Dynamic Mode Decomposition for Perturbation Estimation in Human Robot Interaction

Conference Paper · August 2014

DOI: 10.13140/2.1.2957.7601

Erik Berger
Technische Universität Bergakademie Freiberg

David Vogt
Technische Universität Bergakademie Freiberg

Heni Ben Amor
Arizona State University

Dynamic Mode Decomposition for Perturbation Estimation in Human Robot Interaction

Erik Berger[1], Mark Sastuba[2], David Vogt[1], Bernhard Jung[1], Heni Ben Amor[3]

***Abstract*— In many settings, e.g. physical human-robot interaction, robotic behavior must be made robust against more or less spontaneous application of external forces. Typically, this problem is tackled by means of special purpose force sensors which are, however, not available on many robotic platforms. In contrast, we propose a machine learning approach suitable for more common, although often noisy sensors. This machine learning approach makes use of *Dynamic Mode Decomposition* (DMD) which is able to extract the dynamics of a nonlinear system. It is therefore well suited to separate noise from regular oscillations in sensor readings during cyclic robot movements under different behavior configurations. We demonstrate the feasibility of our approach with an example where physical forces are exerted on a humanoid robot during walking. In a training phase, a snapshot based DMD model for behavior specific parameter configurations is learned. During task execution the robot must detect and estimate the external forces exerted by a human interaction partner. We compare the DMD-based approach to other interpolation schemes and show that the former outperforms the latter particularly in the presence of sensor noise. We conclude that DMD which has so far been mostly used in other fields of science, particularly fluid mechanics, is also a highly promising method for robotics.**

Fig. 1: A NAO robot detects the existence and amount of external perturbations applied by a human interaction partner. The direction and amount of the perturbation is used to infer human guidance, e.g., walk backwards.

I. Introduction

Robots need accurate and efficient sensing capabilities in order to react to influences from the environment. This is particularly true for robots that are engaging in joint physical activities with a human partner. In such scenarios, forces and torques applied by the human can severely perturb the execution of a motor skill and need to be accounted for in the decision making process. In order to appropriately respond to a perturbation, a robot needs to detect both the occurrence of such an event as well as the degree by which it occurred. One way of implementing such detection is to use readings from a special purpose sensor, e.g., force-torque sensor, along with a thresholding method. However, such sensors are often heavy, expensive and prone to error. In practice many sensors return non-zero readings even when the robot merely moves. Distinguishing between external, human perturbations and natural variation in the sensor values can therefore become a challenging task.

In this paper, we present a machine learning approach to robot sensing that is well suited for identifying external influences caused by a human partner as shown in Figure 1. The approach focuses on learning probabilistic, *behavior-specific* models of regular oscillations in sensor readings during motor skill execution. These models are used to (1) identify perturbations by detecting irregularities in sensor readings that cannot be explained by the inherent noise, and (2) to generate a continuous estimate of the amount of external perturbation. Due to the data-driven nature of the approach, no detection threshold needs to be provided by the user. The presented perturbation filter can be regarded as a virtual force sensor that produces a continuous estimate of external forces. To this end, we use Sparse Dynamic Mode Decomposition to learn a model of the system dynamics during the robot execution of a specific motor skill. During human-robot interaction, the model is then used to determine the existence and amount of irregularities in the sensor readings. By modeling the correlations as well as the time-dependent variation in the original sensor values, our filter can robustly deal with uncertainties in estimating the human physical influence on the robot. During task execution, the estimated perturbation value can be used to compensate for the external forces or infer the intended guidance of a human interaction partner. Experiments on a real robot show that

[1]Institute of Computer Science, Technical University Bergakademie Freiberg, Bernhard-von-Cotta-Str. 2, 09599 Freiberg, Germany

[2]Institute of Mechanics and Fluid Dynamics, Technical University Bergakademie Freiberg, Lampadiusstr. 4, 09599 Freiberg, Germany

[3]Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, 801 Atlantic Drive, Atlanta, GA 30332-0280, USA

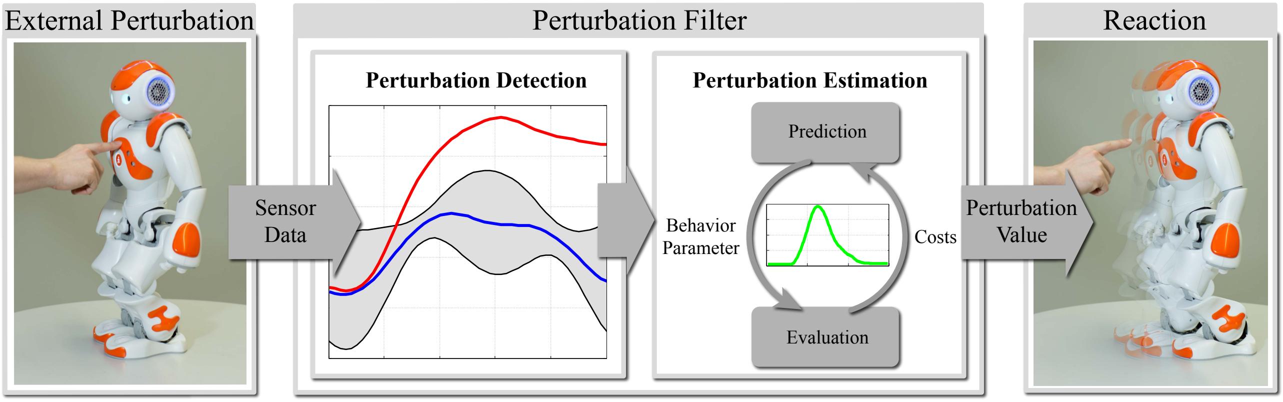


Fig. 2: An overview of the presented machine learning approach. An external perturbation is filtered by using a previously learned predictive model of behavior parameters. After detecting a perturbation the strength and direction is estimated in the behavior parameter space. The resulting perturbation value can be used for an adequate reaction.

learned models can be used to accurately determine even small disturbances.

