Stochastic-Skill-Level-Based Shared Control for Human Training in Urban Air Mobility Scenario

SOOYUNG BYEON, JOONWON CHOI, YUTONG ZHANG, and INSEOK HWANG, Purdue University, USA

This paper proposes a novel stochastic-skill-level-based shared control framework to assist human novices to emulate human experts in complex dynamic control tasks. The proposed framework aims to infer stochastic-skill-levels (SSLs) of the human novices and provide personalized assistance based on the inferred SSLs. SSL can be assessed as a stochastic variable which denotes the probability that the novice will behave similarly to experts. We propose a data-driven method which can characterize novice demonstrations as a novice model and expert demonstrations as an expert model, respectively. Then, our SSL inference approach utilizes the novice and expert models to assess the SSL of the novices in complex dynamic control tasks. The shared control scheme is designed to dynamically adjust the level of assistance based on the inferred SSL to prevent frustration or tedium during human training due to poorly imposed assistance. The proposed framework is demonstrated by a human subject experiment in a human training scenario for a remotely piloted urban air mobility (UAM) vehicle. The results show that the proposed framework can assess the SSL and tailor the assistance for an individual in real-time. The proposed framework is compared to practice-only training (no assistance) and a baseline shared control approach to test the human learning rates in the designed training scenario with human subjects. A subjective survey is also examined to monitor the user experience of the proposed framework.

Additional Key Words and Phrases: human-machine interaction, skill-level inference, shared control, human training

1 INTRODUCTION

We propose a novel framework for inferring human skill-level and dynamically adjusting the assistance based on the inferred skill-level in complex dynamic control tasks. Along with the development of state-of-the-art artificial intelligence technology, questions have been raised about the role, ability, and necessity of humans. The first question about the motivation of human training might be *"Why do we have to train humans in the era of artificial intelligence?"* We can first consider the technical aspects of this question. For instance, the ever-increasing demand for well-trained human pilots in the field of urban air mobility (UAM) has received a lot of attention recently. To promote the safety and efficiency of UAM operations in urban areas, the role of human pilots is actively being discussed [41, 46]. Some tasks such as *detection and avoidance*, *emergency procedures*, *communication*, *takeoff and landings*, and *planning and decision-making* still require well-trained human pilots in specific context [5]. Regulatory and social aspects also provide further context. For example, the Federal Aviation Administration (FAA) pointed out that a human pilot is ultimately responsible for the operation and safety during unmanned aerial vehicle (UAV) flight [18]. A human supervisor with responsibility and authority must have the ability to ensure safe operation [19]. Therefore, well-trained human operators are required, and assisting human training is a significant benefit.

Since human training is important for safe and efficient system operations, the next question to ask is *"How can we train humans efficiently?"* In daily life, humans learn motor skills for vehicle control through repetitive practice

Authors' address: Sooyung Byeon, sbyeon@purdue.edu; Joonwon Choi, choi774@purdue.edu; Yutong Zhang, zhan3606@purdue.edu; Inseok Hwang, ihwang@purdue.edu, Purdue University, 701 W Stadium Ave, West Lafayette, Indiana, USA, 47907.

2573-9522/2023/6-ART
https://doi.org/10.1145/3603194

which induces a physical change in human brains [28]. To shorten the repetitive learning process, humans have sought assistance from experts. This can be analogous to the *parent-child metaphor* [40]. When children learn how to ride a bike, an adult assists to balance that bike. In general, the adult lets the children take control of the bike if it is in a safe operational range so that the children can practice balancing themselves. One important point is that assistance should only be provided as needed. The goal of assistance is to help novices perform their task independently, so the assistant should not have full control of the target system. As the novice's skill-level improves, the assistant needs to reduce intervention so that the novice can practice the task.

The primary motivation of this paper is that if human-human (*parent-child* or *expert-novice*) interaction can expedite human learning for dynamic control tasks, then autonomy can serve as a tutor to build an efficient human-machine interaction (HMI). The expert plays an important role in human-human interaction, but the number of experts is limited. In our proposed approach, autonomy models the performance of experts to build an expert model. The expert models enable autonomy to provide appropriate assistance to novices, as experts do. The past and current performance of novices can be utilized by autonomy to infer their skill-levels online. Then, autonomy can personalize the level of assistance based on the inferred skill-level. This approach reduces the demand for experts for human training while giving novices rapid access to quality training. In the following section, we provide related work. Various HMI approaches have been developed to assist human training such as modulating practice conditions, robot-assisted motor skill rehabilitation, and shared control applications. The skill-level inference methods have been proposed in many applications since they may not be straightforward in complex scenarios.

1.1 Related Work

1.1.1 Modulating Practice Conditions for Human Training. Effective practice conditions have been widely explored, since practice is generally considered the most important factor for motor learning and human training. The *optimal challenge point* (OCP) framework formulates the learning rate as a function of two variables, skill-level and task difficulty [23]. Hence, the learning rate can be optimized by modulating the task difficulty based on the skill-level of a learner. Note that *skill* refers to sensory-motor performance during tasks based on an intention [51]. The OCP framework suggests that as skill-level improves, the task should be more difficult to obtain further learning benefits [60]. In human-computer interaction, the *dynamic difficulty adjustment* (DDA) utilizes the OCP framework. The DDA has been adopted in computer games, healthcare, and education to offer appropriate difficulty to learners based on their dexterity, adapting ability, and personal characteristics [64]. The *intelligent tutoring system* is another approach to utilize the OCP framework [29]. The learners can engage in educational tasks with the intelligent tutor, which provides appropriate interventions to expedite the learning process based on the learner's previous performance [62].

However, inferring skill-level and adjusting task difficulty are task-dependent and may not be straightforward for complex dynamic control tasks. For instance, education games can assess the skill-level using a testing score and provide easy or difficult problems based on the skill-level. On the other hand, the testing score and pre-defined easy or difficult practice conditions may not be available in complex scenarios such as operating UAM vehicles.

1.1.2 Robot-Assisted Motor Skill Rehabilitation. Robotic exoskeleton systems have been used for motor learning and rehabilitation. Recent experimental studies have revealed that the motor learning rate can be improved or hindered by adjusting the assistance, feedback, and training tasks based on the current status of trainees [3, 45, 48, 52, 53]. In various rehabilitation studies, skill-level-based haptic guidance and error augmentation techniques have been demonstrated [3, 54]. Haptic guidance refers to providing appropriate torque to assist the learners. The error augmentation imposes artificial error to intentionally offer variations in training environments [42, 58]. Their studies showed that providing haptic assistance is generally beneficial for more impaired learners and error augmentation is effective for less impaired learners. Their ideas and some of their results coincide

with the OCP framework [23]. One study addressed the most similar approach to this paper, called the *minimal assist-as-needed controller* [48]. As its name implies, the minimal assist-as-needed controller modulates the assistance from a robotic algorithm which is affected by the skill-level of the learner and a pre-defined error bound.

However, there are some concerns over the existing work. The existing algorithms for rehabilitation devices only have considered limited degrees of freedom, and they may not be applicable to high-dimensional tasks. Furthermore, the existing algorithms usually employ simple reference trajectories for human training, for instance, straight lines. In complex high-dimensional scenarios, reference trajectories should consider the dynamics of the target systems (e.g., vehicle dynamics in UAM scenarios). Also, reference trajectories might be situation dependent (e.g., the UAM vehicle should slow down near the landing area). In this regard, data-driven methods could be better options in complex cases (e.g., learning reference trajectories from expert demonstrations).

1.1.3 Shared Control. Shared control has no global definition, but the method of controlling a system by a convex combination of independent control inputs from the human and autonomy has been widely used [40]. In general, autonomy assists humans to achieve goals by making tasks easier [16]. Popular applications include automated vehicles [1], UAVs [33], robot-assisted surgery [57], and manufacturing robots [44]. For enhancing the human learning rate, some applications utilized shared control [1]. Haptic shared control schemes use force or torque inputs from humans to infer their intentions [31]. Interestingly, haptic shared control with fixed gain (i.e., non-adaptive assistance) can decrease the human learning rate [36]. The researchers claimed that ill-designed shared controls may cause humans to adapt to a new task transformed by the shared controller rather than to the actual task [47]. To realize a well-designed shared control scheme for human training, a *progressive shared control* was proposed [35]. The progressive shared controller adjusts its assistance based on the skill-level of the learners. Physically decoupled shared control schemes (i.e., non-haptic shared controllers [1]) have also been studied for human training. Visual and auditory feedback can be provided [45, 52] or a shared controller can be used to provide assistant inputs obtained from expert demonstrations based on inferred skill-level of the learners [7, 9].

Studies on adaptive shared control provide a solid foundation for human training, but their target systems are relatively simple, i.e., controlling mass points in two-dimensional space [31, 35, 49]. Thus, their skill-level inference approaches need to be extended in complex dynamic control tasks. More complex scenarios, such as aerial vehicle landing scenarios, can be found in [7, 9], but only deterministic skill-level inference approaches are available. Those approaches cannot address uncertainties in human behaviors [56]. More details of the skill-level inference will be followed in Section 1.1.4.

1.1.4 Skill-Level Inference. Skill-level inference techniques aim to find quantifiable indicators of humans' skill-level by observing human motion patterns [17]. If a given task is simple, an intuitive indicator such as a trajectory tracking error can be used. On the other hand, for complex tasks such as surgery using robotic arms, human expert demonstrations can be utilized to identify the performance metric. In [20], the authors extracted comparison criteria for novice and expert performance in robot-controlled surgery. However, their comparison criteria depend on tasks, and they cannot perform the skill-level inference online (i.e., the skill-level is inferred after each trial). In [57], a convolution neural network technique was used to divide the skill-level into two categories (adept or rusty). This approach requires labeled human demonstrations for each skill-level. In robotics, human skill-levels were successfully quantified in a complex task and human-exoskeleton interactive system but pre-defined performance features are required [21, 22]. Some machine learning techniques have been investigated to infer the skill-level [38]. They utilize human demonstrations to model human behavioral patterns. For instance, *inverse optimal control* and *inverse reinforcement learning* were used to compare the given demonstrations from novices and experts [8, 9, 38]. These methods can infer and quantify the skill-level as a continuous variable. Nevertheless, these techniques cannot account for the *variability* of human behaviors. The variability refers to the intrinsic uncertainties that can be observed in human behaviors [56]. For instance, when an expert performs a task

multiple times, resultant demonstrations might not be perfectly consistent due to variability. In our pilot study, experts tend to control the multi-rotor vehicle very accurately near an obstacle or waypoint, but relatively larger variations are observed in the open air. Similar observations can be found in civil aircraft pilots who are highly qualified [15]. Even if a novice deviates from the expert nominal trajectory, if there is a contextual situation in which the expert frequently deviates from the nominal trajectory, we do not want to underestimate the skill-level of the novice. These complexities may induce additional problems in assessing skill-level.

1.2 Contributions

Our main contributions are three folds: 1) a novel stochastic-skill-level (SSL) inference method is developed to estimate the SSL of each human in complex dynamic control tasks. The proposed inference method can account for not only the nominal human behavior but also the variability in human behaviors. The SSL can be assessed online; 2) the SSL-based shared control scheme is proposed to adapt the assistance based on the inferred SSL in complex dynamic control tasks. It can personalize the assistance according to the inferred SSL; and 3) a human subject experiment is conducted to demonstrate the proposed framework.

More details are provided to contextualize the contributions. The proposed shared control approach enables us to apply the OCP framework in complex dynamic control tasks. To improve the human learning rate in the dynamic control task, the difficulty of the task is modulated to prevent frustration from tasks being too hard or boredom from tasks being too easy [64]. Consequently, two methods are required to facilitate task difficulty modulation: skill-level inference and skill-level-based task modulation. Some related applications have been investigated in robot-assisted motor skill rehabilitation [3, 48, 53], robot-assisted surgery [20, 57], and shared control [35, 49, 61]. However, their target systems are relatively simple in terms of the system dynamics (e.g., mass points in the two-dimensional space with spring-damper systems) or the degree of freedom (DOF) (e.g., single-axis control of rehabilitation devices). Also, their tasks are simple (e.g., tracking a pre-defined path), and applied skill-level inference methods are only suitable for certain simple tasks (details in Section 1.1.4). Several concerns could arise if the target system and tasks are complex. In our case, the target system is a multi-rotor system with the nonlinear 6-DOF dynamics. The task is flying through the urban area by safely passing through multiple waypoints, which requires more sophisticated path planning rather than simply connecting waypoints by curved lines since the vehicle dynamics adds constraints (e.g., obstacle avoidance maneuvers and acceleration constraints). In the worst case, the waypoints may not be available, but they need to be derived from expert demonstrations. Even if the optimal trajectory is available, it might not be preferred by humans due to the perceived risk [32] (e.g., human pilots do not want to fly close to buildings although autonomy says it is optimal). Thus, a data-driven method can be applied to efficiently generate the nominal trajectory from the expert demonstrations. Furthermore, multiple expert demonstrations also include the variability [56] which represents the inherent uncertainties in human behaviors. For instance, pilots tend to control the vehicle very accurately near obstacles and waypoints, but positional variations in the open air are much larger [15]. The proposed SSL inference method can explicitly account for the variability to identify acceptable or unacceptable deviations from the nominal trajectory. Then, the proposed shared control framework can adjust the assistant scheme based on the inferred SSL. Our human subject study shows how the proposed method works in an illustrative scenario.

