УДК 007.51
DOI: 10.17586/0021-3454-2024-67-6-500-510

MODELING PATTERNS OF SENSORY-MOTOR SKILLS FOR PROGRAMMING ROBOTS IN CONTACT MANIPULATION TASKS

Waddah Ali, S. A. Kolyubin*

ITMO University, St. Petersburg, Russia
** s.kolyubin@itmo.ru*

Abstract. Learning from demonstration approach is gaining interest for programming robot sensory-motor skills. At the same time, most of the works are addressing manipulation scenarios with position-based control, while various application domains and work in dynamic environment require safe and stable physical interaction where assessing proper force/torque profile along motion is crucial. This study is aimed at developing experiment planning and data collection and processing procedure for training robot behavior priors for dynamic interaction tasks. We fuse motion capture and force-torque sensory data within robot-out-of-loop setting to train Gaussian Mixture Model/Gaussian Mixture Regression (GMM/GMR) model as a reference motion generator that takes time and material label as inputs and outputs predicted end-effector's pose, twist, and interaction wrench vectors. For the case-study we considered experiment setting of cutting three different materials like penoplex, cork, and PVC resulting in 120 demonstrations in total (40 for each material). Algorithms for data processing, GMM/GMR model training and verification have been introduced. We achieved RMSEs of 7.12 and 10.69 % for twist and pose predictions respectively and RMSE of 14.33 % for power estimates as a metric to illustrate how accurate twist-wrench correspondences have been captured by our model, which is important for interaction tasks.

Keywords: *learning from demonstration, robot skill transfer, contact manipulation, interaction dynamics, motion capture*

For citation: Waddah Ali, Kolyubin S. A. Modeling patterns of sensory-motor skills for programming robots in contact manipulation tasks. *Journal of Instrument Engineering*. 2024. Vol. 67, N 6. P. 500–510 (in English). DOI: 10.17586/0021-3454-2024-67-6-500-510.

МОДЕЛИРОВАНИЕ ШАБЛОНОВ СЕНСОРНО-МОТОРНЫХ НАВЫКОВ ДЛЯ ПРОГРАММИРОВАНИЯ РОБОТОВ В ЗАДАЧАХ КОНТАКТНОГО МАНИПУЛИРОВАНИЯ

Али Ваддах, С. А. Колюбин*

Университет ИТМО, Санкт-Петербург, Россия
** s.kolyubin@itmo.ru*

Аннотация. Подход к обучению на основе демонстрации привлекает все больше внимания при программировании сенсорно-моторных навыков роботов. В то же время большинство работ сосредоточено на сценариях с управлением по положению, тогда как различные прикладные области и работа в динамической среде требуют безопасного и устойчивого физического взаимодействия, где критически важно оценивать соответствующий профиль силы/момента контакта вдоль траектории. Разработана методика планирования экспериментов и сбора и обработки данных для обучения моделей, кодирующих сенсорно-моторные навыки динамического взаимодействия манипулятора с окружением. Для этих целей комплексируются данные, поступающие от системы оптического захвата движения и силомоментного датчика, измеряемые при выполнении человеком последовательности действий. Рассмотрен пример резки скальпелем различных материалов по заданным траекториям. В качестве генератора эталонного движения используется регрессионная модель на основе смеси гауссиан (GMM/GMR), на вход которой поступают метки времени и материала, а на выходе выводятся предсказанные значения векторов пространственного положения, скоростей и сил и моментов контакта инструмента. Проведено 120 экспериментов с тремя различными материалами (пеноплекс, пробка и ПВХ) — по 40 на каждый материал. Представлены алгоритмы для обработки данных, результаты обучения модели и ее верификации. Для предсказаний скорости и положения инструмента получены значения среднеквадратического отклонения соответственно 7,12 и 10,69 %, а также 14,33 % — для мощности как метрики точности соответствия профиля сил и моментов контакта вдоль движения.

Ключевые слова: *обучение на основе демонстрации, передача сенсорно-моторных навыков, контактная манипуляция, захват движения, GMM/GMR модели*

Ссылка для цитирования: *Ваддах Али, Колюбин С. А. Моделирование шаблонов сенсорно-моторных навыков для программирования роботов в задачах контактного манипулирования* // Изв. вузов. Приборостроение. 2024. Т. 67, № 6. С. 500–510. DOI: 10.17586/0021-3454-2024-67-6-500-510.

Introduction. Learning from demonstration (LfD) is a promising approach to transfer safe and dexterous manipulation skills from human to robots, which is of even higher importance for contact-rich tasks, where coordination between the applied interaction forces/torques and manipulation trajectories is essential. This work is aimed at developing experiment planning and data collection and processing procedure for training robot behavior priors for such tasks.

A recent comprehensive review on transfer learning in robotics, which includes approaches taxonomy, trends and challenges description as well as analysis of more than 150 papers is presented in [1].

Recording and processing data from demonstrations experiments is the first step to encode prior knowledge on human sensory-motor skills that will be used later as behavior priors for accelerating the LfD process [2, 3]. There are many recent works on that subject focusing mostly on pick-and-place tasks, where kinesthetic data limited to position recordings and robot-in-the-loop (teleoperated or hand-guided in admittance control mode) demos are good enough.

In [4], a novel approach to robot learning from human physical feedback is introduced. This method characterizes human skills and tasks by breaking them down into object-centric sub-tasks and interpreting physical interventions from human in relation to specific objects. The task is to adjust nominal behavior priors from corrective movements (perturbations) initiated by human, therefore unlike our approach there were no complete movement skill demonstrations recorded and the task was limited to imitation of trajectories, while the interaction wrench was not considered.

In [5], a hybrid learning and optimization framework for mobile manipulators for complex and physically interactive tasks was proposed. The framework exploits an admittance-type physical interface to obtain intuitive and simplified human demonstrations and Gaussian Mixture Model (GMM)/Gaussian Mixture Regression (GMR) to encode and generate the learned task requirements in terms of position, twist, and wrench profiles. Unlike our proposed approach, this work adopted robot-in-the-loop scenario and uses the resulted behavior priors as constraints for optimization of Cartesian impedance controlled for a specified robotic platform which limits the applicability of the study to address a very specific task with special conditions.

Since we target interaction control scenarios over larger workspaces like material cutting, a novel robot-out-of-loop (ROOL) sensory setup enabling simultaneous motion capture (MoCap) and interaction force-torque (FT) data recording have been designed.

Proposed approach provides a number of advantages:

1) high-accuracy richer sensory data to capture end-effector's 6D pose as well as twist and wrench correspondences to encode by behavior priors' models;

2) better safety and more natural human movements during demonstrations, because there are no constraints due to kinematic singularities and robot inertia for physically operated arms or limited field-of-view and latency during teleoperation;

3) wider workspace not limited by robot reachability constraints.

To collect training datasets, we planned and conducted a series of cutting experiments with three different materials like penoplex, cork, and PVC resulting in 120 demonstrations in total (40 for each material).

GMM/GMR model have been employed as a manipulation skill encoder and behavior prior generator. It accepts time and material labels as inputs and outputs predicted end-effector's trajectory (end-effector's pose and twist) with interaction wrench aligned along the movement. Models have been trained using expectations maximization algorithm with log-likelihood as a loss function.

