# A meta unit for co-constructing a computational scaffold model to guide human motor learning

**Alexandra Moringen**
Institute of Data Science
University of Greifswald
17489 Greifswald
moringena@uni-greifswald.de

**Sascha Fleer**
Independent
sascha.fleer@gmail.com

**Kristina Yordanova**
Institute of Data Science
University of Greifswald
17489 Griefswald

## Abstract

Learning of e.g. a dexterous motor skill, which is common in medical training, sports, or playing a musical instrument, is time-costly and needs supervision from a teacher, to avoid injury and to progress. Importantly, learning a motor skill requires active practice, and adjusting one's own individual embodiment to perform the target movement. We introduce a meta unit for co-constructing a computational scaffold for learning. It targets to optimize the learning process of a student, as well as the intervention of their teacher. This approach aims to enable the student to discover their own optimal learning policy while being guided by the teacher on demand, with the ultimate goal of improving beyond the capacity of the teacher.

## 1 Introduction

Learning of e.g. a dexterous motor skill, which is common in medical training, sports, or playing a musical instrument, is time-costly and requires, on the one hand, supervision from a teacher, to avoid injury and to progress. In most of the above-mentioned scenarios, in particular in medical training and in sports associated with a high risk of injury, it is obligatory to be guided by a teacher, and not sufficient to try out different learning strategies without being led by an expert. On the other hand, learning a complex motor skill requires a lengthy and active practice, training and adjusting one's own individual embodiment[1] to perform the target movement. Parents often observe, how small children insist on self-regulatory learning, and refuse to be helped/guided. They still try to perform the movement, such as eating with a spoon, even though the result is often not robust, as it would be, if guided by the parents. Guidance of babies by their parents is one example in which the teacher and the student strongly differ in embodiment. It is likely that the teacher and the student are rarely characterized by a completely overlapping set of talents, although in this work we do not differentiate between observable and non-observable differences yet. Another prominent example of embodiment differences is in a motor rehabilitation setting. Here in particular, each patient has their own skills that need to be relearned.

Open research questions are: How to adjust the teacher guidance strategy that has been optimized by the teacher to fit their experience and embodiment, to suit the student's own unique talents? How can the student optimally mine their talent and improve *beyond* the performance of the teacher?

The main question that we are interested in exploring is: How to balance self-regulated, exploratory practice, and develop awareness of ones' own talents with teacher-guided practice?

[1] "The word embodied refers to the dual valence of the notion of body: bodiness is a combination of a physical structure (the biological body) and an experiential structure, which corresponds to the living, moving, suffering, and enjoying body. From here it is possible to arrive at the dual acceptation of embodied cognition, which refers, on the one hand, to the grounding of cognitive processes in the brain's neuroanatomical substratum, and on the other, to the derivation of cognitive processes from our organism's sensorimotor experiences". [5]

Accepted at the Intrinsically Motivated Open-ended Learning workshop at NeurIPS 2024.

In this work we propose a self-adapting scaffolding meta unit that uses an *agent* trained with Dyna-Q reinforcement learning (RL) to optimally instruct a simulated student, together with a *coordinator* that manages teacher guidance. This approach aims to find an optimal combination of self-regulatory talent mining of the student and on-demand teacher guidance.

## 2 Related work

Scaffolding for learning [5] is a term that originates from the educational psychology. “Scaffolding is a reciprocal feedback process in which a more expert other (e.g., teacher, or peer with greater expertise) interacts with a less knowledgeable learner, with the goal of providing the kind of conceptual support that enables the learner, over time, to be able to work with the task, content, or idea independently” [5]. While being a rigorously researched topic in educational domains (e.g. [6, 7, 8]), such as mathematics and programming, the domain of computational scaffolding of motor learning is relatively new. The domain has a huge relevance due to its range that stretches from learning musical instruments to relearning during rehabilitation. One of the earlier works [3] describes a setup in which a simulated student is given exercise units, termed "practice modes", that are optimized based on their corresponding utility. A related approach maps both the student state and target parameters of the training to a suitable exercise unit [1]. The model is trained in a supervised manner with the ground truth provided by a teacher and recorded during the training session with the student. This approach is teacher-driven and does not contain any exploratory or self-regulated learning that can be performed by the student by themselves. A common motivation to use a teacher is the acceleration of learning and the prevention of students losing motivation due to poor progress [11]. Another approach uses reinforcement learning to schedule exercises for the student [2] based on the achieved reward and the state of their ability. Here only the agent is used to guide the student and the teacher is not explicitly modeled. Similar to [3], the authors of [4] propose the so-called teachable skills, segments of a motor task that can be selected and recombined to form individual drills for the student. There are common aspects to the above approaches. They aim to assess the capabilities of the student, and to generate an individually tuned sequence of exercise units. The above-mentioned scaffolding principle is inspired by the traditional literature on motor learning with a human teacher (e.g. [9]). Here, it is common for a student to get small exercise units that are derived by simplification or related to the target task. Here we will explore the challenge of embodiment differences between the student and the teacher, and resulting from this, the need to empower the student to learn in self-regulatory manner and mine their own talents, but also get support from the teacher if needed. This work aims at merging the two approaches to computational training, teacher-driven, and the reinforcement learning-driven.