II. Related Work

A popular approach to the design of human-robot interaction (HRI) is the use of mediating artefacts, such as pendants, joysticks or mobile phones [1]. The approach allows a programmer to pre-specify a set of tasks, commands and corresponding robot reactions. Since communication is mediated through the artefact, no filtering or interpretation of the human commands is required.

In recent years, more natural and intuitive approaches to HRI have gained popularity. Various researchers have proposed the so-called *soft robotics* paradigm: compliant robots that "can cooperate in a safe manner with humans" [2]. An important robot control method for realizing such a compliance is impedance control [3]. Impedance control can be used to allow for touch based interaction and human guidance. To this end, impedance controllers require accurate sensing capabilities, in the form of force-torque sensors. However, such sensors are typically heavy, expensive and suffer from noise. Other sensors, such as current based torque sensors are even more prone to issues related to noise and drift.

Still, the ability to sense physical influences is at the core of recent advances made in the field of HRI. For example, Lee et al. [4] use impedance control and force-torque sensors in order to realize human-robot interaction during programming by demonstration tasks. Wang et al. [5] present a robot that can adapt its dancing steps based on the external forces exerted by a human dance partner. Ben Amor et al. [6] uses touch information to teach new motor skills to a humanoid robot. Touch information is therefore only used to collect data for subsequent learning of a robotic motor skill. Robot learning approaches that are based on such kinesthetic teach-in have gained considerable attention in the literature, with similar results reported in [7] and [8]. A different approach aiming at joint physical activities between humans and robots has been reported in [9]. Ikemoto et al. use Gaussian mixture models to adapt the timing of a humanoid robot to that of a human partner in close-contact interaction scenarios. The parameters of the interaction model are updated using binary evaluation information obtained from the human. The approach significantly improves physical interactions, but is limited to learning timing information.

Stückler et al. [10] present a cooperative transportation task where a robot follows the human guidance using arm compliance. In doing so, the robot recognizes the desired walking direction through visual observation of the object being transported. A similar setting has been investigated by Yokoyama et al. [11]. They use a HRP-2P humanoid robot equipped with a biped locomotion controller and an aural human interface to carry a large panel together with a human. Forces measured with sensors on the wrists are utilized to derive the walking direction. Similarly, Bussy et al. [12] also use force-torque sensors on the wrists to adapt the robot behavior during object transportation tasks. Lawitzky et al. [13] also shows how load sharing and role allocation can be used to balance the contribution of each interaction partner depending on the current situation.

The main drawback of the above approaches is that they require special aural and visual input devices or force sensors which are not present on many robot platforms. Additionally, none of the approaches using force-torque sensors addresses the problem of uncertainty in the provided measurements. As a result, all of these approaches assume high-quality sensing capabilities and low-speed execution of the joint motor task. In contrast to the above approaches, we propose a new filtering algorithm that can *learn* the natural variation in sensor values during a motor skill. Using predictive models learned by Dynamic Mode Decomposition, the filtering algorithm also estimates the perturbation which best explains an observed set of new sensor values.

III. Approach

In our approach the robot recognizes and automatically differs between strength and direction of external perturbations which may be caused by a human interaction partner. An overview of the approach can be seen in Figure 2. First,

we record training data for a behavior with different parameter configurations, e.g. walking with varying step lengths, in a controlled environment without external perturbations. The training data is used to learn a dynamic model utilizing a state of the art interpolation method from fluid dynamics (DMD).
During behavior execution an external perturbation is detected by comparing the recorded training data with the current sensor data. For estimating the perturbation value the current sensor readings are compared to new sensor values generated from the learned model. The perturbation value is then calculated from the difference between the current behavior parameter and the behavior parameter of the sensor characteristic with the highest compliance to the current sensor characteristic.
In the following, we will address each step of our approach in more detail. Subsequently, we will describe how perturbation detection, model learning and perturbation estimation are realized in order to allow a whole variety of HRI scenarios.

A. Recording Training Data

The first step in our approach is to record training data that reflects the evolution of sensor values during regular execution of a motor skill. It is important to record several executions of the behavior, since motor skills can often be executed with different parameters, e.g., varying step lengths during walking. However, since we use machine learning methods, we will later see that the number of required training data can be limited to about five examples.
Each recorded example contains training data sampled with $100Hz$ for one repetition of the modelled robot behavior. In our specific case of training a perturbation filter for walking, we record both the *center of mass* (CoM) and the proper acceleration of the robot for four seconds. Acquiring training data requires less than one minute in total.