The rest of the paper is organized as follows. In Section 2, a human training scenario using a UAM simulator is introduced. Section 3 addresses the structure of the proposed framework. The proposed framework is demonstrated via a human subject experiment in Section 4. Discussion and conclusions are given in Section 5 and 6, respectively.

2 HUMAN TRAINING SCENARIO

We introduce our human training scenario that helps readers understand how the proposed framework works. It is used for the human subject experiment in Section 4. In this scenario, a human user is requested to control and move a multi-rotor vehicle from an initial position to a destination for UAM operation training.

A 3-D virtual training environment is constructed in a constrained rectangular region with a take-off pad, a landing pad, two large buildings, and detailed surroundings that imitate an urban environment as shown in Fig. 1. This environment is powered by AirSim, an open-source simulator for physically and visually realistic simulations [55]. Inside the environment, the two buildings lie between the starting and ending points, creating a path shaped like the letter "S" between them (Fig. 1a). Along the path, six virtual waypoints, which have no collision effect, are strategically placed in the air (Fig. 1b). The human users use a physical flight controller (Xbox controller) to control the vehicle's roll, pitch, yaw rate, and thrust. A graphical representation of the vehicle and the current system state information (e.g., current position and velocity) are displayed on a monitor (Fig. 1c). Vibration can be given to the human user through the flight controller to notify changes in the assistant scheme. The vehicle takes off automatically from its initial position to its designated altitude before human users start each trial. When a release signal is triggered by human users via the flight controller, they are requested to trace the path by flying through the waypoints successively to their best ability whether passing through each waypoint or not, i.e., they do not need to retry passing through each waypoint even if they missed it. The last waypoint is three times wider to give flexibility on the final approach to the landing pad. Note that we only consider the skill training in this scenario, but path re-planning (e.g., choosing alternative paths) is not considered. Above the landing pad, there is a target area marked by a semi-transparent sphere, which triggers an automatic landing sequence only if the vehicle approaches the sphere with a speed of less than 7 m/s (Fig. 1d). During the entire training, human users control the vehicle in the first-person view to replicate a remote pilot control situation. A sub-window, which shows beneath the vehicle, is provided to the human user to enhance the user's depth perception. Along with each trial, the vehicle states such as position, velocity, acceleration, attitude, angular velocity, angular acceleration, and human behavior data such as roll, pitch, yaw rate, and thrust inputs are recorded in 20 Hz. The human user has to complete the given task within three minutes time limit, otherwise, the simulation is terminated. Fig. 2 shows recorded trajectories from novice and expert demonstrations using the training environment.

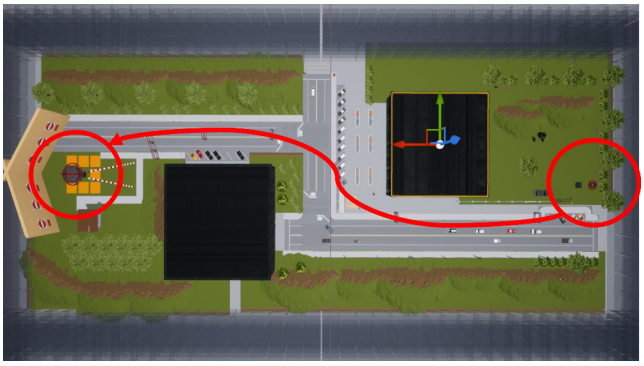
3 STOCHASTIC-SKILL-LEVEL-BASED SHARED CONTROL FRAMEWORK

In the SSL-based shared control framework, we utilize autonomy as a catalyst to expedite skill transfer from experts to novices. The proposed framework assists novices to emulate experts by permitting manual control (no assistance) to the novice unless the predicted behavior is deviating from a certain confidence interval. The probability of conducting the given task properly can be mapped into the SSL and the assistant scheme would be personalized since each novice has a different SSL in various situations. We formally present all necessary technical elements of the SSL inference and SSL-based shared control. We consider a 6-DOF nonlinear dynamical system operated by humans (like the multi-rotor system in Section 2):

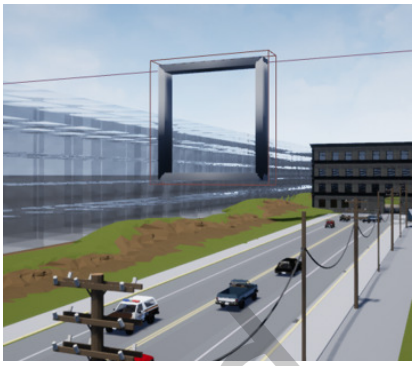
$$[\mathbf{x}_{k+1} \quad \dot{\mathbf{x}}_{k+1} \quad \boldsymbol{\phi}_{k+1} \quad \boldsymbol{\omega}_{k+1}] = f(\mathbf{x}_k, \dot{\mathbf{x}}_k, \boldsymbol{\phi}_k, \boldsymbol{\omega}_k, \mathbf{u}_k) \tag{1}$$

where $\mathbf{x}_k \in \mathbb{R}^3$ is the position, $\dot{\mathbf{x}}_k \in \mathbb{R}^3$ is the velocity, $\boldsymbol{\phi}_k \in \mathbb{R}^3$ is the Euler angle, and $\boldsymbol{\omega}_k \in \mathbb{R}^3$ is the angular velocity of the system, respectively. k denotes the time step. $\mathbf{u}_k \in \mathcal{U} \subset \mathbb{R}^m$ denotes the m-dimensional control input and $f : \mathbb{R}^{12+m} \to \mathbb{R}^{12}$ denotes the nonlinear system dynamics. Throughout this paper, our main interest is to model human behaviors in controlling the trajectory of the system. The system model can be reformulated as:

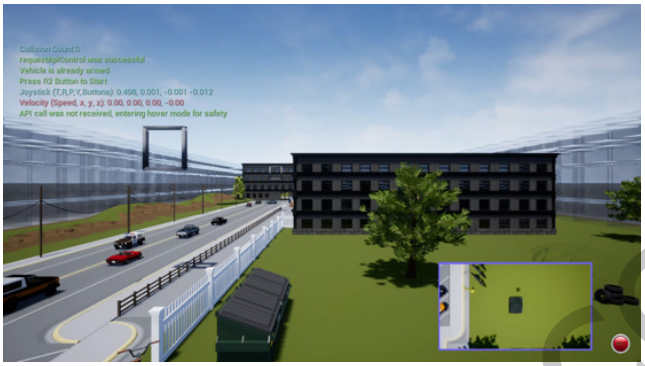
$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \dot{\mathbf{x}}_{k+1} \end{bmatrix} = \begin{bmatrix} I & I\Delta t \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ \dot{\mathbf{x}}_k \end{bmatrix} + \begin{bmatrix} 0 \\ I\Delta t \end{bmatrix} \mathbf{U}(\mathbf{x}_k, \dot{\mathbf{x}}_k, \boldsymbol{\phi}_k, \boldsymbol{\omega}_k, \mathbf{u}_k) \tag{2}$$



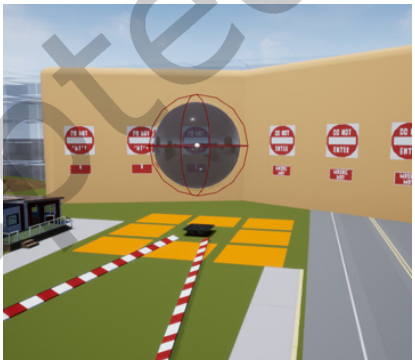
(a) Top view of the 3-D training environment.



(b) The first waypoint and other objects.



(c) The human user view in the training environment.



(d) The target area.

Fig. 1. Human training scenario.

where $\mathbf{U} : \mathbb{R}^{12+m} \rightarrow \mathbb{R}^3$ is the nonlinear mapping function from the position, velocity, Euler angle, angular velocity, and control input into the acceleration input: $\ddot{\mathbf{x}}_k = \mathbf{U}(\mathbf{x}_k, \dot{\mathbf{x}}_k, \boldsymbol{\phi}_k, \boldsymbol{\omega}_k, \mathbf{u}_k)$. I and Δt denote the identity matrix and the time step, respectively. The acceleration mapping function $\ddot{\mathbf{x}}_k = \mathbf{U}(\mathbf{u}_k|\boldsymbol{\phi}_k) = \mathbf{U}_k$ and its inverse function $\mathbf{u}_k = \mathbf{U}^{-1}(\mathbf{U}_k|\boldsymbol{\phi}_k)$ for a multi-rotor system are feasible with high precision [34]. See Appendix A.1 and A.2 for the details.

The structure of the SSL-based shared control is given in Fig. 3. For the SSL inference, the proposed framework requires two trajectory distributions: the expert trajectory distribution and the predicted novice trajectory distribution. Even if experts perform the very same control task multiple times, their input sequences may not be the same multiple times because of unintended variations in behavior, known as the variability of human behavior in motor skills [56]. For novices, the variability is usually greater and their behaviors are unintentionally inconsistent [9]. Thus, we model and estimate the probability distribution of multiple trajectories instead of only considering the single mean trajectory. The expert trajectory distribution can be obtained from expert demonstrations using a data-driven method. Since the expert trajectory distribution is only used as a reference, offline computation is acceptable. The predicted novice trajectory distribution can be computed from novice demonstrations and the current state of the system. By combining the novice's past data and the current state, it is possible to predict how the novice will control the system in the future. This process should be conducted

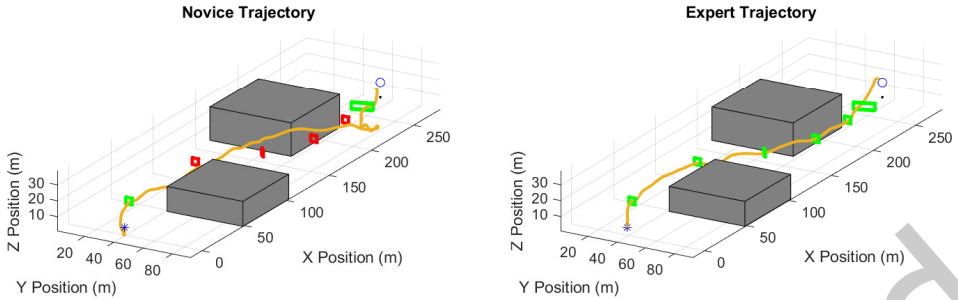


Fig. 2. The recorded vehicle trajectories of human users. The red and green waypoints denote the human user fails and succeeds to pass through them, respectively. The yellow solid lines show the recorded trajectories.

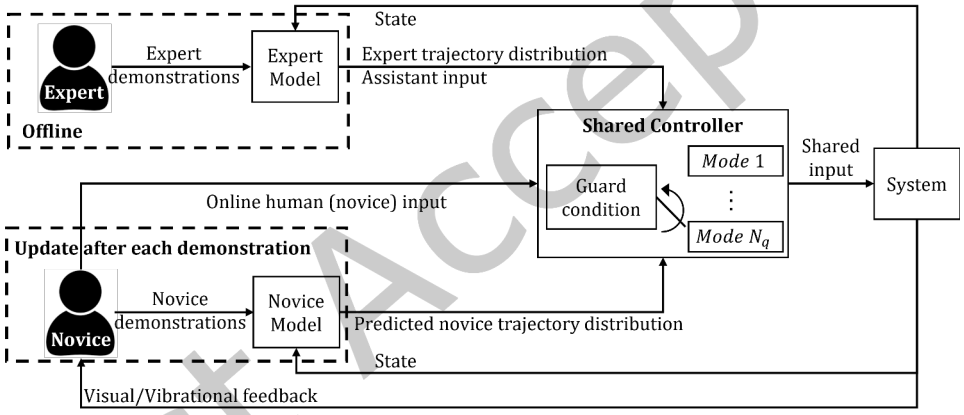


Fig. 3. Structure of the proposed stochastic-skill-level-based shared control.

in real-time to account for the time-varying SSL of the novice. Then, the SSL can be inferred by comparing the expert trajectory distribution and the predicted novice trajectory distribution. Note that the novice model can also be updated after each trial to examine SSL improvement trial by trial as shown in Fig. 3.

A data-driven method is used to reproduce the trajectory distribution from human demonstrations. The proposed framework models the trajectory distribution using the hidden semi-Markov model (HSMM). The HSMM has advantages over the existing trajectory distribution encoding algorithms [25]. The Gaussian mixture model (GMM) can represent the structure of the demonstrations but cannot provide the state transition information (i.e., a transition from one Gaussian mixture to another). The hidden Markov model (HMM) provides the state transition probabilities, but its self-state transitions are known to be inaccurate. The HSMM can encode the state duration probabilities which provide further advantages in accurate learning. See [10] for the details and tutorial source codes.

