IODA-TC: Imaginary Out-of-Distribution Actions with Trajectory-Continuation Mapping

Isabella Bock
Tufts University
Medford USA
Isabella.Bock@tufts.edu

Elaine Schaertl Short
Tufts University
Medford USA
elaine.short@tufts.edu

Abstract

To enable smoother human-robot interactions, a robot's policy must be predictable to users. A recent method, Imaginary Out-of-Distribution Actions (IODA), preserves user expectation by mapping Out-of-Distribution (OOD) states to In-Distribution (ID) states in shared-control settings. However, one limitation of this method is that it uses Euclidean distance which may fail to capture semantic similarity, especially in high-dimensional state spaces. In this report, we analyze limitations of using Euclidean distance for the state mapping and propose a Trajectory-Continuation (TC) mapping designed to preserve predictability by selecting ID states based on local trends.

CCS Concepts

• **Human-centered computing** → **Interaction paradigms**; *Collaborative and social computing*; • **Computing methodologies** → Machine learning approaches; • **Computer systems organization** → *Robotic autonomy*.

Keywords

Human-Robot Interaction (HRI), Shared Control, Predictable Motion, Out-of-Distribution (OOD)

ACM Reference Format:
Isabella Bock and Elaine Schaertl Short. 2026. IODA-TC: Imaginary Out-of-Distribution Actions with Trajectory-Continuation Mapping. In *Companion Proceedings of the 21st ACM/IEEE International Conference on Human-Robot Interaction (HRI Companion '26), March 16–19, 2026, Edinburgh, Scotland, UK*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3776734.3794449

1 Introduction

Predictable robot behavior is important for enabling successful collaboration between humans and robots [3]. One prior approach, the Imagining Out-Of-Distribution Actions (IODA) algorithm, addresses erratic motion in shared-control tasks by mapping out-of-distribution (OOD) states back to in-distribution (ID) states [11]. However, this approach relies on a Euclidean-distance mapping that may fail to preserve human expectations for continuous motion. In this work, we propose a Trajectory-Continuation (TC) mapping from OOD to ID states that incorporates semantic similarity, which we define as the extent to which the ID state produces actions continuing a robot's recent trajectory.

HRI Companion '26, Edinburgh, Scotland, UK

ACM ISBN 979-8-4007-2321-6/2026/03
https://doi.org/10.1145/3776734.3794449

Incorporating semantic similarity into the state-mapping addresses two key limitations of the Euclidean-distance mapping. First, geometric distance doesn't always align with how humans predict motion. People often expect continuity rather than real physical dynamics [7]. For example, many people expect an object that exits a curved tube to continue curving, despite physics dictating that it will travel straight [9]. Analogously, a robot motion selected by Euclidean distance may produce motion that breaks this intuitive sense of continuity. Second, Euclidean distance becomes unreliable in high dimensions [2], making it mathematically unstable for selecting ID states.

We illustrate the TC and Euclidean-distance mappings in a 2D navigation demonstration with a simulated policy failure. While Euclidean distance maps the agent to an ID state that leads to a discontinuous action, the TC mapping selects a state whose action continues a prior motion trend. Although this illustrates preliminary evidence for our algorithm's conceptual contribution, further empirical validation of its effects on predictability, trust, and coordination is required.

2 Background

Previous work has shown that predictable robot motion can be valuable for effective human-robot interaction. Predictability, defined as the extent to which robot motion meets human expectations, allows users to form models of a robot's behavior [3]. Consistency in robot behavior supports coordination [4], increases trust in robotic systems [5], and supports the overall effectiveness of human-robot collaboration [6].

In shared-control tasks, predictability is especially important as users must harness their expectations for robot behavior to anticipate and adapt to robot responses. Shared control allows a user to work alongside an automated robot to perform dynamic tasks [1]. Prior work highlights that user trust is essential in shared-control settings where robots are completing novel tasks [5]. Unexpected motion from a robot causes users to lose confidence and coordination, so a robot's policy must reflect user expectation to enable collaboration [10].

The IODA algorithm supports collaboration in shared-control settings by leveraging a previously trained policy and simulating "imagined states" to help users attempt new tasks [11]. Users harness their predictions for how a robot will behave as they control one axis of robot motion. When the robot is brought into an OOD state s, IODA uses a distance mapping $d(s, s')$ to select the in distribution (ID) state $s' \in D$ that will be used to select the robot action.

