SynIL: Leveraging Synergy for Offline Imitation Learning from Imperfect Demonstration Datasets

Anonymous authors
Paper under double-blind review

Abstract

Imitation learning has undergone significant evolution with the advent of deep learning. Deep neural networks enable the learning of complex policies directly from demonstrations, without relying on traditional handcrafted feature approaches. However, deep learning-based imitation learning requires high-quality demonstrations, as the policies are directly trained on data. Thus, factors such as task difficulty and expert proficiency can lead to contamination of non-optimal demonstrations in reality, resulting in decreased performance. Previous studies have explored methods for evaluating the quality of demonstrations to identify optimal samples of demonstration. However, they often require the hand-selection of expert data in advance, which becomes increasingly challenging as datasets grow larger. This study proposes a method for automated evaluation of demonstration quality based on synergy, a low-dimensional structure observed in biological systems. The proposed method quantifies the degree of synergy manifestation as rewards to perform offline reinforcement learning. Since synergy is reported to relate to proficiency, we expect this to work as an indicator of the motion quality of demonstrations. Results demonstrated that the synergy-based rewards correlated with true rewards, and synergy-based imitation learning outperformed behavior cloning and even offline reinforcement learning with true rewards in some cases. Those results will offer a new framework for enhancing imitation learning systems with demonstrations in which not every sample is optimal. The proposed method will also contribute to computational neuroscience, as well as robotics and machine learning.

1 Introduction

Imitation learning, a method for mimicking expert demonstrations, has gained attention as a new learning paradigm that can potentially outperform manual programming requiring considerable time and a high level of expertise in coding and control. Unlike traditional manual programming methods, imitation learning is a data-driven approach, allowing learning without the need for complex design, programming skills, and specialized knowledge. Imitation learning is highly regarded as a more realistic approach in fields such as robotics, healthcare, and autonomous driving (Kim et al., 2024; Pan et al., 2017). In the field of robotics, imitation learning has successfully automated numerous actions in various robots, including Mobile ALOHA (Fu et al., 2024). Recent robotic foundation models are also based on the framework of imitation learning(Kawaharazuka et al., 2025).

In spite of its potential, the performance of imitation learning is adversely affected by low quality teaching data. A large amount of optimal demonstration data is required to ensure the stable performance of imitation learning. However, due to factors such as the difficulty of the task, the skill level of the expert, and issues like fatigue and concentration when collecting demonstration data, the data obtained may include suboptimal data. As the dataset grows larger, the likelihood of including incomplete data and noise increases. Presence of suboptimal or poor-quality demonstrations gets imitation learning to mimic these poor-quality demonstrations, resulting in decreased performance.

Evaluating the quality of demonstrations will overcome the above issue. For example, DemoDICE (Kim et al., 2022), ISWBC (Li et al., 2024), and DWBC (Xu et al., 2022) pre-divide the training data into optimal data D_E from experts and D_U with unknown proficiency, enhancing imitation learning performance by focusing more on expert data or adjusting the distributional distance with non-optimal data. Additionally, CLUE (Liu et al., 2023) and ORIL (Zolna et al., 2020) address this issue by converting distributional distances with optimal data into rewards and conducting offline reinforcement learning, which considers these rewards. This approach has the potential to enable to learn behaviors beyond the (possibly poor) quality of the demonstrations. However, in all these studies, it is necessary for humans to pre-determine the expert data. Furthermore, when it comes to qualitative evaluation, it may be challenging to discern which instructional actions are optimal depending on the task. Therefore, a method to automatically and quantitatively evaluate optimality is needed.

To realize automated demonstration-quality assessment, we focus on the concept of synergy, which is proposed in neuroscience. Synergy refers to the low-dimensional structure of coordinated muscles and joints, which are often observed in human proficient movements (Latash et al., 2014; Turvey, 1990). Many studies have reported that the degree of synergy manifestation corresponds to the proficiency of movements (Zaal et al., 1999; Gentner et al., 2010). This phenomenon has also been observed from reinforcement learning in locomotion (Chai & Hayashibe, 2020) and reaching movements (Han et al., 2021). Synergies and their degree of manifestation can be obtained algorithmically. Thus, we expect that synergy can be used for automated demonstration-quality assessment.

In this study, we propose a method of imitation learning with automatic assessment of demonstration quality. We propose a method to compute rewards from unlabelled demonstrations based on synergy and use them in offline reinforcement learning. Our contributions are as follows:

- We propose a method to algorithmically compute the quality of demonstrations based on synergy and use it as reward for learning.
- This study is the first application of the concept of synergy to assess the quality of demonstration movements in imitation learning.
- We successfully improved imitation learning performance across various datasets, including real human demonstrations.

