

Adaptive Hint-Level Selection in Intelligent Tutoring Systems: A Double DQN Study of Policy Behavior and Assistance Trade-Offs

Jingwen Dong, Siti Nurulain Mohd Rum*,

Teh Noranis Mohd Aris, Khairul Azhar Kasmiran, Raihani Mohamed

Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang, Malaysia

Abstract—Intelligent Tutoring Systems must determine not only whether assistance is needed, but also how much support should be provided during problem solving. This study investigates adaptive hint-level selection as a sequential decision problem in simulated mathematics tutoring. The action space consists of Strong Guidance, Partial Guidance, and Minimal Scaffold, and a Double Deep Q-Network (DDQN) policy uses learner and task state information to select among these levels. DDQN was evaluated against Random, Fixed, Rule-based, and conventional DQN policies, with DQN and DDQN assessed across 10 independent runs using held-out greedy evaluation. The results showed that DDQN did not demonstrate a statistically supported performance advantage over DQN on the primary evaluation metrics. DDQN selected Strong Guidance less frequently than DQN (45.54% versus 56.05%), a difference of 10.52 percentage points, but the resulting smaller assistance-related reward penalties did not produce higher overall simulated tutoring performance. DDQN also exhibited differentiated hint-selection patterns across mastery and difficulty states, although Low-mastery observations were sparse and supporting run-level contrasts did not survive multiplicity correction. Increasing difficulty was descriptively associated with less Strong Guidance and more Minimal Scaffold. These findings indicate that state-sensitive adaptation does not necessarily imply superior or pedagogically intuitive tutoring behaviour, highlighting the need to evaluate learned instructional behaviour and assistance-performance trade-offs alongside aggregate reward.

Keywords—Adaptive hint selection; instructional assistance; reinforcement learning; Double Deep Q-Network (DDQN); intelligent tutoring systems

I. INTRODUCTION

Intelligent tutoring systems (ITSs) have become an important approach to providing personalized instructional support by adapting learning experiences to individual learner needs. In mathematics education, such systems can support learners during problem solving through individualized feedback, hints, and scaffolding, allowing instructional support to respond to differences in learners' knowledge and performance [1]. However, effective personalization requires more than simply making assistance available. Learners may require different degrees of support depending on their current knowledge, task difficulty, and ongoing problem-solving performance. Providing insufficient assistance may leave learners unable to make productive progress, whereas excessive guidance may

reduce opportunities for independent reasoning and problem solving. Determining how much assistance should be provided under different learning conditions therefore represents an important challenge for adaptive tutoring. This challenge is closely related to the assistance dilemma, which concerns how intelligent tutoring systems can provide appropriate help while avoiding both inadequate and unnecessary instructional support [2].

Existing research has addressed the assistance dilemma through increasingly adaptive forms of tutoring support. Data-driven approaches have been developed to predict when learners are likely to need help and to trigger proactive hints when unproductive problem-solving behaviour is detected. For example, [3] incorporated learners' prior hint usage into a HelpNeed model to improve decisions about when proactive assistance should be provided, while [2] combined predictions of help need with subgoal-oriented hints to support learners during problem solving. Other studies have explored the selection of assistance actions from multiple candidate interventions based on observed learner behaviour and outcomes. These studies demonstrate the value of moving beyond uniform assistance toward context-sensitive tutoring decisions. However, determining whether or when assistance should be provided does not fully address the question of how much assistance is appropriate. Once support is required, learners with different knowledge states and facing tasks of different difficulty may require different levels of guidance. The fine-grained selection of instructional assistance intensity according to evolving learner and task states therefore constitutes a distinct adaptive decision problem.

Reinforcement learning (RL) provides a suitable framework for addressing this problem because tutoring can be viewed as a sequential interaction in which instructional decisions influence both immediate learner responses and subsequent interaction states. Recent studies have demonstrated the potential of RL and deep reinforcement learning (DRL) for learning adaptive pedagogical policies in intelligent tutoring systems. For example, [4] investigated hierarchical reinforcement learning for pedagogical decisions at both problem and step levels, while [5] applied batch DRL to select learning activities within an intelligent tutoring system. [6] further demonstrated the use of deep RL for adaptive pedagogical support in a mathematical learning task. These studies establish RL as a promising approach to sequential instructional decision making, although their decision targets include learning ac-

*Corresponding author.

tivity selection, problem-level pedagogical choices, and other forms of instructional intervention. In contrast, the present study focuses specifically on the sequential adaptation of hint intensity during problem solving.

To investigate this problem, this study formulates adaptive hint-level selection as a sequential decision-making task and employs a Double Deep Q-Network (DDQN) to learn an instructional assistance policy. At each interaction, the policy observes a state characterized by learner mastery, question difficulty, recent performance and interaction information, and the current knowledge point, and selects one of three levels of assistance: Strong Guidance, Partial Guidance, or Minimal Scaffold. The model selects the level of assistance rather than generating the natural-language content of the hint, thereby distinguishing the present problem from automatic hint generation. The learned policy is evaluated in a simulation-based mathematical tutoring environment and compared with random, fixed, rule-based adaptive, and conventional DQN hint-selection strategies. Rather than assuming that a more adaptive or more complex policy necessarily produces superior tutoring outcomes, the evaluation examines policy performance together with interaction efficiency, assistance intensity, and state-dependent hint-selection patterns. This design enables an analysis of both the potential and the limitations of reinforcement learning for fine-grained instructional assistance decisions.

The contributions of this study are threefold. First, it operationalizes fine-grained hint assistance intensity as a sequential decision problem with three ordered levels of instructional support. Second, it provides a controlled comparison of DDQN and conventional DQN, together with non-learning and rule-based strategies, under a common simulated tutoring environment. Third, it analyzes how assistance intensity, simulated tutoring outcomes, and state-dependent policy behaviour interact, highlighting that state-sensitive adaptation does not necessarily imply superior or pedagogically intuitive tutoring behaviour.

