

MuTT: A Multimodal Trajectory Transformer for Robot Skills

Claudius Kienle¹, Benjamin Alt¹, Onur Celik², Philipp Becker², Darko Katic¹
Rainer Jäkel¹ and Gerhard Neumann²

Abstract—High-level robot skills represent an increasingly popular paradigm in robot programming. However, configuring the skills' parameters for a specific task remains a manual and time-consuming endeavor. Existing approaches for learning or optimizing these parameters often require numerous real-world executions or do not work in dynamic environments. To address these challenges, we propose Multimodal Trajectory Transformer (MuTT), a novel encoder-decoder transformer architecture designed to predict environment-aware executions of robot skills by integrating vision, trajectory, and robot skill parameters. Notably, we pioneer the fusion of vision and trajectory, introducing a novel trajectory projection. Furthermore, we illustrate MuTT's efficacy as a predictor when combined with a model-based robot skill optimizer. This approach facilitates the optimization of robot skill parameters for the current environment, without the need for real-world executions during optimization. Designed for compatibility with any representation of robot skills, MuTT demonstrates its versatility across three comprehensive experiments, showcasing superior performance across two different skill representations.

I. INTRODUCTION

The use of robots in industry depends to a large extent on the difficulty of programming the robot for the task to be solved. One programming paradigm that has become increasingly popular is programming with high-level robot skills [1]–[4]. While there exist plentiful skill representations, like Dynamic Movement Primitives (DMPs) [5]–[7] or classical force-controlled skills [8], all of them have in common that they have a set of parameters (e.g. goal pose, robot velocity) to configure them for the task at hand. Choosing the correct parameters can be a time-consuming and complex process, further complicated by the fact that the optimal parameterization of a skill is highly sensitive to both the robot and its environment. Prior work on learning [9] or optimizing task parameters [10] is limited by the need for hours of real-world robot executions or fails to generalize to environment changes.

In light of these challenges, we propose MuTT, an innovative encoder-decoder transformer for environment-aware predictions of robot skill executions. To the best of our knowledge, MuTT stands out as the first multimodal transformer that fuses trajectory and vision modalities, enabling trajectory predictions based on visual reasoning. A trajectory is a defined path in Joint or Cartesian space the robot follows over a specified period of time. Every point in this path may encompass additional information, such as the forces

This work was supported by the German Federal Ministry of Education and Research (grant 01MJ22003B).

¹ArtiMinds Robotics, Karlsruhe, Germany

²Autonomous Learning Robots, KIT, Germany

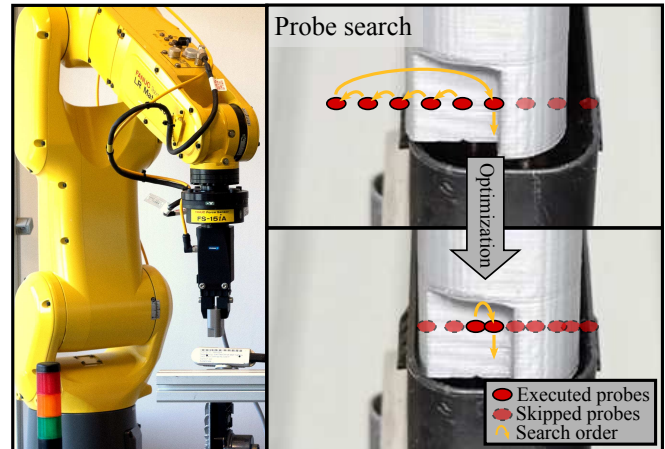


Fig. 1. MuTT is used in the SPI parameter optimizer [10] to refine the initial search pattern (red dots, top). The optimization yields an improved search pattern (red dots, bottom), reducing the required probes from six to two to successfully locate the socket. This significantly decreases the cycle time of the task. As an environment-aware model of robot skills, MuTT enables the optimization of skill parameters for the current environment, alleviating the need for real-world executions during optimization.

and torques experienced, the task success, or the expected reward for Reinforcement Learning (RL) agents. Moreover, MuTT can be used as a predictor of real-world executions in robot skill optimizers [10], enabling the optimization of robot skills for the current environment without requiring real-world executions during optimization.

With this, our contribution is threefold:

- 1) MuTT: A transformer that can easily be finetuned to predict environment-aware trajectories of robot skills. This includes a novel projection for trajectories that retains pose and force information at high temporal resolution, enabling precise prediction of high-frequency features such as sharp peaks.
- 2) Integration of MuTT as a predictive model into an optimizer for skill-based robot programs.
- 3) Evaluation of the prediction and optimization capabilities of MuTT on two real-world industrial tasks and one ManiSkill2 environment [11], using two different skill frameworks [6], [8].

II. RELATED WORK

A. Multimodal Transformer

Transformers were applied to many multimodal application domains in the past, typically combining the modalities vision and text [12]–[15].

Recent approaches are delving into the integration of text and audio [16], as well as audio and vision [17], [18]. Nevertheless, these approaches consistently operate under the assumption of modal synchronization. For instance, in [17], video processing involves aligned audio tracks and video frames. This exploitation of modality synchronization enables the consolidation of synchronized segments from both streams into a singular token. Gemini 1.5 [19] is a notable exception as being one of the first to combine audio and vision without requiring modal synchronization. However, as the architecture of Gemini 1.5 is not publicly available, there is no information about its implementation.

In the robotics domain, [20], [21] have advanced the training of a multimodal transformer architecture for closed-loop robot control. Similarly, [22] integrate language instructions and images, although their model focuses on predicting text-based instructions rather than directly predicting raw actions.