B. Phase Estimation

Since we are dealing with time-varying data, it is important to estimate the phase of the robot during the execution of a motor skill. Depending on the phase, e.g., the left leg is lifted, the variance in the sensor readings can change drastically. To determine the current phase, a time window of sensor values is captured and temporally aligned to the training data. To this end, we use the *dynamic time warping* technique (DTW) [14]. DTW is a time series alignment algorithm for measuring the similarity between two temporal sequences $\mathbf{X} = (x_1, \ldots, x_N)$ and $\mathbf{Y} = (y_1, \ldots, y_M)$ of length $N \in \mathbb{N}$ and $M \in \mathbb{N}$. In our specific case, the goal is to find the optimal correspondence between the sensor data $\mathbf{Y}$ recorded during the training phase and the currently observed sequence $\mathbf{X}$, where M is much larger then N.
Due to this significant difference in length of $\mathbf{X}$ and $\mathbf{Y}$, we formulate our task as finding a subsequence

$$\mathbf{Y}(a^* : b^*) = (y_{a^*}, y_{a^*+1}, \ldots, y_{b^*}) \tag{1}$$

with $1 \leq a^* \leq b^* \leq M$, where a^* is the starting index and b^* is the end index that optimally fits to the corresponding

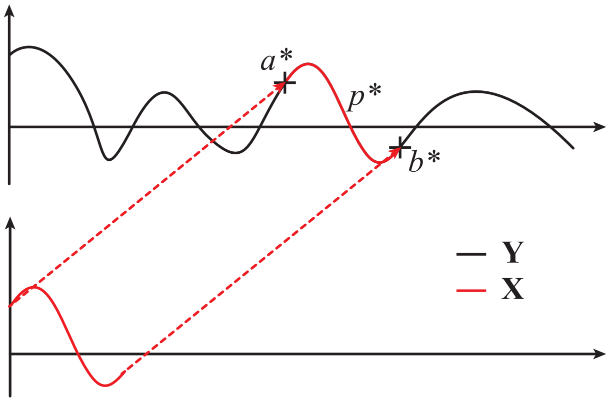


Fig. 3: Given the recorded data (black) and the partial observation (red), we calculate the optimal warping path p^* between a^* and b^* .

subsequence $\mathbf{X}$. This technique is also known as *subsequence dynamic time warping* (SDTW) [15]. To find the optimal subsequence we first have to calculate the accumulated cost matrix D, which for SDTW is defined as

$$D(n,1) = \sum_{k=1}^{n} c(x_k, y_1), n \in [1:N],$$
$$D(1,m) = c(x_1, y_m), m \in [2:M],$$
$$D(n,m) = min\{D(n-1,m-1), D(n-1,m), \\ D(n,m-1)\} + c(x_n, y_m)$$

where c is a local distance measure, which in our case is defined as $c = |x - y|$. The goal of the SDTW algorithm is to determine the path with minimal overall costs C ending at (b^*, M), where b^* is given by

$$b^* = \operatorname*{argmin}_{b \in [1:M]} D(N, b). \tag{2}$$

To determine the warping path $p^* = (p_1, \ldots, p_L)$ starting at $p_1 = (a^*, 1)$ and ending at $p_L = (b^*, M)$ a dynamic programming recursion is used. As illustrated in Figure 3 the resulting path p^* represents the optimal subsequence of $\mathbf{X}$ in $\mathbf{Y}$. As a result SDTW can be used to estimate the current state of a behavior using a subset of temporally measured sensor values which are mapped to the recorded data. In more detail, we use the subsequence p^* as prediction of sensor values at the current state.

C. Perturbation Detection

Due to uncertainties in the real world, a motor skill is never twice executed in exactly the same way. To accommodate for such natural noise in the behavior, we use learned, behavior-specific information about the temporal evolution of sensor variances.
Different approaches can be used to learn such a probabilistic model. One solution is to use *Gaussian Process Regression* (GPR) [16]. An important advantage of GPR is the ability to learn a probabilistic model from a small set of training

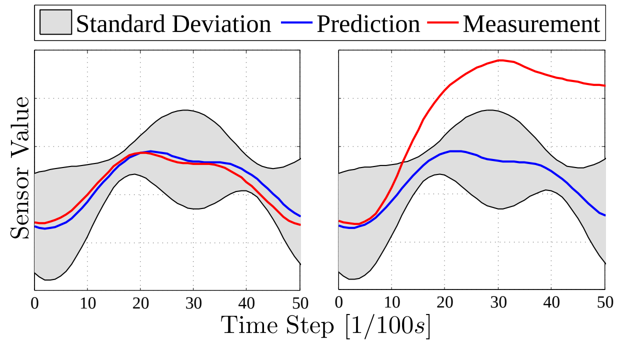


Fig. 4: After estimating the current phase of the behavior the deviation between the measured and predicted sensor values can be used to detect external influences. Left: There is no external perturbation. Right: An external perturbation is detected.

data. The main drawback of this approach is the large computational effort. Another, computationally less expensive solution is to compute the standard deviation σ for each time step of the recorded data separately.

Given a probabilistic model as described above, we can detect a perturbation by calculating the likelihood of the current sensor readings. In our implementation, we trigger a detection when the sensor values are outside of the computed standard deviation σ. Figure 4 shows an example for a regular and a disturbed execution of a behavior.

D. Modelling Robot Dynamics using Dynamic Mode Decomposition

In this section we use *Dynamic Mode Decomposition* (DMD) to learn a predictive model describing the change in sensor values under different behavior parameters. DMD is a novel data processing technique from fluid dynamics and was introduced in [17] and [18]. Once a DMD is learned, it can be applied to simulated sensor values under different parameter conditions.

