\dot{\theta}) : (-\pi,\pi)\times(-2\pi,2\pi)\}$.

Fig. 5. Data-driven identification of exemplar behaviors through the use of dynamical system segmentation and the resulting graphical model from the segmentation of the pendulum inversion task.

basis functions has the greatest effect on the algorithm's performance because of how they reshape the state-space boundaries. The set of basis functions selected for this task were

$$\Psi(x) = [\theta,\ x,\ \dot{\theta},\ \dot{x},\ u,\ u\ cos(\theta),\ u\ cos(\dot{\theta}),\ |u_{sat}|cos^2\left(\frac{u\pi}{|u_{sat}|}\right),\ \dot{x}^2,\ 1] \quad (5)$$

where $|u_{sat}|$ is the optimal controller's saturation limit on the control effort. The basis functions were selected from the set of linear combinations of second order polynomial and sinusoidal functions. Since clustering occurs in $\mathbb{R}^{N^2}$ space, where N is the number of basis functions, we chose a low-dimensional set of representative basis functions from the larger set of linear combinations of polynomial and sinusoidal functions. This dimensionality reduction can be achieved via multiple methods, such as principal component analysis [15].

Figure 5(a) depicts the behaviors identified from the exemplar trial. The identified modes 0, 1 and 2 correspond to energy pumping and swing-up, energy removal and slow-down, and stabilization, respectively. These modes represent a set of behaviors that an expert user should exhibit in succeeding at the task.

We synthesize the optimal graph $\mathbb{G}_{opt}$ using the identified behaviors, and then use the graph's state distribution $p(B_k)$, shown in Fig. 6, as the optimal baseline with which to assess the subjects' task embodiment. The graph $\mathbb{G}_{opt}$ and the segmented behaviors projected onto the $(\theta,\dot{\theta})$ phase portrait by the trained SVM $\Phi_{opt}(\Psi(x))$ is shown in Fig. 5(b).

The human data is analyzed by using the trained SVM $\Phi_{opt}(\Psi(x))$ to detect the identified behaviors in each subject's trials with and without the presence of assistance. By tracking the relative frequencies of behaviors B_k we can generate a distribution $q(B_k)$ with which to compare to $\mathbb{G}_{opt}$'s state distribution $p(B_k)$. We compare the distributions using task embodiment quantified by $D_{KL}(p(B_k)||q(B_k))$, where a lower D_{KL} indicates greater embodiment of the task. This same procedure is applied to the two sets of data from the control group subjects.

IV. RESULTS

We analyzed the human subjects dataset, and found that task embodiment is a reliable predictor of physical assistance. All subjects better embodied the task in their assisted trials, whereas there was no observed difference in the control group. In addition to comparing the groups using

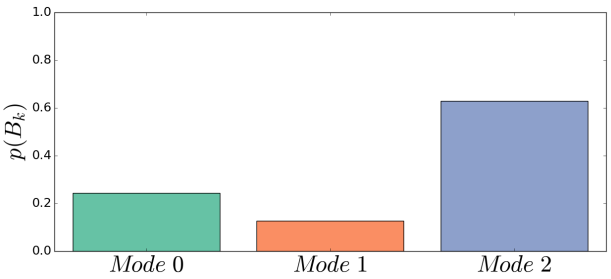


Fig. 6. Optimal behavior mode distribution $p(B_k)$ determined by 30 optimal control solutions to the pendulum inversion task over randomized initial conditions. $p(B_k) = (0.2437,\ 0.1275,\ 0.6288)$.