3 Limitations of Euclidean Mapping

The Euclidean-distance metric introduces limitations to the mapping of OOD states in high-dimensional robot state spaces. The curse of dimensionality leads to mappings that may break human expectations for continuous motion. As a result, the Euclidean distance fails to capture the semantic similarity of robot states and leads to actions that may be unpredictable to users.

The Curse of Dimensionality In high-dimensional robot state spaces that include features such as joint angles and sensor readings, Euclidean distance becomes less reliable. As dimensionality increases, the Euclidean distance to nearest and farthest neighbors becomes less meaningful [2]. Thus, in high-dimensional robot state spaces OOD states have many neighbors that tend to be equally close, diminishing the ability of Euclidean distance to map OOD states to reliable ID states.

For example, small amounts of noise may affect the selected ID state leading to instability in the projection. If an OOD state s has multiple ID neighbors that are geometrically close, a minimal perturbation can cause the projection to flip from an ID region associated with an action A to a region associated with action B. In the case that actions A and B correspond to abruptly different actions, this can break user predictability by disrupting continuous movement.

To demonstrate this instability, we calculated nearest-neighbor distance on a Mahalanobis OOD point (red x) in a 30-dimensional multivariate-normal dataset with $1,000$ samples [8]. After altering the OOD point by $\Delta = 0.5$, the nearest neighbor flipped from index 328 (purple o) to index 537 (blue o) [Figure 1]. The flip in the selection of the ID state translates to potential temporal instability in a robot's motion. For example, if one nearest neighbor corresponds to an action that maintains a forward path, but the other equally close neighbor corresponds to an action that leads the agent to veer off course, the switch in selection of the ID state would break user expectations for continuous motion.

Semantic Similarity The instability of OOD projections arising from the curse of dimensionality leads to a breakdown of semantic similarity. Since users rely on policies to continue patterns of motion, the flip in behavior goes against predictability.

Since Euclidean distance only minimizes spatial differences, it ignores functional or temporal characteristics that define a policy's action. To represent robot motion, trajectory representations [12] take into account velocity and time when defining how a robot should move along a path. These representations show that state transitions and temporal smoothness are critical in defining robot motions. Our TC mapping incorporates this idea by considering trajectory continuity rather than relying solely on Euclidean distance.

4 Trajectory-Continuation Mapping

TC mapping prioritizes trajectory continuation by estimating the local trajectory of the OOD state. Prior work has defined a trajectory as a function of time $q(t)$ in a robot's configuration space Q [12]. We extend this definition from the configuration space Q to the state space S where state vectors $\mathbf{s}$ may contain features beyond kinematics.

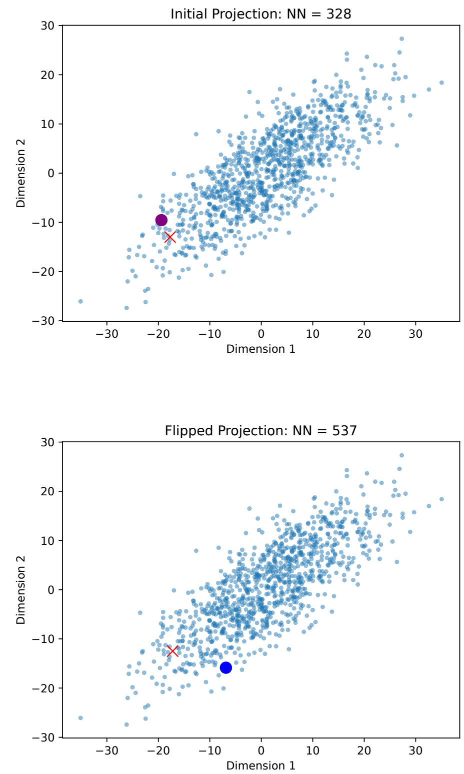


Figure 1: Demonstration of Euclidean Mapping Instability. *Top:* The nearest neighbor (purple o) at index 328 to an OOD point (red x). *Bottom:* After altering the OOD point by $\Delta = 0.5$, the nearest neighbor (blue o) flips to index 537.

We define a path as a continuous map, $\tau : [0, 1] \rightarrow S$ with $\tau(0) = s_{init}$ and $\tau(1) = s_{final}$. The trajectory is a function $s(t)$ such that $s(t_0) = s_{init}$ and $s(t_f) = s_{final}$. We estimate local trajectory from finite differences in states.