Accordingly, this study investigates how a DDQN-based policy selects different levels of instructional assistance according to learner and task states, and what trade-offs emerge between tutoring performance and assistance intensity. Three research questions guide the investigation:

- RQ1: How does the DDQN-based adaptive hint-selection policy perform compared with random, fixed, rule-based adaptive, and conventional DQN strategies in simulated mathematics tutoring?
- RQ2: What trade-offs emerge between tutoring performance, interaction efficiency, and instructional assistance intensity under the DDQN policy?
- RQ3: How does the learned DDQN policy vary its hint-level selections across different learner mastery and question difficulty states?

The remainder of this paper is organized as follows. Section II reviews related work, Section III describes the methodology and experimental design, Section IV presents the results, and Sections V and VI discuss the findings and conclude the study, respectively.

II. RELATED WORK

Adaptive support is a central function of intelligent tutoring systems (ITSs), but the adaptation target varies considerably across systems. A long-standing issue is the assistance dilemma, which concerns whether and when support should be provided without making assistance either insufficient or unnecessarily intensive. Recent work has increasingly used learner interaction data to make these decisions more responsive. Maniktala et al. developed HelpNeed models for proactive problem-solving assistance, using recent learner behaviour and prior hint use to identify situations in which intervention may be appropriate [3]. Alam et al. combined HelpNeed prediction with subgoal hints in a logic tutor, showing how adaptive delivery can link assistance to predicted need [2]. These studies mainly address whether or when support should be triggered rather than treating assistance intensity itself as the primary sequential decision.

A second line of work focuses on what form of feedback or assistance should be delivered. Kochmar et al. investigated automated generation and personalization of hints and explanations in a large-scale tutoring system [7], whereas Schmucker et al. learned question-specific assistance policies using a multi-armed bandit formulation to select among candidate actions such as hints, vocabulary support, and other feedback types [8]. Chevalère et al. examined a tutoring strategy that varied levels of hint detail according to task performance and self-reported emotions [9]. Together, these studies show that adaptive feedback can vary in timing, content, type, and detail. However, these formulations differ from the present problem, which holds the hint modality fixed and asks the policy to select among predefined levels of instructional assistance. The decision target is therefore not hint generation or arbitrary feedback selection, but the intensity of a text-based hint at each tutoring interaction.

Reinforcement learning has also been used to learn sequential pedagogical decisions in ITSs. Zhou et al. applied hierarchical reinforcement learning to decisions at problem and step levels, including choices such as worked examples, problem solving, eliciting a step, or telling a step [4]. Sanz Ausin et al. used batch deep reinforcement learning to select learning activities [5], while Ruan et al. applied deep reinforcement learning to adaptive pedagogical support in a mathematical task [6]. Gao et al. further studied on-demand selection among previously induced pedagogical policies for different learner subgroups [10]. These studies support the use of reinforcement learning for state-dependent educational decisions, but their action spaces primarily represent broader pedagogical activities, intervention types, or whole policies rather than ordered hint-assistance levels.

Deep Q-learning has also appeared in educational recommendation and pedagogical planning. Fahid et al. used a deep Q-learning-based planner in a simulated narrative-centered learning environment [11], while Ren et al. incorporated Double DQN within a framework for adaptive exercise recommendation [12]. Offline reinforcement learning has similarly been explored for learning-path recommendation [13]. These applications are important for positioning the present study because they show that DQN and DDQN are not new to educational decision making. Their adaptive targets, however, are mainly learning content, activities, or paths. Double DQN

itself is therefore not treated as the contribution of this work. Its relevance lies in providing a value-based mechanism in which action selection and action evaluation are separated, as originally proposed to mitigate potential value overestimation [14].

The literature therefore suggests a narrower gap than a general lack of adaptive hints or reinforcement learning in ITSs. Existing research has extensively examined help-need detection, proactive hint delivery, personalized feedback, assistance-action selection, learning-activity choice, pedagogical policy induction, and educational recommendation. Comparatively less attention has been given to treating hint assistance intensity itself as an ordered, fine-grained sequential decision and examining its trade-offs with simulated tutoring performance and interaction efficiency. This distinction is particularly relevant when the instructional content is held fixed and the adaptive variable is how strongly the learner is guided.

Accordingly, the present study formulates adaptive hint-level selection as a three-action decision problem consisting of Strong Guidance, Partial Guidance, and Minimal Scaffold. A DDQN policy is evaluated alongside non-adaptive, rule-based, and conventional DQN strategies in a simulated middle-school mathematics environment. The focus is not on establishing DDQN superiority, but on examining how a learned policy behaves under different learner mastery and question difficulty states, and what trade-offs emerge between tutoring outcomes and assistance intensity.

III. METHODOLOGY

A. Adaptive Hint-Level Selection Framework

Adaptive hint-level selection is formulated as a Markov decision process,

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \gamma), \quad (1)$$

where the tutoring agent observes the current learner and task state, selects a hint-level action, receives the resulting learner response and reward, and transitions to the next state. The adaptive target is assistance intensity rather than hint generation. After a hint level is selected, the corresponding natural-language text is retrieved from a predefined hint bank.

At interaction t , the policy observes the six-dimensional state

$$s_t = [m_t, \tilde{d}_t, \rho_t, \tilde{\tau}_{t-1}, \tilde{n}_{t-1}, \tilde{k}_t], \quad (2)$$

where m_t is a global scalar simulated mastery variable. The remaining state features summarize question difficulty, recent correctness, previous response time, previous attempts, and the current knowledge point, as shown in Table I.

The discrete action space is

$$\mathcal{A} = \{0, 1, 2\}, \quad (3)$$

with 0 = Strong Guidance, 1 = Partial Guidance, and 2 = Minimal Scaffold. There is no no-hint action, so every tutoring

interaction receives one of the three predefined assistance levels.