We aim to model the trajectory distribution using HSMM with a given set of human demonstrations $\{\mathbf{x}_k, \dot{\mathbf{x}}_k, \ddot{\mathbf{x}}_k\}_{k=1}^{N}$ where $N = \sum_{m=1}^{M} T_m$ denotes the total number of data points from M demonstrations with T_m data points for each demonstration. Let $\boldsymbol{\zeta} \in \mathbb{R}^{9N}$ and $\mathbf{x} \in \mathbb{R}^{3N}$ be the concatenated states as:

$$\boldsymbol{\zeta}_k = \begin{bmatrix} \mathbf{x}_k^T & \dot{\mathbf{x}}_k^T & \ddot{\mathbf{x}}_k^T \end{bmatrix}^T \in \mathbf{Z} \subset \mathbb{R}^9, \qquad \boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\zeta}_1^T & \boldsymbol{\zeta}_2^T & \cdots & \boldsymbol{\zeta}_N^T \end{bmatrix}^T, \qquad \mathbf{x} = \begin{bmatrix} \mathbf{x}_1^T & \mathbf{x}_2^T & \cdots & \mathbf{x}_N^T \end{bmatrix}^T. \tag{3}$$

The given training data points $\boldsymbol{\zeta}$ can be encoded into a GMM using the expectation maximization (EM) algorithm [10, 43], i.e., $P(\boldsymbol{\zeta}) \sim \sum_{i=1}^{K} w_i \mathcal{N}(\boldsymbol{\mu}_i, \Sigma_i)$ where K denotes the number of Gaussian mixture component, w_i is the weight of each Gaussian mixture component with $\sum_{i=1}^{K} w_i = 1$, $\mathcal{N}(\cdot)$ is the normal distribution, and $\boldsymbol{\mu}_i$ and Σ_i denote the mean and covariance of each Gaussian mixture component. In the next step, the HSMM parameters $\{\pi_i, \boldsymbol{\mu}_i, \Sigma_i, \{\alpha_{i,j}\}_{j=1}^{K}, \mu_i^D, \Sigma_i^D\}_{i=1}^{K}$ are trained and the mixture duration is post-estimated from the training data points [12], where π_i is the initial probability of being the Gaussian mixture component i and $\alpha_{i,j}$ is the transition probability from the Gaussian mixture component i to that of j. μ_i^D and Σ_i^D are the mean and variance of the mixture duration time.

Based on the trained HSMM parameters, the expert trajectory distribution can be reconstructed as a sequence of Gaussian states $s = \{s_1, s_2, \cdots, s_T\}$ where $s_k \in \{1, 2, \cdots, K\}$ and T denotes the trajectory reconstruction horizon. Let $d_{\max} = \sigma_d T/K$ be the maximum duration of each mixture where $\sigma_d \in \mathbb{R}^+$ denotes the safety factor and $\sigma_d > 1$. The likelihood of the mixture duration for the Gaussian mixture i is given as:

$$P_d(i, d) = \frac{\mathcal{N}(d \mid \mu_i^D, \Sigma_i^D)}{\sum_{k=1}^{d_{\max}} \mathcal{N}(k \mid \mu_i^D, \Sigma_i^D)} \tag{4}$$

where $d \in \{1, \cdots, d_{\max}\}$ and $\mathcal{N}(x \mid \mu, \Sigma)$ denotes the evaluated Gaussian function with input x, mean μ, and covariance Σ. Then, the sequence of Gaussian mixture can be found by:

$$h(i, t) = \frac{\gamma(i, t)}{\sum_{j=1}^{K} \gamma(j, t)}, \qquad \gamma(i, t) = \sum_{j=1}^{K} \sum_{d=1}^{\mathcal{D}} \gamma(j, t - d) \alpha_{j,i} P_d(i, d) \tag{5}$$

where $\mathcal{D} = \min(d_{\max}, t - 1)$ with initialization $\gamma(i, 1) = \pi_i$. Then, (5) can be recursively computed. We pick the maximum probability Gaussian component to reconstruct the trajectory distribution [12]:

$$s_k = \max_i h(i, k). \tag{6}$$

The likelihood of trajectory $\boldsymbol{\zeta}$ given the Gaussian mixture sequence s is described as:

$$P(\boldsymbol{\zeta} \mid s) = \prod_{k=1}^{T} \mathcal{N}(\boldsymbol{\zeta}_k \mid \boldsymbol{\mu}_{s_k}, \Sigma_{s_k}) = \mathcal{N}(\boldsymbol{\zeta} \mid \boldsymbol{\mu}_s, \Sigma_s) \tag{7}$$

where $\boldsymbol{\mu}_s = \begin{bmatrix} \boldsymbol{\mu}_{s_1}^T, \cdots, \boldsymbol{\mu}_{s_T}^T \end{bmatrix}^T$ and $\Sigma_s = \text{diag}\,(\Sigma_{s_1}, \cdots, \Sigma_{s_T})$. $\text{diag}(\cdot)$ is the square diagonal matrix with the elements on the diagonal. The log-likelihood of $P(\boldsymbol{\zeta} \mid s)$ can be obtained with a large sparse matrix $\Phi \in \mathbb{R}^{9N \times 3N}$.

$$\Phi = \begin{bmatrix} & \vdots & \vdots & \vdots & \vdots & \\ \cdots & I & 0 & 0 & 0 & \cdots \\ \cdots & -\frac{1}{\Delta t} I & \frac{1}{\Delta t} I & 0 & 0 & \cdots \\ \cdots & \frac{1}{\Delta t^2} I & -\frac{2}{\Delta t^2} I & \frac{1}{\Delta t^2} I & 0 & \cdots \\ \cdots & 0 & I & 0 & 0 & \cdots \\ \cdots & 0 & -\frac{1}{\Delta t} I & \frac{1}{\Delta t} I & 0 & \cdots \\ \cdots & 0 & \frac{1}{\Delta t^2} I & -\frac{2}{\Delta t^2} I & \frac{1}{\Delta t^2} I & \cdots \\ & \vdots & \vdots & \vdots & \vdots & \end{bmatrix}. \tag{8}$$

Then, $P(\boldsymbol{\zeta}|s) = P(\Phi\mathbf{x}|s)$ and the trajectory mean $\hat{\boldsymbol{\mu}}^{\mathbf{x}}$ and covariance $\hat{\Sigma}^{\mathbf{x}}$ are given as a least-square solution [10]:

$$\hat{\boldsymbol{\mu}}^{\mathbf{x}} = \arg\max_{\mathbf{x}} \log P(\Phi\mathbf{x} \mid s) = \left(\Phi^T\Sigma_s^{-1}\Phi\right)^{-1}\Phi^T\Sigma_s^{-1}\boldsymbol{\mu}_s = \left[(\hat{\boldsymbol{\mu}}_1^{\mathbf{x}})^T \quad \cdots \quad (\hat{\boldsymbol{\mu}}_T^{\mathbf{x}})^T\right]^T \tag{9}$$

$$\hat{\Sigma}^{\mathbf{x}} = \sigma\left(\Phi^T\Sigma_s^{-1}\Phi\right)^{-1} = \mathrm{diag}\left(\hat{\Sigma}_1^{\mathbf{x}}, \cdots, \hat{\Sigma}_T^{\mathbf{x}}\right) \tag{10}$$

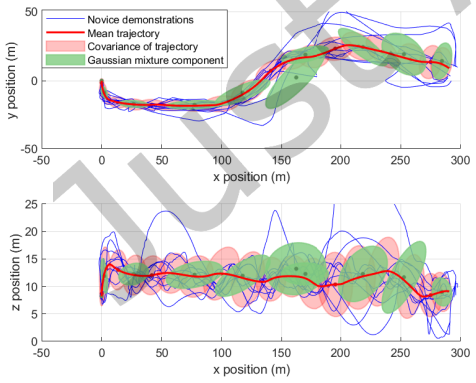
where $\sigma > 0$ denotes the scale factor. Then, the reproduced trajectory distribution is given by $\hat{\mathbf{x}}_k \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_k^{\mathbf{x}}, \hat{\Sigma}_k^{\mathbf{x}})$. Fig. 4 shows exemplar trajectory distributions from human demonstrations. Note that 10 Gaussian components are used in our training environment ($K = 10$). This number can be determined based on the Bayesian information criterion (BIC). One can find the best fitting number of Gaussian components by selecting K that yields the lowest information criterion [59]. In the novice case, the trajectory distribution is relatively wide since the novice demonstrations are inconsistent due to the lack of skill. The expert can perform the task relatively accurately and consistently, and thus the trajectory distribution is relatively narrow. At the final stage of the trajectory (marked by black squares in Fig. 4b), the last waypoint is three times wider than the others. The wider waypoint is designed to induce more variability in trajectories.

The proposed framework needs an assistant policy to provide expert-like assistant input. The assistant policy can be represented as a state feedback controller which can reproduce the learned expert nominal trajectory. Let $\hat{\mathbf{x}}_k^e \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}}, \hat{\Sigma}_k^{e,\mathbf{x}})$ be the expert trajectory distribution. Then, a trajectory tracking cost can be formulated as a quadratic cost function c_k for all $k \in \{0, \cdots, T\}$ [10]:

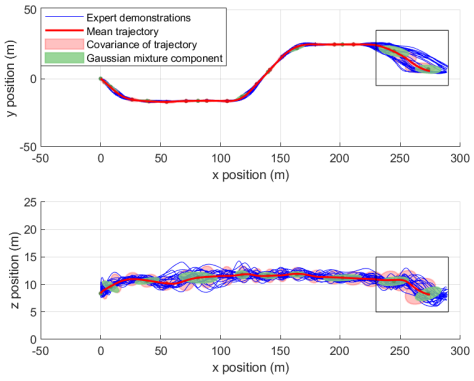
$$c_k = \left(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}}\right)^T\left(\hat{\Sigma}_k^{e,\mathbf{x}}\right)^{-1}\left(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}}\right) + \mathbf{U}_k^T R\mathbf{U}_k \tag{11}$$

where $R = \rho I$ is the control input cost with the scale factor $\rho > 0$. We consider a constrained optimization problem with the constrained acceleration input $A^{\mathbf{u}}\mathbf{U}_k \leq b^{\mathbf{u}}$ to conduct realistic and safe operations. This optimization problem can be solved using the finite-time horizon model predictive control (MPC) [30]. Let the current time step $k = 1$, without loss of generality, and the finite-time horizon l. The cost function C is given by:

$$\boldsymbol{\zeta}_{[1:l]} = S^{\mathbf{x}}\boldsymbol{\zeta}_1 + S^{\mathbf{u}}\tilde{\mathbf{U}} \tag{12}$$



(a) Novice demonstrations and corresponding trajectory distribution ($M = 10, K = 10$).



(b) Expert demonstrations and corresponding trajectory distribution ($M = 24, K = 10$).

Fig. 4. Trajectory distributions from novice and expert demonstrations.

$$S^{\mathbf{x}} = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{l-1} \end{bmatrix}, \qquad S^{\mathbf{u}} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ A^{l-2}B & A^{l-3}B & \cdots & B \end{bmatrix}, \qquad \tilde{\mathbf{U}} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_{l-1} \end{bmatrix} \tag{13}$$

$$C = \left(\tilde{\boldsymbol{\mu}}_l - S^{\mathbf{x}}\boldsymbol{\zeta}_1 - S^{\mathbf{u}}\tilde{\mathbf{U}}\right)^T \left(\hat{\Sigma}^{e,\mathbf{x}}\right)^{-1} \left(\tilde{\boldsymbol{\mu}}_l - S^{\mathbf{x}}\boldsymbol{\zeta}_1 - S^{\mathbf{u}}\tilde{\mathbf{U}}\right) + \tilde{\mathbf{U}}^T \tilde{R} \tilde{\mathbf{U}} \tag{14}$$

where

$$A = \begin{bmatrix} I & I\Delta t & 0 \\ 0 & I & I\Delta t \\ 0 & 0 & I \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 0 \\ I\Delta t \end{bmatrix} \tag{15}$$

$$\tilde{\boldsymbol{\mu}}_l = \begin{bmatrix} \boldsymbol{\mu}_1^T & \cdots & \boldsymbol{\mu}_l^T \end{bmatrix}^T, \qquad \hat{\Sigma}^{e,\mathbf{x}} = \text{diag}\left(\hat{\Sigma}_1^{e,\mathbf{x}}, \cdots, \hat{\Sigma}_l^{e,\mathbf{x}}\right), \qquad \tilde{R} = \text{diag}\left(R, \cdots, R\right). \tag{16}$$

Then, the MPC problem can be solved using the quadratic programming (QP) [6].

$$\min_{\tilde{\mathbf{U}}} \left(\frac{1}{2}\tilde{\mathbf{U}}^T H \tilde{\mathbf{U}} + g^T \tilde{\mathbf{U}}\right) \quad \text{subject to} \quad A^{\mathbf{u}}\mathbf{U}_k \leq b^{\mathbf{u}} \tag{17}$$

where

$$H = 2(W^T W + \tilde{R}), \qquad W = LS^{\mathbf{u}}, \qquad L = \text{diag}\left(L_1, \cdots, L_l\right), \qquad \left(\hat{\Sigma}_k^{e,\mathbf{x}}\right)^{-1} = L_k^T L_k \tag{18}$$

$$g = -2v^T W, \qquad v = L\left(\tilde{\boldsymbol{\mu}}_l - S^{\mathbf{x}}\zeta_1\right) \tag{19}$$