The rest of the paper is organized in the following way. At first, we formulate the problem of behavior model training from ROOL demonstration data as a general optimization task adapted for our case study. Next, we describe the experimental setup design including custom tooling and optimal sensory infrastructure configuration. After we describe the MoCap and FT data processing approaches followed by model structure and training algorithm explanation. Finally, we analyze obtained results to justify training data quality, convergence of the training process, and consistency of the learned manipulation skills with human demonstrations and conclude our work with discussion on future steps of how obtained results can be incorporated to robot interaction control systems.

Problem Statement. Fig. 1 illustrates the suggested approach for training behavior priors' generator from ROOL demonstrations and further use of this data within interaction control scenarios. So, the core problem here is generator's parameters training.

We introduced the GMM/GMR model [6–8] of components as a generator due to its adaptability in capturing intricate structures, handling nonlinear relationships between inputs and outputs, effectively estimating continuous variables from complex input-output mappings and the simplicity in hyperparameter adjustment as we only need to fine-tune the number of Gaussian components.

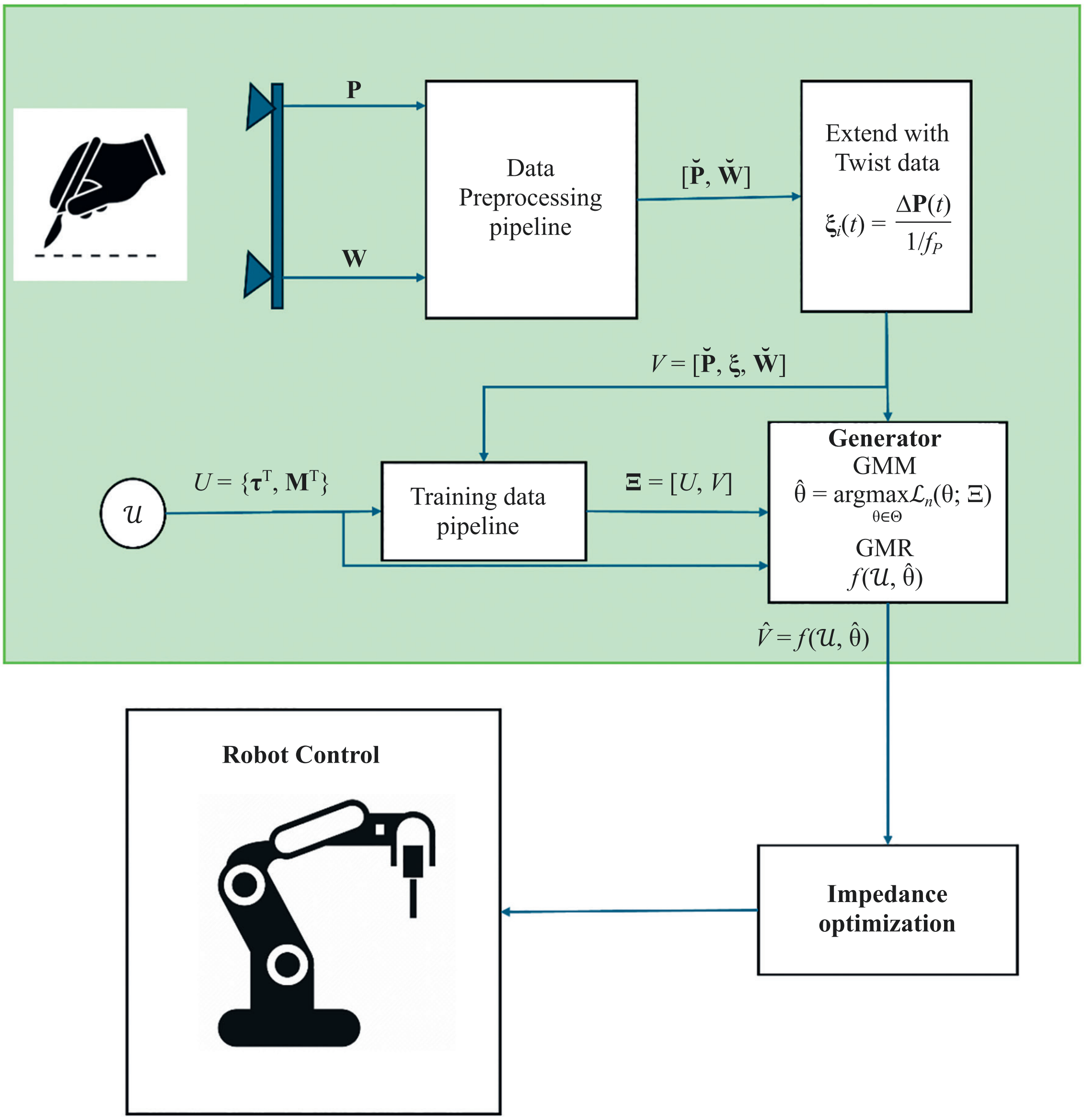

Fig. 1

We will refer to the GMM/GMR model parameters as $\theta = [\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}]$, where $\boldsymbol{\mu} \subset R^{K \times D}$ is the means of Gaussians vector, $\boldsymbol{\Sigma} \subset R^{K \times D \times D}$ is the covariance matrix, and $\boldsymbol{\pi} \subset R^{K}$ is the mixing weights that indicate how much each Gaussian component contributes to the model.

Then, generator training from a given pair of input-output data $\boldsymbol{\Xi} = [U, V]$ can be formalized as optimization problem:

$$\hat{\theta}^* = \underset{\hat{\theta}}{\operatorname{argmin}} \|V - \hat{V}\|, \text{ s. t.}$$

$$0 \le \pi_k \le 1;\ \Sigma_k \pi_k = 1,\ V_i^{\mathrm{T}} \Sigma_k V_i > 0,$$

$$V_i^{\mathrm{T}} \Sigma_k V_i > 0,$$

where V and $\hat{V}$ are recorded and predicted by the model $f(U, \hat{\theta})$ output values respectively given inputs U, V_i is i-th column of V, Σ_k is k-th covariance matrix from $\boldsymbol{\Sigma}$, $\hat{\theta}^*$ is the vector of desired values of model parameters' estimates that minimize the error between the desired output values.

Experimental setup. The experimental setup is depicted at Fig. 2. Here we introduce the following frames: stationary base frame {B} and two moving tool-attached frames {P} for scalpel and {W} for FT sensor with both origins located at the tool central point (TCP, we select it at the scalpel's blade tooltip), but different orientations.

The OptiTrack system with 8 cameras, positioned around the working space in a way that tool-attached markers are visible along its entire movement range by most of the cameras, was calibrated to capture the central area among all cameras. The base frame {B} was set on a fixed table in the center to guarantee precise capturing measurements for the frames {P} and {W}.

The FT sensor was attached to the scalpel through a custom-designed adapter consisting of two parts: a grip attached to the back of the FT sensor, held by a human hand, and a scalpel handler fastening the scalpel to the frontal side of the FT sensor. Eight markers were attached to the adapter in a way to ensure distribution along the entire body for better tracking accuracy.

Within this experimental setup, we introduced three different materials (cork — CRK, penoplex — PNX, and PVC). To ensure better generalization capabilities of the GMM/GMR, while still capturing shape preservation capabilities we performed a series of straight line parallel cuts of different length, 40 trials for each material resulting in a total of 120 demonstrations.