DMD presents a modal decomposition for nonlinear flows and features the extraction of coherent structures with a single frequency and growth/decay rate. It computes a linear model which approximates the underlying nonlinear dynamics. Given is an equidistant snapshot sequence $N+1$ of an observable $x = (u_1, \ldots, u_M)^* \in \mathbb{C}^{M\times 1}$ which is stacked into two matrices $K_1 = [x_0 \ldots, x_{N-1}] \in \mathbb{C}^{M\times N}$ and $K_2 = [x_1 \ldots, x_N] \in \mathbb{C}^{M\times N}$. The matrices K_1 and K_2 are shifted by one time step Δt and can be linked via the mapping matrix (system matrix) $A \in \mathbb{C}^{M\times M}$ such that $K_2 = AK_1 = K_1 S$. Since the data stem from experiments, the system matrix A is unknown and for a very large system it is computationally impossible to solve the eigenvalue problem directly as well as to fulfill the storage demand [19]. The idea is to solve an approximate eigenvalue problem by projecting A onto an N-dimensional Krylow subspace and to compute the eigenvalues and eigenvectors of the resulting low-rank operator as described in [20]. One type of Krylow methods is the Arnoldi algorithm and the knowledge of A is not required for the following variant: $x_N = a_0x_0 + a_1x_1 + \cdots + a_{N-1}x_{N-1} + r$. The final snapshot x_N can be expressed as a linear combination of the previous ones $[x_0, \ldots, x_{N-1}]$ by computing the weighting factors $[a_0, \ldots, a_{N-1}]$, considering the residual r is minimized in a least squares sense, to form the companion matrix

$$S = \begin{pmatrix} 0 & & & & a_0 \\ 1 & 0 & & & a_1 \\ & \ddots & \ddots & & \vdots \\ & & 1 & 0 & a_{N-2} \\ & & & 1 & a_{N-1} \end{pmatrix} \in \mathbb{C}^{N\times N}. \tag{3}$$

In [17] the author describes a more robust solution, which is achieved by applying a *singular value decomposition* on K_1 such that $K_1 = U\Sigma W^*$. The *full-rank* matrix $\tilde{S} \in \mathbb{C}^{N\times N}$ is determined on the subspace spanned by the orthogonal basis vectors U of K_1, described by $\tilde{S} = U^* K_2 W \Sigma^{-1}$. Solving the eigenvalue problem $\tilde{S}\mu = \lambda\mu$ leads to a subset of complex eigenvectors μ. The DMD modes are defined by $\Phi = U\mu$, which implies a mapping of the eigenvectors $\mu \in \mathbb{C}^{N\times N}$ from a lower dimensional space to a higher dimensional space $\mathbb{C}^{M\times N}$. The complex eigenvalues λ contain growth/decay rates $\delta = \Re[\log(\lambda)]/\Delta t$ and frequencies $f = \Im[\log(\lambda)]/(2\pi\Delta t)$ of the corresponding DMD modes Φ. The temporal behavior of the DMD modes is contained in the Vandermonde matrix V_{and}, which is formed by

$$V_{and} = \begin{bmatrix} 1 & \lambda_1 & \cdots & \lambda_1^{N-1} \\ 1 & \lambda_2 & \cdots & \lambda_2^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_N & \cdots & \lambda_N^{N-1} \end{bmatrix}. \tag{4}$$

The DMD modes Φ must be scaled in order to perform a data recalculation of the first snapshot sequence $K_1 = \Phi D_\alpha V_{and}$. Therefore, having a look into V_{and} shows that the first snapshot x_0 is independent from temporal behavior since $\lambda = [\lambda_1, \ldots, \lambda_N] = 1$. The scaling factors $\alpha = [\alpha_1 \ldots \alpha_N]^*$ are calculated by $\Phi D_\alpha = x_0$, where $D_\alpha = diag\{a\}$.

A new solution to find the scaling vectors α was introduced in [21]. Here, α is obtained by considering the temporal growth/decay rates of the DMD modes in order to approximate the entire data sequence K_1 optimally. Therefore the problem can be brought into the following form

$$\min_\alpha J(\alpha) = |\Sigma W^* - \mu D_\alpha V_{and}|_F^2 \tag{5}$$

which is a convex optimization problem. Its solution leads to

$$\alpha = ((\mu^*\mu) \circ (\overline{V_{and}V_{and}^*}))^{-1}\overline{diag(V_{and}W\Sigma^*\mu)} \tag{6}$$

where the over line denotes the complex-conjugate of a vector/matrix. However, the key challenge is to identify a subset of DMD modes that captures the most important dynamic structures in order to achieve a good quality approximation. To solve that problem, the *sparsity-promoting dynamic mode decomposition* (SDMD) [21] was developed. The sparsity structure of the vector of amplitudes α is fixed in order

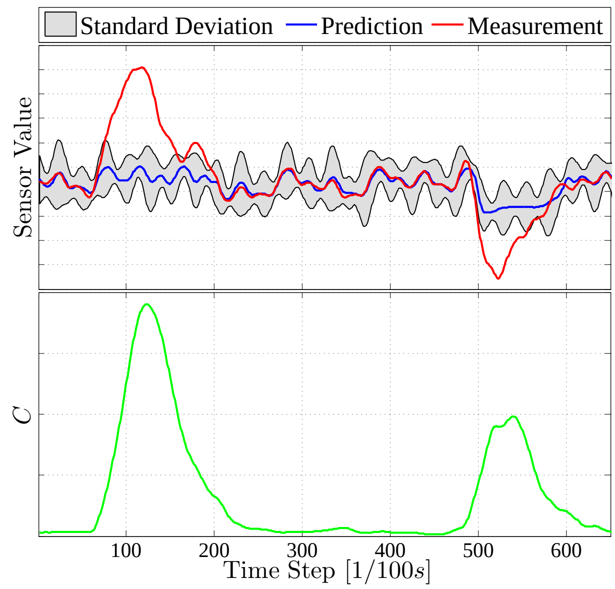


Fig. 5: External perturbations which differ in strength and direction are increasing the overall warping costs C during behavior execution.