The simulated domain covers middle-school Quadratic Equations with eight knowledge points: Concept of Quadratic Equations, Standard Form, Direct Square Root Method, Factoring, Completing the Square, Quadratic Formula, Discriminant, and Applications. Each knowledge point is associated with Low, Medium, and High difficulty conditions, producing $8 \times 3 = 24$ question conditions. The internal simulator uses difficulty values $d_t \in \{0.3, 0.6, 0.9\}$, which are distinct from the normalized state encoding 0, 0.5, 1. All episodes begin with $m_0 = 0.5$. The predefined hint effects are $h_t = 0.30, 0.20$, and 0.10 for Strong Guidance, Partial Guidance, and Minimal Scaffold, respectively.

Correctness is sampled from:

$$P(c_t = 1 \mid m_t, d_t, h_t) = \sigma(4(m_t - d_t + h_t)), \quad (4)$$

$$\sigma(x) = \frac{1}{1 + e^{-x}}.$$

Thus, higher mastery and stronger simulated support increase the modeled correctness probability, whereas greater difficulty reduces it. For a correct response,

$$\Delta m_t = 0.04 [0.5 + \max(0, d_t - m_t) - 0.4h_t] - 0.01(n_t - 1), \quad (5)$$

whereas an incorrect response gives

$$\Delta m_t = -0.01(0.5 + d_t). \quad (6)$$

Mastery is then updated as

$$m_{t+1} = \text{clip}(m_t + \Delta m_t, 0, 1). \quad (7)$$

The transition therefore incorporates task challenge, assistance intensity, and additional attempts. Response time is generated from mastery, difficulty, hint effect, and Gaussian noise, using a 45-second base time and a 5-second minimum. Attempts are generated from simulated struggle determined by the mismatch among difficulty, mastery, and hint support.

The simulator provides controlled learner-response dynamics for policy comparison. Its parameters are predefined assumptions rather than empirically calibrated educational effects, so the findings concern policy behavior in a simulated tutoring environment rather than direct real-student learning effects.

B. DDQN-Based Hint-Level Policy Learning

The adaptive hint-level policy is learned using a value-based neural-network approximation of the action-value function. The Q-network receives the six-dimensional state defined in Section III.A and outputs three Q-values corresponding to Strong Guidance, Partial Guidance, and Minimal Scaffold. The network architecture is $6 \rightarrow 64 \rightarrow 64 \rightarrow 3$, with two hidden layers of 64 ReLU units and a three-unit linear output layer.

TABLE I. STATE REPRESENTATION USED BY THE HINT-SELECTION POLICY

Feature	Symbol	Encoding / Range
Learner mastery	m_t	$[0, 1]$
Question difficulty	$\tilde{d}_t$	Low = 0, Medium = 0.5, High = 1
Recent accuracy	ρ_t	Mean correctness over previous three interactions
Previous response time	$\tilde{\tau}_{t-1}$	$\min(\tau_{t-1}/120, 1)$
Previous attempts	$\tilde{n}_{t-1}$	$\min(n_{t-1}/3, 1)$
Knowledge point	$\tilde{k}_t$	$(k_t - 1)/7$

For conventional DQN, the temporal-difference target is

$$y_t^{\text{DQN}} = r_t + \gamma(1 - z_t) \max_{a'} Q_{\text{target}}(s_{t+1}, a'), \quad (8)$$

where z_t is the terminal indicator. This target uses the maximum action value estimated by the target network, following the standard DQN formulation [15]. In DDQN, action selection and action evaluation are separated. The online network first selects

$$a^* = \arg \max_{a'} Q_{\text{online}}(s_{t+1}, a'), \quad (9)$$

and the target network evaluates that action:

$$y_t^{\text{DDQN}} = r_t + \gamma(1 - z_t) Q_{\text{target}}(s_{t+1}, a^*). \quad (10)$$

This decoupled target was proposed to mitigate potential Q-value overestimation associated with conventional DQN [14]. DDQN is adopted here for this value-estimation mechanism rather than from an assumption that it must outperform DQN. The present study also does not directly measure overestimation bias. Its contribution is the application and evaluation of DDQN for adaptive hint-level selection rather than the proposal of a new reinforcement-learning algorithm.

Both learned policies use experience replay and a periodically synchronized target network. DQN and DDQN are kept identical in state representation, action space, reward function, network architecture, optimizer, hyperparameters, replay settings, target-update frequency, and linear ϵ -greedy exploration. The intended algorithmic difference is therefore the TD target calculation. Table II summarizes the common training configuration.

TABLE II. TRAINING CONFIGURATION OF THE DQN AND DDQN POLICIES

Parameter	Value
Network	6-64-64-3
Optimizer	Adam
Learning rate	0.001
Discount factor	0.99
Replay buffer	10,000
Batch size	64
Loss	Mean squared TD error
Target-network update	Hard update every 20 optimizer updates
Exploration	Linear ϵ -greedy
Initial ϵ	0.80
Final ϵ	0.05

C. Experimental Design and Evaluation

The reward function combines six components so that policy learning reflects simulated task performance, mastery progression, interaction efficiency, and assistance intensity:

$$r_t = r_t^{\text{cor}} + r_t^{\text{imp}} + r_t^{\text{time}} + r_t^{\text{att}} + r_t^{\text{hint}} + r_t^{\text{strong}}. \quad (11)$$

The individual components are summarized in Table III. Correct responses and positive simulated mastery change contribute positively, while additional attempts and more intensive assistance introduce predefined costs. These assistance-related terms are design assumptions used to represent an assistance trade-off within the simulator and should not be interpreted as empirically calibrated evidence that stronger hints are harmful.

Five hint-selection policies are compared, as summarized in Table IV. Random and Fixed provide non-adaptive references, while the Rule-based policy uses predefined state-dependent thresholds. Specifically, Strong Guidance is selected when

$$m_t < 0.4 \vee \rho_t < 0.34 \vee \tilde{n}_{t-1} > 0.67. \quad (12)$$

If this condition is not satisfied, Partial Guidance is selected when

$$m_t < 0.7 \vee \tilde{d}_t \geq 0.5 \vee \rho_t < 0.67. \quad (13)$$

Otherwise, Minimal Scaffold is selected. Conventional DQN and DDQN provide learned state-dependent policies under the common configuration described in Section III.B.