While numerous approaches exist that combine vision, audio, or text [14]–[25], there currently lacks a multimodal transformer that effectively merges trajectory and vision modalities. Despite the potential similarity between the audio and trajectory modalities due to both being continuous-valued time-series, trajectories require a more precise resolution which hinders the use of discretization, often done for the projection of audio data [16]. Additionally, the trajectory length is crucial, which for audio is often not considered [24], [16].

B. Forward Dynamics Models

Previous approaches [26]–[28] develop a forward dynamics model to predict the state of the robot or environment given a robot action. The learned model is then utilized to determine the action that yields the optimal state, using either sample-based methods or gradient-based optimization. For instance, [28] propose a forward model that, given a text-based description of a task, predicts a video depicting the robot performing the task. From the video, they derive the robot’s action sequence. However, most forward dynamics models are tailored to specific tasks, such as predicting the cable tip in [26]. In contrast, MuTT receives and predicts trajectories, allowing for its application across a wide range of tasks and robot skill representations. Furthermore, MuTT is the first forward model to incorporate an environment image, enabling it to predict the robot state conditioned on the robot’s current environment.

C. Robot Skill Optimization

Alt et al. [10] propose a model-based optimizer for skill-based robot programs that learns a differentiable model of a sequence of robot skills. This “shadow program” predicts the expected robot execution given the skills’ parameters, enabling the optimization of robot skill parameters via gradient descent without additional executions on the real robot. Alternative approaches optimize skill parameters with Reinforcement Learning [29], Bayesian Optimization [1], or Evolutionary Algorithms [2] by repeatedly executing the

program with varying parameterizations on a real robot or in a simulated environment for evaluation.

Most of the proposals for robot skill optimization [1], [2] require executing candidate parameterizations on the real robot to assess them for the current environment. This significantly slows down the optimization and necessitates a real robot during optimization. Additionally, it poses the risk of potentially damaging the robot or its surroundings, as the chosen program parameterizations are solely determined by the optimizer, potentially leading to unforeseen consequences during execution. A simulated environment can be beneficial, but introduces the Sim2Real Gap [30], which complicates the transfer of learned skills to the real robot. While [10] avoids this by learning a surrogate model that predicts the robot execution, this predictor has no sense of the current environment the robot operates in, restricting its application to static environments.

Our contribution incorporates an environmental signal and enables the prediction of accurate robot trajectories for the current environment. With this, it is not tailored to one skill representation or one specific optimization algorithm, but can be integrated as an environment-aware predictor in gradient-based [10], [31], gradient-free [32]–[34] and model-based [35], [36] optimization strategies. This eliminates the need to perform real robot executions during optimization of the robot skill.

III. MULTIMODAL TRAJECTORY TRANSFORMER

MuTT is an encoder-decoder transformer [37] that predicts the execution of robot skills for the current environment. The encoder fuses the different modalities into one hidden representation, based on which the decoder predicts the environment-aware trajectory autoregressively. The model architecture is depicted in Figure 2.

A. Encoder

The encoder of MuTT follows the *single-stream* approach [38] and uses a minimal embedding pipeline to project the trajectories, environment images and skill parameters into a token sequence. As we are the first to consider the trajectory modality, we propose a novel trajectory embedding further detailed in III-C. An image of the environment is split into 16x16 patches and linearly projected into tokens as in [39]. We interpolate over the position encoding for the image tokens to enable the processing of differently sized images [39]. The robot skill parameters are embedded by a single linear layer. All embedded tokens are coded with modality-specific positional encoding and their respective token type before they are passed through the encoder transformer to obtain a fused hidden representation.

B. Decoder

The MuTT decoder predicts the trajectory segment-wise in an autoregressive manner, while feeding the already predicted trajectory segments back into the decoder alongside the encoder’s hidden representation, as in [16]. We decide on a segment-wise prediction and not a prediction of individual