to determine the optimal values of the non-zero amplitudes. Therefore the objective function $J(\alpha)$ is extended with an additional term such that

$$\min_{\alpha} J(\alpha) + \gamma \sum_{i=1}^{N} |\alpha_i| \,, \tag{7}$$

where γ denotes a regularization parameter that focuses on the sparsity of the vector α. As a result, instead of considering only the modes with largest amplitudes, the sparsity-promoting DMD aims to identify the modes that have the strongest influence on the entire time sequence. The lower the number of non-zero amplitudes, the more the sparse-promoting DMD concentrates on the low-frequency modes. As already mentioned, the data presented here stems from low cost sensors which may be affected by disturbance. Hence, forcing a low number of non-zero amplitudes in α can reduce the influence of noise in the approximation.

For our implementation of DMD in a human robot interaction scenario, the snapshot data $N+1$ is represented by the sensor data recorded during training data acquisition. Each column of the snapshot matrices K_1 and K_2 contains a fixed number of sensor values, i.e., the longitudinal CoM.

E. Calculating a Continuous Measure of Perturbation

If the deviation between measured and predicted sensor value is larger than the allowed variance σ we assume that an external perturbation is influencing the execution of the behavior. However, the question remains: how strong is the external perturbation?

To estimate the strength of the perturbation, we simulate different behavior parameters using the learned DMD model and select the one that produces sensor values similar to our current readings. For this task, we make use of the previously described SDTW method. As mentioned the SDTW finds the optimal warping path p^* for a currently measured subsequence $\mathbf{X}$ to a previous recorded dataset $\mathbf{Y}$. Whenever a perturbation is detected, we perform iterative optimization by generating predictions using a DMD model and calculating the warping costs using SDTW. The goal of this optimization process is to identify the behavior parameter that would best explain the currently observed sensor values. Optimization is performed using a stochastic optimization technique, i.e, *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES). The warping costs C generated by SDTW are used as objective function. Figure 5 shows the warping costs C calculated during a walking task.

The behavior parameter which produces least costs C is regarded as the true behavior parameter if human forces are taken into account. Accordingly, we can generate an estimate

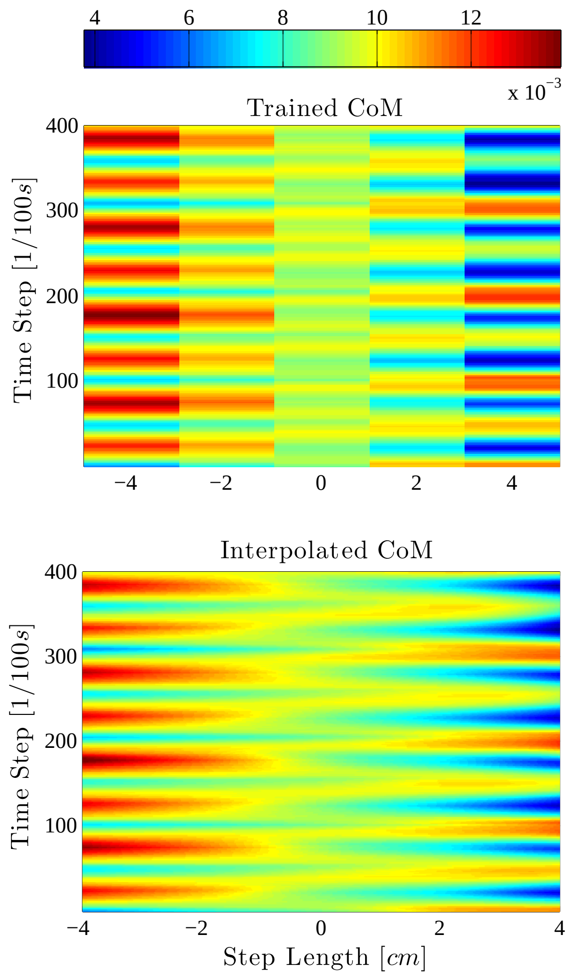


Fig. 6: DMD is used to generate new sensor values for unknown parameter settings. Top: The training data which consists of five equidistant samples of the longitudinal CoM during walking. Bottom: The longitudinal CoM is interpolated with an interval of $0.01cm$ resulting in predictions for 800 possible parameter configurations.

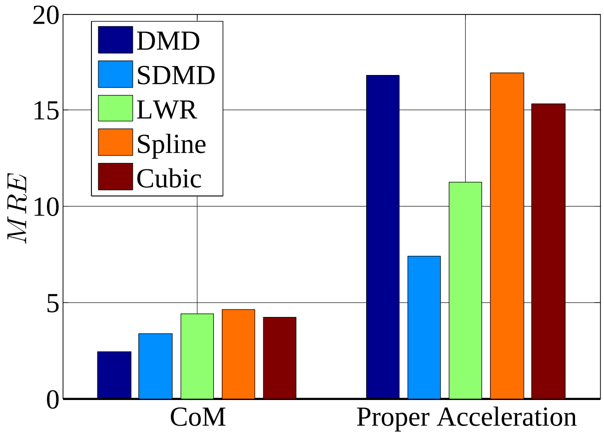


Fig. 7: The DMD techniques are compared with a set of classical interpolation schemes. Left: The DMD shows the highest accuracy for the CoM. Right: In presence of high noise, which is the case for proper acceleration estimates, SDMD produces higher accuracy than DMD or classical interpolation schemes.