As defined in prior work, the system is modeled as an MDP with states $S \subset \mathbb{R}^n$, actions A, reward function r, and transition function T. The robot has learned an optimal policy $\pi^* : S \rightarrow A$, and the rollout history D is defined as the set of in-distribution states under π^* informing a user's prediction [11].

To calculate the ID trajectory **v'** that will most closely continue the OOD trajectory $\mathbf{v}$, we define the OOD velocity vector:

$$\mathbf{v} = \mathbf{s}_t - \mathbf{s}_{t-1} \tag{1}$$

where $\mathbf{s}_t$ and $\mathbf{s}_{t-1}$ are consecutive OOD states. For a subset of candidate ID states $\mathbf{s'} \in D$ we calculate the policy's velocity vector **v'**.

$$\mathbf{a} = \pi^*(\mathbf{s'}) \tag{2}$$

$$\mathbf{v'} = T(\mathbf{s'}, \mathbf{a}) - \mathbf{s'} \tag{3}$$

In order to find the semantic similarity of the two state vectors, we use cosine similarity and transform it to a positive measurable distance:

$$d_{\text{traj}} = 1 - \frac{\mathbf{v}' \cdot \mathbf{v}}{\|\mathbf{v}\| \cdot \|\mathbf{v}'\|} \tag{4}$$

Next, we define a cost function C for $\mathbf{s}'$ where α is a weight to adjust the importance of Euclidean distance versus trajectory similarity. We define d_{euclid} to be the normalized Euclidean distance to match the scaling of d_{traj}.

$$C = \alpha \cdot d_{\text{traj}} + (1 - \alpha) \cdot d_{\text{euclid}} \tag{5}$$

Lastly, we select the ID state from a set of candidate states $\mathbf{s'} \in D$ that minimizes C for IODA. By incorporating the trajectory-alignment distance d_{traj} alongside the geometric distance d_{euclid}, the TC mapping reflects semantic similarity. This approach is designed to prevent selection of ID states that result in abrupt change in robot actions in order to reflect user prediction for smooth movement.

Algorithm 1 Trajectory-Continuation Mapping

Require: Current OOD state $\mathbf{s}_t$, Previous OOD state $\mathbf{s}_{t-1}$, ID dataset D, Policy π, Transition function T, Weight α.

$\mathbf{v} \leftarrow \mathbf{s}_t - \mathbf{s}_{t-1}$
min_cost $\leftarrow \infty$
$\mathbf{s}_t^* \leftarrow$ None
for each candidate state $\mathbf{s}'$ in $\mathcal{D}$ **do**
  $\mathbf{a}' \leftarrow \pi(\mathbf{s}')$
  $\mathbf{s}_{\text{next}} \leftarrow T(\mathbf{s}', \mathbf{a}')$
  $\mathbf{v}' \leftarrow \mathbf{s}_{\text{next}} - \mathbf{s}'$
  $d_{\text{traj}} \leftarrow 1 - \frac{\mathbf{v}' \cdot \mathbf{v}}{\|\mathbf{v}\| \cdot \|\mathbf{v}'\|}$
  $d_{\text{euclid}} \leftarrow \text{normalize}(\|\mathbf{s}_t - \mathbf{s'}_t\|)$
  current_cost $\leftarrow \alpha \cdot \mathrm{d}_{\text{traj}} + (1 - \alpha) \cdot \mathrm{d}_{\text{euclid}}$
  if current_cost $<$ min_cost **then**
    min_cost $\leftarrow$ current_cost
    $\mathbf{s}_t^* \leftarrow \mathbf{s}'$
  end if
end for
return $\mathbf{s}_t^*$