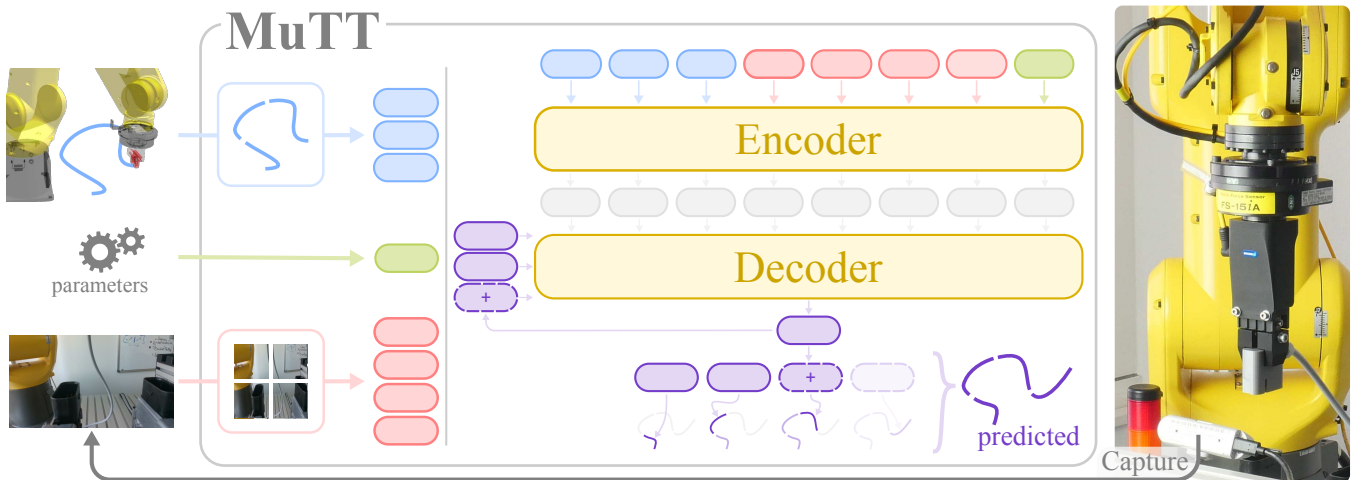


Fig. 2. Multimodal Trajectory Transformer architecture: modality specific embedding of the simulated trajectory (blue), skill parameters (green) and environment image (red) into tokens, which are concatenated to one token sequence. All tokens are coded with modality specific positional and token-type encoding and passed through an encoder transformer. The decoder predicts the real-world trajectory (purple) in an autoregressive manner.

points to utilize the hidden state effectively and reduce the inference duration and number of autoregressive iterations. A segment size of 20 points proved to be the best, as it did not reduce the prediction accuracy, but significantly reduced the memory requirements and inference time of MuTT. The decoder should predict the trajectory that matches the execution of the robot skill in a particular environment. Both the encoder and decoder have a hidden size of 768, the encoder consists of 12 attention layers, while the decoder consists of 6 attention layers. Every attention layer uses 12 attention heads.

We chose an encoder-decoder architecture to enable the prediction of variable-length trajectories and to support a variable number of image patches. An encoder-only or decoder-only transformer would require padding input or output to a fixed length, artificially enlarging the data that must be processed. In Experiment IV-A.1, we compare the encoder-decoder transformer with an encoder-only transformer. The encoder-only transformer predicted trajectories whose length deviated significantly more from the execution than the predicted trajectories of the encoder-decoder transformer. Moreover, a fixed input or prediction size limits the ability to transfer a trained MuTT instance to a new task that would require processing longer trajectories.

C. Trajectory Projection

Trajectories are continuous-valued time-series with a fixed temporal sampling interval. In comparison to many other modalities, trajectories need a high prediction accuracy and also require predicting the exact length of the trajectory. We normalize trajectories by the dataset mean and standard deviation. Since the trajectories in a batch can be of different length, we pad them to the length of the longest in the batch by replicating the last point. Additionally, the batch is padded to a multiple of 20 points. Padding a trajectory involves annotating every point with a binary flag that indicates whether the point serves as padding or not. We

split the trajectories along the time dimension into equal sized segments of 20 points. The trajectory embedding follows the minimal embedding pattern used in [39] and projects the segments with a single linear layer in a token sequence. In contrast to related approaches working with actions [20], [40], the trajectories are not discretized before projection, which would reduce their resolution. We also do not apply any smoothing or additional pre- or postprocessing to the trajectories. Furthermore, we set the attention mask value for a token to zero if the entire segment consists of padding points only and otherwise to one. The position of every token is encoded with absolute position encoding [41]. Relative positional encoding or interpolation would violate the invariant that there is a fixed time interval between two successive data points in a trajectory.

D. Pre-training

MuTT must be trained to predict the real execution of a robot skill. Since collecting a real-world dataset of robot executions is costly, we aim to minimize the data required to train the model effectively. Consequently, we compared initializing the transformer encoder with weights from related models [15], [16], [24], [39]. Nearly all initialization variants improved the performance of the finetuned MuTT significantly, except for [24]. The initialization with Vision Transformer (ViT) [39] resulted in the best performance. We use the weights of SpeechT5 [16] for the decoder transformer due the similarity between the audio and trajectory modality, which stems from their shared characteristic as continuous-valued time series.

IV. EXPERIMENTS

We demonstrate the capability of MuTT to accurately predict the real-world execution of robot skills by evaluating it in three different manipulation scenarios, using two different robot skill representations. This showcases MuTT's design to seamlessly work with various robot skill