## 3 A self-adapting scaffolding meta unit to guide human learning

When a teacher teaches their student a new skill, the teacher contributes the knowledge of the domain/skill to the learning process, whereas the student contributes the capability to explore their individual talents. During the learning process the teacher tries to guide the student by supporting them by utilizing a set of *exercise units* that enables the learning or deepening of a new skill. There are, however two hindering factors:

**The teacher and student differ in their embodiment** Be it a difference in a physical ability or due to the large experience gap and thus a lack of practice, there is - in most cases - a huge difference how a student or their teacher tackles the task. As a result, the exercise units the teacher chooses for the student might not be tailord to their current skill level and, therefore, too hard or impossible to solve.

**The teacher has to choose the order and the content of the exercise units and when to support the learning process by an educated guess** When to intervene in the student’s self-learning process and how to choose the best exercises for their current skill level are delicate questions the teacher has to answer based on its teaching experience and known educational studies.

Our approach tackles the stated problems by proposing a self-adapting scaffolding meta-unit that manages the interaction of the teacher with the student, while additionally guiding the choice of the students’ exercise units for self-guided study. The agent is inspired by previous work in [2], in which the authors pursue an adaptive and patient-customized therapeutic intervention for rehabilitation of reaching movement. In this setting, the simulated patients interact with a Q-learning algorithm that adjusts the virtual reality (VR) game according to the performance of the patient. Building upon this approach, in our work a state corresponds to the skill level, and an action corresponds to an exercise unit. Optimization criterion is designed to reflect how efficient the student is training, reflected in the speed of improvement in the skill level. Extending upon the work performed in [2], our agent is trained by $\epsilon$-greedy Dyna-Q RL [10] from both the student training process and the teacher guidance data. Dyna-Q blends model-free and model-based RL approaches and is employed due to its known sample efficiency. In our approach, it integrates the teacher and the student embodiments (see next section for a detailed description), and results in a policy that optimizes the improvement rate of the student under this condition. During training of the policy, the teacher guides the student only in the situation when the student's own learning strategy is not successful. Otherwise, the student queries the agent to provide them with an exercise unit (action), given their level of skill (state). The quality of the agents policy is measured via the students *improvement rate*. It is calculated based on how quickly the student moves from one skill state to the next, while only such transitions are associated with a positive reward.

### 3.1 Designing the meta unit

The proposed meta unit combines an *scaffolding agent* for learning the best choice of suitable exercise units with a self-adapting *coordinator* that manages the student-teacher interaction.

The **scaffolding agent** is a Dyna-Q reinforcement learner whose main function is to choose suitable exercise units for the student based on their current skill level. Dyna-Q integrates model-based elements into Q-learning. It has a model-free table $q(s, a)$ for choosing of the action $a^*$ with the highest Q-value, given state $s$; and a model part for simulation, consisting of the transition function $\tau(s, a)$ that maps to $(s', r)$, i.e. for a state $s$ and action $a$ it models the state after transition $s'$ and the corresponding reward $r$. The agent samples its policy and offers the student an exercise unit $a_i$ that either maximizes the Q-value or is random, given the current state of the student and the available policy. According to their improvement rate (described above) and in the case in which the action corresponds to the optimal action defined in $\pi^*_S$, the student performs the action, moves to the next skill level, receives the positive reward $r = 1$. The resulting data $(s_t, a_t, s_{t+1}, r)$ is provided to the agent for training. In the other case, in which the sampled value is below the improvement rate, or the action does not match the ground truth, a reward $r = 0$ is emitted. As the last step, the Dyna-Q agent updates all state-action-reward information that the experimental process yields. The agent is implemented with an $\epsilon$-greedy approach.