For both DQN and DDQN, 10 independent training runs are conducted. Each run contains 500 episodes of 50 tutoring interactions, corresponding to 250,000 training interactions per algorithm and 500,000 across both learned policies. Independent seeds are used across runs, and training seeds are separated from evaluation seeds. After training, each checkpoint is frozen and evaluated greedily with $\epsilon = 0$. No learning, optimizer updates, parameter changes, or replay-buffer updates occur during held-out evaluation. Across the five policies, the held-out evaluation comprised 2,300 episodes and 115,000 interactions in total. Random, Fixed, and Rule-based policies do not require training and are evaluated on fixed held-out evaluation cohorts rather than being treated as having independent training replications.

The simulated curriculum comprised 24 question conditions, covering eight knowledge points (KP1–KP8) and three difficulty levels (Low, Medium, and High) per knowledge point. The conditions followed a fixed sequence, with difficulty progressing from Low to High within each knowledge point,

and the sequence repeated cyclically. Each episode terminated after 50 interactions, corresponding to two complete curriculum cycles followed by the first two conditions. DQN and DDQN used training seeds 42–51, while all five policies were evaluated using the same 100 initial environment seeds (10000–10099). These shared seeds provided comparable starting scenarios, although subsequent simulated trajectories could differ across policies.

Policy evaluation uses mean step reward, correctness rate, mean attempts, mean response time, simulated mastery gain, and hint-level distribution. These measures characterize tutoring performance, interaction efficiency, assistance intensity, and state-dependent policy behaviour. Hint selections are additionally analyzed across three mastery groups: Low ($m_t < 0.4$), Medium ($0.4 \leq m_t < 0.7$), and High ($m_t \geq 0.7$), as well as Low, Medium, and High question-difficulty conditions.

Replay minibatches were sampled uniformly without replacement from the current buffer. For both DQN and DDQN, terminal transitions were handled by setting the bootstrapped component of the temporal-difference target to zero. Evaluation metrics were first aggregated across the 50 interactions within each episode and then averaged over 100 evaluation episodes per policy. For DQN and DDQN, the reported means and standard deviations were calculated from the 10 independent training-run aggregates, with equal weight assigned to each run. Hint-selection proportions were calculated over 5,000 evaluation interactions per trained run. The experiments used Python 3.7.2, PyTorch 1.13.1, and NumPy 1.21.6.

The primary inferential comparison is between DQN and DDQN using the 10 matched independent runs as the replication unit. Paired t -tests are used together with mean paired differences, 95% confidence intervals, and Cohen's d_z . Statistical significance is evaluated after Holm correction for multiple comparisons. For state-conditioned hint distributions, pooled chi-square tests and Cramer's V are used descriptively to characterize associations, with run-level supporting contrasts considered where appropriate. Because interaction records are nested within runs and episodes, pooled interaction-level p -values are not interpreted as independent learner-level inference. Non-significant comparisons are likewise not interpreted as evidence of equivalence.

An exploratory sensitivity analysis was conducted to examine the dependence of the trained DDQN policies on selected simulator and reward parameters. The 10 frozen DDQN policies were evaluated using 100 matched episodes per run, with 50 steps per episode. Four single-factor perturbations were considered: reducing all hint-effect coefficients by 20%, increasing the magnitude of the hint-intensity penalty by 20%, changing the additional Strong Guidance penalty at high mastery from -0.10 to -0.12 , and increasing the corresponding mastery threshold from 0.75 to 0.80 . The hint-effect perturbation required new simulated interactions, whereas the reward-only perturbations were evaluated by rescaling the original transitions. No policies were retrained under the modified parameters.

IV. RESULTS

A. Overall Policy Performance

Table V summarizes the held-out performance of the five hint-selection policies. Across the descriptive comparison, the Rule-based policy achieved the highest mean step reward (0.840), followed by DQN (0.832), DDQN (0.813), Fixed (0.811), and Random (0.799). DQN produced the highest correctness rate (0.743) and the lowest mean number of attempts (1.411) and response time (33.37 s). The Rule-based policy achieved the largest simulated mastery gain (0.445). DDQN achieved a mean step reward of 0.813, with correctness of 0.728, mean attempts of 1.437, mean response time of 34.17 s, and simulated mastery gain of 0.424. Because Random, Fixed, and Rule-based policies were evaluated on fixed held-out cohorts rather than independent training replications, comparisons with these baselines are interpreted descriptively only.

Variability across the 10 independent learned-policy runs was also examined. DQN obtained a mean step reward of 0.832 with $SD = 0.055$ and a 95% confidence interval of $[0.793, 0.871]$. DDQN obtained a mean of 0.813 with $SD = 0.074$ and a 95% confidence interval of $[0.760, 0.866]$. Mean episode return was 41.600 for DQN and 40.638 for DDQN. These values indicate noticeable between-run variability for both learned policies, particularly for DDQN, but confidence-interval overlap alone is not used as an inferential criterion.

The matched DQN–DDQN comparisons are reported in Table VI. Using DDQN minus DQN as the paired difference, DDQN showed a mean step-reward difference of -0.019 , with a 95% confidence interval spanning zero and a Holm-adjusted $p = 1.000$. The corresponding effect size was small ($d_z = -0.202$). Similar patterns were observed for correctness, attempts, response time, and simulated mastery gain. For example, the correctness difference was -0.015 , whereas the response-time difference was $+0.81$ s. After Holm correction, no statistically significant DQN–DDQN difference was detected for any of the five primary metrics.

Overall, the held-out results do not support a claim that DDQN was superior to conventional DQN in this experiment. Descriptively, DDQN also did not achieve the highest overall reward, correctness, efficiency, or simulated mastery gain among the evaluated policies. At the same time, the absence of statistically significant DQN–DDQN differences should not be interpreted as evidence that the two methods are equivalent. These findings answer RQ1 by showing that the DDQN policy achieved performance within the range of the evaluated strategies but did not demonstrate a statistically supported performance advantage over conventional DQN.