Note that L_k for $k \in [1, l]$ can be computed using the Cholesky factorization [24]. This QP problem can be solved very efficiently using the standard QP algorithm, and the assistant acceleration input $\mathbf{U}_k^a$ computation is tractable in real-time. Then, the assistant policy Π_a is given by solving the QP problem (17):

$$\mathbf{U}_k^a = \Pi_a\left(\zeta_k, \hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}}, \hat{\Sigma}_k^{e,\mathbf{x}}\right) \tag{20}$$

which is a deterministic feedback policy with respect to the current state ζ_k and the data-driven expert model $\{\hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}}, \hat{\Sigma}_k^{e,\mathbf{x}}\}$. Finally, the assistant input $\mathbf{u}_k^a$ for a multi-rotor system can be obtained using the mapping function from acceleration to control input (see Appendix A.2).

$$\mathbf{u}_k^a = \mathbf{U}^{-1}(\mathbf{U}_k^a | \boldsymbol{\phi}_k). \tag{21}$$

Next step is to obtain the novice control policy using the novice demonstrations (past data) and system state (current data). This step aims to obtain the *predicted novice trajectory distribution* to assess the novice's SSL based on the observed variability. Assuming that the current state $\zeta_k = [\mathbf{x}_k^T, \dot{\mathbf{x}}_k^T, \ddot{\mathbf{x}}_k^T]^T$ of the system is available. Let $\{\hat{\boldsymbol{\mu}}_k^{n,\mathbf{x}}, \hat{\Sigma}_k^{n,\mathbf{x}}\}_{k=1}^T$ be a set of the inferred trajectory distribution using the novice demonstrations and (9)-(10). Then, the current phase $j \in \{1, \cdots, T\}$ of the system needs to be identified to align time series data:

$$j = \arg\min_i \|\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i^{n,\mathbf{x}}\| \tag{22}$$

where $\|\cdot\|$ denotes the L_2 norm and $i \in \{1, \cdots, T\}$ denotes the time step. Then, we pick the Gaussian component that has the maximum probability as similar to (6) with the finite-time horizon l:

$$s_k^n = \max_i h(i, k) \tag{23}$$

where $k \in \{j, \cdots, j + l - 1\}$. The trajectory distribution can be obtained using (7)-(10) and the initial condition

$$\boldsymbol{\mu}_{s_1} = \zeta_j, \qquad \Sigma_{s_1} = \Sigma_j \tag{24}$$

where Σ_j denotes the initial state covariance. Note that if the state ζ_j is perfectly known without uncertainty, the initial state covariance can be $\Sigma_j = \epsilon I$ where $\epsilon < 1$ denotes the small positive scale factor. The novice

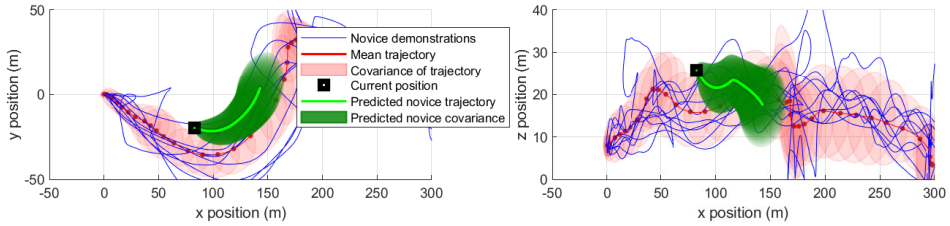


Fig. 5. Predicted trajectory distribution from current state and novice model (finite-time horizon $l = 200$ or 10 *sec*).

model $\{s^n, \boldsymbol{\mu}_{s^n}, \Sigma_{s^n}\}$ is defined as a set of sequence, mean, and covariance of each Gaussian component, as shown in (7). The prediction result $\{\hat{\boldsymbol{\mu}}_k^{np,\mathbf{x}}, \hat{\Sigma}_k^{np,\mathbf{x}}\}$ is given by the mean and covariance of the novice trajectories with respect to the current time step k. In other words, the predicted novice trajectory distribution is given as $\mathbf{x}_k^{np} \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_k^{np,\mathbf{x}}, \hat{\Sigma}_k^{np,\mathbf{x}})$, as shown in (9)-(10). The novice control policy Π_n represents the predicted novice trajectory distribution as a stochastic policy:

$$\left(\hat{\boldsymbol{\mu}}_k^{np,\mathbf{x}}, \hat{\Sigma}_k^{np,\mathbf{x}}\right) = \Pi_n\left(\zeta_k, s^n, \boldsymbol{\mu}_{s^n}, \Sigma_{s^n}\right). \tag{25}$$

Fig. 5 shows the predicted novice trajectory distribution $(\hat{\boldsymbol{\mu}}_k^{np,\mathbf{x}}, \hat{\Sigma}_k^{np,\mathbf{x}})$ from the current state ζ_k and the novice model $\{s^n, \boldsymbol{\mu}_{s^n}, \Sigma_{s^n}\}$, using an example from the human training environment in Section 2.

3.1 Shared Controller Design

We present a formal representation of the SSL-based shared control in this section. The primary role of the shared control is to determine the *control authority* $\alpha_k \in [0, 1]$ at each time step k. The control authority is defined as a weight value of the human input on the shared input space [2]. The shared control input $\mathbf{u}_k$ is then a convex combination of the human input $\mathbf{u}_k^h \in \mathcal{U}$ and assistant (autonomy) input $\mathbf{u}_k^a \in \mathcal{U}$ as

$$\mathbf{u}_k = \alpha_k \mathbf{u}_k^h + (1 - \alpha_k)\mathbf{u}_k^a. \tag{26}$$

Note that $\alpha_k = 1$ indicates the full manual control and $\alpha_k = 0$ denotes the full autonomous control, respectively. In the proposed shared control framework, we only use a set of finite parameters such as $\alpha_k \in \mathcal{A} = \{0.1, 0.5, 1.0\}$, since it is observed in our previous study [9] that continuous control authority change may provoke mode confusion to the human. The control authority is allocated in a discretized manner and the corresponding control authority is mapped into the *discrete mode* of the *hybrid system* [37]. The hybrid systems refers to the dynamic systems that involve the interaction of continuous states and discrete states. Accordingly, the SSL can be inferred in a discretized manner as well, and the SSL and the control authority are one-to-one mapped. Note that $\alpha_k = 0.1$, $\alpha_k = 0.5$ and $\alpha_k = 1.0$ correspond to the assistant schemes for low-skill-level, intermediate-skill-level, and high-skill-level human users, respectively. Also, the mapping can be modified depending on the specific task and context (e.g., more finely discretized skill-level). The hybrid shared controller is formalized as a tuple of the following elements [37], $\mathcal{H} = (\mathcal{Q}, \mathcal{I}, \mathcal{O}, r, \mathcal{G})$ where each element is defined as follows.

- $\mathcal{Q} = \{1, 2, \cdots, N_q\}$ denotes the set of discrete modes. $q_k \in \mathcal{Q}$ denotes the discrete mode at time step k. Note that $N_q = 3$ for the application in this paper.
- $\mathcal{I} = \mathbf{Z} \times \Pi_a \times \Pi_n$ denotes the input of the hybrid system, which consists of the state space $\mathbf{Z}$, the assistant policy Π_a, and the novice control policy Π_n.
- $\mathcal{O} = \mathcal{U}$ is the output of the hybrid system, i.e., the shared input $\mathbf{u}_k$.
- $r : \mathcal{Q} \times \mathcal{I} \to \mathcal{O}$ denotes the shared control law:

$$\mathbf{u}_k = r\left(q_k, \zeta_k, \Pi_a, \Pi_n\right) \triangleq \alpha(q_k)\mathbf{u}_k^h + (1 - \alpha(q_k))\mathbf{u}_k^a \tag{27}$$

where the control authority α_k is an one-to-one mapping from the discrete mode q_k, i.e., $\alpha_k = \alpha(q_k)$, where $\alpha : \mathcal{Q} \to [0, 1]$ is the one-to-one mapping function.

- $\mathcal{G} : \mathcal{Q} \times \mathcal{Q} \to 2^{\mathbf{Z} \times \Pi_a \times \Pi_n}$ denotes the guard condition which assigns the previous discrete mode q_{k-1} to the current discrete mode q_k if the state $\mathbf{Z}$, the assistant policy Π_a, and the novice control policy Π_n satisfy the guard condition. The guard condition is covered in detail in Section 3.3.

3.2 Stochastic-Skill-Level Inference

We define the SSL to quantify the guard condition which determines the mode change of the hybrid system or the control authority allocation in the proposed framework. The SSL represents the probability that a novice emulates an expert's behavior while performing a given task. With the control authority being allocated based on the SSL, the proposed framework is named the *SSL-based shared control*. The proposed framework uses the Mahalanobis distance [50] as a performance measure to determine the assistant scheme by considering the variability. Let the mean of expert trajectory distribution $\hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}}$ be the reference trajectory and the covariance $\hat{\Sigma}_k^{e,\mathbf{x}}$ be the weighting matrix for the Mahalanobis distance.

$$D_k = \left[\left(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}} \right)^T \left(\hat{\Sigma}_k^{e,\mathbf{x}} \right)^{-1} \left(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_k^{e,\mathbf{x}} \right) \right]^{\frac{1}{2}}. \tag{28}$$

Let the current time step $k = 1$ without loss of generality. If the discrete mode is fixed for a finite-time horizon L, i.e., $q_k = q$ for all $k \in \{1, \cdots, L\}$, the predicted Mahalanobis distance $\hat{D}_L(q)$ of the system, controlled by the shared control law (27), is given by:

$$\hat{D}_L(q) = \left[\left(\mathbf{x}_L(q) - \hat{\boldsymbol{\mu}}_L^{e,\mathbf{x}} \right)^T \left(\hat{\Sigma}_L^{e,\mathbf{x}} \right)^{-1} \left(\mathbf{x}_L(q) - \hat{\boldsymbol{\mu}}_L^{e,\mathbf{x}} \right) \right]^{\frac{1}{2}} \tag{29}$$

where the future predicted state $\mathbf{x}_L(q)$ with the constant discrete mode q is given as:

$$\mathbf{x}_L(q) = \alpha(q) \hat{\boldsymbol{\mu}}_L^{np,\mathbf{x}} + (1 - \alpha(q)) \mathbf{x}_L^a \tag{30}$$

subject to the state prediction under the assistant acceleration input $\mathbf{U}_k^a$

$$\begin{bmatrix} \mathbf{x}_{k+1}^a \\ \dot{\mathbf{x}}_{k+1}^a \end{bmatrix} = \begin{bmatrix} I & I\Delta t \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{x}_k^a \\ \dot{\mathbf{x}}_k^a \end{bmatrix} + \begin{bmatrix} 0 \\ I\Delta t \end{bmatrix} \mathbf{U}_k^a \tag{31}$$

for all $k \in \{1, \cdots, L\}$ and the initial condition $\{\mathbf{x}_1^a, \dot{\mathbf{x}}_1^a\} = \{\mathbf{x}_1, \dot{\mathbf{x}}_1\}$ is known. The predicted Mahalanobis distance $\hat{D}_L(q)$ for the fixed discrete mode q is a stochastic value that represents the skill-level. Indeed, if a novice performs a given task similar to an expert, then $\hat{D}_L(q)$ would be a small value which indicates that the skill-level of that novice is high, and vice versa.

The covariance matrix of the expert trajectory distribution $\hat{\Sigma}_k^{e,\mathbf{x}}$ is used as a weighting matrix of the performance measure $\hat{D}_L(q)$ to model the uncertainty of the expert demonstrations. If the expert demonstrations show a smaller covariance at a region, it implies that the region is highly constrained. Thus, any small deviation $(\mathbf{x}_L(q) - \hat{\boldsymbol{\mu}}_L^{e,\mathbf{x}})$ may cause higher $\hat{D}_L(q)$ which would induce a smaller control authority to the novice, and vice versa. This design ensures that the shared controller reflects the variability in the expert demonstrations.