Out of demonstrations we can record the following set of data: trajectory of coordinate frame {P} with respect to the base frame {B} registered by MoCap system and denoted as $\mathbf{P} = [\mathbf{X}^{\mathrm{T}}, \mathbf{Y}^{\mathrm{T}}, \mathbf{Z}^{\mathrm{T}}, \mathbf{R}_x^{\mathrm{T}}, \mathbf{R}_y^{\mathrm{T}}, \mathbf{R}_z^{\mathrm{T}}]$, and the wrench measurements expressed in coordinate frame {W} and denoted as $\mathbf{W} = [\mathbf{F}_x^{\mathrm{T}}, \mathbf{F}_y^{\mathrm{T}}, \mathbf{F}_z^{\mathrm{T}}, \mathbf{T}_x^{\mathrm{T}}, \mathbf{T}_y^{\mathrm{T}}, \mathbf{T}_z^{\mathrm{T}}]$.

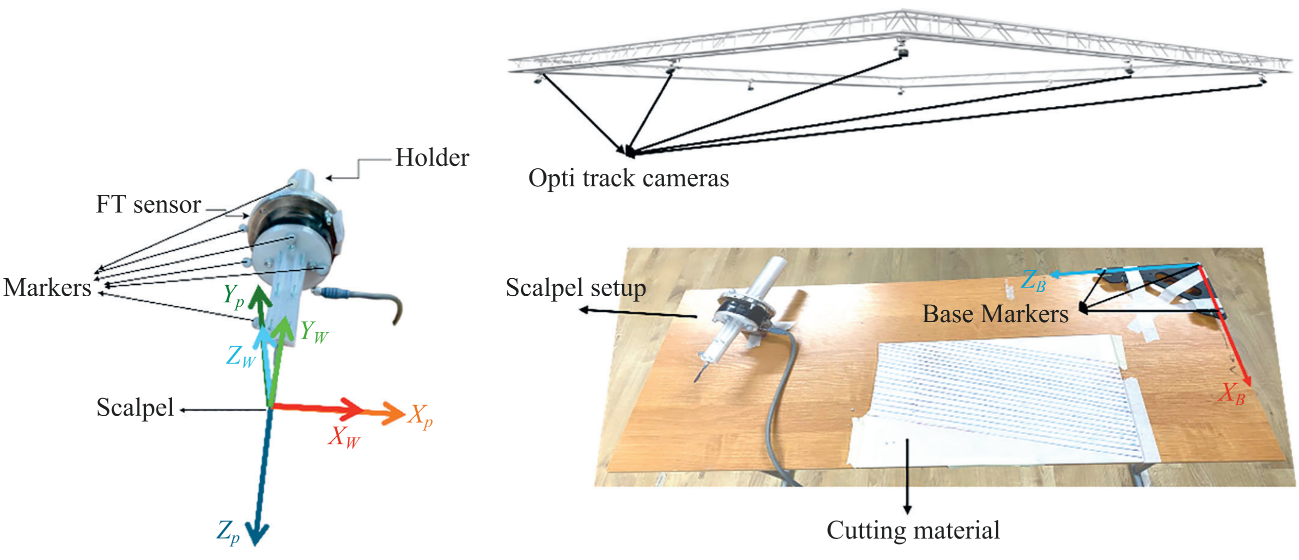

Fig. 2

Data processing. We introduced data processing procedure consisting from three steps.

1. Cutting phase slicing. Recorded data contains three phases: reaching, cutting and releasing. Since we are interested in training cutting skill in particular, we sliced entire sequence and extracted only data corresponding to that stage by detecting, when absolute value of contact forces $\mathbf{F}_x^{\mathrm{T}}$, $\mathbf{F}_z^{\mathrm{T}}$ change above a specified threshold (see Fig. 3: red is trajectory on X, Y, Z and green is F_x, F_y, F_z respectively).

2. Data imputation. We applied forward and backward fill techniques to guarantee the absence of missing values in recorded data.

3. Data filtering. As obtained position data are for cutting straight lines, we filtered outliers by fitting recorded sequences by a linear regression (see Fig. 4):

$$\check{\mathbf{Y}} = \alpha_y + \beta_y \mathbf{X},$$

where $\check{\mathbf{Y}}$ is the fitted value for $\mathbf{Y}$ measurements, $\alpha_y = \overline{\mathbf{Y}} - \beta_y \overline{\mathbf{X}}$, $\beta_y = \dfrac{\sum_{j=1}^{N}(X_j - \overline{\mathbf{X}})(Y_j - \overline{\mathbf{Y}})}{\sum_{j=1}^{N}(X_j - \overline{\mathbf{X}})}$, $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$ are mean values, N is the number is samples in the measurement's sequence. We fit Z measurements the same way.

We extended our training data by calculating from recorded MoCap trajectory data $\mathbf{P}$ the associated twist $\boldsymbol{\xi} = [\mathbf{V}_x^{\mathrm{T}}, \mathbf{V}_y^{\mathrm{T}}, \mathbf{V}_z^{\mathrm{T}}, \boldsymbol{\omega}_x^{\mathrm{T}}, \boldsymbol{\omega}_y^{\mathrm{T}}, \boldsymbol{\omega}_z^{\mathrm{T}}]$. It was done by numeric differentiation:

$$\xi_i(t) = \frac{\Delta \mathbf{P}(t)}{1/f_P},$$

where ξ_i, is the i-th column of ξ, $\xi_i(0) = 0$, $\Delta\mathbf{P}(t)$ is the change in position between two consecutive measurements, f_P is the measurements frequency.

To filter wrench measurements $\mathbf{W}$ we applied Exponential Moving Average (EMA) filter, which is a type of infinite impulse response filter that applies weighting factors which decrease exponentially (see Fig. 5):

$$W_i^t = \alpha_{\mathrm{EMA}} W_i^t + (1 - \alpha_{\mathrm{EMA}}) W_i^{t-1},$$

where W_i is the i-th column of $\mathbf{W}$, W_i^t is the exponential moving average value of W_i at time t, $\alpha_{\mathrm{EMA}} = \dfrac{2}{n_s + 1}$ is the filter's smoothing factor with ns being the desired number of periods or the span of