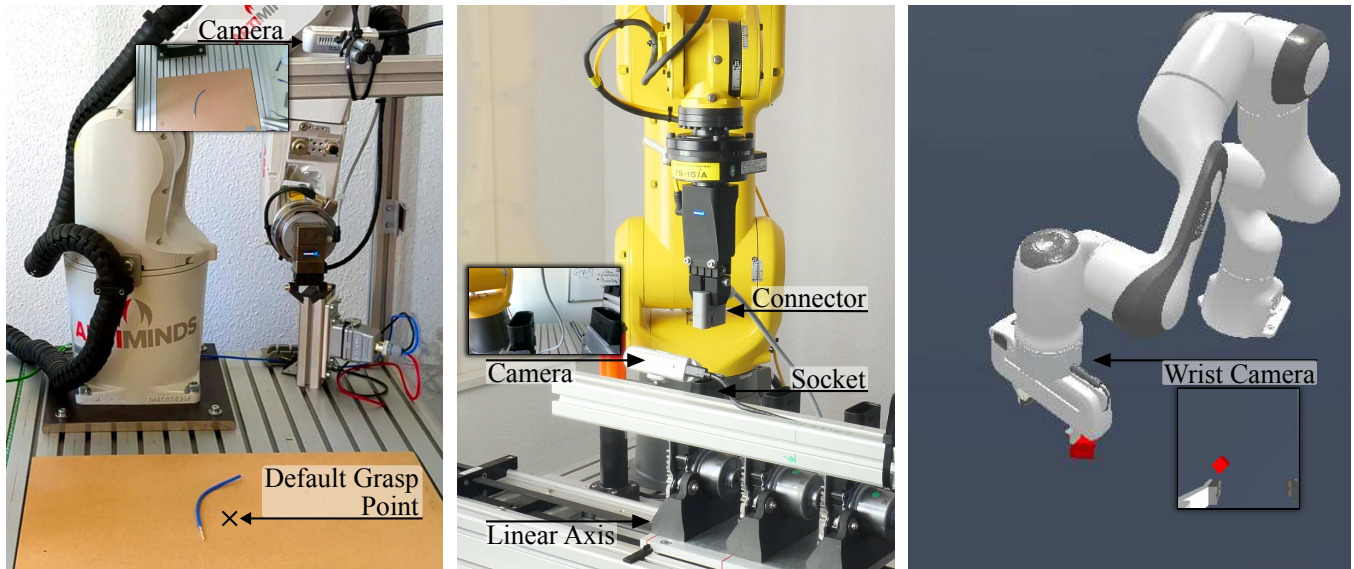


Fig. 3. Evaluation scenarios: Real-world grasping of deformable cables (Experiment IV-A.1, left), real-world force-controlled plug insertion under uncertainty (Experiment IV-A.2, middle), and simulated grasping in the ManiSkill2 benchmark (Experiment IV-B.1, right).

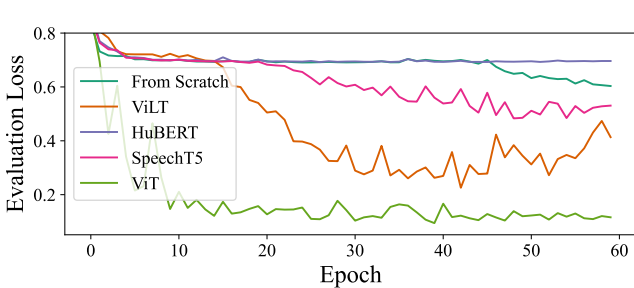


Fig. 4. Comparison of training MuTT on the dataset of Experiment IV-A.1 with different initial weights from related applications [15], [16], [24], [39]. Initialization with ViT [39] resulted in the best evaluation performance.

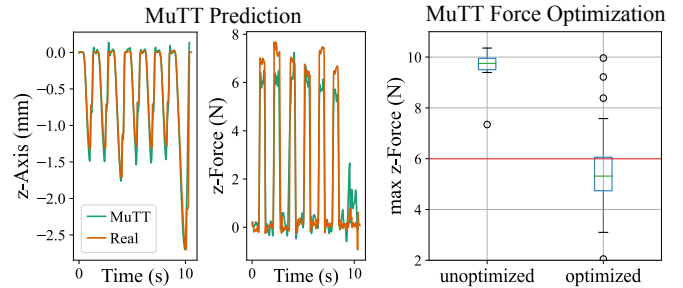


Fig. 5. Experiment IV-A.2: End-effector pose (along Z axis, left) and force (along Z axis, middle) prediction of MuTT for a probe skill. MuTT predicts forces accurately, enabling the optimization with SPI [10] of the robot skill to adhere to the user-defined force limit of 6 N, while the unoptimized skill greatly exceeds that limit (right).

representations, without necessitating any modifications to the model architecture or training algorithm. An overview of the experiments is shown in Figure 3. In all experiments, we embed MuTT as an environment-aware predictor of real robot-skill executions in existing frameworks [10], [11] to optimize the robot skills for the current environment.

A. Model-Based Optimization of Industrial Robot Skills

We use MuTT to optimize industrial robot skills [8] for grasping deformable cables and force-controlled plug insertion. In both experiments, MuTT receives a simulated trajectory of the skill, the skill’s parameters, and an unprocessed RGB environment image from a RealSense D435. Based on these inputs, the model must predict the real-world trajectory resulting from the execution of the skill. The trained MuTT instances are then used in the robot program parameter optimization framework SPI [10] to optimize the program for the current environment.

1) *Grasping of Deformable Cables:* This experiment is designed to clearly demonstrate that MuTT can be used as forward model in existing optimization frameworks, in

this case for the optimization of robot skill parameters with SPI [10]. In the experiment, the robot tries to grasp an industrial cable from a table with a grasp skill parameterized by a grasp pose. The cable’s position varies by up to 5 centimeters in each direction. MuTT is trained on 5,000 randomly selected grasp executions. We compare different initialization strategies for MuTT [15], [16], [24], [39], with ViT [39] yielding the best evaluation performance, as seen in Figure 4. MuTT predicts the execution of the grasp skill, including whether the cable was grasped successfully. Subsequently, we utilized MuTT to predict the real robot executions within the parameter optimizer SPI [10], optimizing the grasp pose for successful cable grasping. While the robot grasped the cable in 2 of 100 evaluation runs with initial parameterizations sampled from the region of feasible grasp poses, parameter optimization with MuTT and SPI led to grasping the cable in 67 of the same 100 evaluation runs.

This demonstrates MuTT’s capability to integrate the environment image and program parameters to accurately predict the likelihood of grasping in the current environment. With

TABLE I
EXPERIMENT IV-A.2: RESULTS

	Unoptimized	Optimized
Avg number of probes	14	3
Avg search duration (s)	25.3	7.5
Success probability	0.55	0.96
Avg exceeded forces by (N)	3.5	0.7

this, the grasp pose can be optimized until MuTT predicts a successful grasp of the cable.