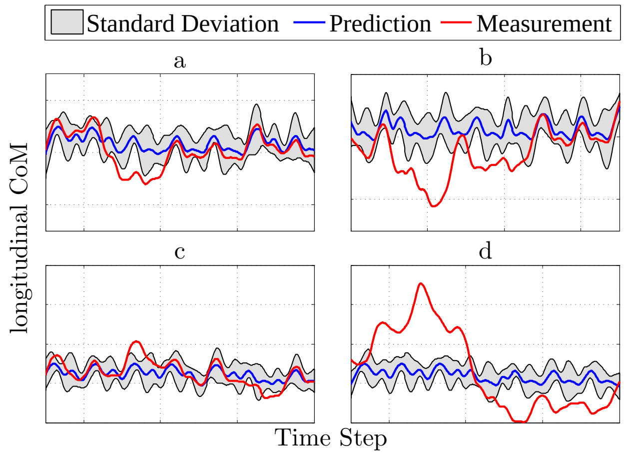


Fig. 9: Perturbation detection during walking using the robot's longitudinal CoM. Top Left: Slight push from the front. Top Right: Strong push from the front. Bottom Left: Slight push from the back. Bottom Right: Strong push from the back.

for the human forces by calculating the difference between the behavior parameter used to control the robot and the behavior parameter identified by the learned model.

IV. Experiments

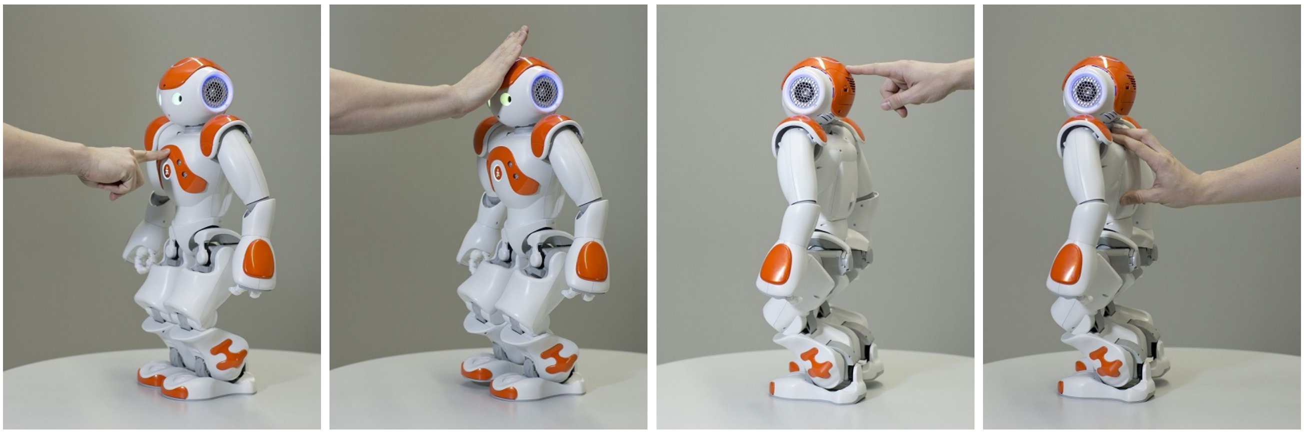
In the following experiments we use DMD, SDMD and classical interpolation schemes to learn a model of a robot's walking gait. Furthermore, we evaluate and compare the quality of each of these models. The best model is then used to detect and estimate external perturbations during a human robot interaction task.

A. *Prediction Quality*

For the evaluation of DMD and SDMD we make use of a walking dataset recorded on a Nao robot. The longitudinal CoM was recorded for a walking gait with five different equidistant step lengths between $-4cm$ and $+4cm$. The data is recorded with $100Hz$ for four seconds. Both the DMD and SDMD algorithms were applied on this dataset, resulting in four DMD modes. Given the learned models, the goal is to generate new sensor values for step lengths that were not recorded during training. Figure 6 shows the five training samples of the longitudinal CoM and the generated model which was interpolated with an interval of $0.01cm$. To evaluate the precision of the generated data we additionally recorded test samples with step lengths in an interval of $1cm$ and measured their mean relative error MRE w.r.t. the corresponding generated data. We also compared the results with a set of classical interpolation schemes. For the CoM, Figure 7 shows that DMD results in the highest accuracy among all methods. SDMD reduces the number of used modes to three and results in a slightly less accurate model. Since, we want to work with low cost sensors which may have significant noise, we additionally recorded the robot's longitudinal acceleration and applied the same data generation techniques as above. Again, the original DMD uses all extracted modes to predict new sensor values. However, these predictions are corrupted by the fact that some of these extracted modes mainly contain noise. As a result, the prediction performance of DMD deteriorates to about the same level as classical interpolation schemes. In contrast, SDMD concentrates on the three DMD modes that best approximate the sensor data. In this case, one mode was set to zero which obviously contained strong noise. However, because of its smaller MRE, we use the DMD model in conjunction with the CoM for the following experiments.