2 Methodology

2.1 Overview

This section describes the proposed method, SynIL, which derives its name from synergy-based imitation learning. This method allows a policy to imitate demonstrations by performing offline reinforcement learning based on rewards computed from the degree of synergy manifestation. In SynIL, demonstrations do not have to include rewards; instead, rewards are computed from demonstrations using synergy, a concept proposed in neuroscience that quantifies the degree of movement coordination. Unlike most conventional methods, SynIL does not require manual evaluation of demonstrations, except for setting several hyperparameters.

An overview of SynIL is provided in Figure 1. First, the degree of synergy manifestation is quantified as a synergy score ξ. Next, the synergy scores ξ are converted into rewards by using self-supervised reward regression (SSRR) (Chen et al., 2021). Finally, offline reinforcement learning is performed using the synergy-based rewards associated with state-action pairs.

2.2 Acquisition of Synergy Scores based on Synergy Manifestation

As the first step of SynIL, the quality of demonstration is quantified based on synergy. Among various types of synergy, we employ the spatial synergy, which represents low-dimensional coordination patterns in multi-dimensional spatial data. The spatial synergy is

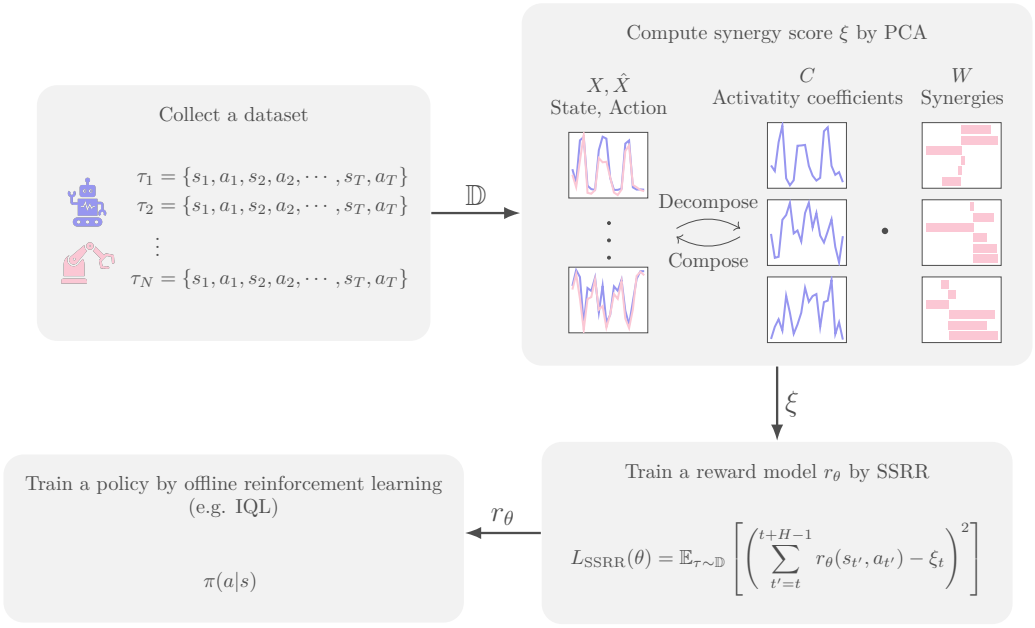

Figure 1: Overview of SynIL. Synergy scores ξ are calculated from trajectories of a dataset $\mathbb{D}$ (top-right), and a reward model is trained based on the synergy scores (bottom-right). Offline reinforcement learning is finally performed with rewards computed by the trained reward model (bottom-left).

defined by the following equation:

$$\boldsymbol{X}_t = \boldsymbol{W}_t^n \boldsymbol{C}_t^n + \bar{\boldsymbol{X}}_t + \text{residuals}, \tag{1}$$

where

$$\boldsymbol{X}_t = \begin{bmatrix} \boldsymbol{s}_t & \boldsymbol{s}_{t+1} & \cdots & \boldsymbol{s}_{t+(H-1)} \\ \boldsymbol{a}_t & \boldsymbol{a}_{t+1} & \cdots & \boldsymbol{a}_{t+(H-1)} \end{bmatrix}. \tag{2}$$

Here, data $\boldsymbol{X}_t$ is the H-step trajectory of the state-actions cut from a demonstraion, n is the number of spatial synergies, $\boldsymbol{W}_t^n$ corresponds to the spatial synergies, and $\boldsymbol{C}_t^n$ represents the activation level of the spatial synergies at each time. Also, $\bar{\boldsymbol{X}}_t$ incidates the row-wise mean values of $\boldsymbol{X}_t$. Equation 1 is computed by the principle component analysys (PCA), which computes $\boldsymbol{W}_t^n$ and $\boldsymbol{C}_t^n$ to minimize residuals. It can be said that the smaller the residuals are, the more significant the spatial synergies are manifest.