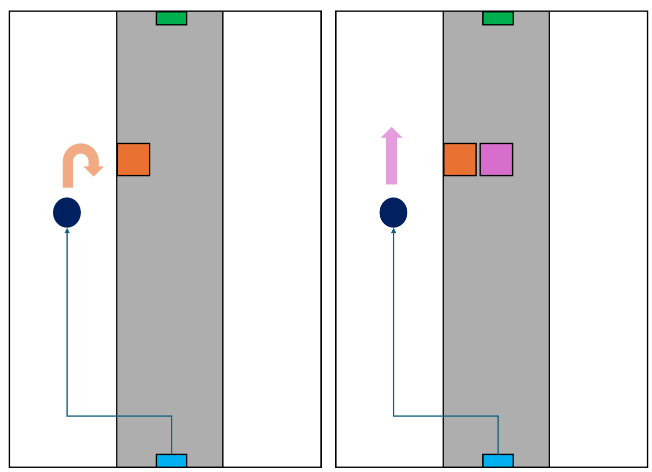
5 2D Navigation Demonstration

We illustrate the performance of the TC mapping compared to the Euclidean-distance mapping in a 2D navigation demonstration [Figures 2a and 2b]. The environment shares a similar visual structure to the demonstration in the original IODA work, but our agent (circle) was trained with Q-learning in a discrete action space. The environment contains a central ID region (gray) and a surrounding OOD region (white). To create OOD conditions, the agent was trained for every discrete location within the 5×15 gray workspace. The white region was held out during training so that the agent has no learned policy for those states. The agent's goal position (top, green rectangle) is set to be straight across from the starting position (bottom, blue rectangle).

During the demonstration, the user controls the horizontal direction of the policy to bring the agent from the ID region (gray) to the OOD region (white). When in the OOD region, IODA maps the agent to actions based on the selected ID state. For this demonstration, when using TC mapping, the candidate set for any OOD state $\mathbf{s}_t$ was restricted to all ID states with the same vertical coordinate as the agent. This set is used to minimize the cost function C and select the ID state.

To illustrate the performance of the TC mapping compared to Euclidean mapping, a policy failure was simulated at a specific ID location (orange, left square), where an unpredictable action was injected into the policy causing the agent to go down, away from the goal. In response to the injected policy failure, the Euclidean-distance mapping selects the ID state that leads to the downward action (orange arrow) [Figure 2a]. However, the TC mapping selects an ID state (pink, middle square) maintaining continuous, upward motion (pink, arrow) [Figure 2b]. This demonstration shows preliminary evidence that TC mapping creates the trajectory alignment necessary to create continuous, predictable motions.

In the TC mapping for this demonstration, we used min-max scaling to normalize d_{euclid}, ensuring comparable scales between d_{euclid} and d_{traj}. Additionally, we divided d_{traj} by 2 because it is bounded in $[0, 2]$. This produces values for d_{euclid} and d_{traj} that are bounded in $[0, 1]$. In our demonstration we set $\alpha = 0.8$, giving d_{traj} more importance over d_{euclid}. Sensitivity analysis indicated that values of $\alpha > 0.3$ provided stable projections to candidate states.



(a) Euclidean-Distance Mapping **(b) TC Mapping**

Figure 2: In a 2D navigation task with a simulated policy failure point (orange, left square), a user controls the x-axis of the agent (circle) while the policy controls the y-axis. The gray region indicates ID states, and the white region indicates OOD states. (a) Euclidean distance selects a nearby state (orange, left square) leading to a downward action (orange arrow). (b) TC mapping selects the state (pink, middle square) that leads to an upward, continuous action (pink arrow).

6 Discussion

Our work contributes a TC mapping from OOD to ID states that prioritizes state-change alignment. We hypothesized that the Euclidean-distance mapping does not capture the semantic similarity required to make robot actions predictable to users. We found behaviors supportive to this hypothesis in a 2D navigation demonstration, where the TC mapping selected an ID state leading to continuous forward motion, while the Euclidean-distance mapping selected an ID state leading to discontinuous motion. Given that our analysis was limited to a simulation environment, further evidence is needed for the generalization of the method and for user-perceived predictability. Future work should focus on user validation and generalization, user prediction models, and scalability.

User studies on cases where Euclidean-distance mapping fails in real robotic shared-control scenarios are necessary to evaluate whether the TC mapping leads to tangible improvements in user trust, coordination, and perceived predictability. We propose an experimental setup where a robot is trained to perform a learned policy and used with IODA to complete a novel task, such as the flower watering task used in the original IODA work [11]. Users will control one axis of a robot's motion using the IODA algorithm with both the Euclidean-distance and TC mappings. Likert-scale ratings for expectation alignment can be collected to evaluate trust and predictability, and task performance metrics can be collected to evaluate coordination.