Fig. 1 shows the training trajectories of DQN and DDQN over 500 episodes. Both algorithms exhibited fluctuations in episode rewards, while their training losses decreased after an initial increase and remained relatively stable during later episodes. The exploration rate decreased linearly from 0.80 to 0.05 , reaching its minimum at episode 500. Although the training results showed no clear evidence of sustained divergence, they do not establish that the learned policies had converged.

TABLE III. REWARD COMPONENTS USED IN POLICY LEARNING

Component	Definition
Correctness	+1.0 if correct; -0.5 otherwise
Mastery improvement	+0.3 if $m_{t+1} > m_t$; 0 otherwise
Response time	$0.2(1 - \min(\tau_t/120, 1))$
Additional attempts	$-0.1 \max(n_t - 1, 0)$
Hint intensity	Strong: -0.1000; Partial: -0.0667; Minimal: -0.0333
High-mastery Strong Guidance	-0.1 if Strong Guidance and $m_t \geq 0.75$; 0 otherwise

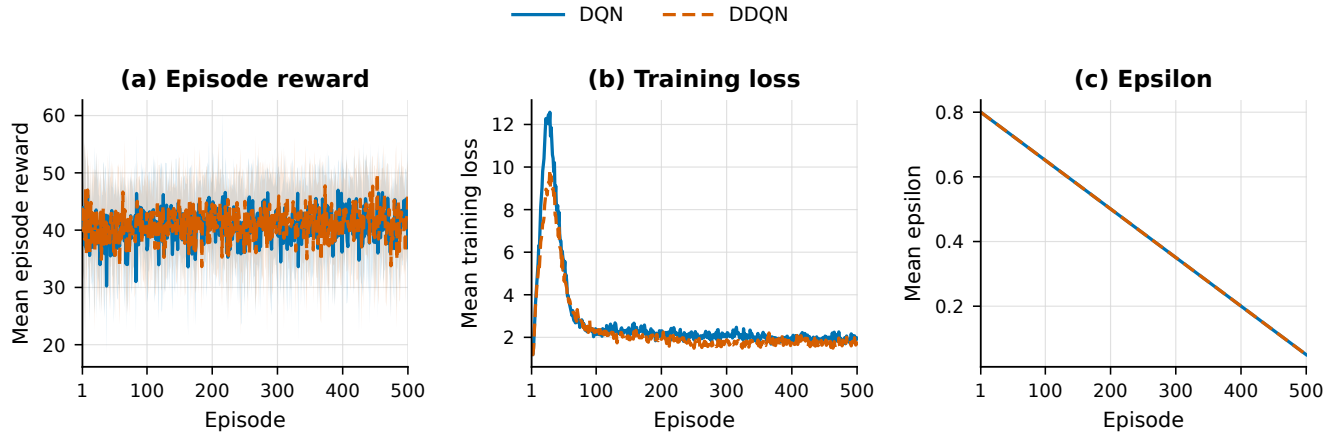


Fig. 1. Training trajectories of DQN and DDQN across 500 episodes, averaged over 10 independent runs: (a) Episode reward, (b) Training loss, and (c) Exploration rate.

TABLE IV. COMPARISON POLICIES USED IN THE EXPERIMENT

Policy	Selection mechanism
Random	Uniform random selection among three hint levels
Fixed	Always selects Partial Guidance
Rule-based	Predefined state-dependent heuristic thresholds
DQN	Learned Q-value policy using the conventional DQN target
DDQN	Learned Q-value policy with decoupled action selection and evaluation

B. Assistance Intensity and Performance Trade-offs

Fig. 2 shows the distinct assistance-intensity profiles of the five policies. Random produced an approximately even distribution across Strong Guidance (33.86%), Partial Guidance (33.20%), and Minimal Scaffold (32.94%), whereas the Fixed policy selected Partial Guidance exclusively. The Rule-based policy relied mainly on Partial Guidance (67.76%) and used Strong Guidance in 23.98% of interactions. In contrast, both learned policies selected Strong Guidance more frequently. DQN used Strong Guidance in 56.05% of interactions, compared with 45.54% for DDQN, a descriptive reduction of 10.52 percentage points. DDQN correspondingly selected Partial Guidance and Minimal Scaffold more often than DQN. These differences indicate distinct assistance-intensity behaviours, but a lower proportion of Strong Guidance is not interpreted as inherently better.

The hint-selection distributions also varied considerably across the 10 independent DDQN runs. The mean proportions of Strong Guidance, Partial Guidance, and Minimal Scaffold were 45.54%, 34.83%, and 19.64%, with between-run standard deviations of 28.73, 23.46, and 21.90 percentage points, respectively. These results indicate substantial variability in the learned assistance policies across training seeds. Consequently, the pooled distributions should not be interpreted as represent-

ing a consistent hint-selection strategy across all runs.



Fig. 2. Distribution of hint-level selections across the five comparison policies.

The reward decomposition in Table VII clarifies the trade-off between assistance intensity and simulated tutoring outcomes. Compared with DQN, DDQN incurred a smaller hint-intensity cost (-0.0753 versus -0.0799) and a smaller high-mastery Strong-Guidance cost (-0.0108 versus -0.0165). Within the predefined reward structure, these values reflect DDQN's lower reliance on the most intensive assistance level.