Based on Equation 1, the degree of synergy manifestation is quantified by a metric denoted by $(R^2)_t^n$, as follows:

$$(R^2)_t^n = 1 - \frac{\|\boldsymbol{X}_t - \bar{\boldsymbol{X}}_t - \boldsymbol{W}_t^n \boldsymbol{C}_t^n\|_F^2}{\|\boldsymbol{X}_t - \bar{\boldsymbol{X}}_t\|_F^2}. \tag{3}$$

Here, $\| \bullet \|_F$ represents the Frobenius norm. $(R^2)_t^n$ ranges from 0 to 1, and the larger the value of (R^2) is, the higher the degree of synergy manifestation is.

However, the above $(R^2)_t^n$ metric varies depending on the number of synergies, n. Instead, this study uses the synergy level(Chai & Hayashibe, 2020), ζ_t, as an indicator of synergy manifestation independent of n, defined as follows:

$$\zeta_t = \frac{1}{N} \sum_{n=1}^{N} (R^2)_t^n, \tag{4}$$

where N indicates the degree of freedom of demonstrations. Note that $(R^2)_t^0 \equiv 0$, and $(R^2)_t^n \equiv 1$ for $n \geq N$, so the synergy score sums up with a range of $n = 1, \ldots, N$.

In addition to the synergy level, ζ, we also consider the standard deviation of the demonstrations. This is because only use of the synergy level can lead to the problem of high evaluation of demonstrations with overly simplistic states and actions, such as a demonstration with no movement. It is undesirable in many motor tasks to highly valuing the movement of doing nothing. To address this issue, we use the following synergy score, ξ_t, that considers the standard deviation of state-action pairs as well as the synergy level:

$$\xi_t = \log \zeta_t + \eta \log \bar{\sigma}_t. \tag{5}$$

Here, $\bar{\sigma}_t$ is the mean value of the standard deviation of columns of $\boldsymbol{X}_t$. η is a hyperparameter to weight the standard deviation of the demonstration action.

2.3 Reward Estimation Based on Synergy

As shown in Equations 2–5, the synergy score ξ_t are computed from an H-step state-action trajectory. However, to use offline reinforcement learning, we need rewards associated with each state-action pair. To bridge this gap, we consider that the synergy score ξ_t, computed from an H-step state-action trajectory, approximates the total reward of the H-step state-action trajectory, as expressed in the following equation:

$$\xi_t \approx \sum_{t'=t}^{t+H-1} r_{t'}, \tag{6}$$

where $r_{t'}$ indicates a reward at time t'.

To acquire such rewards, we construct a reward model that maps a state-action pair to a reward value. Let a reward model be $r_\theta(\boldsymbol{s}_t, \boldsymbol{a}_t)$, where $\boldsymbol{s}_t$ and $\boldsymbol{a}_t$ denote a state and action at time t, respectively, and θ indicates the parameters of the reward model. The reward model is trained to minimize the following loss function:

$$L_{\mathrm{SSRR}}(\theta) = \mathbb{E}_{\tau \sim \mathbb{D}} \left[\left(\sum_{t'=t}^{t+H-1} r_\theta(s_{t'}, a_{t'}) - \xi_t \right)^2 \right]. \tag{7}$$

Minimizing this loss function gets the reward model to generate rewards which satisfy Equation 6. This method is similar to the self-supervised reward regression (SSRR) (Chen et al., 2021), with the noise level replaced with the synergy score.

3 Evaluation

3.1 Dataset

To evaluate the proposed method, we used D4RL(Fu et al., 2020) and Robomimic(Mandlekar et al., 2021) datasets as demonstration datasets. D4RL is a dataset for imitation learning and offline reinforcement learning, offering datasets for various environments and problem settings. Among various tasks, we specifically used the locomotion dataset as a typical motor task. The locomotion dataset includes three types of robots: Hopper, HalfCheetah, and Walker2d. D4RL contains several types of datasets with varying quality in terms of the performance and rewards of the expert policies. The "medium" dataset contains demonstrations with moderate quality, the "medium-replay" dataset is a mixed dataset of low to moderate quality demonstrations, and the "medium-expert" dataset is also a mix dataset of medium to high quality demonstrations. Each dataset was generated by policies at different training stages of the reinforcement learning algorithm, soft actor-critic(Haarnoja et al., 2018). These datasets are suitable for our problem setting, as they involve varying-quality demonstrations. It is known that typical imitation learning, such as behavior cloning (BC), decreases the performance with these types of datasets.

Robomimic, on the other hand, is an imitation-learning dataset with situations similar to real-world applications. The dataset is composed of demonstraions in various tasks with a seven degrees-of-freedom manipulator with a gripper. The demonstrations have been acquired by human operators with tele-operation in a simulator. Among datasets in

Robomimic, we used the multi-human (MH) datasets of Can and Square tasks. The MH dataset contains a total of 300 demonstrations, with 50 collected from each of six individuals with varying levels of expertise.