2) *Force-Controlled Plug Insertion*: The second experiment is representative of many current applications of robots in industry and shows that MuTT is able to predict complex skill executions. In this experiment, the predictions not only include the end-effector pose, but also the forces and torques that occur at the end-effector during execution. Specifically, we study the insertion of an industrial connector into the matching socket with a force-controlled probe-search skill. In real-world industrial applications, the exact positioning of parts such as the socket is often subject to process noise. We simulate random deviations of the positioning of the socket by up to 1 cm with a linear axis. The robot searches for the socket along the linear axis. In this experiment, MuTT predicts the end-effector motion during search, which deviates from the planned motion due to environment interaction. This includes the unknown position of the socket, on which the robot’s search motion depends. Additionally, MuTT predicts the end-effector forces and torques during search, and a success token indicating whether the robot successfully plugged the connector into the socket. MuTT must adjust its prediction based on the socket’s position depicted in the environment image and the search pattern given by the skill parameters and simulated trajectory.

We train MuTT on 5,000 randomly parameterized robot skill executions. The trained MuTT instance accurately predicts the search motion of the end-effector with an average deviation to the real execution by 0.3 mm and an average force deviation of 0.5 N. The success of the search was predicted correctly with an F1 score of 0.99. Figure 5 depicts an exemplary prediction of MuTT alongside the real execution of that same robot skill.

We show that the accurate prediction of MuTT can be used in SPI to optimize the search skill, including the search pattern for a fast and successful search, as well as the contact forces during the search. In force-controlled skills, reaction delays consistently lead to the robot exceeding pre-set force limits. Consequently, optimizing the search skill to adhere to a user-defined force-threshold safeguards hardware (such as the grasped plug) from damage by exceeding the force limit.

Table I compares the execution of the initial robot skill and the optimized robot skill on the real robot. MuTT predicts the real execution accurately for the current deviation of the socket. This enabled SPI to optimize the robot skill parameters resulting in a 70 % faster search, doubling the success probability while adhering to the force limit 78 % more accurately. The maximal forces experienced during

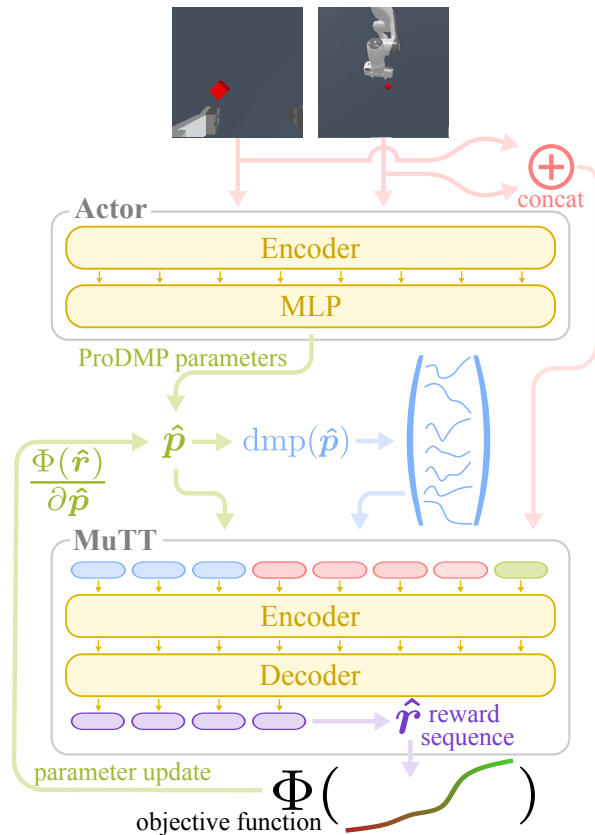


Fig. 6. Architecture of Experiment IV-B.1. The BC actor predicts ProDMP parameters \hat{p} that define a trajectory to pick up the red cube shown in the environment images. MuTT predicts the expected reward \hat{r} the agent would receive for every action in this trajectory. Φ computes the mean squared distance to the maximal reward the actor can receive for an action. Via gradient descent, the ProDMP parameters \hat{p} are optimized to minimize Φ and consequently maximize the predicted reward \hat{r} , resulting in improved ProDMP parameters \hat{p} .

search with unoptimized and optimized parameters can be seen in Figure 5. While using the unoptimized parameters resulted in exceeding the user-defined force threshold by about 4 N, the optimized programs kept to the force limit.

B. Model-Based Optimization of Probabilistic Dynamic Movement Primitive (ProDMP) Skills

In a second series of experiments, we apply MuTT to ProDMPs, demonstrating that its architecture is not limited to one specific robot skill representation. ProDMPs provide a representation capable of generating smooth trajectories from any initial state while capturing higher-order motion statistics. This experiment assesses MuTT’s prediction capabilities on the *PickCube-v0* environment of the ManiSkill2 Benchmark [11]. We focus solely on picking up the cube from the surface without evaluating whether the cube was lifted to the correct position. Here, MuTT’s predictions are used to optimize the trajectory suggested by an episodic RL agent, resulting in an increase of task success compared to the raw agent. For this, we learn a robot skill with an episodic Behaviour Cloning (BC) actor that predicts parameters \hat{p} for 7 ProDMPs [6], 6 of which define the end-effector pose (po-