However, the reduction in assistance-related penalties was accompanied by weaker contributions from several outcome and efficiency components. DDQN obtained a lower correctness reward than DQN (0.5925 versus 0.6150) and a slightly lower simulated mastery-improvement reward (0.2070 versus 0.2101). Its attempt penalty was also more negative (-0.0437 versus -0.0411), while its response-time reward

TABLE V. HELD-OUT PERFORMANCE OF THE FIVE HINT-SELECTION POLICIES

Policy	Mean step reward	Correctness	Attempts	Response time (s)	Mastery gain
Random	0.799	0.720	1.457	35.14	0.421
Fixed	0.811	0.718	1.450	35.13	0.426
Rule-based	0.840	0.737	1.418	34.22	0.445
DQN	0.832	0.743	1.411	33.37	0.437
DDQN	0.813	0.728	1.437	34.17	0.424

TABLE VI. PAIRED COMPARISON BETWEEN DQN AND DDQN ACROSS 10 INDEPENDENT RUNS

Metric	DDQN - DQN	95% CI	Holm-adjusted p	Cohen's d_z
Mean step reward	-0.019	[-0.088, 0.049]	1.000	-0.202
Correctness	-0.015	[-0.055, 0.025]	1.000	-0.267
Attempts	0.026	[-0.053, 0.106]	1.000	0.237
Response time (s)	0.808	[-1.267, 2.883]	1.000	0.279
Mastery gain	-0.013	[-0.051, 0.025]	1.000	-0.239

was slightly lower (0.1430 versus 0.1444). Consequently, the smaller assistance-related penalties did not translate into a higher total reward: DDQN achieved 0.8128, compared with 0.8320 for DQN. This decomposition is descriptive and does not constitute a separate significance test.

The Rule-based policy provides additional descriptive context. Despite selecting Strong Guidance in only 23.98% of interactions and relying primarily on Partial Guidance (67.76%), it achieved the highest mean step reward among the evaluated policies. This pattern further shows that assistance intensity alone does not determine overall policy performance.

Overall, RQ2 reveals a trade-off rather than a uniformly favorable assistance pattern. DDQN selected Strong Guidance less frequently than DQN and consequently incurred smaller assistance-related penalties, but these gains were accompanied by lower correctness and mastery-improvement rewards and less favorable interaction-efficiency components. Thus, within the present simulator and predefined reward structure, reducing assistance intensity alone was insufficient to improve overall policy performance.

C. State-Dependent Hint Selection

Fig. 3 shows the DDQN hint-level distribution across learner mastery groups. The clearest descriptive contrast occurred between the Medium- and High-mastery conditions. For Medium mastery, Strong Guidance was selected in 55.26% of interactions, compared with 20.06% for Partial Guidance and 24.69% for Minimal Scaffold. For High mastery, the proportion of Strong Guidance decreased to 33.94%, while Partial Guidance increased to 52.62% and Minimal Scaffold accounted for 13.45%. This indicates that the learned policy changed its assistance-intensity distribution across mastery states.

The Low-mastery condition should be interpreted with considerable caution. Only 35 of the 50,000 DDQN evaluation interactions, corresponding to 0.07%, fell into this group. All 35 selected Minimal Scaffold, but 34 of these interactions came from one training seed and one from another, while 8 of the 10 DDQN runs never entered the Low-mastery condition. This category is therefore too sparse to support a reliable conclusion about DDQN behaviour for low-mastery learners.

A pooled contingency analysis showed a descriptive association between mastery group and selected hint level,

$\chi^2(4) = 5947.88$, with nominal $p < 10^{-300}$ and Cramer's $V = 0.244$. However, the pooled interaction rows are nested within runs and should not be interpreted as independent learner-level observations. Across the 10 runs, High mastery involved less Strong Guidance than Medium mastery in 8 runs and more Partial Guidance in 9 runs. Nevertheless, the corresponding whole-run sign-flip tests did not survive the full Holm correction, with adjusted $p = 0.191$ for the Strong-Guidance contrast and $p = 0.070$ for the Partial-Guidance contrast.

Fig. 4 shows a second state-dependent pattern across question difficulty. Strong Guidance decreased from 58.16% under Low difficulty to 42.12% under Medium difficulty and 35.76% under High difficulty. In contrast, Minimal Scaffold increased from 12.25% to 19.72% and then to 27.39%. Thus, the observed policy did not follow a simple monotonic pattern in which harder questions automatically received stronger assistance.

The pooled difficulty-conditioned association was also substantial descriptively, with $\chi^2(4) = 2143.46$, nominal $p < 10^{-300}$, and Cramer's $V = 0.146$. Across runs, High difficulty showed less Strong Guidance than Low difficulty in 9 of 10 runs and more Minimal Scaffold in 8 of 10 runs. However, the run-level contrasts again did not survive multiplicity correction, with Holm-adjusted $p = 0.0996$ for Strong Guidance and $p = 0.176$ for Minimal Scaffold.

The state-conditioned hint-selection patterns also showed substantial between-run variability. Across the 10 DDQN runs, the mean proportions of Strong Guidance were 58.79% (SD = 36.22 percentage points) for Medium mastery and 32.07% (SD = 28.38 percentage points) for High mastery. The corresponding values were 58.16% (SD = 35.07 percentage points) for Low difficulty and 35.76% (SD = 26.90 percentage points) for High difficulty. Although the directional differences were observed in most runs, the large standard deviations indicate considerable variation in hint-selection behaviour across independently trained policies.

Overall, the results for RQ3 show that the DDQN policy exhibited differentiated hint-selection patterns across mastery and difficulty states, providing descriptive evidence of state-sensitive behaviour. However, the Low-mastery category was severely underrepresented, the supporting run-level contrasts did not survive multiplicity correction, and the difficulty-conditioned pattern was not pedagogically monotonic. The

TABLE VII. REWARD-COMPONENT DECOMPOSITION ACROSS POLICIES

Policy	Correctness	Improvement	Response time	Attempts	Hint cost	Strong cost	Total
Random	0.5800	0.2027	0.1414	-0.0457	-0.0670	-0.0120	0.7994
Fixed	0.5773	0.2039	0.1415	-0.0450	-0.0667	0.0000	0.8110
Rule-based	0.6055	0.2090	0.1430	-0.0418	-0.0719	-0.0038	0.8401
DQN	0.6150	0.2101	0.1444	-0.0411	-0.0799	-0.0165	0.8320
DDQN	0.5925	0.2070	0.1430	-0.0437	-0.0753	-0.0108	0.8128

results therefore indicate state-sensitive policy behaviour, but do not establish that the learned policy produced an optimal or pedagogically appropriate adaptation strategy.