3.3 Stochastic-Skill-Level-Based Assistance

The core part of designing the hybrid shared control is to determine the guard condition, which represents the mode transition condition in the hybrid system. The discrete mode q_k is one-to-one mapped into the control authority α_k and the mode transition from q_{k-1} to q_k is governed by the guard condition. Note that if $\alpha_k = 0$ for every time step k, then the system is fully controlled by the autonomy, and the performance measure is optimized. However, this is not a desirable case for human training. The proposed framework is designed to

minimize the intervention from the autonomy during the human training so that novices can learn through their own control experience. Thus, the control authority α_k needs to be maintained high enough when the novice can perform a given task *similar enough* to the expert performance. In this context, the guard condition is designed to guarantee a certain confidence level of similarity between the expert performance and the novice performance while maximizing the control authority of the novice:

$$\alpha_k = \max \alpha(q) \quad \text{such that} \quad P(\hat{D}_L(q) < \bar{D}) > \bar{P},\ q \in \{1, \cdots, N_q\} \tag{32}$$

where $\bar{D}$ denotes the largest Mahalanobis distance of the expert demonstrations. $\bar{P} \in [0, 1]$ is the confidence level threshold and it is a tunable design parameter. Note that $\bar{P}$ can be tuned in different training situations: a higher $\bar{P}$ will impose more assistance and vice versa. Then, the discrete mode q_k at time step k is determined by an inverse mapping

$$q_k = \alpha^{-1}(\alpha_k) \tag{33}$$

since $\alpha_k = \alpha(q_k)$ is an one-to-one mapping. Note that the control authority can be updated in real-time. The probability $P(\hat{D}_L(q) < \bar{D})$ in (32) can be computed using the generalized Chi-square (GCS) method which provides a numerical probability density function (PDF) [14]. To compute the PDF, the Mahalanobis distance can be reformulated as:

$$\left(\hat{D}_L(q)\right)^2 = \mathbf{x}_L(q)^T Q_2 \mathbf{x}_L(q) + Q_1 \mathbf{x}_L(q) + Q_0 \tag{34}$$

where

$$\mathbf{x}_L(q) \sim \mathcal{N}\left(\alpha(q)\hat{\boldsymbol{\mu}}_L^{np,\mathbf{x}} + (1-\alpha(q))\mathbf{x}_L^a, \alpha(q)^2 \hat{\Sigma}_L^{np,\mathbf{x}}\right) \tag{35}$$

$$Q_2 = \left(\hat{\Sigma}_L^{e,\mathbf{x}}\right)^{-1}, \qquad Q_1 = -2\left(\hat{\boldsymbol{\mu}}_L^{e,\mathbf{x}}\right)^T \left(\hat{\Sigma}_L^{e,\mathbf{x}}\right)^{-1}, \qquad Q_0 = \left(\hat{\boldsymbol{\mu}}_L^{e,\mathbf{x}}\right)^T \left(\hat{\Sigma}_L^{e,\mathbf{x}}\right)^{-1} \hat{\boldsymbol{\mu}}_L^{e,\mathbf{x}} \tag{36}$$

and the standard GCS method can provide the PDF of the Mahalanobis distance which represents the SSL in the proposed framework. Then, the computed PDF can be used to determine the control authority α_k by solving (32). Note that $(\hat{D}_L(q) < \bar{D}) \Leftrightarrow ((\hat{D}_L(q))^2 < \bar{D}^2)$ since $\hat{D}_L(q)$ and $\bar{D}$ are positive.

4 HUMAN SUBJECT EXPERIMENT

4.1 Purpose and Hypothesis

The purpose of the human subject experiment is to demonstrate the proposed framework in the complex UAM vehicle control scenario with the environment described in Section 2. Participants are randomly divided into three groups. The first group is a control group that is not assisted by any method during their practice. The second group is assisted by a baseline shared control approach called Maxwell's Demon Algorithm (MDA), which is known to aid human learning [7]. The baseline approach is a deterministic and non-personalized hybrid shared control method. The MDA has only two modes: MDA fully accepts human input if it is in the same half-hyperplane (in $\mathbb{R}^m$) with assistant input (i.e., $\alpha_k = 1$), or MDA discards human input otherwise (i.e., $\alpha_k = 0$ and set $\mathbf{u}_k = 0$). The MDA only blocks particularly bad input from the human but does not provide any assistant input. The MDA is formally presented in Appendix B. The third group is assisted by the proposed framework. A hypothesis to be investigated is given as follows.

- Hypothesis: the proposed framework can improve the human learning rate, in terms of the immediate skill retention, in the complex UAM vehicle control scenario in Section 2 compared to the practice-only approach and the baseline approach.

4.2 Participants

A total of 40 human subjects were selected as novices for the human subject study, which is approved by the Institutional Review Board at Purdue University (IRB-2020-755). Participants who can pass through more than 50% of waypoints at their initial trial were excluded. Novice participants were randomly divided into three groups.

- Group 1 (practice-only group): 10 novices who are not assisted during training.
- Group 2 (baseline group): 15 novices who are assisted by the baseline approach during training.
- Group 3 (proposed group): 15 novices who are assisted by the proposed framework during training.

Two human subjects who can pass through all waypoints without failure for more than 10 consecutive trials and touch the target area in Fig. 1d with stable speed and attitude were selected as the experts to provide expert demonstrations.

4.3 Procedures

An experiment instruction was explained in advance to all participants so that they had the same information regarding the task, basic control technique, and expectations. All participants were allowed three minutes to familiarize themselves with the flight controller and the environment. We did not request the participants to fly fast but fly accurately. The only constraint for speed was to complete each trial within three minutes, otherwise, the trial was terminated in the middle and data was stored up to that point. There were only three trials out of 1,000 trials of terminating in the middle due to timeout, and all of them were included in the experimental data. The experimental procedure for each individual was separated into the following three phases:

- Phase 1 (Initial performance): All groups conduct the task 10 times by manual control. Their initial performance measures before any training are recorded. The proposed framework exploits the participant demonstrations to estimate parameters for the novice model. A five minutes break follows Phase 1.
- Phase 2 (Training): All groups conduct the task 10 times. Group 1 performs manual control. Group 2 is assisted by the baseline approach. Group 3 is assisted by the proposed framework during their trials. For Group 3, their novice models are automatically updated after each trial by incorporating the latest trial and excluding the oldest trial. A five minutes break follows Phase 2.
- Phase 3 (Final performance): All groups conduct the task five times by manual control. It is expected that their performance measures are improved compared to their initial performance measures.

The control authority was allocated from finite sets for Group 2 and Group 3. For Group 2, the control authority was determined in a finite set $\mathcal{A}_{g2} = \{0.0, 1.0\}$: full autonomous control when $\alpha_k = 0.0$ or full manual control when $\alpha_k = 1.0$. Participants can recognize the current control authority via vibration cues: when the baseline approach intervenes a participant ($\alpha_k = 0.0$), the flight controller vibrates, and there is no vibration for the manual control mode ($\alpha_k = 1.0$). For Group 3, the control authority was determined in a finite set $\mathcal{A}_{g3} = \{0.1, 0.5, 1.0\}$. Note that the minimum control authority $\alpha_k = 0.1$ was set to prevent abuse of the autonomy. Vibration cues were given with respect to the control authority: a high-frequency vibration for the first mode ($\alpha_k = 0.1$), a low-frequency vibration for the second mode ($\alpha_k = 0.5$), and no vibration for the manual control mode ($\alpha_k = 1.0$). All participants in Group 3 reported that they can distinguish the vibration cues during the instruction. The threshold for the Mahalanobis distance $\bar{D} = 9.65$ in (32) was obtained from the worst Mahalanobis distance of expert demonstrations. The confidence level threshold $\bar{P} = 0.5$ was selected by our pilot study. These thresholds can be interpreted as the assistance will be given only if the confidence level that the novice is worse than the expert's worst case is more than $1 - \bar{P}$. The finite-time horizon for the shared control scheme was set to $l = 40$ which is equivalent to two seconds in real-time. A dwell-time of three seconds was imposed on the mode transitions in Group 2 and Group 3 to prevent frequent mode changes.

After each trial, the participants were instructed to answer the subjective survey form that evaluates their *self-confidence* and *workload* for each trial on an integer scale of 1-10 [26, 63]. During the experiment, physical variables such as position, velocity, and acceleration were recorded, as well as human input variables such as roll, pitch, yaw rate, and thrust input. The SSL-based shared control parameters (e.g., α_k and D_k) were also recorded.

4.4 Performance Measures

The following performance measures are used to quantify the results.

- The Mahalanobis distance D_k in (28), for all k, quantifies the weighted tracking error to the expert nominal trajectory. This value is computed after applying the dynamic time wrapping (DTW) [11] so that the expert's and novice's trajectories are temporally aligned with each other.
- The minimum distance to each waypoint, d_i^w for all $i \in \{1, 2, \cdots, 6\}$, measures how well novices perform the high-level task of passing through the waypoints.
- The final stage performance denotes the Mahalanobis distance D_k after passing the last waypoint. It represents the ability of the novice to stabilize and slow down the vehicle for safe landing near the final target.
- The learning rate is used to fit the performance improvement over the training trials. The power function ($y = ax^{-b}$) is used for the regression analysis [4]. A larger b value indicates a faster learning rate.
- The averaged control authority $\alpha_{\text{avg}} \triangleq 1/T \sum_{k=0}^{T} \alpha_k$ represents control authority of the novices for their training in Phase 2. A higher averaged control authority means that more human control is engaged in Phase 2.
- Two self-reported cognitive states, self-confidence and workload, are used to measure the user experience [63].
- The human control efforts $\mathbf{u}_{\text{eff}} \triangleq \sum_{k=0}^{T} \mathbf{u}_k^T \mathbf{u}_k$ during the task is recoreded. The control input $\mathbf{u}_k \in [-1, 1]^4$ is constrained in roll, pitch, yaw rate, and throttle commands.
- Mean and standard deviation of the mission completion time T in each phase are used to examine execution time and consistency.

4.5 Results

The Mahalanobis distance D_k is compared in three groups to answer the research question of whether the proposed framework can enhance the human learning rate. In Fig. 6, the box plots are used to visualize the means of the Mahalanobis distance of each participant. The analysis of variance (ANOVA) is used for statistical testing [27]. The two-way mixed ANOVA is used to determine how performance is affected by two variables, phase and group. One-way repeated measures ANOVAs are used to examine the training results between phases within each group. Note that the one-way repeated measures ANOVAs are presented in each figure. Conventional symbols are used to represent p-values in figures: $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$, and $^{****}p < 0.0001$. The ANOVA result reveals that there is no significant two-way interaction between group and phase on the Mahalanobis distance, $F(2.63, 48.71) = 0.94, p = 0.42, \eta^2 = 0.005$. There is a significant main effect of phase, $F(1.32, 48.71) = 55.755, p < 0.0001, \eta^2 = 0.135$. All pairwise comparisons are analyzed between each group. Only Group 1 and Group 3 show a significant difference of the Mahalanobis distance ($p = 0.0162$). Group 2 is not significantly different from Group 1 ($p = 0.374$) and Group 3 ($p = 0.0863$). The one-way repeated measures ANOVAs in Fig. 6 show that Group 3 has better training results compared to the others.

The minimum distance to each waypoint d_i^w is used to compare three groups in terms of the performance of the high-level task, passing through the waypoints. The expert performance was 0.689 m on average. As participants are trained through Phase 2, their mean, median, and variance are all reduced in Fig. 7. There is no significant two-way interaction between group and phase on the minimum distance to waypoints, $F(3.25, 60.2) = 0.93, p = 0.44, \eta^2 = 0.006$. There is a significant main effect of phase, $F(1.63, 60.2) = 41.948, p < 0.0001, \eta^2 = 0.117$. The pairwise comparison results between each group reveal that Group 3 is significantly different from Group 1 ($p = 0.018$) and Group 2 ($p = 0.0194$). Group 1 and Group 2 are not significantly different ($p = 0.78$). The one-way repeated measures ANOVAs results in Fig. 7 indicate that Group 3 outperforms other groups.

The final stage performance is a subset of the Mahalanobis distance result. The final stage is different from the rest of the trajectory since it requires slowing down the vehicle to gently touch the target. In Fig. 8, all groups show improvement from Phase 1 to Phase 3, but only Group 3 shows a significant difference. There is no significant two-way interaction between group and phase on the final stage performance, $F(3.03, 55.98) = 1.25, p = 0.3, \eta^2 = 0.02$. There is a significant main effect of phase, $F(1.51, 55.98) = 14.167, p < 0.0001, \eta^2 = 0.093$. The pairwise comparison results show that Group 1 and Group 3 are significantly different ($p = 0.0485$). Group 2 is not significantly different from Group 1 ($p = 0.44$) and Group 3 ($p = 0.176$). The one-way repeated ANOVAs in Fig. 8 show that Group 3 performs better than Group 1 and Group 2.

A regression analysis using the power function fitting technique is used to quantify the human learning rate. The power function $y = ax^{-b}$ is widely adopted to measure the learning rate [4], and it shows the highest R^2 value among the linear, power, and exponential function fitting for the presented experiment results. A bigger b indicates a higher learning rate. The learning rate is estimated using each raw data over a total of 25 trials. The one-way ANOVA method is used for comparing each group in Fig. 9. The results show that the learning rate differences (i.e., fitted b values) are not significant except for the final stage, between Group 1 and Group 3 ($p = 0.0293$). The results show that the fitted power functions are not significantly different, even if the performance improvements show significant differences in Fig. 6-8. Nevertheless, Group 3 still shows higher mean and median values of b for all the cases.

Fig. 10a represents the averaged control authority α_{avg} over Phase 2 of each group. Larger values indicate that the humans take more control authority over the system. Group 1 has full control authority (no assistance), i.e.,

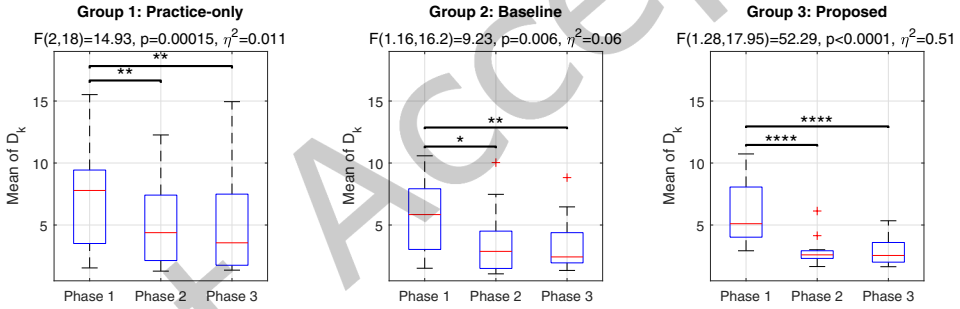


Fig. 6. Box plots of mean of the Mahalanobis distance (D_k) for each participant.