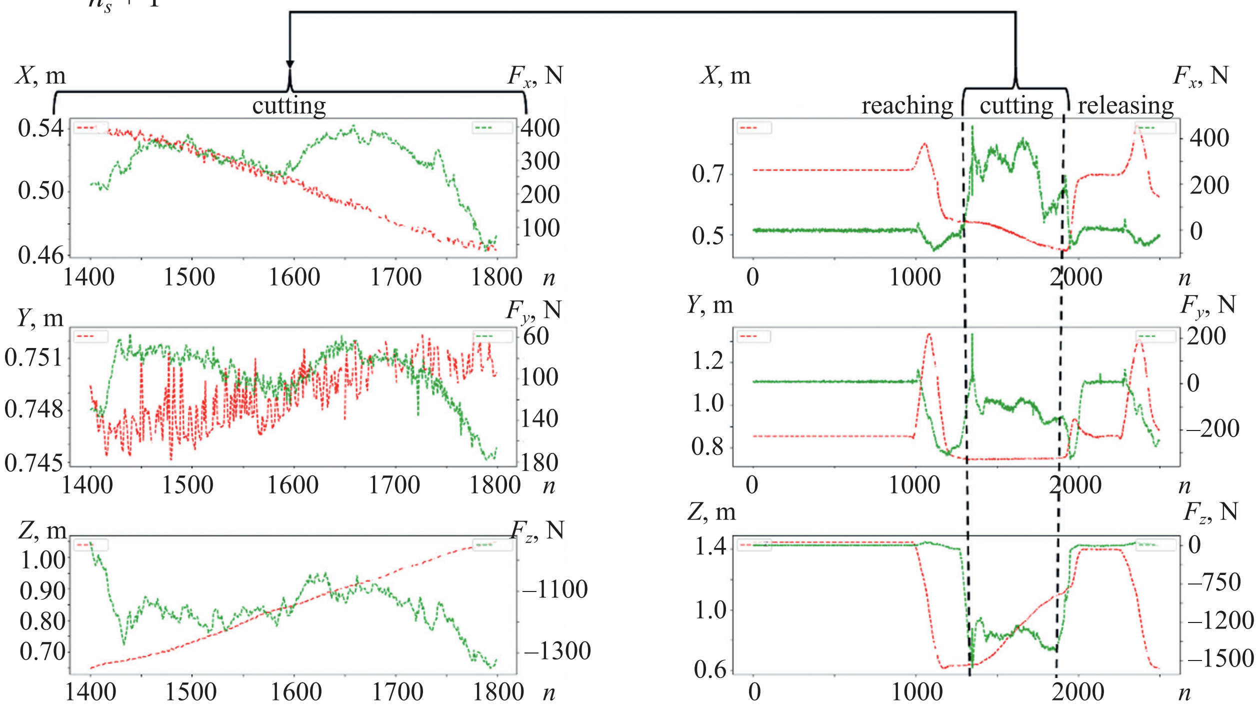

Fig. 3

the EMA. The choice of affects the sensitivity of the EMA to changes in the data: a smaller n_s makes the EMA more responsive to new data, while a larger n_s makes smoothing stronger.

Model training. We define generator input as a concatenation of time and material label sequences $\mathbf{U} = [\boldsymbol{\tau}^{\mathrm{T}}, \mathbf{M}^{\mathrm{T}}] \subset \mathcal{R}^{2\times N}$ and the output is defined as $V = [\check{\mathbf{P}}\ , \xi, \check{\mathbf{W}}] \subset \mathcal{R}^{18\times N}$.

GMMs assume that the training data $\boldsymbol{\Xi} = [U, V]$ are generated from a mixture of several Gaussian distributions, each characterized by parameters mean μ_k covariance Σ_k, and a mixing coefficient π_k which represents the weight of the k-th Gaussian component in the mixture. Then, the model training problem formulated above can be resolved by applying expectation maximization algorithm with log-likelihood maximization criteria:

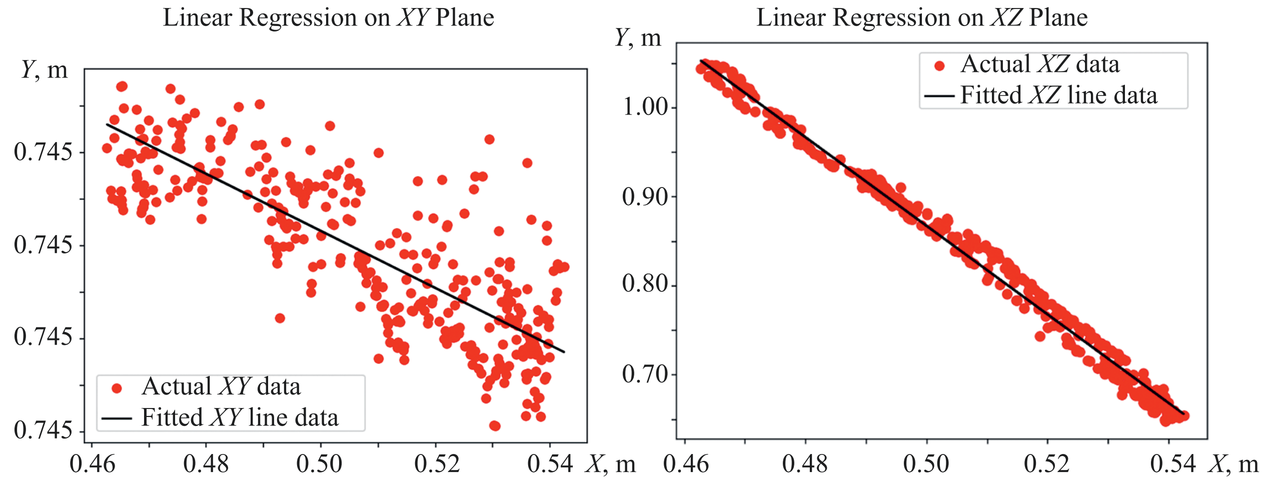

Fig. 4

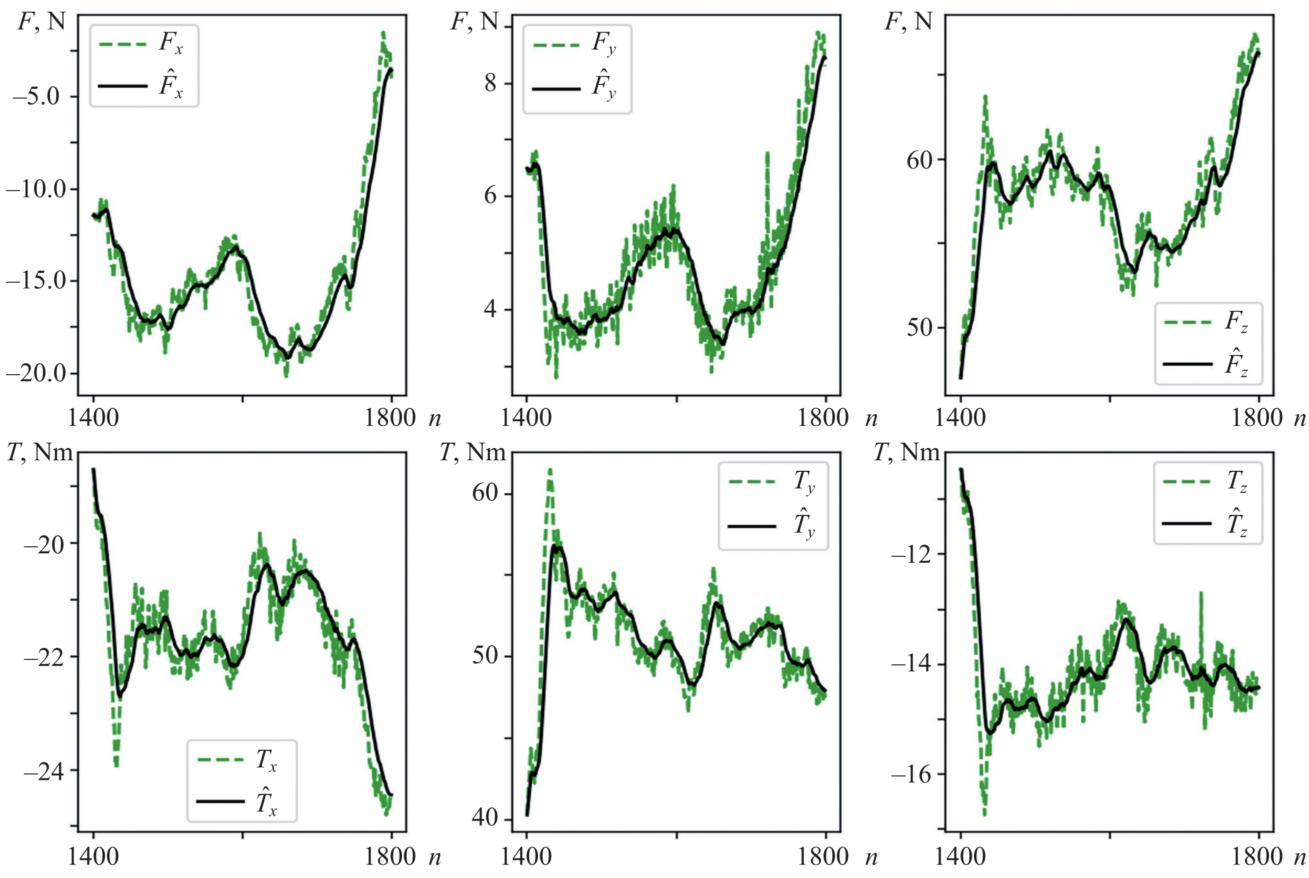

Fig. 5

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmax}} \mathcal{L}(\theta; \Xi),$$

where log-likelihood cost function

$$\mathcal{L}(\theta; \Xi) = \log(P(\Xi, \rho|\theta)) = \sum_{n=1}^{N} \sum_{k=1}^{K} I(\rho_n = k)(\log(\pi_k) + \log(N(\Xi_n|\mu_k, \Sigma_k))),$$

ρ_n is a latent variable indicating the component that generated the (n_{th}) data point, ($I(\rho_n = k)$) is an indicator function that is 1 if ($\rho_n = k$) and 0 otherwise, ($N(\Xi_n|\mu_k, \Sigma_k)$) is the probability density of (Ξ_n) under the Gaussian distribution with parameters (μ_k) and (Σ_k).

The algorithm consists of two steps. At the expectation step we compute the expected value of $\mathcal{L}(\theta; \Xi)$

$$E_{Z|\Xi}[\mathcal{L}(\theta; \Xi)] = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{\rho_n}(k)(\log(\pi_k) + \log(N(\Xi_n|\mu_k, \Sigma_k))),$$

where $\gamma_{\rho_n}(k) = E[I(\rho_n = k)] = P(\rho_n = k|\Xi) = \dfrac{\pi_k N(\Xi_n|\mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k N(\Xi_n|\mu_k, \Sigma_k)}$ represents the posterior probability that the (n_{th}) data point was generated by the (k_{th}) component, given the current parameter estimates.

At the maximization step we maximize the expected complete log-likelihood obtained at the previous step with respect to parameters θ keeping ($\gamma_{\rho_n}(k)$) fixed.

So, we update parameters' estimates the following way:

$$\mu_k^{\text{new}} = \frac{\sum_{n=1}^{N} \gamma_{\rho_n}(k) \Xi_n}{\sum_{n=1}^{N} \gamma_{\rho_n}(k)},$$

$$\Sigma_k^{\text{new}} = \frac{\sum_{n=1}^{N} \gamma_{\rho_n}(k)(\Xi_n - \mu_k^{\text{new}})(\Xi_n - \mu_k^{\text{new}})^{\mathrm{T}}}{\sum_{n=1}^{N} \gamma_{\rho_n}(k)},$$

$$\pi_k^{\text{new}} = \frac{\sum_{n=1}^{N} \gamma_{\rho_n}(k)}{N}.$$

Estimates convergence is assessed based on the change in log-likelihood between successive iterations. Specifically, the algorithm is considered to have converged when the change in log-likelihood is below a predefined tolerance level , i.e.,

$$\Delta\mathcal{L} = |\mathcal{L}(\theta^{(t+1)}; \Xi) - \mathcal{L}(\theta^{(t)}; \Xi)| < \alpha,$$

where: $\theta^{(t)}$ and ($\theta^{(t+1)}$) are the parameter sets from consecutive iterations (t) and ($t + 1$) respectively.

Next, at the regression prediction step we calculate parameters of the GMR model out of GMM representation. At first, we decompose GMM parameters for input and output dimensions:

$$\mu_k = \begin{bmatrix} \mu_k^{\mathcal{U}} \\ \mu_k^{\mathcal{V}} \end{bmatrix}, \Sigma k = \begin{bmatrix} \Sigma_k^{\mathcal{U}\mathcal{U}} & \Sigma_k^{\mathcal{U}\mathcal{V}} \\ \Sigma_k^{\mathcal{V}\mathcal{U}} & \Sigma_k^{\mathcal{V}\mathcal{V}} \end{bmatrix},$$

where $\mu_k^{\mathcal{U}} \subset \boldsymbol{\mathcal{R}}^{\mathcal{D}_u}$, $\mu_k^{\mathcal{V}} \subset \boldsymbol{\mathcal{R}}^{\mathcal{D}_v}$, $\Sigma_k^{\mathcal{U}\mathcal{U}} \subset \boldsymbol{\mathcal{R}}^{\mathcal{D}_u \times \mathcal{D}_u}$, $\Sigma_k^{\mathcal{V}\mathcal{V}} \subset \boldsymbol{\mathcal{R}}^{\mathcal{D}_v \times \mathcal{D}_v}$, $\Sigma_k^{\mathcal{V}\mathcal{U}} \subset \boldsymbol{\mathcal{R}}^{\mathcal{D}_v \times \mathcal{D}_u}$, $\Sigma_k^{\mathcal{U}\mathcal{V}} \subset \boldsymbol{\mathcal{R}}^{\mathcal{D}_u \times \mathcal{D}_v}$, for each component k, $\mu_k^{\mathcal{V}|\mathcal{U}} = \mu_k^{\mathcal{V}} + \Sigma_k^{\mathcal{V}\mathcal{U}}(\Sigma_k^{\mathcal{U}})^{-1}(\mathcal{U} - \mu_k^{\mathcal{U}})$, $\Sigma_k^{\mathcal{V}|\mathcal{U}} = \Sigma_k^{\mathcal{V}\mathcal{V}} - \Sigma_k^{\mathcal{V}\mathcal{U}} (\Sigma_k^{\mathcal{U}\mathcal{U}})^{-1}\Sigma_k^{\mathcal{U}\mathcal{V}}$.

Then, we calculate the predicted output as

$$\hat{\mathcal{V}} = \gamma_\rho(K)^{\mathcal{U}} \mu_k^{\mathcal{V}|\mathcal{U}},$$

where $\gamma_{\rho}(K)^{\mathcal{U}} = \dfrac{\pi_k N(\mathcal{U};\, \mu_k^{\mathcal{U}},\, \Sigma_k^{\mathcal{UU}})}{\sum_{k=1}^{K} \pi_k N(\mathcal{U};\, \mu_k^{\mathcal{U}},\, \Sigma_k^{\mathcal{UU}})}$ is the responsibility.

Training results. Figures 6–9 illustrate the resulted generated trajectory-wrench data by GMM/GMR compared to expected values. Fig. 6 — expected vs generated cutting trajectories on XZ plane using trained GMM/GMR for 3 different materials: *a* — cork, *b* — PVC, *c* — PNX. Fig. 7 — generated from GMM/GMR vs Expected Wrench data for cork; Fig. 8 — for penoplex; Fig. 9 — for PVC.