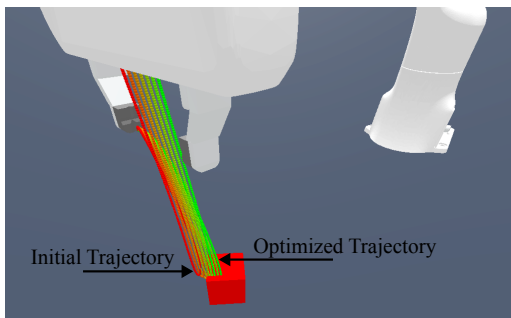


Fig. 7. Robot skill optimization in Experiment IV-B.1. Evolution of the trajectory generated from the ProDMP parameters throughout the optimization process. The robot would not have grasped the cube with the initial robot skill parameters (red). Over the course of the optimization, the trajectory is gradually aligned with the cube’s position. Ultimately, with the optimized ProDMP parameters, the robot successfully grasps the cube (green).

sition and rotation) and one the gripper configuration. MuTT receives the ProDMP parameters, the simulated trajectory generated by the ProDMPs, and concatenated images of two cameras as input. In contrast to the first two experiments, the real-world trajectory predicted by MuTT also contains the reward \hat{r} the agent would receive for every action in the given ProDMP trajectory. This underscores MuTT’s versatility, as the predicted trajectory is not limited to the robots motion, but can contain any sequence of data associated with the execution of the robot skill. Via gradient descent, the parameters \hat{p} are optimized to maximize the reward \hat{r} predicted by MuTT. The architecture is illustrated in Figure 6.

1) *ManiSkill2 Benchmark*: We train a BC actor [42] on a dataset \mathcal{D}_{demo} consisting of 1,000 optimal demonstrations and use the trained actor to generate an additional dataset \mathcal{D}_{im} consisting of 30,000 imperfect executions. Subsequently, MuTT is trained on \mathcal{D}_{im} to predict the expected reward for every action. Finally, we employ the trained MuTT instance to optimize the ProDMPs predicted by the BC actor. To achieve this, we compute a loss $\Phi(\hat{r})$ based on the reward \hat{r}

$$\Phi(\hat{r}) := \frac{1}{|\hat{r}|} \sum_{i=1}^{|\hat{r}|} (r_{max} - \hat{r}_i)^2$$

and update the ProDMP parameters \hat{p} via gradient descent. r_{max} is the maximal reward the actor can receive for one action, in this experiment set to 1. This aims to maximize the reward \hat{r} predicted by MuTT.

Table II compares the BC actor with and without optimization by MuTT to the offline episodic RL algorithm Advantage Weighted Regression (AWR) [43] trained on \mathcal{D}_{im} and to the state-action based offline RL algorithm Implicit Q-Learning (IQL) [44], trained on 66,000 state-action transitions generated from the \mathcal{D}_{im} episodic dataset. All algorithms are evaluated on the same 100 environments they have not seen during training.

Optimizing the ProDMPs predicted by the BC actor with MuTT increases the number of successful task executions by 6 %. Figure 7 displays how the optimization improves the end-effector trajectory leading to successful grasping of

TABLE II
EXPERIMENT IV-B.1: RESULTS

	Time steps	Dataset size	Success prob.
BC [42]	60k	1k	0.33
BC + MuTT (Ours)	7.3M	30k	0.39
AWR [43]	15M	30k	0.02
IQL [44]	9.3M	66k	0.35

the cube, while the robot does not grasp the cube with the trajectory predicted by the BC actor. MuTT notably excels the AWR algorithm that struggles to learn the task based on the same dataset \mathcal{D}_{im} MuTT was trained on, likely due to only 30 % of the samples in this dataset successfully solving the task. MuTT also dominates IQL, which was trained on a dataset more than twice as large. While it is difficult to compare episodic and state-action based algorithms, it demonstrates that MuTT outperforms state-action based algorithms that were trained for a comparable number of steps on a dataset of comparable size. Additionally, state-action based algorithms can adapt their prediction for every new state online, while MuTT predicts the entire trajectory given the initial state, without any online adaptation during execution.

V. CONCLUSION AND OUTLOOK

We introduce MuTT, a Multimodal Trajectory Transformer that accurately predicts robot skill executions aligned with the robot’s current environment by integrating vision, trajectory, and robot skill parameters. MuTT is representation-agnostic with respect to the robot skill and can be applied to near-arbitrary skill representations. To that end, we developed a novel trajectory projection that retains important properties such as the trajectories’ temporal resolution and length.

MuTT’s architecture allows for efficient training with relatively small datasets of random skill executions, making it a promising foundation model with quick adaptation to specific robot skills. Furthermore, MuTT’s compatibility with any robot skill optimizer enables the optimization of robot program parameters tailored to the current environment.

While MuTT as predictor for model-based robot skill optimization offers significant advantages over traditional optimization methods, such as not requiring real-world executions during optimization and precise trajectory prediction in dynamic environments, some challenges remain. The prediction accuracy of MuTT on out-of-distribution samples should be further analyzed. First tests showed that MuTT’s capability to generalize well to such samples is limited, indicating room for improvement through future research. Additionally, the speed of parameter optimization relies heavily on the inference speed of MuTT. While we engineered the model for fast and efficient inference, the optimization of robot skills currently takes about 20 to 40 seconds. Optimizing the model architecture to decrease inference duration and consequently enhance optimization speed is open for future research.