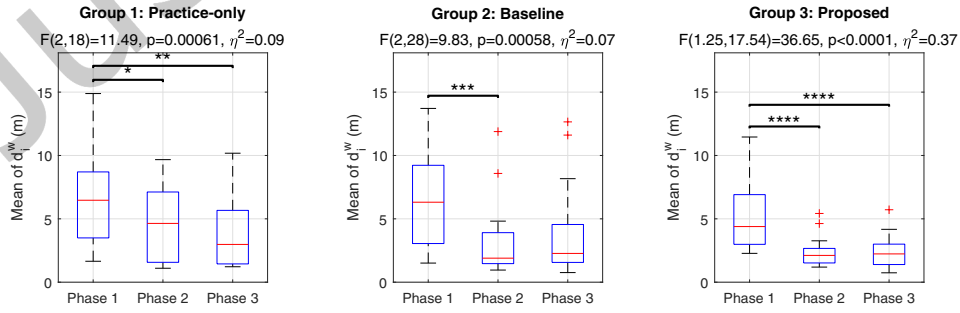


Fig. 7. Box plots of mean of minimum distance to waypoints (d_i^w) for each participant.

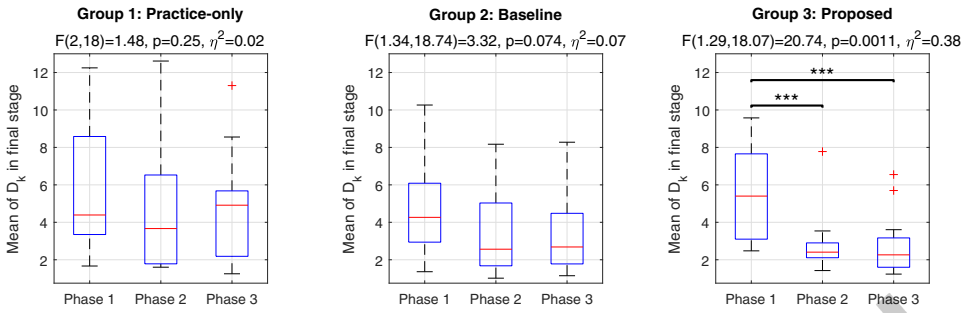


Fig. 8. Box plots of mean of final stage performance for each participant.

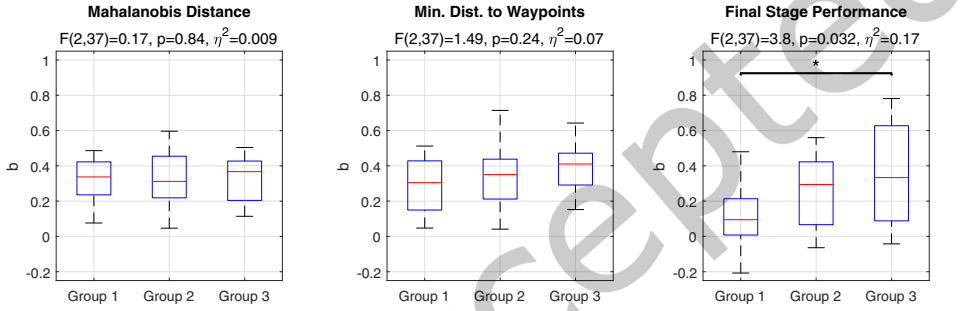
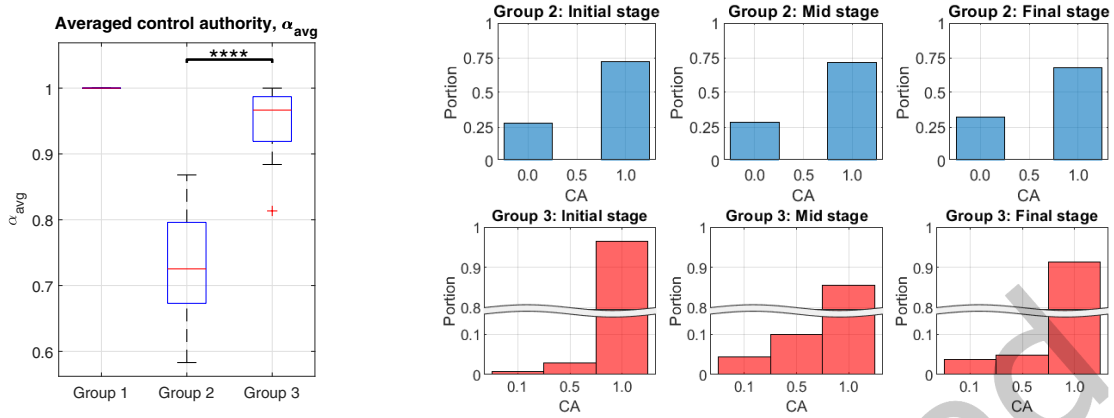


Fig. 9. Box plots of the learning rate of each group using the power function $y = ax^{-b}$.

$\alpha_k = 1.0$ for all k. Group 2 takes significantly less control authority during the training (lower α_{avg}) compared to Group 3. However, its performance improvement is still worse than that of Group 3. This fact indicates that the proposed framework can infer the SSL and adjust the level of assistance appropriately such that the human learning rate is increased while allowing more control authority to the humans. Fig. 10b shows the time portion of each control authority (discrete mode) in Group 2 and Group 3. Each trial is divided into three segments to get more detailed information, from the start to the third waypoint (initial stage), from the third waypoint to the last waypoint (mid stage), and from the waypoint to landing area (final stage). Some participants reported that a curved trajectory in the mid stage is difficult compared to the initial and final stage with the straight trajectories. The proposed framework can reflect the task difficulties since it provides more assistance (i.e., less control authority to the humans) in the mid stage to Group 3 as shown in Fig. 10b.

The cognitive states are investigated using a subjective survey form. The surveyed values are normalized using the z-score method [63]. Fig. 11 shows similar trends in all groups: self-confidence is increasing, and workload is decreasing over phases, respectively. In the self-confidence survey, there is no statistically significant two-way interaction between group and phase, $F(2.97, 54.95) = 0.21, p = 0.89, \eta^2 = 0.003$. There is a significant main effect of phase, $F(1.49, 54.95) = 77.682, p < 0.0001, \eta^2 = 0.385$. One observed point is that Group 3 is the only group that shows a significant difference in post-training, from Phase 2 to Phase 3 ($p = 0.018$). In the workload survey, there is no significant two-way interaction between group and phase, $F(4, 74) = 0.98, p = 0.42, \eta^2 = 0.02$. However, there is a significant main effect of phase, $F(2, 74) = 29.172, p < 0.0001, \eta^2 = 0.249$. Group 3 reported a significantly lower workload in Phase 2 compared to Group 1 ($p = 0.0238$) and Group 2 ($p = 0.0277$). Some



(a) Averaged control authority.

(b) Portion of times of the different discrete modes (control authority).

Fig. 10. Control authority (CA) in Phase 2.

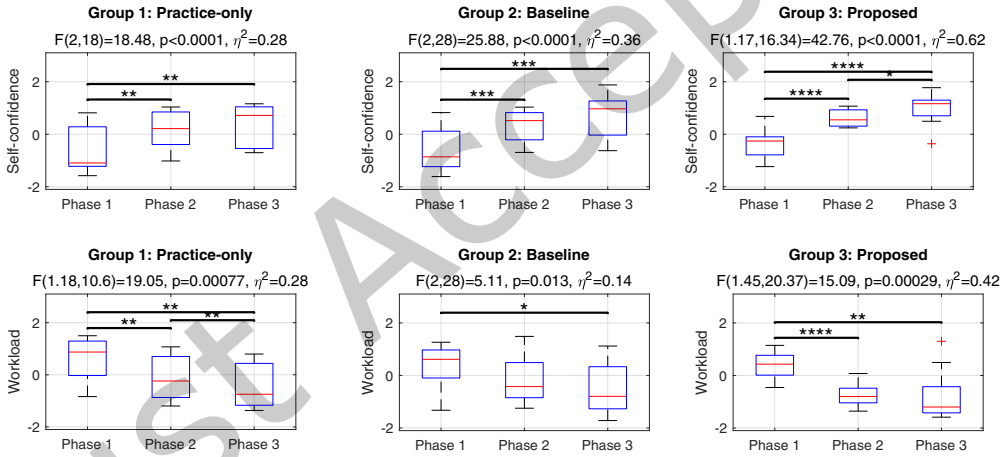


Fig. 11. Self-reported cognitive states: self-confidence and workload.

participants in Group 2 described that the baseline shared controller intervenes too frequently, and they tried to explore the right control input to avoid the intervention. This may cause a higher workload in Phase 2 for Group 2. This observation coincides with the average control authority result in Fig. 10a since Group 2 shows the lower averaged control authority (more intervention from autonomy) in Phase 2. The results indicate that the proposed framework could reduce workload during the training. Another interesting observation is that Group 3 shows lower variability in self-confidence and workload. This observation can be interpreted as a result of the personalized nature of the proposed method, but further investigation is required to conclude. We plan to examine the correlation between workload and training with further experiments.

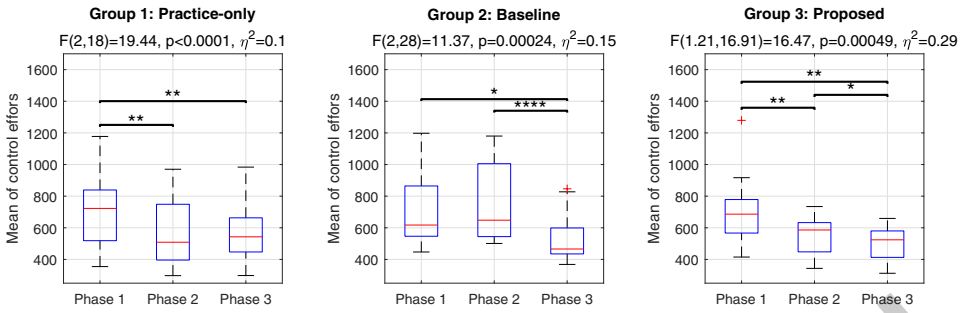


Fig. 12. Mean of control effort ($\mathbf{u}_{\text{eff}}$).

The control effort is inspected to monitor the behavioral changes of the participants over the phases. There is significant two-way interaction between group and phase, $F(2.78, 51.49) = 6.06, p = 0.002, \eta^2 = 0.06$. All groups show significant main effects of the phase as shown in Fig. 12. Group 2 demonstrates significantly higher control efforts in Phase 2 compared to Group 1 ($p = 0.0195$) and Group 3 ($p = 0.0102$). Some Group 2 participants reported that the assistant scheme was too restrictive for them, and they tried to avoid vibration which was regarded as *punishment* instead of assistance (details in Section 5). They also pointed out that they had explored the control space to stop the assistance and vibration in Phase 2. The exploring behaviors can be interpreted as adaptive behaviors to the changed nature of the task due to the baseline approach. This aspect is also reflected in the workload survey in Fig. 11 since Group 2 reported a higher workload in Phase 2 compared to Group 3. On the other hand, Group 3 made significantly lower control efforts in Phase 2 compared to Phase 1. This fact shows that the proposed framework does not change the nature of the training task, and thus, it did not cause the exploring behaviors in Phase 2.

The mission completion time should be carefully analyzed since the participants were not requested to finish the task fast but accurately. In our pilot study, some participants tended to find shortcuts while ignoring the waypoints. Those strategies are not appropriate for UAM operations especially near the landing area due to safety concerns. Thus, we particularly requested the participants to follow the nominal trajectory. Nevertheless, our intuitive prediction was that the participants will fly faster and be consistent in multiple trials as they have more experience. In Fig. 13, all groups show decreasing trends in terms of the mean and the standard deviation of the mission completion time. There is no significant two-way interaction on the mean mission completion time, $F(3.22, 59.65) = 1.32, p = 0.28, \eta^2 = 0.01$. Only Group 2 does not show a significant difference between Phase 1 and Phase 3. The results show that Group 3 performed better than other groups in mission completion time reduction. In the bottom of Fig. 13, the standard deviations of the mission completion time decrease over phases. There is no significant two-way interaction on the standard deviation of the mission completion time, $F(4, 74) = 2.22, p = 0.075, \eta^2 = 0.04$. Nevertheless, Group 3 shows the most significant main effects of the phase compared to Group 1 and Group 2 as shown in Fig. 13. Since consistency in time is another feature of expert demonstrations (e.g., the standard deviation of one expert was 2.18 *sec*), this result indicates that Group 3 successfully emulated expert demonstrations even without a specific direction in mission completion time.