The **coordinator** is currently implemented as a Bernoulli distribution, and is directly parameterized by the mean success of the student. This can be formalized by the average reward achieved $\langle R_\mathrm{S} \rangle$ during the previous $t$ steps by the student. An exception is the first $t$ steps. Here, the student is given a chance to learn completely by themselves. The parameter that determines the teacher intervention probability is calculated as $(1 - \langle R_\mathrm{S} \rangle)$ in each step. Thus, the higher the average reward $\langle R_\mathrm{S} \rangle$ of the student, the smaller the probability of the teacher intervention. The smaller the average student reward, corresponding to an unsuccessful self-guided learning, the higher the probability of the teacher intervention. In case the coordinator does not schedule the teacher to intervene and guide the learning, the action is generated by the agent. This enables the student to either learn by themselves or get guidance from the teacher if they seem not to improve quickly enough. The pseudo-code for the training of the meta unit is displayed in appendix B.

In the long-term we aim to transfer the simulation described above to a human study. Each step $t$ is one training session of the exercise unit selected by the scaffolding agent or the teacher. If the student successfully performs the unit, they receive the reward $r = 1$, otherwise $r = 0$.

## 4 Simulation and Environment

We tested our proposed approach by creating a simulated learning environment that is complex enough to grasp the stated problems of teacher-student guidance but is otherwise designed in the most simple way. Therefore, it is possible to set the focus on evaluating the proposed meta unit while not being distracted by a convoluted experimental setup.

**States, actions, rewards** The experimental setup consists of a set of states $S = \{s_1, \ldots, s_N\}$, which represents different skill levels for a task. The goal of the student is to move from the start state (i.e. their initial skill level as a beginner) $s_1 \in S_0$ to the end state $s_N$ (representing the skill level of an intermediate student). In our work we assume that the states are chained to a fixed sequence $s_1, \ldots, s_N$ with length $N$. Thus, only two state-transitions are possible: from $s_i$ to $s_{i+1}$ corresponding to skill improvement, or staying in same state $s_i$, which corresponds to staying on the same skill level. In order to trigger a state transition, the student has a set of actions $A = \{a_1, \ldots, a_M\}$ at its disposal. Each action $a_i$ represents an exercise unit, or another type of training that the student can perform, either under guidance or by themselves. The reward is set to $r = 1$, in case the chosen action results in a transition from the current state $s_i$ to $s_{i+1}$, corresponding to a skill improvement. Otherwise, the reward is $r = 0$.

**Simulated Student** The **student** $P^a_{\vec{s},\vec{s'}}(\text{student})$ is simulated with an improvement probability threshold[2] $\theta$ with $\theta \in [1, 0]$. We sample from the uniform distribution $x \sim \text{Uniform}(0, 1)$ and check if $x > \theta$. If the condition holds the process results in a state transition (either to the next or recursively to the current state) and a corresponding reward. The uniform distribution simulates that even correct practice does not result in an improvement directly.

**Simulated Teacher** The **teacher** has one functionality - to guide the student according to their optimal policy $\pi^*_T$. Both student and teacher share the same set of actions (exercise units). In order to model the differences in embodiment for the teacher and the student, the transition matrices of both are not equal, i.e. $P^a_{\vec{s},\vec{s'}}(\text{teacher}) \neq P^a_{\vec{s},\vec{s'}}(\text{student})$. This leads to different optimal training policies[3] $\pi^*_T$ and $\pi^*_S$ of the teacher and the student respectively.

$$\pi^*_T := \{(s_{T,1}, a_{T,1}), \ldots, (s_{T,N}, a_{T,N})\} \neq \{(s_{L,1}, a_{L,1}), \ldots, (s_{L,N}, a_{L,N})\} := \pi^*_S \quad (1)$$

Thus, the teachers guiding does not always lead to a successful state-transition - i.e. skill improvement - of the student.

**Experiment** During the experiment, the scaffolding agent learns to support the simulated student by choosing the correct sequence of actions to reach the final state $s_N$. Aid for this task is given by the teacher that can help by choosing an action for the student according to their own optimal policy $\pi^*_T$ in which 25% of the optimal actions are different from the optimal actions of the student. In each step, the coordinator decides, based on the average reward analysis, whether the teacher should guide, or the student should query the scaffolding agent to generate an action (exercise unit). Each experiment is repeated 40 times. The results are then averaged. A summary of all parameters can be found in Table 1 (see Appendix A).