3.2 Problem Setting

In the evaluation, we assume a dataset can be expressed as follows:

$$\mathbb{D} = \{(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}) | t = 0, \ldots, T-1\}, \tag{8}$$

where $\boldsymbol{s}_t$ and $\boldsymbol{a}_t$ indicates a state and action at time t. We assume that the dataset does not have rewards or any sort of quality values for each tuple of $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1})$. Although the D4RL datasets contains rewards, we excluded them to emulate the sitations where only demonstrations were provided. Thus, reinforcement learning methods cannot be applied unless we compute reward values.

We compared the proposed method with BC, a common imitation learning method. Additionally, for reference, we also compared them with the implicit Q-learning (IQL)(Kostrikov et al., 2021) with datasets containing the true rewards. This comparison was conducted to discuss how our method, which uses estimated rewards, is comparable with offline reinforcement learning with true rewards. The algorithm for SynIL is presented in Algorithm 1. Detail hyperparameters are provided in Appendix.

Algorithm 1 SynIL

Require: dataset $\mathbb{D}$ of state-action trajectories, as defined in Equation 8
Ensure: a policy π and a reward model r_θ
  for each trajectory $\tau \sim \mathbb{D}$ do
    Compute ζ_t and $\bar{\sigma}_t$ from H-step trajectories cut from τ by Equation=4
    Compute ξ_t by Equation 5
  end for
  Initialize reward function parameters θ
  for each epoch do
    for each trajectory τ in $\mathbb{D}$ do
      Compute loss $\mathcal{L}_{\mathrm{SSRR}}(\theta)$ by Equation 7
      Update θ to minimize $\mathcal{L}_{\mathrm{SSRR}}(\theta)$ through gradient descent
    end for
  end for
  Train a policy π by offline reinforcement learning with the trained reward model r_θ

3.3 Results

3.3.1 Correlation between Estimated Reward and Ground-Truth Reward

First, we investigated the relationship between the synergy-based rewards and ground-truth rewards in D4RL datasets. Figure 2 shows scatter plots of the rewards estimated by the SSRR (vertical axis, cumulative reward over 200 steps) and true rewards (horizontal axis, cumulative reward over 200 steps). Additionally, Table 1 presents the correlation coefficients between the rewards estimated by the SSRR and true rewards.

As a result, the correlation coefficients were more than 0.8 for all datasets except for the walker2d-medium-v2 and hopper-medium-replay-v2. Particularly, the correlation coefficients were more than 0.86 in all medium-expert datasets.

3.3.2 Performance Comparison

We evaluated the performance of imitation learning of SynIL (the proposed method), BC, and IQL in D4RL datasets. Table 2 shows the performance, which was computed as the average sum of rewards over 10 episodes. The values were normalized according to Fu et al. (2020).