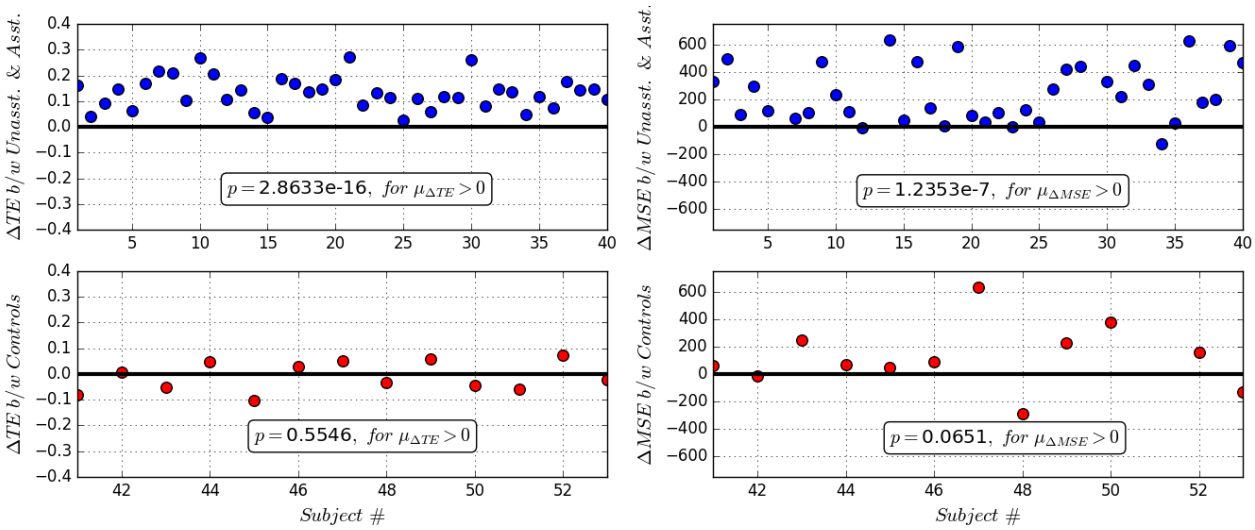


Fig. 7. **Summary of experimental results:** subjects in the experimental group who received assistance (blue) were compared to their own unassisted trials. The control group subjects (red) were compared from their initial session to their final session. The pair of plots to the left show the difference in task embodiment between the sessions of the experimental and control groups. The plots to the right show the difference between the same groups using the integrated MSE instead. We note that task embodiment is a better predictor of whether or not a subject received assistance than integrated MSE.

task embodiment, we also evaluated a standard metric for assessing task performance, the integrated MSE. Specifically, we calculated the integrated MSE with respect to a goal state of $(\theta, \dot{\theta}) = (0,0)$. We found that integrated MSE is also a good predictor of assistance, but at a lower significance level, and lower effect size than task embodiment.

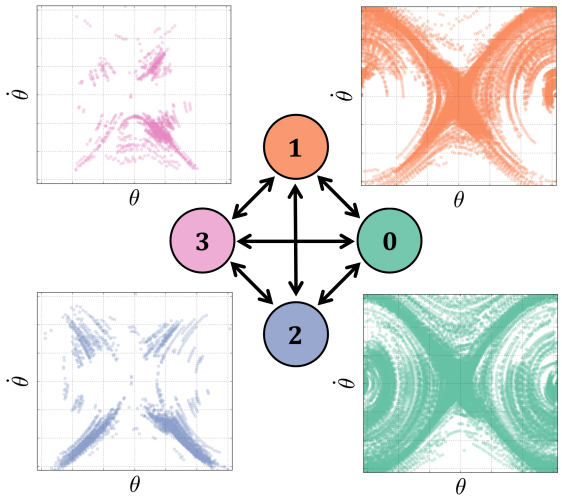
A paired two-sample t-test on the task embodiment of each subject with and without assistance showed that the subjects' sessions with assistance ($\mu = 0.0756,\ \sigma = 0.0436$) significantly outperformed the sessions without assistance ($\mu = 0.2084,\ \sigma = 0.0560$), with $p = 2.8633\text{e-}16,\ t(39) = 13.4876$, and an effect size of $d = 2.1326$. In contrast, there was no significant difference between the first session ($\mu = 0.2039,\ \sigma = 0.0406$) and the second session ($\mu = 0.1943,\ \sigma = 0.0400$) of the control group when a paired two-sample t-test was performed $p = 0.5564,\ t(12) = -0.6051$. These results indicate that task embodiment reliably captures assistance and lack thereof.