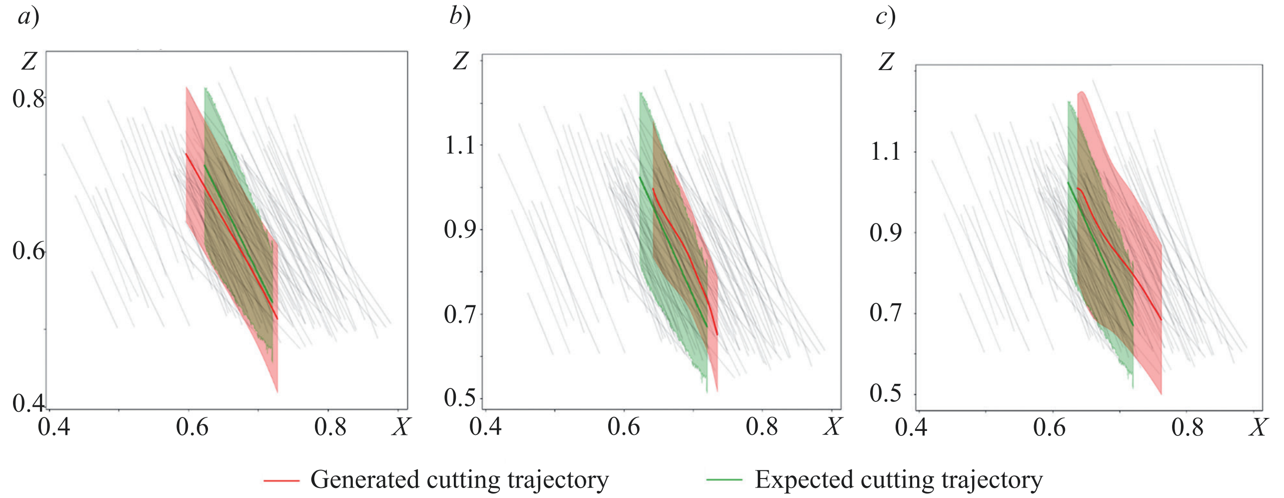

Fig. 6

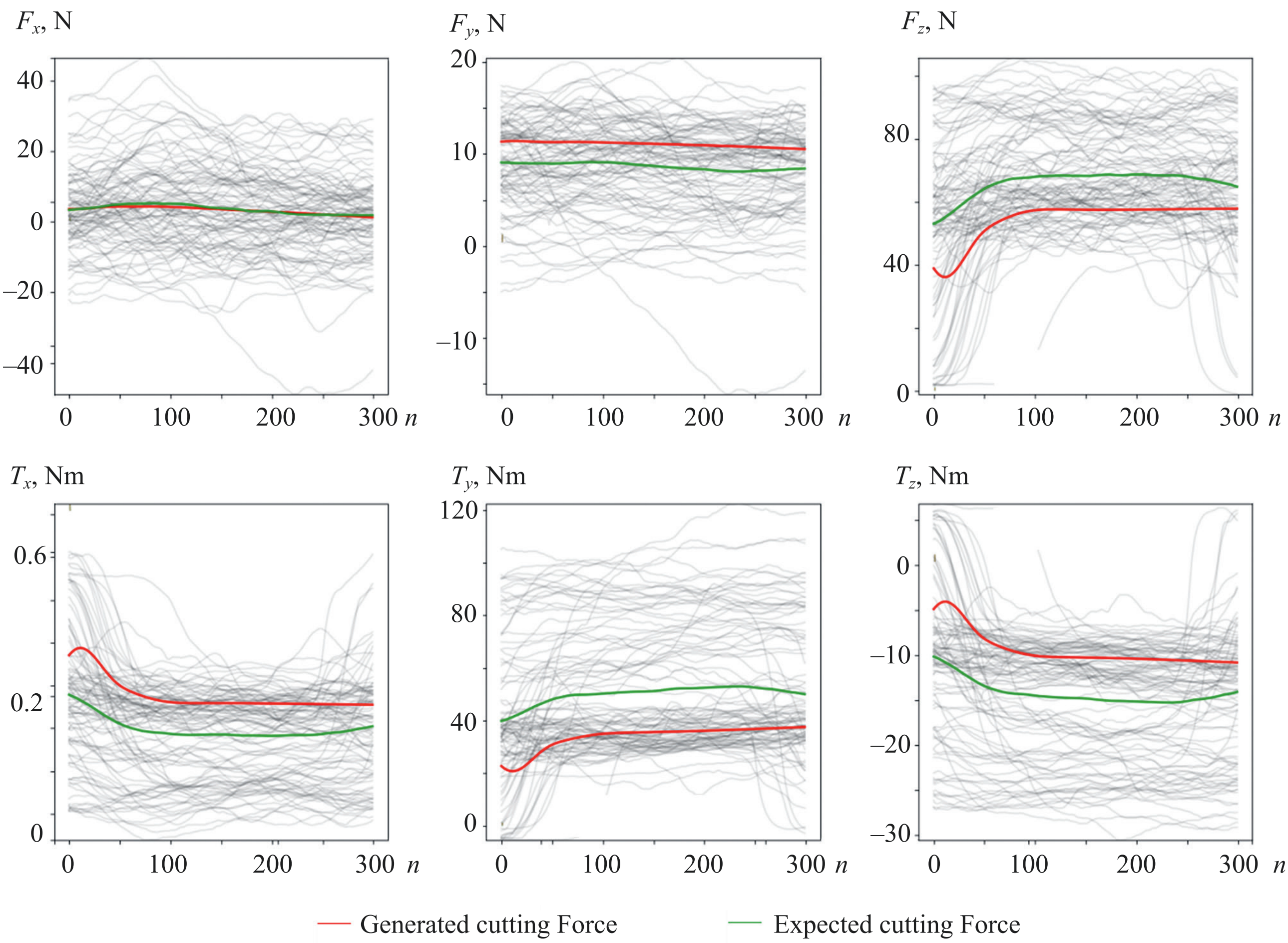

Fig. 7

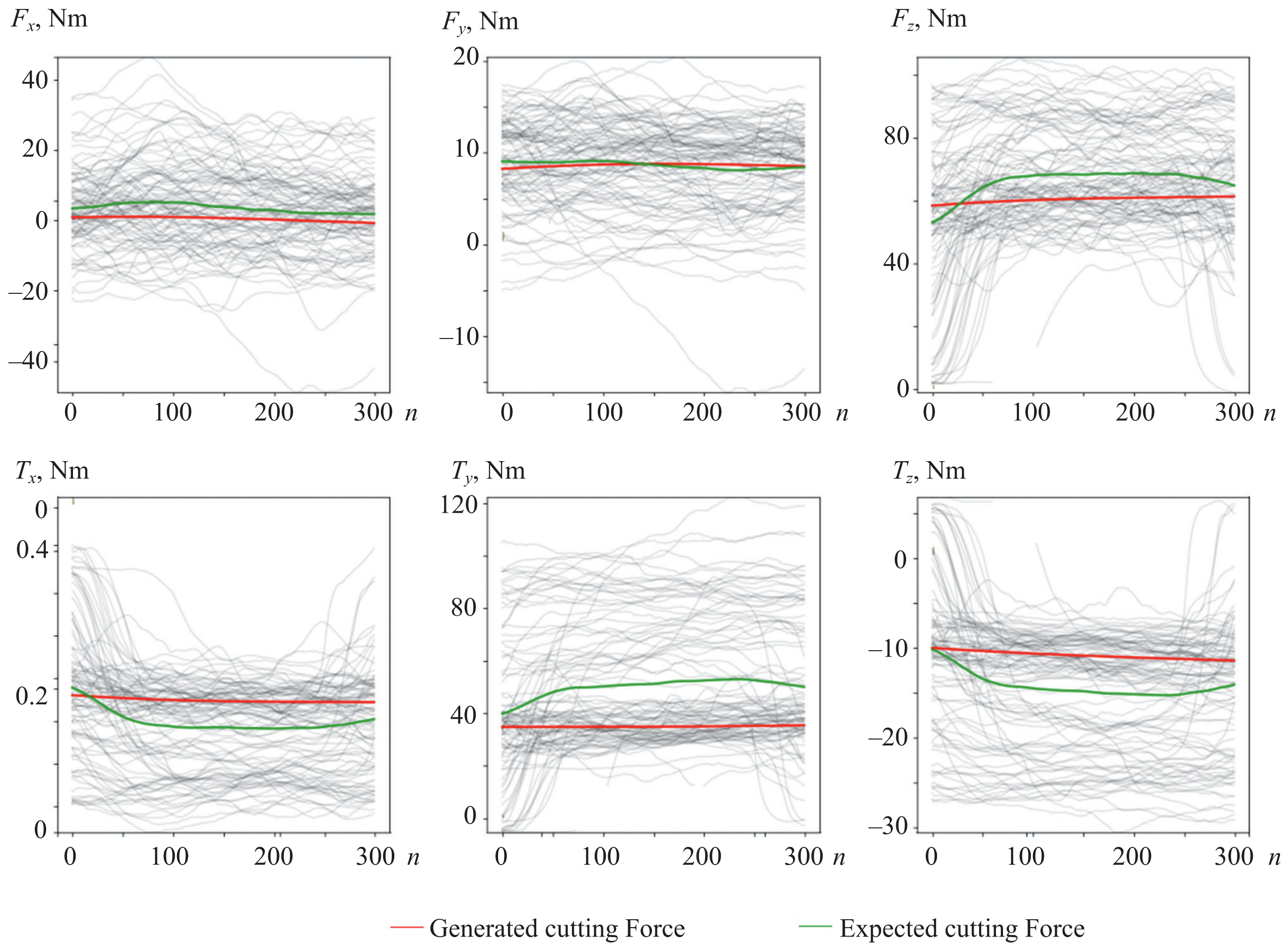

Fig. 8

Fig. 10 illustrates how model parameters (θ) converge to fit the training trajectory-wrench data (Ξ) using change in log-likelihood criteria ($\Delta\mathcal{L}$) which reaches the threshold ($\alpha = 1^{-5}$) within 100 epochs (training iterations).

The results of training the GMM/GMR model on the preprocessed collected data and evaluating the generated trajectory, twist and wrench data compared to expected means of training trajectory, twist and wrench data Ξ. To assess the model's performance, metrics such as root mean squared error (RMSE)) for Twist ξ and Position **P**, was employed. These evaluation metrics are detailed in Table.

$$\epsilon = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\overline{\mathcal{V}}_i - \hat{\mathcal{V}}_i)},$$