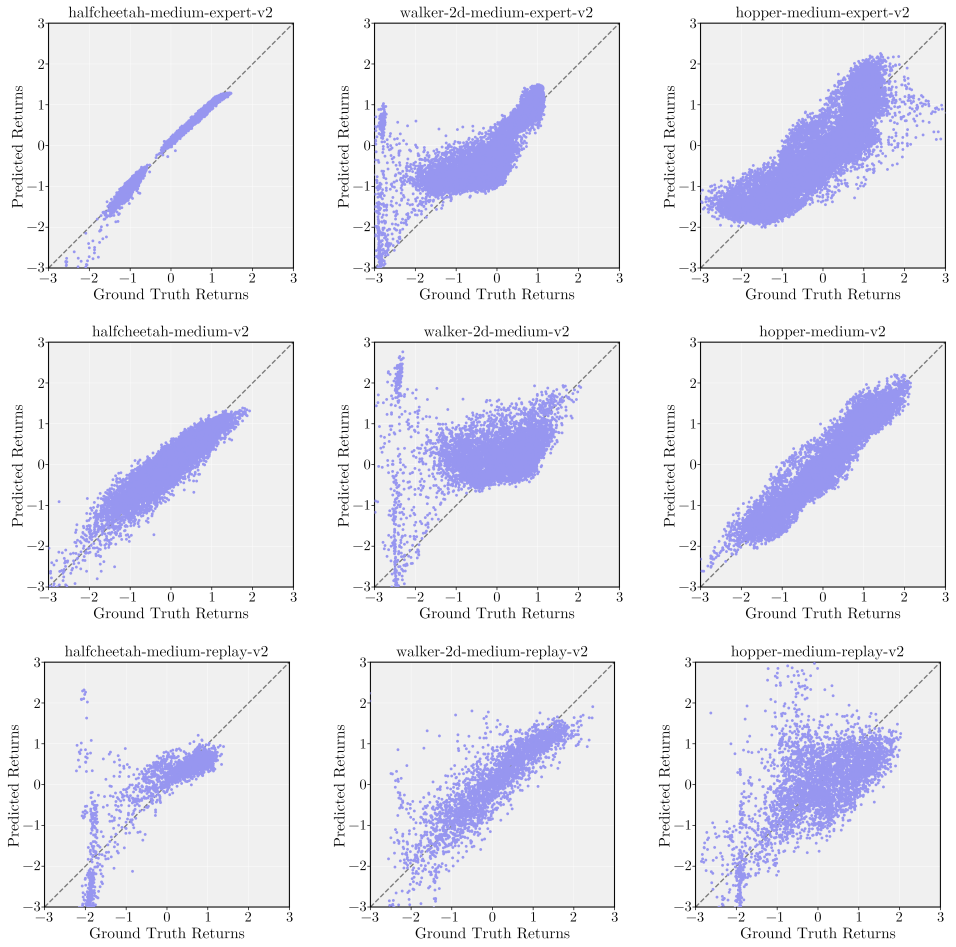

Figure 2: Comparison of synergy-based/true rewards for different datasets in D4RL. The vertical axes correspond to the cumulative rewards over 200 steps estimated by SSRR, and the horizontal axes correspond to the true cumulative rewards over 200 steps.

Table 1: Correlation coefficients between the synergy-based and true rewards in D4RL

Dataset	Correlation coefficient
halfcheetah-medium-expert-v2	0.9959
walker2d-medium-expert-v2	0.8655
hopper-medium-expert-v2	0.8718
halfcheetah-medium-v2	0.9482
walker2d-medium-v2	0.5809
hopper-medium-v2	0.9526
halfcheetah-medium-replay-v2	0.8039
walker2d-medium-replay-v2	0.8263
hopper-medium-replay-v2	0.5314

Table 2: Performance of each algorithm in D4RL (Average sum of rewards over 10 episodes)

Dataset	BC	SynIL (Ours)	IQL
halfcheetah-medium-v2	42.7 ± 0.3	$\mathbf{45.4 \pm 0.2}$	47.4 ± 0.2
hopper-medium-v2	$\mathbf{54.4 \pm 1.7}$	46.6 ± 3.8	54.4 ± 3.5
walker2d-medium-v2	71.7 ± 6.3	$\mathbf{72.0 \pm 5.4}$	80.0 ± 4.9
halfcheetah-medium-replay-v2	37.1 ± 2.4	$\mathbf{40.6 \pm 1.4}$	44.3 ± 0.5
hopper-medium-replay-v2	14.4 ± 2.4	$\mathbf{99.9 \pm 0.5}$	66.5 ± 8.1
walker2d-medium-replay-v2	15.5 ± 6.4	$\mathbf{74.5 \pm 3.0}$	69.5 ± 7.9
halfcheetah-medium-expert-v2	60.9 ± 5.0	$\mathbf{90.9 \pm 1.3}$	87.0 ± 3.1
hopper-medium-expert-v2	51.4 ± 1.8	$\mathbf{80.9 \pm 8.0}$	2.2 ± 0.0
walker2d-medium-expert-v2	80.0 ± 5.8	$\mathbf{108.9 \pm 0.3}$	110.7 ± 0.5

Table 3: Performance comparison in Robomimic (average success rate, 3 seeds $\times$ 50 rollouts)

Dataset	BC	SynIL (Ours)	IQL (sparse rewards)
Can (MH)	56.7 ± 10.0	$\mathbf{94.0 \pm 3.3}$	67.3 ± 5.0
Square (MH)	24.0 ± 4.3	$\mathbf{49.3 \pm 8.2}$	16.7 ± 5.2

As a result, according to Table 2, SynIL outperformed BC on most datasets and was comparable to IQL with true rewards. This is particularly evident in datasets of varying quality such as medium-replay and medium-expert. On the other hand, BC, which performs imitation by supervised learning, shows decreased performance on datasets with diverse quality like medium-expert.

We then evaluated the performance of imitation learning in Robomimic environments. Table 3 presents the performance, which was evaluated as the mean success rates for 50 trials across three different initial values (i.e. a total of 150 trials). Table 4 shows the execution time in seconds. Note that, since the Robomimic datasets did not contain reward data, we used sparse rewards, where a reward of 1 was applied at the final steps of successful episodes, and 0 was applied otherwise.

As a result, according to Table 3, SynIL significantly outperformed both BC and even IQL. Additionally, from Table 4, it is evident that SynIL had a shorter execution time than both BC and IQL.

Learning curves in D4RL and Robomimic are shown in Figures 3 and 4.

4 Discussion

4.1 Effectiveness of Synergy Rewards in Imitation Learning

The results suggest that the synergy-based rewards, calculated from the degree of synergy manifestation, can be used as surrogate rewards quantifying the quality of demonstrations. The synergy-based rewards have high correlation with the ground-truth rewards, as shown in Tables 1. Also, the use of the synergy-based rewards resulted in high performance in imitation learning, as shown in Tables 2, and 3. Those results demonstrate the effectiveness of the proposed method, that is, the use of synergy for evaluating the quality of demonstrations.