REFERENCES

- [1] L. Johannsmeier, M. Gerchow, and S. Haddadin, "A Framework for Robot Manipulation: Skill Formalism, Meta Learning and Adaptive Control," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 5844–5850, iSSN: 2577-087X.
- [2] J. A. Marvel, W. S. Newman, D. P. Gravel, G. Zhang, Jianjun Wang, and T. Fuhlbrigge, "Automated learning for parameter optimization of robotic assembly tasks utilizing genetic algorithms," in *2008 IEEE International Conference on Robotics and Biomimetics*, Feb. 2009, pp. 179–184.
- [3] U. Thomas, G. Hirzinger, B. Rumpe, C. Schulze, and A. Wortmann, "A new skill based robot programming language using UML/P Statecharts," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 461–466, iSSN: 1050-4729.
- [4] M. R. Pedersen, L. Nalpanitidis, R. S. Andersen, C. Schou, S. Bøgh, V. Krüger, and O. Madsen, "Robot skills for manufacturing: From concept to industrial deployment," *Robotics and Computer-Integrated Manufacturing*, vol. 37, pp. 282–291, Feb. 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0736584515000575>
- [5] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural computation*, vol. 25, no. 2, pp. 328–373, 2013, publisher: MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info
- [6] G. Li, Z. Jin, M. Volpp, F. Otto, R. Lioutikov, and G. Neumann, "ProDMP: A Unified Perspective on Dynamic and Probabilistic Movement Primitives," *IEEE Robotics and Automation Letters*, vol. 8, no. 4, pp. 2325–2332, 2023, publisher: IEEE. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/10050558>
- [7] S. Schaal, "Dynamic Movement Primitives -A Framework for Motor Control in Humans and Humanoid Robotics," in *Adaptive Motion of Animals and Machines*, H. Kimura, K. Tsuchiya, A. Ishiguro, and H. Witte, Eds. Tokyo: Springer, 2006, pp. 261–280. [Online]. Available: <https://doi.org/10.1007/4-431-31381-8-23>
- [8] H. Bruyninckx and J. De Schutter, "Specification of force-controlled actions in the "task frame formalism"-a synthesis," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 581–589, Aug. 1996, conference Name: IEEE Transactions on Robotics and Automation. [Online]. Available: <https://ieeexplore.ieee.org/document/508440>
- [9] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation Learning: A Survey of Learning Methods," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–35, Mar. 2018. [Online]. Available: <https://dl.acm.org/doi/10.1145/3054912>
- [10] B. Alt, D. Katic, R. Jäkel, A. K. Bozcuoglu, and M. Beetz, "Robot Program Parameter Inference via Differentiable Shadow Program Inversion," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, May 2021, pp. 4672–4678, iSSN: 2577-087X.
- [11] J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Mu, et al., "ManiSkill2: A Unified Benchmark for Generalizable Manipulation Skills," Feb. 2023, arXiv:2302.04659 [cs]. [Online]. Available: <http://arxiv.org/abs/2302.04659>
- [12] T. Le, H. T. Nguyen, and M. L. Nguyen, "Vision And Text Transformer For Predicting Answerability On Visual Question Answering," in *2021 IEEE International Conference on Image Processing (ICIP)*, Sept. 2021, pp. 934–938, iSSN: 2381-8549.
- [13] J. Wu, Y. Peng, S. Zhang, W. Qi, and J. Zhang, "Masked Vision-Language Transformers for Scene Text Recognition," Nov. 2022, arXiv:2211.04785 [cs]. [Online]. Available: <http://arxiv.org/abs/2211.04785>
- [14] Y. Khare, V. Bagal, M. Mathew, A. Devi, U. D. Priyakumar, and C. Jawahar, "MMBERT: Multimodal BERT Pretraining for Improved Medical VQA," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, Apr. 2021, pp. 1033–1036, iSSN: 1945-8452.
- [15] W. Kim, B. Son, and I. Kim, "Vilt: Vision-and-language transformer without convolution or region supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5583–5594.
- [16] J. Ao, R. Wang, L. Zhou, C. Wang, S. Ren, Y. Wu, et al., "SpeechT5: Unified-Modal Encoder-Decoder Pre-Training for Spoken Language Processing," May 2022, arXiv:2110.07205 [cs, eess]. [Online]. Available: <http://arxiv.org/abs/2110.07205>
- [17] B. Shi, W.-N. Hsu, K. Lakhota, and A. Mohamed, "Learning Audio-Visual Speech Representation by Masked Multimodal Cluster Prediction," Mar. 2022, arXiv:2201.02184 [cs, eess]. [Online]. Available: <http://arxiv.org/abs/2201.02184>
- [18] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "VisualBERT: A Simple and Performant Baseline for Vision and Language," Aug. 2019, arXiv:1908.03557 [cs]. [Online]. Available: <http://arxiv.org/abs/1908.03557>
- [19] M. Reid, N. Savinov, D. Teplyashin, D. Lepikhin, T. Lillicrap, J.-b. Alayrac, et al., "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," Mar. 2024, arXiv:2403.05530 [cs]. [Online]. Available: <http://arxiv.org/abs/2403.05530>
- [20] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Chormanski, et al., "RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control," July 2023, arXiv:2307.15818 [cs]. [Online]. Available: <http://arxiv.org/abs/2307.15818>
- [21] D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, et al., "Octo: An Open-Source Generalist Robot Policy."
- [22] D. Driess, F. Xia, M. S. M. Sajjadi, K. Lynch, A. Chowdhery, B. Ichter, et al., "PaLM-E: An Embodied Multimodal Language Model," Mar. 2023, arXiv:2303.03378 [cs]. [Online]. Available: <http://arxiv.org/abs/2303.03378>
- [23] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, et al., "Learning Transferable Visual Models From Natural Language Supervision," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, July 2021, pp. 8748–8763, iSSN: 2640-3498. [Online]. Available: <https://proceedings.mlr.press/v139/radford21a.html>
- [24] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhota, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021, publisher: IEEE.
- [25] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, et al., "RT-1: Robotics Transformer for Real-World Control at Scale," Aug. 2023, arXiv:2212.06817 [cs]. [Online]. Available: <http://arxiv.org/abs/2212.06817>
- [26] V. Lim, H. Huang, L. Y. Chen, J. Wang, J. Ichnowski, D. Seita, et al., "Real2Sim2Real: Self-Supervised Learning of Physical Single-Step Dynamic Actions for Planar Robot Casting," in *2022 International Conference on Robotics and Automation (ICRA)*, May 2022, pp. 8282–8289. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9811651>
- [27] B. Sukhija, N. Köhler, M. Zamora, S. Zimmermann, S. Curi, A. Krause, and S. Coros, "Gradient-Based Trajectory Optimization With Learned Dynamics," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 1011–1018.
- [28] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, et al., "Learning Universal Policies via Text-Guided Video Generation," *Advances in Neural Information Processing Systems*, vol. 36, pp. 9156–9172, Dec. 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/1d5b9233ad716a43be5c0d3023cb82d0-Abstract-Conference.html
- [29] T. Zhang, C. Yuan, and Y. Zou, "Online Optimization Method of Controller Parameters for Robot Constant Force Grinding Based on Deep Reinforcement Learning Rainbow," *Journal of Intelligent & Robotic Systems*, vol. 105, no. 4, p. 85, Aug. 2022. [Online]. Available: <https://doi.org/10.1007/s10846-022-01688-z>
- [30] S. Höfer, K. Bekris, A. Handa, J. C. Gamboa, M. Mozifian, F. Golemo, et al., "Sim2Real in Robotics and Automation: Applications and Challenges," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 2, pp. 398–400, Apr. 2021, conference Name: IEEE Transactions on Automation Science and Engineering.
- [31] B. Alt, D. Katic, R. Jäkel, and M. Beetz, "Heuristic-free Optimization of Force-Controlled Robot Search Strategies in Stochastic Environments," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2022, pp. 8887–8893, iSSN: 2153-0866. [Online]. Available: <https://ieeexplore.ieee.org/document/9982093>
- [32] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, Nov. 1995, pp. 1942–1948 vol.4.
- [33] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the Human Out of the Loop: A Review of Bayesian Optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016, conference Name: Proceedings of the IEEE.