where $\mathcal{V}$ and $\overline{\mathcal{V}}$ denote predicted and averaged recorded values for $\mathcal{V} \subset [\mathrm{P}, \xi]$ at time t_i, ϵ refers to RMSE.

To evaluate the correspondence of the generated wrench to position data, we calculated the root mean squared error for the calculated power values from both predicted and expected Wrench and Position data ($\overline{\mathbf{\Pi}}$, $\hat{\mathbf{\Pi}}$), respectively. The results under different materials and cutting shapes are illustrated in Table:

$$\overline{\mathbf{\Pi}}_i = \overline{\mathbf{W}}_i \overline{\xi}_\iota,$$

$$\hat{\mathbf{\Pi}}_i = \hat{\mathbf{W}}_i \hat{\xi}_\iota,$$

where i refers to axes (X, Y, Z)

$$\epsilon_\Pi = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\overline{\mathbf{\Pi}}_i - \hat{\mathbf{\Pi}}_i)}.$$

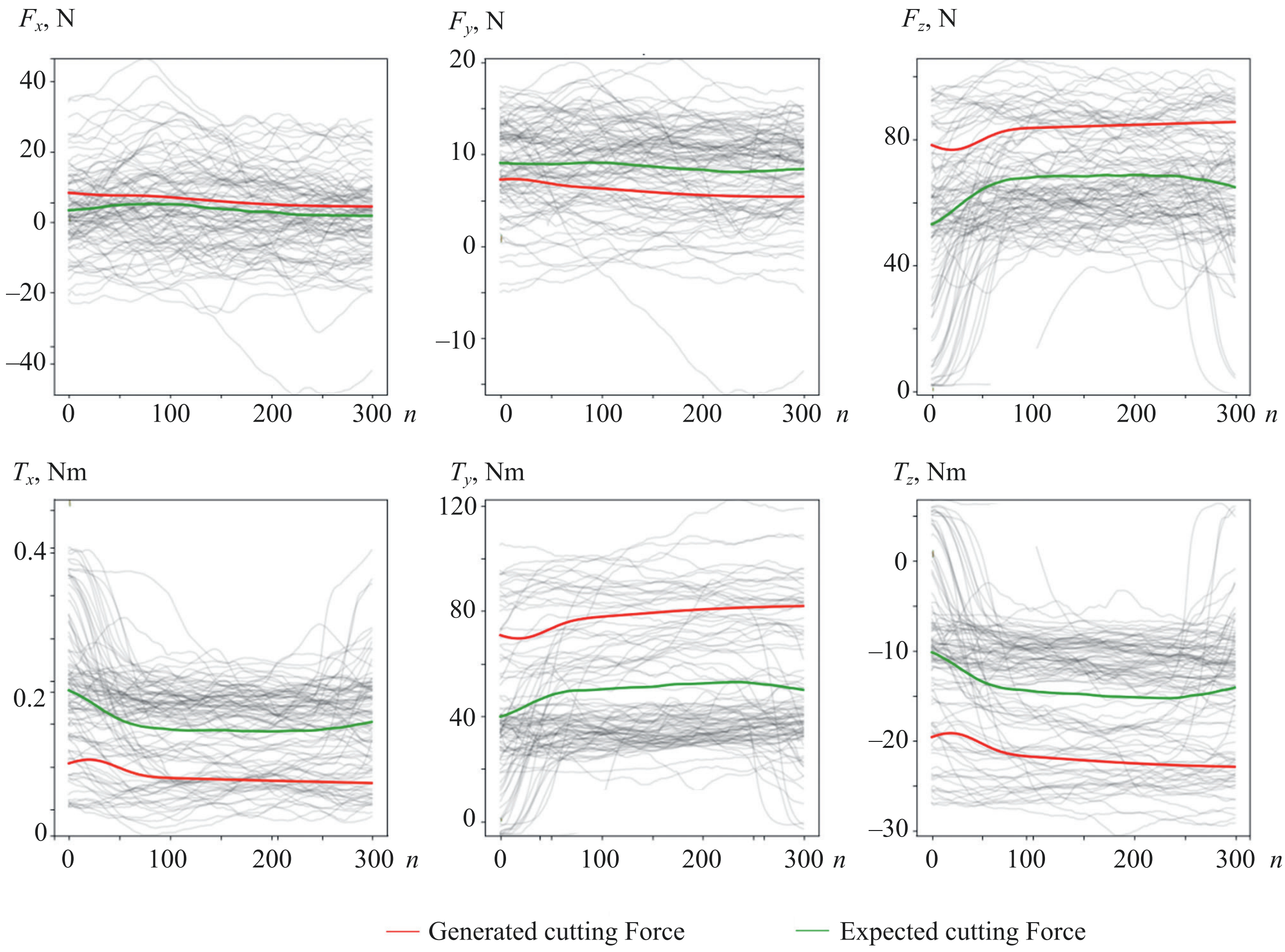

Fig. 9

RMSE Evaluation metric for training the GMM/GMR model for all data where ϵ_P, ϵ_ξ, ϵ_Π, represent the RMSE for Position, twist and Power, respectively