Table 4: Execution time in Robomimic (seconds)

Dataset	BC	SynIL (Ours)	IQL (sparse rewards)
Can (MH)	2.1 ± 0.2	$\mathbf{1.3 \pm 0.0}$	2.2 ± 0.2
Square (MH)	2.9 ± 0.0	$\mathbf{2.0 \pm 0.1}$	3.4 ± 0.2

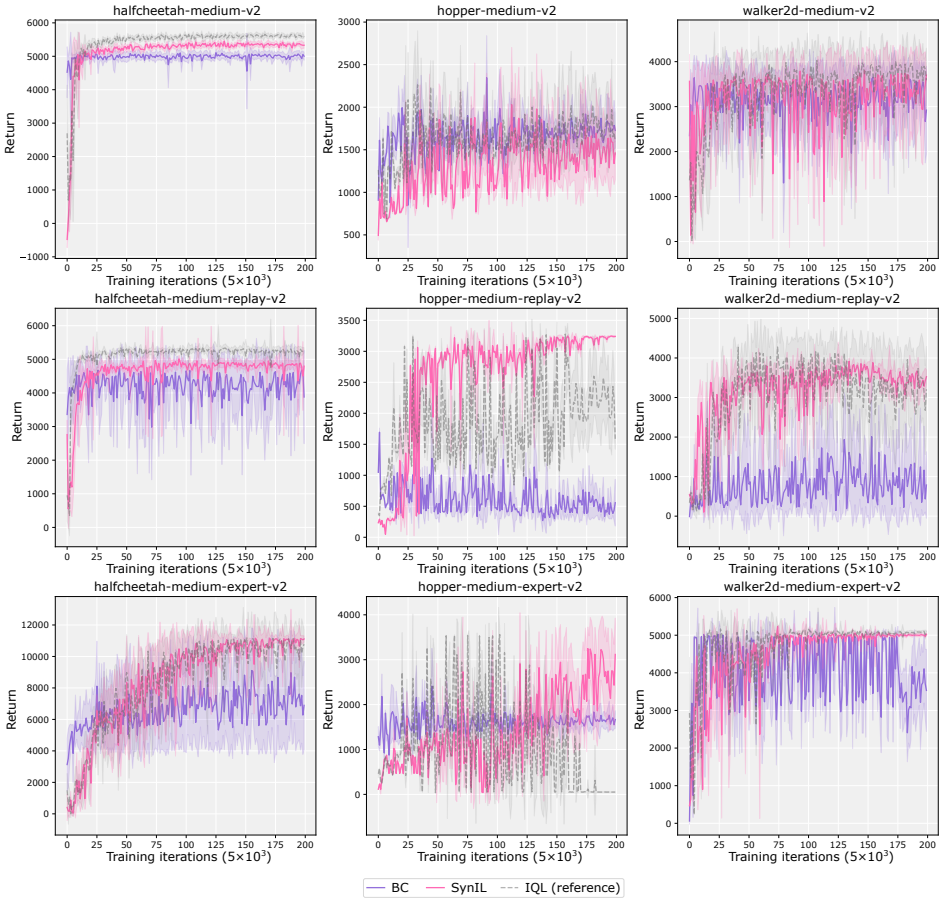

Figure 3: Learning curves of all environments.

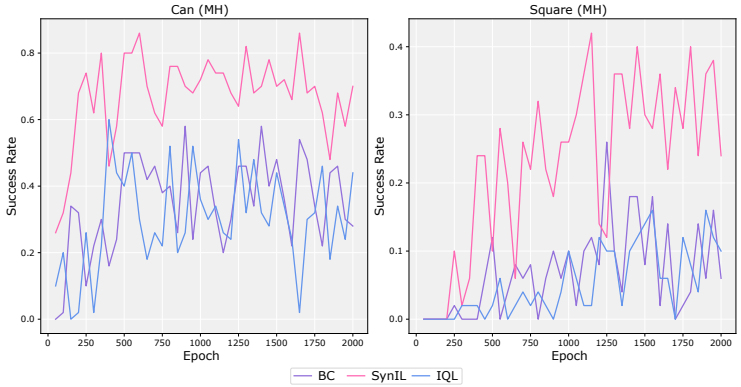

Figure 4: Learning curves of Robomimic.

The use of synergy-based rewards even outperformed IQL with ground-truth rewards in some cases of D4RL and in Robomimic. This is because the ground-truth rewards we used in Robomimic were sparse, which was hard to train a well-performing policy. The synergy-based rewards, in contrast, could densely provide more information and function as an important guide for successful completion in the task. This also highlights the advantage of the synergy-based rewards that densely evaluate movement coordination.

4.2 Relationship Between Motor Learning and Synergy

The results shown in Tables 2 and 3 suggest that the synergy-based reward was effective in improving the performance of imitation learning. Here, although it has not fully been proven, we discuss why the synergy worked well in D4RL and Robomimic.