When scaling the TC mapping to real robotic systems, we must address the complexity of calculating the transition function for each candidate ID state. The 2D navigation demonstration selects a subset of ID states using a task-relevant constraint, resulting in a computation cost that scales linearly with the candidate set size. In practice, a candidate set constructed with methods such as geometric nearest neighbors could also limit the computation cost of the transition function. Although candidate selection may still rely on geometric constraints, we mitigate the effects of trajectory discontinuity by selecting the state whose motion best continues the local trajectory. Future work will investigate how the proposed optimization techniques scale with larger state spaces. We will focus on techniques to optimize the search process for optimal ID states and to optimize transition function evaluations.

7 Conclusion

In this paper, we analyzed scenarios where Euclidean distance may not capture user predictions for robot actions in high-dimensional state spaces. When using the IODA algorithm to collaborate with a robotic policy, a Euclidean-distance mapping may fail to capture characteristics of the data that make robot motions predictable, leading to unexpected behaviors. To address this, we introduced a Trajectory-Continuation (TC) mapping. The TC mapping asserts that the action at a selected ID state continues the trend of motion at an OOD state, which is hypothesized to create more predictable and continuous motions.

Acknowledgments

This work was supported in part by the National Science Foundation (IIS-2132887, 2417014).

References

[1] David A. Abbink, Tom Carlson, Mark Mulder, Joost C. F. de Winter, Farzad Aminravan, Tricia L. Gibo, and Erwin R. Boer. 2018. A Topology of Shared Control Systems—Finding Common Ground in Diversity. *IEEE Transactions on Human-Machine Systems* 48, 5 (2018), 509–525. doi:10.1109/THMS.2018.2791570

[2] Charu C. Aggarwal, Alexander Hinneburg, and Daniel A. Keim. 2001. On the Surprising Behavior of Distance Metrics in High Dimensional Space. In *Database Theory — ICDT 2001*, Jan Van den Bussche and Victor Vianu (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 420–434. doi:10.1007/3-540-44503-X_27

[3] Anca D. Dragan, Kenton C.T. Lee, and Siddhartha S. Srinivasa. 2013. Legibility and predictability of robot motion. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 301–308. doi:10.1109/HRI.2013.6483603

[4] Miguel Faria, Francisco S. Melo, and Ana Paiva. 2021. Understanding Robots: Making Robots More Legible in Multi-Party Interactions. In *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*. 1031–1036. doi:10.1109/RO-MAN50785.2021.9515485

[5] Peter A. Hancock, Deborah R. Billings, Kristin E. Schaefer, Jessie Y. C. Chen, Ewart J. de Visser, and Raja Parasuraman. 2011. A Meta-Analysis of Factors Affecting Trust in Human-Robot Interaction. *Human Factors* 53, 5 (2011), 517–527. doi:10.1177/0018720811417254

[6] Guy Hoffman. 2019. Evaluating Fluency in Human–Robot Collaboration. *IEEE Transactions on Human-Machine Systems* 49, 3 (2019), 209–218. doi:10.1109/THMS.2019.2904558

[7] James R Kubricht, Keith J Holyoak, and Hongjing Lu. 2017. Intuitive physics: Current research and controversies. *Trends in cognitive sciences* 21, 10 (2017), 749–759. doi:10.1016/j.tics.2017.06.002

[8] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2018/file/abdeb6f575ac5c6676b747bca8d09cc2-Paper.pdf

[9] Michael McCloskey, Alfonso Caramazza, and Bert Green. 1980. Curvilinear Motion in the Absence of External Forces: Naïve Beliefs About the Motion of Objects. *Science* 210, 4474 (1980), 1139–1141. doi:10.1126/science.210.4474.1139

[10] Stefanos Nikolaidis, David Hsu, and Siddhartha Srinivasa. 2017. Human-robot mutual adaptation in collaborative tasks: Models and experiments. *The International Journal of Robotics Research* 36, 5-7 (2017), 618–634. doi:10.1177/0278364917690593

[11] Isaac Sheidlower, Emma Bethel, Douglas Lilly, Reuben M. Aronson, and Elaine Schaertl Short. 2024. Imagining In-distribution States: How Predictable Robot Behavior Can Enable User Control Over Learned Policies. In *2024 33rd IEEE International Conference on Robot and Human Interactive Communication (ROMAN)*. 1308–1315. doi:10.1109/RO-MAN60168.2024.10731233

[12] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Oriolo. 2009. *Trajectory Planning*. Springer London, London, 161–189. doi:10.1007/978-1-84628-642-1

Received 2025-12-08; accepted 2026-01-12