5 DISCUSSION

The SSL-based shared control framework was demonstrated in the UAM vehicle control training task. The human subject experiment results reveal that the proposed framework can expedite human training in a specific scenario compared to practice-only (no assistance) scheme and the baseline shared control scheme. In particular, similarity

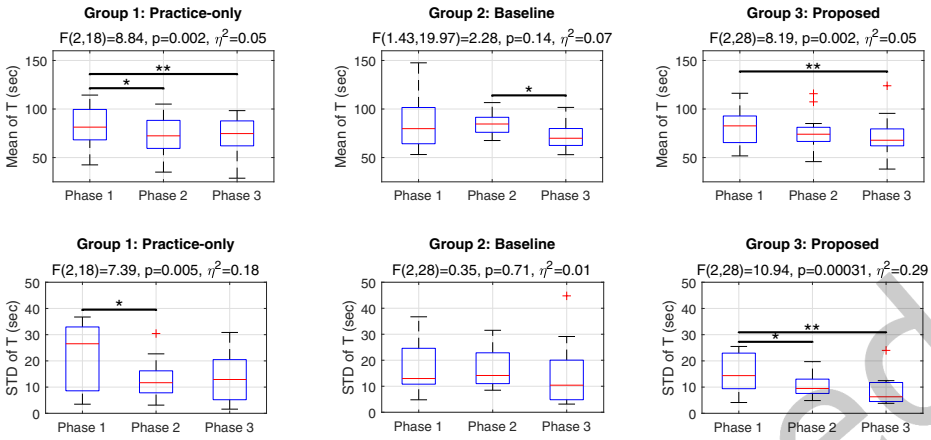


Fig. 13. Mission completion time (T) in mean and standard deviation (STD).

to expert performance increased more significantly in Group 3 (assisted by the proposed framework) accompanied by an improvement in other metrics like control effort and mission completion time. In subjective cognitive states report, the proposed framework further reduced the training workload compared to the baseline approach. The results reveal that our SSL inference approach and SSL-based shared control scheme can facilitate better training environments by realizing the OCP [23] in complex dynamic control tasks. The human subject experiment results are valuable since the adaptive training environment methods are not necessarily result in improving the human learning rate [3]. Our results present significant differences between the control group, baseline group, and the proposed framework group. This fact shows that the proposed framework can be a foundation for further investigations in SSL-based approaches for human training.

However, we note that the proposed framework may not be always effective for general training scenarios. It is fair to say that the proposed framework tested a specific hypothesis as described in Section 4.1. Indeed, our primary contributions are the design and demonstration of the SSL-based shared control for complex dynamic tasks. Any claim on the human learning rate, on the other hand, may require more experimental data and depend on tasks [49]. We provide several points which can be tested with further investigations in skill training: long-term skill retention, dynamic training environments, and multi-modal feedback. Long-term skill retention can be tested by collecting performance measures after a certain time after the training [35]. It can test whether human subjects maintain and retain their skills. Dynamic training environments can be imposed to test whether human subjects can transfer their capabilities to cope with different situations, such as tracking different trajectories and avoiding dynamic obstacles [61]. Random errors in the experiment environments (e.g., disturbance) can be imposed to investigate the human learning rate under error-augmented situations [61]. Multi-modal feedback, including haptic feedback [49, 61] and auditory feedback [52], has been examined in human training. Especially, various haptic feedback schemes have been implemented to assist human learning [1]. Interesting results might be obtained if an experiment is conducted to examine how the human learning rate changes when all the factors mentioned are involved. Nevertheless, the proposed framework is still valuable since the inferred SSL can be useful for further experiments.

In future work, we may investigate the transparency [13, 39] between humans and autonomy for a comprehensive SSL inference. Autonomy may be designed in a better way if it can directly measure, infer, and consider

the human cognitive state. Especially, physio-psychological instruments including but not limited to eye-tracker, electroencephalography (EEG), and Galvanic skin response (GSR). Some participants wanted explanations on how autonomy determined the control authority. If autonomy can convey the information concisely, the acceptance of assistance could be further improved. In the human subject experiment, five participants in Group 2 and Group 3 reported that they want to have more information, for instance, the reasoning for the control authority decision. However, increasing the transparency requires an additional consideration since the higher transparency may induce higher workload to the humans [13]. Finally, we can investigate an online human model update scheme (i.e., update the HSMM online) to cope with potentially fast-changing human characteristics. In our experiments, updating the HSMMs after each trial was enough. However, further fast-changing cases may require online model update [25].

6 CONCLUSION

A stochastic-skill-level-based shared control framework is proposed and a human subject experiment was conducted. The purpose of the proposed framework is to help a human novice emulate a human expert in complex dynamic control tasks. The proposed framework consists of three steps to transfer skills from the expert to the novice. In the first step, the autonomy learns how the expert performs the given task. Then, the autonomy infers the stochastic-skill-level of the novice by comparing the demonstrations of the expert and those of the novice. In the third step, the autonomy allocates the control authority based on the time-varying stochastic-skill-level of the novice. The stochastic-skill-level inference and the control authority allocation are both based on probabilistic approaches which account for the uncertainty of human behavior (i.e., variability). The assistance is given to the novices only when their current stochastic-skill-levels are poor such that the performance measure is not satisfying a stochastic threshold. The human subject experiment results show that the proposed framework can provide a solid foundation for human training in complex dynamic control tasks.

ACKNOWLEDGMENTS

The authors would like to acknowledge that this work is supported by NSF CNS-1836952.

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. https://doi.org/10.1109/THMS.2018.2791570

[2] David A. Abbink, Mark Mulder, and Erwin R. Boer. 2012. Haptic Shared Control: Smoothly Shifting Control Authority? *Cognition, Technology & Work* 14, 1 (Mar 2012), 19–28. https://doi.org/10.1007/s10111-011-0192-5

[3] Priyanshu Agarwal and Ashish D. Deshpande. 2019. A Framework for Adaptation of Training Task, Assistance and Feedback for Optimizing Motor (Re)-Learning With a Robotic Exoskeleton. *IEEE Robotics and Automation Letters* 4, 2 (2019), 808–815. https://doi.org/10.1109/LRA.2019.2891431

[4] Negin Amirshirzad, Asiye Kumru, and Erhan Oztop. 2019. Human Adaptation to Human–Robot Shared Control. *IEEE Transactions on Human-Machine Systems* 49, 2 (2019), 126–136. https://doi.org/10.1109/THMS.2018.2884719

[5] General Aviation Manufacturers Association. 2019. *A Rational Construction for Simplified Vehicle Operations (SVO).* GAMA EPIC SVO Subcommittee Whitepaper, Version 1.0, Copyright 2019, Washington, DC, USA.

[6] Stephen Boyd and Lieven Vandenberghe. 2004. *Convex Optimization.* Cambridge University Press.

[7] Alexander Broad, Ian Abraham, Todd Murphey, and Brenna Argall. 2020. Data-driven Koopman operators for model-based shared control of human–machine systems. *The International Journal of Robotics Research* 39, 9 (2020), 1178–1195. https://doi.org/10.1177/0278364920921935

[8] Sooyung Byeon, Wanxin Jin, Dawei Sun, and Inseok Hwang. 2021. Human-Automation Interaction for Assisting Novices to Emulate Experts by Inferring Task Objective Functions. In *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC).* 1–6. https://doi.org/10.1109/DASC52595.2021.9594324

[9] Sooyung Byeon, Dawei Sun, and Inseok Hwang. 2021. Skill-level-based Hybrid Shared Control for Human-Automation Systems. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC).* 1507–1512. https://doi.org/10.1109/SMC52423.2021.9658994

[10] Sylvain Calinon. 2016. A Tutorial on Task-Parameterized Movement Learning and Retrieval. *Intelligent Service Robotics* 9, 1 (2016), 1–29. https://doi.org/10.1007/s11370-015-0187-9

[11] Sylvain Calinon, Florent D'halluin, Eric L. Sauser, Darwin G. Caldwell, and Aude G. Billard. 2010. Learning and Reproduction of Gestures by Imitation. *IEEE Robotics & Automation Magazine* 17, 2 (2010), 44–54. https://doi.org/10.1109/MRA.2010.936947

[12] Sylvain Calinon, Antonio Pistillo, and Darwin G. Caldwell. 2011. Encoding the Time and Space Constraints of a Task in Explicit-Duration Hidden Markov Model. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 3413–3418. https://doi.org/10.1109/IROS.2011.6094418

[13] Jessie Y. C. Chen, Shan G. Lakhmani, Kimberly Stowers, Anthony R. Selkowitz, Julia L. Wright, and Michael Barnes. 2018. Situation Awareness-based Agent Transparency and Human-Autonomy Teaming Effectiveness. *Theoretical Issues in Ergonomics Science* 19, 3 (2018), 259–282. https://doi.org/10.1080/1463922X.2017.1315750

[14] Abhranil Das and Wilson S. Geisler. 2021. A Method to Integrate and Classify Normal Distributions. *Journal of Vision* 21, 10 (Sep 2021), 1–1. https://doi.org/10.1167/jov.21.10.1

[15] Chuhao Deng, Hong-Cheol Choi, Hyunsang Park, and Inseok Hwang. 2022. Trajectory Pattern Identification and Classification for Real-Time Air Traffic Applications in Area Navigation Terminal Airspace. *Transportation Research Part C: Emerging Technologies* 142 (2022), 103765. https://doi.org/10.1016/j.trc.2022.103765

[16] Anca D Dragan and Siddhartha S Srinivasa. 2013. A Policy-Blending Formalism for Shared Control. *The International Journal of Robotics Research* 32, 7 (2013), 790–805. https://doi.org/10.1177/0278364913490324

[17] Mustafa Suphi Erden and Ho-Tak D. Chun. 2018. Muscle Activity Patterns Change with Skill Acquisition for Minimally Invasive Surgery: A Pilot Study. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*. 960–965. https://doi.org/10.1109/BIOROB.2018.8487865

[18] FAA. 2020. *UAM Concept of Operations (ConOps) version 1.0.* Federal Aviation Administration.

[19] Frank Flemisch, Matthias Heesen, Tobias Hesse, Johann Kelsch, Anna Schieben, and Johannes Beller. 2012. Towards a Dynamic Balance between Humans and Automation: Authority, Ability, Responsibility and Control in Shared and Cooperative Control Situations. *Cognition, Technology & Work* 14 (2012), 3–18. https://doi.org/10.1007/s10111-011-0191-6

[20] Benjamin Gautier, Harun Tugal, Benjie Tang, Ghulam Nabi, and Mustafa Suphi Erden. 2019. Laparoscopy Instrument Tracking for Single View Camera and Skill Assessment. In *2019 International Conference on Robotics and Automation (ICRA)*. 5039–5045. https://doi.org/10.1109/ICRA.2019.8794038

[21] Keya Ghonasgi, Reuth Mirsky, Adrian M. Haith, Peter Stone, and Ashish D. Deshpande. 2022. Quantifying Changes in Kinematic Behavior of a Human-Exoskeleton Interactive System. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 10734–10739. https://doi.org/10.1109/IROS47612.2022.9981032

[22] Keya Ghonasgi, Reuth Mirsky, Sanmit Narvekar, Bharath Masetty, Adrian M. Haith, Peter Stone, and Ashish D. Deshpande. 2021. Capturing Skill State in Curriculum Learning for Human Skill Acquisition. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 771–776. https://doi.org/10.1109/IROS51168.2021.9636850

[23] Mark A Guadagnoli and Timothy D Lee. 2014. Challenge Point: A Framework for Conceptualizing the Effects of Various Practice Conditions in Motor Learning. *Journal of motor behavior* 36, 2 (2014), 212–224. https://doi.org/10.3200/JMBR.36.2.212-224

[24] Caroline N. Haddad. 2008. *Encyclopedia of Optimization: Cholesky Factorization.* Springer, Boston, MA. 374–376 pages. https://doi.org/10.1007/978-0-387-74759-0_67

[25] Ioannis Havoutis, Ajay Kumar Tanwani, and Sylvain Calinon. 2018. Online Incremental Learning of Manipulation Tasks for Semi-Autonomous Teleoperation. *International Conference on Intelligent Robots and Systems, Workshop on Closed Loop Grasping and Manipulation: Challenges and Progress.*

[26] David Kaber and Mica Endsley. 2004. The Effects of Level of Automation and Adaptive Automation on Human Performance, Situation Awareness and Workload in a Dynamic Control Task. *Theoretical Issues in Ergonomics Science* 5 (Mar 2004), 113–153. https://doi.org/10.1080/1463922021000054335

[27] Damir Kalpić, Nikica Hlupić, and Miodrag Lovrić. 2011. *Student's t-Tests.* Springer Berlin Heidelberg, Berlin, Heidelberg, 1559–1563. https://doi.org/10.1007/978-3-642-04898-2_641

[28] Avi Karni, Gundela Meyer, Christine Rey-Hipolito, Peter Jezzard, Michelle M. Adams, Robert Turner, and Leslie G. Ungerleider. 1998. The Acquisition of Skilled Motor Performance: Fast and Slow Experience-Driven Changes in Primary Motor Cortex. *Proceedings of the National Academy of Sciences* 95, 3 (1998), 861–868. https://doi.org/10.1073/pnas.95.3.861