The success of the synergy-based method in D4RL would be caused by the nature of reinforcement learning. The demonstraions in D4RL datasets have been generated by neural-network policies trained by deep reinforcement learning. In reinforcement learning, an agent initially takes random state-action pairs, but as learning progresses, it takes more regular states and actions. During this learning process, the dimensionality of the state-action distribution decreases. This links to the increase of synergy manifestation. Therefore, learning state-action relationship where synergy is highly manifested will result in the selective learning of more proficient behaviors. This would be a reason why the proposed method resulted in a high performance in the D4RL datasets. A similar phenomenon has been reported in Chai & Hayashibe (2020); the degree of synergy manifestation increases as deep reinforcement learning progresses. In future work, these findings may lead to a new reinforcement learning framework that prefers to learn coordinated movements to accelerate motor learning. Moreover, previous works have reported that synergies obtained in some tasks can generalize to new tasks(Al Borno et al., 2020; Kutsuzawa & Hayashibe, 2022). These findings also suggest that the combination of synergy and reinforcement learning can improve the ability of motor learning.

On the other hand, the reason of the success of the synergy-based method in Robomimic would relate to the nature of the human motor learning. It is known that many human movements become more regular, stable, and low-dimensional as learning progresses (Davids et al., 2012). Even in unpredictable tasks, diverse actions are initially taken, but ultimately they converge to typical patterns of adaptive responses due to the principle of optimality (Braun et al., 2009). In such cases, it is considered that synergy in state actions also manifests.

In addition to the above, the effectiveness of synergy can also be considered from the perspective of computational neuroscience. A recent theory in computational neuroscience argues that human's central nervous systems perform recognition and action generation so as to minimize the variational free energy (Friston & Stephan, 2007). Minimizing the variational free energy over time requires to decrease the amount of uncertainty, resulting in reduction of the entropy of the sensory signals (Friston, 2012). Reduction in the entropy will bring about more coordinated distributions of the sensory signals and actions, resulting in the increase of the degree of synergy manifestation. Therefore, it is possible that as a human becomes more proficient in a task, the variational free energy decreases more efficiently, thereby the synergy is highly manifested.

From these perspectives and the results of this study, it is suggested that the synergy score indicates the optimality of behaviors.

5 Conclusion

In this study, we aimed to address the issue in imitation learning that the performance degrades due to varying qualities of demonstrations. We proposed a method of computing the quality of demonstrations based on synergy, enabling to use offline reinforcement learning. We expected that synergy, a low-dimensional structure related to skill proficiency, can be used to improve imitation learning performance. As a result, we reported that the synergy-based imitation learning method outperformed the behavior cloning, a supervised

learning-based method. Although the relationship between synergy and demonstration quality has not yet been fully proven, we demonstrated the effectiveness of the use of synergy across various datasets. We expect that our findings contribute to not only imitation learning but robotics and even computational neuroscience. Future work will focus on further clarifying the scope of this method's effectiveness and deepening the understanding of the relationship between synergy and skill proficiency, potentially revealing the mechanisms of skill acquisition in humans and other organisms.

References

Mazen Al Borno, Jennifer L. Hicks, and Scott L. Delp. The effects of motor modularity on performance, learning and generalizability in upper-extremity reaching: a computational analysis. Journal of The Royal Society Interface, 17(167):20200011, June 2020. doi: 10.1098/rsif.2020.0011.

Daniel A Braun, Ad Aertsen, Daniel M Wolpert, and Carsten Mehring. Learning optimal adaptation strategies in unpredictable motor tasks. Journal of Neuroscience, 29(20): 6472–6478, 2009.

Jiazheng Chai and Mitsuhiro Hayashibe. Motor synergy development in high-performing deep reinforcement learning algorithms. IEEE Robotics and Automation Letters, 5(2): 1271–1278, 2020.

Letian Chen, Rohan Paleja, and Matthew Gombolay. Learning from suboptimal demonstration via self-supervised reward regression. In Conference on robot learning, pp. 1262–1277. PMLR, 2021.

Keith Davids, Duarte Araújo, Robert Hristovski, Pedro Passos, and Jia Yi Chow. Ecological dynamics and motor learning design in sport. Skill acquisition in sport: Research, theory and practice, 2:112–130, 2012.

Karl Friston. A free energy principle for biological systems. Entropy, 14(11):2100–2121, 2012.

Karl J Friston and Klaas E Stephan. Free-energy and the brain. Synthese, 159:417–458, 2007.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. arXiv preprint arXiv:2004.07219, 2020.

Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. arXiv preprint arXiv:2401.02117, 2024.

Reinhard Gentner, Susanne Gorges, David Weise, Kristin aufm Kampe, Mathias Buttmann, and Joseph Classen. Encoding of motor skill in the corticomuscular system of musicians. Current Biology, 20(20):1869–1874, 2010.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In International conference on machine learning, pp. 1861–1870. PMLR, 2018.