Material	ϵ_P, m	ϵ_ξ, ms	ϵ_Π, W
Cork	0.1165	0.0714	0.1408
Penoplex	0.1031	0.0711	0.1462
PVC	0.1012	0.0710	0.1430

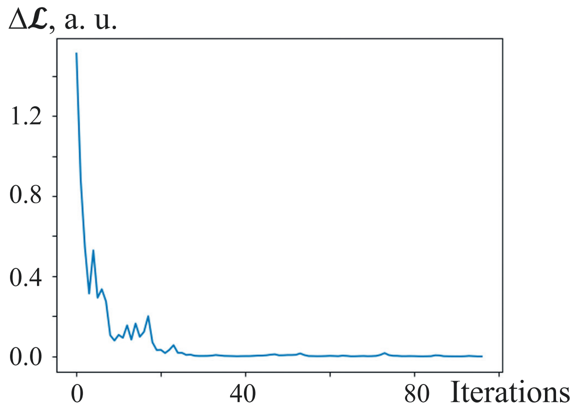

Fig. 10

Conclusions and future work. In this study, our primary objective was to train behavior priors models for robotic manipulation in interactive tasks, where both trajectory and force profiles hold significant importance. Initially, we conducted simultaneous collection of trajectory and force/torque data across multiple trials for the material cutting scenario. This involved designing a custom setup with a scalpel attached to an force-torque sensor to capture interaction wrench data and a motion capture system to record associated cutting trajectories. Subsequently, we underwent a data preprocessing phase to ready the dataset for training a GMM/GMR model, as a generator for behavior priors. We validated the convergence of the proposed model training and verified its performance on tests datasets, which demonstrated high accuracy. The novelty of this work is also in using GMM/GMR model with extended input that includes material labels, which opens opportunity for implementing an approach with a mixture of behavior prior generators specific for different materials. Future steps involve utilizing obtained behavior priors for regularization to improve robot interaction control systems that can be based on Reinforcement Learning (RL) policies training or on optimization-based modern indirect force control algorithms similar to VIC (Variable Impedance Controllers) to train nonlinear functions for stiffness and damping tuning.

REFERENCES

1. Noémie J., Welle M. C., Gams A., Yao K., Fichera B., Billard A., Ude A., Asfour T., and Kragić D. *arXiv preprint arXiv*:2311.18044, 2023.
2. Jauhri S., Peters J., and Chalvatzaki G. *IEEE Robotics and Automation Letters*, 2022, no. 3(7), pp. 8399–8406.
3. Liu P., Zhang K., Tateo D., Jauhri S., Hu Zh., Peters J., and Chalvatzaki G. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023, pp. 9449–9456.
4. Shek A., Su B. Y., Chen R, and Liu Ch. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023, pp. 9910–9916.
5. Zhao J., Giammarino A., Lamon E., Gandarias J., De Momi E., and Ajoudani A. *IEEE Robotics and Automation Letters*, 2022, no. 7, pp. 1–8.
6. Calinon S. *Intelligent service robotics*, 2016, no. 9, pp. 1–29.
7. Fabisch A. *Journal of Open Source Software*, 2021, no. 6 (62), pp. 3054.
8. Sung H. G. *Gaussian mixture regression and classification*, Rice University, 2004.

DATA ON AUTHORS

Waddah Ali — Post-Graduate Student; ITMO University, Faculty of Control Systems and Robotics, International Laboratory of Biomechatronics and Energy-Efficient Robotics; Engineer

Sergey A. Kolyubin — Dr. Sci., ITMO University, Faculty of Control Systems and Robotics, International Laboratory of Biomechatronics and Energy-Efficient Robotics; Professor; Chief Researcher; E-mail: s.kolyubin@itmo.ru

Поступила в редакцию 15.03.2024; одобрена после рецензирования 28.03.2024; принята к публикации 16.04.2024.

СПИСОК ЛИТЕРАТУРЫ

1. *Noémie J., Welle M. C., Gams A., Yao K., Fichera B., Billard A., Ude A., Asfour T., and Kragić D.* Transfer Learning in Robotics: An Upcoming Breakthrough? A Review of Promises and Challenges // arXiv preprint arXiv:2311.18044. 2023.
2. *Jauhri S., Peters J., and Chalvatzaki G*. Robot learning of mobile manipulation with reachability behavior priors // IEEE Robotics and Automation Letters. 2022 Vol. 3, N 7. P. 8399–8406.
3. *Liu P., Zhang K., Tateo D., Jauhri S., Hu Zh., Peters J., and Chalvatzaki G*. Safe reinforcement learning of dynamic high-dimensional robotic tasks: navigation, manipulation, interaction // 2023 IEEE Intern. Conf. on Robotics and Automation (ICRA). IEEE, 2023. P. 9449–9456.
4. *Shek A., Su B. Y., Chen R., and Liu Ch.* Learning from physical human feedback: An object-centric one-shot adaptation method // 2023 IEEE Intern. Conf. on Robotics and Automation (ICRA). IEEE, 2023. P. 9910–9916.
5. *Zhao J., Giammarino A., Lamon E., Gandarias J., De Momi E., and Ajoudani A*. A hybrid learning and optimization framework to achieve physically interactive tasks with mobile manipulators // IEEE Robotics and Automation Letters. 2022. N 7. P. 1–8.
6. *Calinon S*. A tutorial on task-parameterized movement learning and retrieval // Intelligent service robotics. 2016. N 9. P. 1–29.
7. *Fabisch A*. gmr: Gaussian mixture regression // J. of Open Source Software. 2021. Vol. 62, N 6. P. 3054.
8. *Sung H. G*. Gaussian mixture regression and classification. Rice University, 2004.

СВЕДЕНИЯ ОБ АВТОРАХ

Ваддах Али — аспирант; Университет ИТМО, факультет систем управления и робототехники, Международная лаборатория Биомехатроники и энергоэффективной робототехники; инженер: E-mail: waddahkh.ali94@gmail.com

Сергей Алексеевич Колюбин — докт. техн. наук; факультет систем управления и робототехники, Международная лаборатория Биомехатроники и энергоэффективной робототехники; профессор, г.н.с; E-mail: s.kolyubin@itmo.ru

Received 15.03.2024; approved after reviewing 28.03.2024; accepted for publication 16.04.2024