B. *Perturbation Detection*

In the following experiments we detect external perturbations while the robot performs a walking gait with a step length of $0.5cm$ for 35 seconds. During slowly walking the human perturbs the robot by touching and pushing it as shown in Figure 8. While Figure 8a, 8c show slight pushes, which just marginally disturb the walking gait. We also applied strong pushes as shown in Figure 8b, 8d. Especially, the strong push from the back as shown in Figure 8d significantly affected the robot‘s stability during walking.
We use the DMD model to generate the predicted sensor values for the current step length. During behavior execution the longitudinal CoM is measured with $100Hz$ and saved in a sliding window with 10 measurements. To estimate the current walking phase, we calculate the optimal warping path from this subsequence in the predicted data using SDTW. The resulting path is used as time dependent prediction of the longitudinal CoM for the currently measured values. Figure 9 shows the measured and predicted longitudinal CoM for the external perturbations a-d as shown in Figure 8. A perturbation is detected when the measured longitudinal CoM is outside the variance of the predicted one.



(a) Slight Push Front (b) Strong Push Front (c) Slight Push Back (d) Strong Push Back

Fig. 8: The human touches and pushes the robot during the execution of a walking behavior. The estimated perturbation values differ in strength and direction and reflect the amount of force applied on the robot.

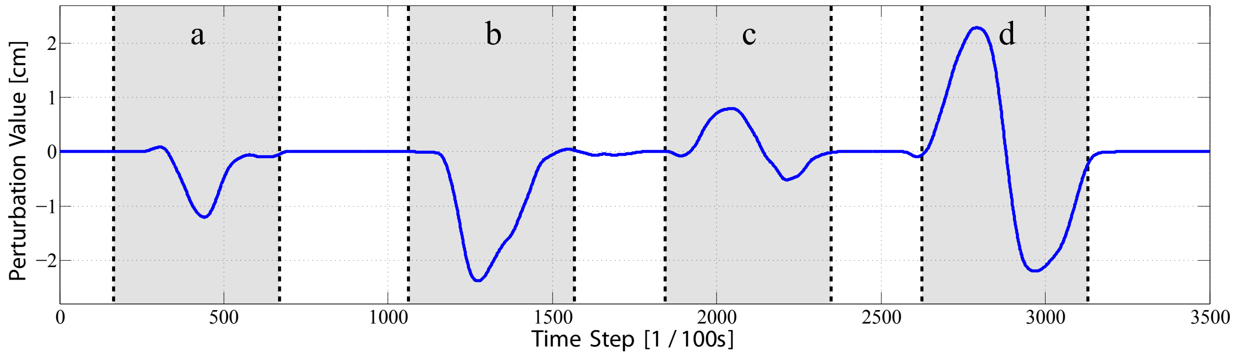


Fig. 11: The perturbation value for the external perturbation a-d is the difference between the predicted parameter with minimal costs and the current behavior parameter. Perturbation d produces a large oscillation which is dampened over time.

C. Perturbation Estimation

If a perturbation is detected, we have to find another behavior parameter and its corresponding sensor evolution, which has minimal mapping costs C for the SDTW. As mentioned before, there are several different approaches for this minimization problem. However, to prove the correctness of our approach we compute C for each step length of the learned DMD model. Figure 10 shows the overall costs C for all possible step lengths of our DMD model during the peeks of the external perturbations as shown in Figure 8a-d. Obviously, pushes from the front produces minimal costs for negative step lengths while pushes from the back lead to positive step lengths. As a result, the parameters with minimal costs can be seen as behavior parameters which counteract the external perturbation. Finally, the perturbation value is calculated from the difference of the current step length of $0.5cm$ and the predicted step length. Since the behavior parameter is specified in cm the measuring unit for the perturbation value is also in cm. The perturbation value for the complete behavior execution is shown in Figure 11.

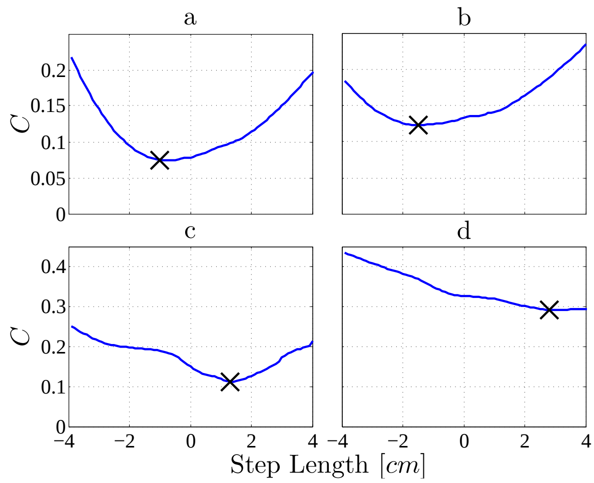


Fig. 10: The overall costs C for all possible parameters during the peaks of the external perturbations a-d. The step length which produces the minimal costs (black crosses) is the predicted step length which is used to calculate the perturbation value.

Fig. 12: During a cooperative transportation task, a humanoid robot continuously estimates the amount and direction of external perturbations in order to react to human guidance.