We also performed a paired two-sample t-test on the MSE of each subject with and without assistance, and found that the session with assistance ($\mu = 124.66,\ \sigma = 119.96$) significantly outperformed the session without assistance ($\mu = 428.88,\ \sigma = 307.46$), but with a lower significance and effect size than task embodiment, with $p = 1.2195\text{e-}7,\ t(39) = 6.4526$ and an effect size of $d = 1.0202$. Again, we applied the paired two-sample t-test to the control group and found that the first session ($\mu = 352.83,\ \sigma = 217.67$) did not significantly outperform the second ($\mu = 546.31,\ \sigma = 446.10$), had $p = 0.0649,\ t(12) = 2.0320$. These results indicate that MSE can also predict the presence of assistance, but not as reliably as task embodiment. The task embodiment measure has both a significance level several orders of magnitude greater than that of integrated MSE, and showed an effect size that was twice as large as integrated MSE. This demonstrates that task embodiment captures the large difference between the assisted and unassisted trials. These results are summarized in Fig. 7.

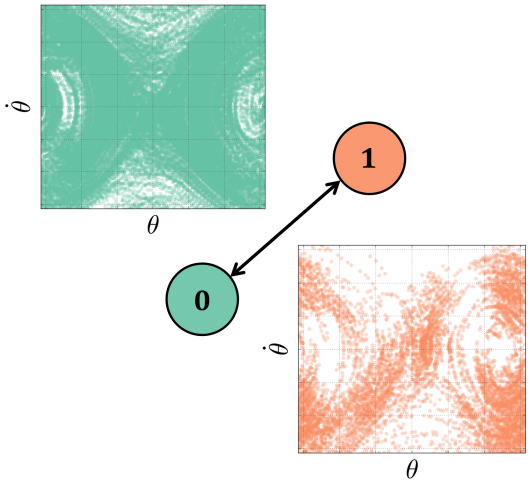
The experimental methodology presented in this study analyzed subject data by means of comparison to an optimal baseline. While the methodology is informative, it cannot detail subject performance without comparison to the optimal agent. Given the same choice of basis functions and algorithm parameters, we can use DSS to generate graphs of each subject with and without assistance, and analyze the identified behaviors in each graph directly. This alternative methodology allows us to take human motion data and represent it graphically, which creates the opportunity for analyzing human motion using graph-theoretic principles. Figures 8(a) & 8(b) illustrate the graphical models constructed from the assisted and unassisted trials of a representative subject. We note that the extracted behaviors from the unassisted trials in Fig. 8(b) lack structure, and more closely resemble noise-driven behaviors. In contrast, by inspecting the graph from the subject's assisted trials in Fig. 8(a), we observe the emergence of finite structure in the identified behaviors.

V. CONCLUSIONS

In this study, we proposed an information-theoretic approach to human motion analysis. The DSS algorithm formulated in Section 2 produces graphical models that encode task-specific information. By tracking the degree of task embodiment, we are able to decode complex relationships in human motion. We validated our approach through an analysis of data from human subjects performing a virtual cart-pendulum inversion task with and without assistance. We determined that task embodiment is a better predictor of the presence of assistance than integrated MSE. Task embodiment identified the presence of task assistance at a higher significance level and with a larger effect size than integrated MSE. Thus, the experimental results provide



(a) Resulting graph and state-space projections from the dynamical system segmentation of subject 16's assisted trials of the pendulum inversion task.



(b) Resulting graph and state-space projections from the dynamical system segmentation of subject 16's unassisted trials of the pendulum inversion task.

Fig. 8. Constructed graphical models from the dynamical system segmentation of a representative experimental subject with and without assistance. The graph's nodes project onto the system's $(\theta, \dot{\theta})$ phase portrait over the domain $\{(\theta, \dot{\theta}) : (-\pi, \pi) \times (-2\pi, 2\pi)\}$ of the state-space manifold according to the phase portraits shown alongside the nodes. We note that the behaviors of the unassisted subject's phase portraits are noise-driven and show no discernible structure.

strong support for the use of information measures in human motion analysis.