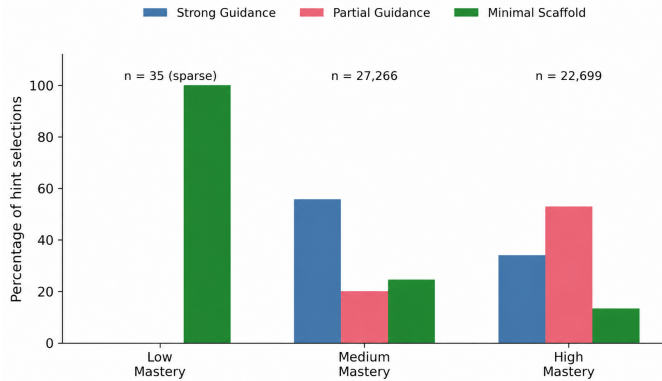


Fig. 3. DDQN hint-level distribution across learner mastery groups.

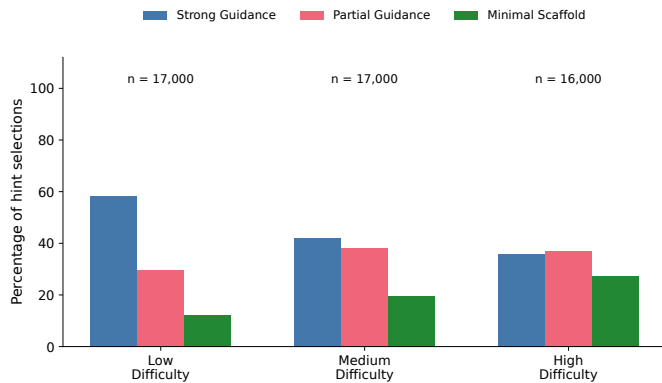


Fig. 4. DDQN hint-level distribution across question-difficulty levels.

D. Sensitivity Analysis

Table VIII presents the exploratory sensitivity results for the frozen DDQN policies. Reducing the hint-effect coefficients by 20% decreased mean episode reward by 6.19% and correctness from 72.83% to 70.42%. Increasing the hint-intensity penalty magnitude by 20% reduced the reported reward by 1.85%, while changes to the additional Strong Guidance penalty and the high-mastery threshold produced smaller reward differences of -0.26% and $+0.42\%$, respectively. The latter three comparisons were based on reward rescaling and therefore do not represent changes in policy actions or simulated learner outcomes.

TABLE VIII. EXPLORATORY SENSITIVITY ANALYSIS OF FROZEN DDQN POLICIES

Condition	Reward (Mean $\pm$ SD)	Change (%)
Baseline	40.638 $\pm$ 3.694	—
Hint effects -20%	38.124 $\pm$ 3.708	-6.19
Intensity penalty $+20\%$	39.885 $\pm$ 3.558	-1.85
Strong penalty -0.12	40.530 $\pm$ 3.649	-0.26
Mastery threshold 0.80	40.808 $\pm$ 3.759	$+0.42$

V. DISCUSSION

The present study does not support a claim that DDQN provides superior tutoring performance to conventional DQN. Instead, the results show that DDQN can learn differentiated hint-level selection behaviour, while such state-sensitive adaptation does not necessarily lead to better simulated outcomes or pedagogically intuitive assistance patterns. This distinction is important because the contribution of the study lies in examining fine-grained assistance-intensity decisions rather than demonstrating algorithmic superiority.

For RQ1, the Rule-based policy achieved the highest descriptive mean step reward, while DQN achieved the highest correctness rate. DDQN obtained a lower mean reward than DQN, and none of the paired DQN–DDQN differences across the five primary metrics remained statistically significant after Holm correction. The corresponding effect sizes were also small. These results indicate that DDQN’s decoupled action-selection and action-evaluation mechanism did not yield a statistically supported performance advantage in the present environment. Although Double DQN was originally designed to mitigate potential value overestimation [14], overestimation bias was not directly measured here. The findings therefore neither confirm nor reject that mechanism. More generally, they suggest that a theoretically motivated value-estimation modification does not guarantee better tutoring outcomes under a specific state, reward, and simulator configuration.

RQ2 reveals a clearer assistance-performance trade-off. DDQN selected Strong Guidance in 45.54% of interactions, compared with 56.05% for DQN, a difference of 10.52 percentage points. This lower use of intensive assistance reduced both the general hint-intensity penalty and the additional high-mastery Strong-Guidance penalty. However, these gains were accompanied by lower correctness and mastery-improvement contributions, a more negative attempt component, and a slightly lower response-time contribution. As a result, the lower assistance-related penalties did not produce a higher total reward. This finding should not be interpreted as evidence that stronger assistance is pedagogically preferable. Rather, it shows that, within the predefined reward structure, reducing assistance intensity alone was insufficient to improve simulated tutoring performance. The balance between task-success incentives, efficiency terms, and assistance costs therefore remains

a critical reward-design issue.

The RQ3 findings further illustrate the difference between state sensitivity and pedagogical appropriateness. DDQN selected Strong Guidance more frequently for Medium-mastery states than for High-mastery states, while Partial Guidance became more common at High mastery. This provides descriptive evidence that the learned policy responded differently to mastery information. However, the Low-mastery category contained only 35 of 50,000 interactions, and the supporting run-level contrasts did not survive multiplicity correction. The difficulty-conditioned pattern was even less intuitive: Strong Guidance decreased from Low to High difficulty, while Minimal Scaffold increased. Thus, the learned policy did not follow the intuitive monotonic pattern in which increasing question difficulty is accompanied by more intensive assistance. This counterintuitive pattern may reflect limitations in policy learning, reward specification, or the simulated learner dynamics. Therefore, the observed behaviour should not be interpreted as evidence of an optimal long-term instructional strategy.