[29] Shreeharsh Kelkar. 2022. Between AI and Learning Science: The Evolution and Commercialization of Intelligent Tutoring Systems. *IEEE Annals of the History of Computing* 44, 01 (Jan 2022), 20–30. https://doi.org/10.1109/MAHC.2022.3143816

[30] Basil Kouvaritakis and Mark Cannon. 2016. *Model Predictive Control: Classical, Robust and Stochastic.* Springer, Cham, Switzerland. 13–64 pages. https://doi.org/10.1007/978-3-319-24853-0

[31] Ayse Kucukyilmaz, Tevfik Metin Sezgin, and Cagatay Basdogan. 2013. Intention Recognition for Dynamic Role Exchange in Haptic Collaboration. *IEEE Transactions on Haptics* 6, 1 (2013), 58–68. https://doi.org/10.1109/TOH.2012.21

[32] Minae Kwon, Erdem Biyik, Aditi Talati, Karan Bhasin, Dylan P. Losey, and Dorsa Sadigh. 2020. When Humans Aren't Optimal: Robots That Collaborate with Risk-Aware Humans. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction* (Cambridge, United Kingdom) *(HRI '20)*. Association for Computing Machinery, New York, NY, USA, 43–52. https://doi.org/10.1145/3319502.3374832

[33] Dongjun Lee, Antonio Franchi, Hyoung Il Son, ChangSu Ha, Heinrich H. Bülthoff, and Paolo Robuffo Giordano. 2013. Semiautonomous Haptic Teleoperation Control Architecture of Multiple Unmanned Aerial Vehicles. *IEEE/ASME Transactions on Mechatronics* 18, 4 (2013), 1334–1345. https://doi.org/10.1109/TMECH.2013.2263963

[34] Seung Jae Lee, Seung Hyun Kim, and Hyoun Jin Kim. 2020. Robust Translational Force Control of Multi-Rotor UAV for Precise Acceleration Tracking. *IEEE Transactions on Automation Science and Engineering* 17, 2 (2020), 562–573. https://doi.org/10.1109/TASE.2019.2935792

[35] Yanfang Li, Joel C. Huegel, Volkan Patoglu, and Marcia K. O'Malley. 2009. Progressive Shared Control for Training in Virtual Environments. In *World Haptics 2009 - Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. 332–337. https://doi.org/10.1109/WHC.2009.4810873

[36] Yanfang Li, Volkan Patoglu, and Marcia K. O'Malley. 2009. Negative Efficacy of Fixed Gain Error Reducing Shared Control for Training in Virtual Environments. *ACM Transactions on Applied Perception* 6, 1, Article 3 (Feb 2009), 21 pages. https://doi.org/10.1145/1462055.1462058

[37] Daniel Liberzon. 2003. *Switching in Systems and Control*. Springer. arXiv:https://link.springer.com/book/10.1007/978-1-4612-0017-8

[38] Jonathan Feng-Shun Lin, Pamela Carreno-Medrano, Mahsa Parsapour, Maram Sakr, and Dana Kulić. 2021. Objective Learning from Human Demonstrations. *Annual Reviews in Control* 51 (2021), 111–129. https://doi.org/10.1016/j.arcontrol.2021.04.003

[39] Joseph B. Lyons, Katia Sycara, Michael Lewis, and August Capiola. 2021. Human–Autonomy Teaming: Definitions, Debates, and Directions. *Frontiers in Psychology* 12 (2021). https://doi.org/10.3389/fpsyg.2021.589585

[40] Mauricio Marcano, Sergio Díaz, Joshué Pérez, and Eloy Irigoyen. 2020. A Review of Shared Control for Automated Vehicles: Theory and Applications. *IEEE Transactions on Human-Machine Systems* 50, 6 (2020), 475–491. https://doi.org/10.1109/THMS.2020.3017748

[41] Akshay Mathur, Karanvir Panesar, Joseph Kim, Ella M. Atkins, and Nadine Sarter. 2019. Paths to Autonomous Vehicle Operations for Urban Air Mobility. In *AIAA Aviation 2019 Forum*. Dallas, Texas, USA. https://doi.org/10.2514/6.2019-3255

[42] Marie-Hélène Milot, Laura Marchal-Crespo, Christopher S. Green, Steven C. Cramer, and David J. Reinkensmeyer. 2010. Comparison of Error-Amplification and Haptic-Guidance Training Techniques for Learning of a Timing-based Motor Task by Healthy Individuals. *Experimental Brain Research* 201, 2 (2010), 119–131. https://doi.org/10.1007/s00221-009-2014-z

[43] Todd K. Moon. 1996. The Expectation-Maximization Algorithm. *IEEE Signal Processing Magazine* 13, 6 (1996), 47–60. https://doi.org/10.1109/79.543975

[44] Christopher E Mower, Joao Moura, and Sethu Vijayakumar. 2021. Skill-based Shared Control. In *Robotics: Science and Systems*. 1–10. https://doi.org/10.15607/RSS.2021.XVII.028

[45] Sarah M. O'Meara, John A. Karasinski, Casey L. Miller, Sanjay S. Joshi, and Stephen K. Robinson. 2021. Effects of Augmented Feedback and Motor Learning Adaptation on Human–Automation Interaction Factors. *Journal of Aerospace Information Systems* 18, 6 (2021), 377–390. https://doi.org/10.2514/1.I010915

[46] Karanvir Panesar, Akshay Mathur, Ella Atkins, and Nadine Sarter. 2021. Moving From Piloted to Autonomous Operations: Investigating Human Factors Challenges in Urban Air Mobility. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 65, 1 (2021), 241–245. https://doi.org/10.1177/1071181321651143

[47] Volkan Patoglu, Yvonne Li, and Marcia K. O'Malley. 2009. On the Efficacy of Haptic Guidance Schemes for Human Motor Learning. In *World Congress on Medical Physics and Biomedical Engineering, September 7 - 12, 2009, Munich, Germany*, Olaf Dössel and Wolfgang C. Schlegel (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 203–206.

[48] Ali Utku Pehlivan, Dylan P. Losey, and Marcia K. O'Malley. 2016. Minimal Assist-as-Needed Controller for Upper Limb Robotic Rehabilitation. *IEEE Transactions on Robotics* 32, 1 (2016), 113–124. https://doi.org/10.1109/TRO.2015.2503726

[49] Dane Powell and Marcia K. O'Malley. 2012. The Task-Dependent Efficacy of Shared-Control Haptic Guidance Paradigms. *IEEE Transactions on Haptics* 5, 3 (2012), 208–219. https://doi.org/10.1109/TOH.2012.40

[50] José Ramón Medina, Dongheui Lee, and Sandra Hirche. 2012. Risk-Sensitive Optimal Feedback Control for Haptic Assistance. In *2012 IEEE International Conference on Robotics and Automation*. 1025–1031. https://doi.org/10.1109/ICRA.2012.6225085

[51] Jens Rasmussen. 1983. Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models. *IEEE Transactions on Systems, Man, and Cybernetics* SMC-13, 3 (1983), 257–266. https://doi.org/10.1109/TSMC.1983.6313160

[52] Giulio Rosati, Fabio Oscari, Simone Spagnol, Federico Avanzini, and Stefano Masiero. 2012. Effect of Task-Related Continuous Auditory Feedback During Learning of Tracking Motion Exercises. *Journal of NeuroEngineering and Rehabilitation* 9, 6 (2012). https://doi.org/10.1186/1743-0003-9-79

[53] Albert Sans-Muntadas, Jaime E. Duarte, and David J. Reinkensmeyer. 2014. Robot-Assisted Motor Training: Assistance Decreases Exploration During Reinforcement Learning. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 3516–3520. https://doi.org/10.1109/EMBC.2014.6944381

[54] Reza Shadmehr, Maurice A. Smith, and John W. Krakauer. 2010. Error Correction, Sensory Prediction, and Adaptation in Motor Control. *Annual Review of Neuroscience* 33, 1 (2010), 89–108. https://doi.org/10.1146/annurev-neuro-060909-153135 PMID: 20367317.

[55] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. 2017. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. In *Field and Service Robotics*. arXiv:arXiv:1705.05065 https://arxiv.org/abs/1705.05065
[56] Thomas J. Smith. 2014. Variability in Human Performance – the Roles of Context Specificity and Closed-Loop Control. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 58, 1 (2014), 979–983. https://doi.org/10.1177/1541931214581205
[57] Kai-Tai Song and Ping-Jui Hsieh. 2022. Shared-Control Design for Robot-Assisted Surgery using a Skill Assessment Approach. *Asian Journal of Control* 24, 3 (2022), 1042–1058. https://doi.org/10.1002/asjc.2746
[58] Ken Takiyama, Masaya Hirashima, and Daichi Nozaki. 2015. Prospective Errors Determine Motor Learning. *Nature Communications* 6, 1 (2015). https://doi.org/10.1038/ncomms6925
[59] Davood Tofighi and Craig K. Enders. 2007. *Identifying the Correct Number of Classes in a Growth Mixture Model.* Information Age, Greenwich. 317–341 pages.
[60] James V. Wertsch. 1984. The Zone of Proximal Development: Some Conceptual Issues. *New Directions for Child and Adolescent Development* 1984, 23 (1984), 7–18. https://doi.org/10.1002/cd.23219842303
[61] Camille K Williams, Luc Tremblay, and Heather Carnahan. 2016. It Pays to Go Off-Track: Practicing with Error-Augmenting Haptic Feedback Facilitates Learning of a Curve-Tracing Task. *Frontiers in psychology* 7, 2010 (2016). https://doi.org/10.3389/fpsyg.2016.02010
[62] Chunsheng Yang, Feng-Kuang Chiang, Qiangqiang Cheng, and Jun Ji. 2021. Machine Learning-Based Student Modeling Methodology for Intelligent Tutoring Systems. *Journal of Educational Computing Research* 59, 6 (2021), 1015–1035. https://doi.org/10.1177/0735633120986256
[63] Madeleine Yuh, Sooyung Byeon, Neera Jain, and Inseok Hwang. 2022. A Heuristic Strategy for Cognitive State-based Feedback Control to Accelerate Human Learning. In *4th IFAC Workshop on Cyber-Physical & Human-Systems (CPHS 2022)*.
[64] Mohammad Zohaib. 2018. Dynamic Difficulty Adjustment (DDA) in Computer Games: A Review. *Advances in Human-Computer Interaction* 2018, 5681652 (2018), 1687–5893. https://doi.org/10.1155/2018/5681652

A MAPPING FUNCTION OF MULTI-ROTOR SYSTEM

A.1 Mapping Function From Control Input to Acceleration

Let the current attitude in Euler angle be $\boldsymbol{\phi}_k = [\phi_k, \theta_k, \psi_k]^T$ and let T_k be the current thrust input. Then, the corresponding translational acceleration $\ddot{\mathbf{x}}_k$ is given as [34]:

$$\ddot{\mathbf{x}}_k = -\frac{1}{m_u} R(\psi_k) \begin{bmatrix} \cos\phi_k \sin\theta_k \\ -\sin\phi_k \\ \cos\phi_k \cos\theta_k \end{bmatrix} T_k + g$$

where $R(\psi_k) \in \mathbb{R}^{3\times3}$ denotes the yaw rotation matrix. m_u and g denote the mass of the multi-rotor and the gravitational acceleration, respectively.

A.2 Mapping Function From Acceleration to Control Input

Let the current attitude in Euler angle be $\boldsymbol{\phi}_k = [\phi_k, \theta_k, \psi_k]^T$. Let $\ddot{\mathbf{x}}_{d,k} = [\ddot{x}_{d,k}, \ddot{y}_{d,k}, \ddot{z}_{d,k}]^T$ be the desired acceleration at time step k. Then, the desired control input in roll, pitch, yaw rate, and thrust, i.e., $\mathbf{u}_{d,k} = [\phi_{d,k}, \theta_{d,k}, \dot{\psi}_{d,k}, T_{d,k}]^T \in \boldsymbol{\mathcal{U}} \subset \mathbb{R}^4$ to realize the desired acceleration $\ddot{\mathbf{x}}_{d,k}$ at the current state and time is given as [34]:

$$\theta_{d,k} = \arctan\left(\frac{\ddot{x}_{d,k}}{\ddot{z}_{d,k}}\right), \qquad \phi_{d,k} = \arctan\left(\frac{\ddot{y}_{d,k}}{\sqrt{\ddot{x}_{d,k}^2 + \ddot{z}_{d,k}^2}}\right), \qquad T_{d,k} = -\frac{m_u \ddot{z}_{d,k}}{\cos\phi_k \cos\theta_k}$$

where m_u denotes the mass of the multi-rotor system. Note that yaw can remain zero to obtain the desired translational acceleration $\ddot{\mathbf{x}}_{d,k}$ or be aligned with the heading angle of the multi-rotor system if necessary.

B BASELINE SHARED CONTROL APPROACH

The Maxwell's Demon Algorithm (MDA) is a deterministic and non-personalized shared control scheme [7]. Assume $\mathbf{u}_k^a$ in (21) is available. Then, the shared control law is given by:

$$\mathbf{u}_k = \begin{cases} \mathbf{u}_k^h & \text{if} \quad \mathbf{u}_k^h \times \mathbf{u}_k^a > 0, \\ 0 & \text{otherwise} \end{cases} \tag{37}$$

where $\times$ denotes the dot product.