## 5 Results and Discussion

We conducted three different experiments. In the first one, the student's learning is guided by the overall meta unit (scaffolding agent and coordinator). In the second experiment, the student is guided by the scaffolding agent only. In the last experiment, the student is guided by the scaffolding agent but receives help randomly from the teacher in 20% of time steps, without the reward analysis performed by the coordinator. The results (see Figure 1 Appendix A) show that the higher number of training episodes performed by the meta unit result in higher improvement rate achieved by the student, which is calculated as an average of the received rewards given the number of learning steps until the end state is achieved. Very early in the process, we observe a saturation. The resulting improvement rate oscillates around the level of 0.3, 0.4 and 0.5 for the scaffolding agent only, added 20% random teacher interventions, and for the meta unit, respectively. The gap in performance between the models grows the less efficiently the simulated student learns, determined by the improvement threshold. Importantly, using the scaffolding-agent without integrating the expert knowledge of a teacher is not applicable in motor learning because of a possible risk of injury.

**Discussion** We presented a method for co-constructing a scaffold to guide the student, with a focus on the differences between the teacher's and student's embodiments. Since we assume that the overlap between the optimal actions of the teacher and student is unknown, our current approach utilizes

[2] The higher the improvement threshold $\theta$ of the student, the more optimal actions lead to zero reward, and therefore, to a need for a higher intervention rate by the teacher.

[3] The policy maps a skill level to an optimal exercise unit.

all available information, i.e. state, transition, action, and reward from both teacher guidance and student actions, to train the scaffolding-agent. In future work, we aim to implement a more advanced training method for the agent, which can identify instances where the teacher's guiding actions do not overlap with the student's optimal actions and exclude such guidance from agent training. The current approach models both the teacher's and the student's embodiment differences in a symmetric way, with non-overlapping policies. The formalism in future work will differentiate between the asymmetric functionalities of the student and the teacher, which can be roughly described as mining of own talents and guiding based on experience, respectively.

For long-term future work, there exists an approach that, to our knowledge, has not yet found realization in the computational motor learning research: we term it *adversarial scaffolding*. This term describes giving exercises to the student that are slightly or more beyond their current capacity, which, therefore, lead the student to get outside their comfort zone. For example, the musicians practice playing a piece in the reverse order, or with an extended rhythmic structure. This is a potentially interesting topic for computational modeling and we will attempt to integrate this idea into the simulated experiment in the next step of our work.

## References

[1] Generating Piano Practice Policy with a Gaussian Process. Alexandra Moringen, Elad Vromen, Helge Ritter, Jason Friedman Proceedings of the 2024 AAAI Conference on Artificial Intelligence, PMLR 257:151-161, 2024.

[2] Personalized rehabilitation approach for reaching movement using reinforcement learning. Pelosi, A.D., Roth, N., Yehoshua, T. et al. Sci Rep 14, 17675 (2024). https://doi.org/10.1038/s41598-024-64514-6

[3] Optimizing piano practice with a utility-based scaffold. Alexandra Moringen, Sören Rüttgers, Luisa Zintgraf, Jason Friedman, Helge Ritter, (2020) https://doi.org/10.48550/arXiv.2106.12937

[4] Assistive Teaching of Motor Control Tasks to Humans. Megha Srivastava, Erdem Biyik, Suvir Mirchandani, Noah Goodman, Dorsa Sadigh, Neurips (2022).

[5] Encyclopedia of the Sciences of Learning. Editor: Seel, Norbert M., 2012, doi = 10.1007/978-1-4419-1428-6

[6] Towards the Future of AI-Augmented Human Tutoring in Math Learning. Vincent Aleven, Richard Baraniuk, Emma Brunskill, Scott Crossley, Dora Demszky, Stephen Fancsali, Shivang Gupta, Kenneth Koedinger, Chris Piech, Steve Ritter, Danielle R. Thomas1, Simon Woodhead, and Wanli Xing. International Conference on Artificial Intelligence in Education (2023)

[7] Progress Networks as a Tool for Analysing Student Programming Difficulties McBroom J, Paaßen B, Jeffries B, Koprinska I, Yacef K (2021) In: Proceedings of the Twenty-Third Australasian Computing Education Conference (ACE '21). Szabo C, Sheard J (Eds); Association for Computing Machinery: 158–167.

[8] Real-Time AI-Driven Assessment and Scaffolding that Improves Students' Mathematical Modeling during Science Investigations Authors: Amy Adair, Michael Sao Pedro, Janice Gobert, Ellie SeganAuthors Info & Claims. Artificial Intelligence in Education: 24th International Conference, AIED 2023.