Jihui Han, Jiazheng Chai, and Mitsuhiro Hayashibe. Synergy emergence in deep reinforcement learning for full-dimensional arm manipulation. IEEE Transactions on Medical Robotics and Bionics, 3(2):498–509, 2021.

Kento Kawaharazuka, Jihoon Oh, Jun Yamada, Ingmar Posner, and Yuke Zhu. Vision-language-action models for robotics: A review towards real-world applications. IEEE Access, pp. 1–1, 2025. doi: 10.1109/ACCESS.2025.3609980.

Geon-Hyeong Kim, Seokin Seo, Jongmin Lee, Wonseok Jeon, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Demodice: Offline imitation learning with supplementary imperfect demonstrations. In International Conference on Learning Representations, 2022.

JW Kim, TZ Zhao, S Schmidgall, A Deguet, M Kobilarov, C Finn, and A Krieger. Surgical robot transformer (srt): Imitation learning for surgical tasks. arxiv 2024. arXiv preprint arXiv:2407.12998, 2024.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. arXiv preprint arXiv:2110.06169, 2021.

Kyo Kutsuzawa and Mitsuhiro Hayashibe. Motor synergy generalization framework for new targets in multi-planar and multi-directional reaching task. Royal Society Open Science, 9(5):211721, 2022.

Mark L Latash, Nicholai A Bernstein, and Michael T Turvey. Dexterity and its development. Psychology Press, 2014.

Ziniu Li, Tian Xu, Zeyu Qin, Yang Yu, and Zhi-Quan Luo. Imitation learning from imperfection: Theoretical justifications and algorithms. Advances in Neural Information Processing Systems, 36, 2024.

Jinxin Liu, Lipeng Zu, Li He, and Donglin Wang. Clue: Calibrated latent guidance for offline reinforcement learning. In Conference on Robot Learning, pp. 906–927. PMLR, 2023.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In arXiv preprint arXiv:2108.03298, 2021.

Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Agile autonomous driving using end-to-end deep imitation learning. arXiv preprint arXiv:1709.07174, 2017.

Michael T Turvey. Coordination. American psychologist, 45(8):938, 1990.

Haoran Xu, Xianyuan Zhan, Honglei Yin, and Huiling Qin. Discriminator-weighted offline imitation learning from suboptimal demonstrations. In International Conference on Machine Learning, pp. 24725–24742. PMLR, 2022.

Frank TJM Zaal, Kristin Daigle, Gerald L Gottlieb, and Esther Thelen. An unlearned principle for controlling natural movements. Journal of Neurophysiology, 82(1):255–259, 1999.

Konrad Zolna, Alexander Novikov, Ksenia Konyushkova, Caglar Gulcehre, Ziyu Wang, Yusuf Aytar, Misha Denil, Nando de Freitas, and Scott Reed. Offline learning from demonstrations and unlabeled experience. arXiv preprint arXiv:2011.13885, 2020.

Table 5: Hyperparameters for SSRR

	Hyperparameter	Value in D4RL	Value in Robomimic
RewardNet	Hidden layers	2	2
	Hidden Size	512	1024
	Activation	ReLU	ReLU
	Last Activation	Linear	Linear
	Epochs	500	10000
	Batch Size	64	64
	Learning Rate	1×10^{-4}	1×10^{-3}
	Weight Decay	1×10^{-2}	0
	Criterion	MSELoss	MSELoss
	Optimizer	Adam	Adam
Synergy Score	Horizon H	200	20
	Stride S	100	20
	η	0.05	0.05

Table 6: Hyperparameters for imitation learning

Hyperparameter	SynIL and IQL	BC
Hidden size	256	256
Hidden layers	2	2
Activation	ReLU	ReLU
Optimizer	Adam	Adam
Epochs	10^6	10^6
Evaluate period	5000	5000
Batch size	256	256
Learning rate	3×10^{-4}	3×10^{-4}
Discount	0.99	-
α	0.005	-
τ	0.7	-
β	3.0	-

A Appendix

A.1 Hyperparameters

The neural network and Synergy Score hyperparameters used in the SSRR are listed in Table 5. The input dimension of the reward estimation model (RewardNet) was equal to the sum of the state and action dimensions, and the output dimension was one-dimensional since it represents the reward. H and S indicate the horizon and stride; H-step trajectories are cut from a demonstration while shifting the start position by S-steps. Additionally, η was set through hand-tuning to achieve the highest correlation with the true reward.

Hyperparameters for SynIL, IQL, and BC are shown in Table 6. IQL becomes equivalent to the BC algorithm by setting the hyperparameter β to 0. Therefore, in D4RL, BC experiments were conducted by setting IQL's β to 0. The hyperparameters were hand-tuned to achieve the highest performance for each algorithm.