- [34] T. Bäck and H.-P. Schwefel, "An overview of evolutionary algorithms for parameter optimization," *Evol. Comput.*, vol. 1, no. 1, pp. 1–23, Mar. 1993. [Online]. Available: <https://dl.acm.org/doi/10.1162/evco.1993.1.1.1>
- [35] F. Berkenkamp, A. Krause, and A. P. Schoellig, "Bayesian Optimization with Safety Constraints: Safe and Automatic Parameter Tuning in Robotics," *arXiv:1602.04450 [cs]*, Apr. 2020. [Online]. Available: <http://arxiv.org/abs/1602.04450>
- [36] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "Bayesian optimization for learning gaits under uncertainty," *Ann Math Artif Intell*, vol. 76, no. 1, pp. 5–23, Feb. 2016. [Online]. Available: <https://doi.org/10.1007/s10472-015-9463-9>
- [37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, *et al.*, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html
- [38] E. Bugliarelli, R. Cotterell, N. Okazaki, and D. Elliott, "Multimodal pretraining unmasked: A meta-analysis and a unified framework of vision-and-language BERTs," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 978–994, 2021, publisher: MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info [Online]. Available: <https://direct.mit.edu/tacl/article-abstract/doi/10.1162/tacl.a.00408/107279>
- [39] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," June 2021, arXiv:2010.11929 [cs]. [Online]. Available: <http://arxiv.org/abs/2010.11929>
- [40] M. Janner, Q. Li, and S. Levine, "Offline Reinforcement Learning as One Big Sequence Modeling Problem," Nov. 2021, arXiv:2106.02039 [cs]. [Online]. Available: <http://arxiv.org/abs/2106.02039>
- [41] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *International conference on machine learning*. PMLR, 2017, pp. 1243–1252. [Online]. Available: <http://proceedings.mlr.press/v70/gehring17a.html?ref=https://githubhelp.com>
- [42] S. Karamcheti, S. Nair, A. S. Chen, T. Kollar, C. Finn, D. Sadigh, and P. Liang, "Language-Driven Representation Learning for Robotics," Feb. 2023, arXiv:2302.12766 [cs]. [Online]. Available: <http://arxiv.org/abs/2302.12766>
- [43] D. Wierstra, T. Schaul, J. Peters, and J. Schmidhuber, "Episodic Reinforcement Learning by Logistic Reward-Weighted Regression," in *Artificial Neural Networks - ICANN 2008*, V. Kůrková, R. Neruda, and J. Koutník, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, vol. 5163, pp. 407–416, iSSN: 0302-9743, 1611-3349 Series Title: Lecture Notes in Computer Science. [Online]. Available: http://link.springer.com/10.1007/978-3-540-87536-9_42
- [44] I. Kostrikov, A. Nair, and S. Levine, "Offline Reinforcement Learning with Implicit Q-Learning," Oct. 2021, issue: arXiv:2110.06169 arXiv:2110.06169 [cs]. [Online]. Available: <http://arxiv.org/abs/2110.06169>