References

[1] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[2] F. Mussa–Ivaldi and E. Bizzi, "Motor learning through the combination of primitives," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 355, no. 1404, pp. 1755–1769, 2000.

[3] D. D. Vecchio, R. M. Murray, and P. Perona, "Primitives for human motion: A dynamical approach," *IFAC Proceedings*, vol. 35, no. 1, pp. 25 – 30, 2002.

[4] D. D. Vecchio, R. M. Murray, and P. Perona, "Decomposition of human motion into dynamics-based primitives with application to drawing tasks," *Automatica*, vol. 39, no. 12, pp. 2085 – 2098, 2003.

[5] C. Gonzalez, D. Svenkeson, D. J. Kim, M. J. McKeown, and M. Oishi, "Detection of manual tracking submovements in parkinson's disease through hybrid optimization," *IFAC Proceedings*, vol. 48, no. 27, pp. 291 – 297, 2015.

[6] G. Mallapragada, I. Chattopadhyay, and A. Ray, "Automated behaviour recognition in mobile robots using symbolic dynamic filtering," *Journal of Systems and Control Engineering*, vol. 222, pp. 409–424, 2008.

[7] C. Rao, A. Ray, S. Sarkar, and M. Yasar, "Review and comparative evaluation of symbolic dynamic filtering for detection of anomaly patterns," *Signal, Image and Video Processing*, vol. 3, no. 2, pp. 101–114, 2009.

[8] B. Koopman, "Hamiltonian systems and transformation in Hilbert space," *Proceedings of the National Academy of Sciences*, vol. 17, no. 5, pp. 315–318, 1931.

[9] J. Tu, C. Rowley, D. Luchtenburg, S. Brunton, and J. Kutz, "On dynamic mode decomposition: Theory and applications," *Journal of Computational Dynamics*, vol. 1, p. 391, 2014.

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

[11] I. Abraham, G. Torre, and T. Murphey, "Model-based control using Koopman operators," *Robotics: Science and Systems*, 2017.

[12] R. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Advances in Knowledge Discovery and Data Mining*. Springer, 2013, pp. 160–172.

[13] L. McInnes, J. Healy, and S. Astels, "HDBSCAN: Hierarchical density based clustering," *The Journal of Open Source Software*, vol. 2, no. 11, 2017.

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[15] C. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.

[16] A. H. Stienen, J. G. McPherson, A. C. Schouten, and J. P. Dewald, "The ACT-4D: a novel rehabilitation robot for the quantification of upper limb motor impairments following brain injury," in *IEEE Int. Conf. on Rehabilitation Robotics*, 2011, pp. 1–6.

[17] M. D. Ellis, Y. Lan, J. Yao, and J. P. Dewald, "Robotic quantification of upper extremity loss of independent joint control or flexion synergy in individuals with hemiparetic stroke: a review of paradigms addressing the effects of shoulder abduction loading," *Journal of NeuroEngineering and Rehabilitation*, vol. 13, no. 1, p. 95, 2016.

[18] E. Tzorakoleftherakis and T. D. Murphey, "Controllers as filters: Noise-driven swing-up control based on Maxwell's demon," in *IEEE Conf. on Decision and Control (CDC)*, 2015, pp. 4368–4374.

[19] K. Fitzsimons, E. Tzorakoleftherakis, and T. D. Murphey, "Optimal human-in-the-loop interfaces based on Maxwell's demon," in *American Control Conference (ACC)*, 2016, pp. 4397–4402.

[20] A. Kalinowska, K. Fitzsimons, J. P. Dewald, and T. D. Murphey, "Online user assessment for minimal intervention during task-based robotic assistance," in *Robotics: Science and Systems*, 2018.

[21] A. Ansari and T. D. Murphey, "Sequential action control: Closed-form optimal control for nonlinear and nonsmooth systems," *IEEE Transactions on Robotics*, vol. 32, 2017.