D. Reaction

Our approach can be used in scenarios where a robot has to detect and react to external perturbations in order to fulfill a specified task. As shown in Figure 12 it can be used to follow the human guidance in a cooperative transportation task as investigated in our previous publication [22]. A video of the functionality can be found under this link[1]. Furthermore, our approach can be used to implement collision detection and safety constrains. In addition, the method can also be used to measure the weight of a carried object during a manipulation task. In general, behavior specific filtering allows for a variety of close contact interactions with the environment.

V. Conclusion

In this paper, we presented a new approach for learning behavior-specific filters that can be used to accurately identify human physical influences on a robot. The approach uses DTW and DMD/SDMD in order to (1) detect an external perturbation, and (2) to quantify the amount of external perturbations. The generated *perturbation value* can then be used by a robot to adapt its movements to the applied forces or interpret a human command such as "walk backwards". In our experiments we showed that the learned perturbation filter can be used to accurately estimate touch information from noisy, low-cost sensors. Our approach produces a continuous perturbation value that can be used to detect even subtle physical interactions with a human partner. Since we are using a data-driven approach, no thresholds need to be defined by the user. At the core of our approach lies Dynamic Mode Decomposition, which so far has mostly been used in other fields of science, particularly fluid mechanics. We conclude that DMD is also a highly promising method for robotics. For future work, we hope to hierarchically combine several filters in a mixture-of-experts approach, in order to generalize perturbation estimation to new, untrained behaviors. We are currently also investigating the application of this approach to industry-grade robots and collaborative assembly tasks.

[1] http://youtu.be/48y0hEix2fY

References

[1] P. Rouanet, P.-Y. Oudeyer, F. Danieau, and D. Filliat, "The Impact of Human-Robot Interfaces on the Learning of Visual Objects," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 525–541, Apr. 2013.

[2] A. Albu-Schäffer, O. Eiberger, M. Fuchs, M. Grebenstein, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, S. Wolf, C. Borst, and G. Hirzinger, "Anthropomorphic soft robotics from torque control to variable intrinsic compliance," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Springer Berlin Heidelberg, 2011, vol. 70, pp. 185–207.

[3] S. Haddadin, *Towards Safe Robots - Approaching Asimov's 1st Law*. Springer, 2014.

[4] D. Lee and C. Ott, "Incremental kinesthetic teaching of motion primitives using the motion refinement tube," *Autonomous Robots*, vol. 31, no. 2-3, pp. 115–131, 2011.

[5] H. Wang and K. Kosuge, "Control of a robot dancer for enhancing haptic human-robot interaction in waltz," *IEEE Trans. Haptics*, vol. 5, no. 3, pp. 264–273, Jan. 2012.

[6] H. Ben Amor, E. Berger, D. Vogt, and B. Jung, "Kinesthetic bootstrapping: Teaching motor skills to humanoid robots through physical interaction," in *KI 2009: Advances in Artificial Intelligence*. Springer Berlin Heidelberg, 2009, pp. 492–499.

[7] S. Calinon, *Robot programming by demonstration: A probabilistic approach*. EPFL Press, 2009.

[8] J. Kober and J. Peters, "Policy search for motor primitives in robotics," *Machine Learning*, vol. 84, no. 1, pp. 171–203, 2011.

[9] S. Ikemoto, H. Ben Amor, T. Minato, B. Jung, and H. Ishiguro, "Physical human-robot interaction: Mutual learning and adaptation," *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 24–35, 2012.

[10] J. Stückler and S. Behnke, "Following human guidance to cooperatively carry a large object," in *Humanoids'11*, 2011, pp. 218–223.

[11] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, "Cooperative works by a human and a humanoid robot," in *Proceedings. ICRA '03. IEEE International Conference on Robotics and Automation 2003.*, vol. 3, 2003, pp. 2985–2991 vol.3.

[12] A. Bussy, P. Gergondet, A. Kheddar, F. Keith, and A. Crosnier, "Proactive behavior of a humanoid robot in a haptic transportation task with a human partner," in *RO-MAN, 2012 IEEE*. IEEE, 2012, pp. 962–967.

[13] M. Lawitzky, A. Mortl, and S. Hirche, "Load sharing in human-robot cooperative manipulation," in *RO-MAN, 2010 IEEE*, 2010, pp. 185–191.

[14] H. Sakoe, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, pp. 43–49, 1978.

[15] M. Müller, *Information Retrieval for Music and Motion*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

[16] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[17] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, pp. 5–28, 8 2010.

[18] C. W. Rowley, I. Igor Mez, I. Shervin Bagher, R. Philipp Schlatte, and D. S. Henningson, "Spectral analysis of nonlinear flows," *Journal of Fluid Mechanics*, vol. 641, no. -1, pp. 115–127, 2009.

[19] S. Bagheri, "Analysis and control of transitional shear flows using global modes," Ph.D. dissertation, KTH, Mechanics, 2010.

[20] K. Chen, J. Tu, and C. Rowley, "Variants of dynamic mode decomposition: Boundary condition, koopman, and fourier analyses," *Journal of Nonlinear Science*, vol. 22, no. 6, pp. 887–915, 2012.

[21] M. R. Jovanovi, P. J. Schmid, and J. W. Nichols, "Sparsity-promoting dynamic mode decomposition," *Physics of Fluids (1994-present)*, vol. 26, no. 2, 2014.

[22] E. Berger, D. Vogt, N. Haji-Ghassemi, B. Jung, and H. Ben Amor, "Inferring guidance information in cooperative human-robot tasks," in *Proceedings of the International Conference on Humanoid Robots (HUMANOIDS)*, 2013.