[9] Chopin: 12 Studies for Piano, Op. 10. Alfred Cortot, editor. BMG Ricordi, 1986. ISBN 978-1-4803-0456-7. Editor: Cortot, Alfred

[10] Dyna-Q, http://www.incompleteideas.net/book/ebook/node96.html

[11] Towards Suggesting Actionable Interventions for Wheel-Spinning Students Mu, Tong; Jetten, Andrea; Brunskill, Emma, EDM, 2020

# A Training & Results

Table 1 lists all parameters of the experiments. The Figure 1 illustrate the achieved results when training with/without the meta unit. We evaluate the students policy w.r.t. improvement ratio for multiple training runs. The improvement rate of the student ($y$-axis) achieved for different number of training iterations while scaffolded by the the meta unit ($x$-axis) is illustrated in Figure 1 (solid line). In addition, we trained the scaffolding agent without the coordinator and evaluated the performance of its policy (dotted line). This means that just the trained Q-learner generates the exercise units and the student either succeeds in transition or not, determined by the correctness of the action and the improvement threshold of the student. We have trained and evaluated the meta unit for the case in which 1/4th of the optimal actions were different for the teacher and the student ($\pi_T^*$ and $\pi_S^*$, respectively). The results in the gray plot show that the higher number of training episodes performed by the meta unit result in higher improvement rate achieved by the student, which is calculated as an average of the received rewards given the number of learning steps until the end state is achieved.

Table 1: List of all parameters and their value if no *entity* is specified the parameter applies to all

| **Entity** | **Parameter** | **Value** |
|---|---|---|
| student | Improvement Threshold $\theta$ | 0.65 |
| – | Number of states $N$ | 24 |
| – | Number of actions $M$ | 4 |
| Agent | Exploration Factor $\epsilon$ | 0.1 |
| Agent | Learning Rate $\alpha$ | 0.1 |
| Agent | Discount Factor $\gamma$ | 0.95 |
| Agent | Planning Steps | 5 |

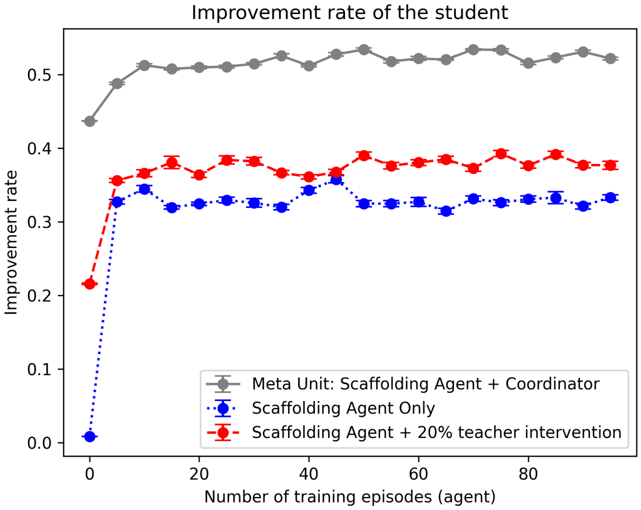

Figure 1: Comparison of improvement rate resulting with increasing number of training iterations averaged over 40 runs. The solid line (gray) illustrates the mean and variance of improvement rate of the student with the growing number of training iterations guided by the meta unit. The dotted line (blue) illustrates the policy evaluation of the scaffolding agent alone. The dashed line (red) illustrates the policy evaluation of the scaffolding agent, together with a 20% interference rate of the teacher.

## B Pseudo code for training with the meta unit (MU)

**Algorithm 1** Training with the MU

```
MU calculates average reward of the student return $p$
MU parameterizes Bernoulli distribution with $1 - p$ and samples from it return $s$
if $s == 1$ then
    MU schedules teacher intervention return teacher selects $a_T^*$
    student performs exercise $a_T^*$
    return student gets reward $r = 1$; and improves skill level from $s_t$ to $s_{t+1}$
else
    MU queries Q-learner return $a_Q^*$
    $x \sim Uniform(0, 1)$ return $x$
    if $x$ > improvement threshold and $a_Q^* == a_S^*$ then
        perform $a_Q^*$
        return student gets reward $r = 1$, and improves skill level from $s_t$ to $s_{t+1}$
    else
        return $r = 0$
    end if
end if
update the Q-learner with the values of $r, a, s_t,$ and $s_{t+1}$
```