These results also highlight the interaction between simulator assumptions and reward design. Stronger hints increase modeled immediate support, influence simulated mastery transitions, and simultaneously incur larger reward costs, with an additional penalty applied to Strong Guidance at high mastery. The learned policy therefore responds to competing incentives involving correctness, efficiency, simulated mastery change, and assistance intensity. The resulting behaviour may differ from human-intuitive scaffolding patterns. This does not imply that the reward formulation is invalid, but it demonstrates the importance of checking whether an RL objective produces educationally meaningful behaviour rather than evaluating reward alone.

Relative to prior work on help-need detection, proactive hints, feedback selection, and broader pedagogical policies [3], [2], [8], [6], the present study focuses more narrowly on assistance intensity as the adaptive decision variable. Closely related work has also varied hint detail [9]. The present contribution is therefore threefold: formulating Strong Guidance, Partial Guidance, and Minimal Scaffold as sequential RL actions; providing a controlled DQN-DDQN comparison under identical conditions; and showing that state-sensitive adaptation within simulation does not necessarily imply superior or pedagogically intuitive behaviour.

The main limitations are the simulation-based evaluation, the use of a single global mastery variable and fixed curriculum traversal, predefined hint effects and reward weights, sparse Low-mastery observations, and the absence of direct overestimation measurements or real-student validation. Future work should therefore examine empirically calibrated learner models and reward structures, richer state representations, and real-user evaluation to determine whether the observed policy behaviours remain meaningful beyond the present simulator. The exploratory sensitivity analysis further showed that evaluation outcomes depended on selected simulator and reward parameters, particularly the assumed hint effects. However, the analysis was restricted to previously trained policies and a small set of parameter perturbations. It therefore does not establish how policies would behave if retrained under alternative parameter settings, nor does it validate the educational realism of the selected parameter values. The six-dimensional state

representation was not independently validated for Markov sufficiency, and omitted simulator information may introduce partial observability. Moreover, although the three assistance levels have a natural ordering, the Q-network treats them as separate discrete actions without explicitly exploiting their ordinal structure.

VI. CONCLUSION

This study investigated adaptive hint-level selection in a simulated mathematics Intelligent Tutoring System by formulating assistance intensity as a sequential decision problem with three actions: Strong Guidance, Partial Guidance, and Minimal Scaffold. A DDQN-based policy was evaluated against Random, Fixed, Rule-based, and conventional DQN strategies.

The results showed that DDQN did not demonstrate a statistically supported performance advantage over DQN across the primary evaluation metrics. DDQN selected Strong Guidance less frequently and consequently incurred smaller assistance-related reward penalties, but this reduction did not produce better overall simulated tutoring performance, revealing an assistance-performance trade-off under the predefined reward structure. The learned DDQN policy also exhibited differentiated hint-selection patterns across learner mastery and question difficulty states. However, Low-mastery observations were extremely sparse, supporting run-level contrasts did not survive multiplicity correction, and the difficulty-conditioned pattern was not pedagogically monotonic.

Overall, the findings indicate that fine-grained assistance-intensity selection is a meaningful sequential decision problem, but state sensitivity does not automatically imply better or more pedagogically intuitive tutoring behaviour. Adaptive RL policies should therefore be evaluated not only by aggregate reward, but also by the instructional behaviours and trade-offs they produce. Future work should validate these findings using empirically calibrated learner models and reward structures, richer learner-state representations, and real student interaction data.

ACKNOWLEDGMENT

This research was funded by Universiti Putra Malaysia under Grant Geran Inisiatif Putra Siswazah (GP-IPS)/2023/9748300. The authors are grateful for this financial support, which made the completion of this study possible. AI-assisted tools were employed exclusively for linguistic refinement of the manuscript, while all research data, analyses, interpretations, and conclusions are entirely attributable to the authors.

CODE AVAILABILITY

The source code for this study is publicly available at https://github.com/DongJingwen/ITS_DDQN_TextHint.

REFERENCES

- [1] T. Son, "Intelligent tutoring systems in mathematics education: A systematic literature review using the substitution, augmentation, modification, redefinition model," *Computers*, vol. 13, no. 10, 2024. [Online]. Available: <https://www.mdpi.com/2073-431X/13/10/270>

- [2] N. Alam, M. Maniktala, B. Mostafavi, M. Chi, and T. Barnes, "Does knowing when help is needed improve subgoal hint performance in an intelligent data-driven logic tutor?" in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 13, 2023, pp. 15 895–15 902.
- [3] M. Maniktala, M. Chi, and T. Barnes, "Enhancing a student productivity model for adaptive problem-solving assistance," *User Modeling and User-Adapted Interaction*, vol. 33, no. 1, pp. 159–188, 2023.
- [4] G. Zhou, H. Azizsoltani, M. S. Ausin, T. Barnes, and M. Chi, "Leveraging granularity: Hierarchical reinforcement learning for pedagogical policy induction," *International Journal of Artificial Intelligence in Education*, vol. 32, no. 2, pp. 454–500, 2022.
- [5] M. Sanz Ausin, M. Maniktala, T. Barnes, and M. Chi, "The impact of batch deep reinforcement learning on student performance: A simple act of explanation can go a long way," *International Journal of Artificial Intelligence in Education*, vol. 33, no. 4, pp. 1031–1056, 2023.
- [6] S. Ruan, A. Nie, W. Steenbergen, J. He, J. Zhang, M. Guo, Y. Liu, K. Dang Nguyen, C. Y. Wang, R. Ying *et al.*, "Reinforcement learning tutor better supported lower performers in a math task," *Machine Learning*, vol. 113, no. 5, pp. 3023–3048, 2024.
- [7] E. Kochmar, D. D. Vu, R. Belfer, V. Gupta, I. V. Serban, and J. Pineau, "Automated data-driven generation of personalized pedagogical interventions in intelligent tutoring systems," *International Journal of Artificial Intelligence in Education*, vol. 32, no. 2, pp. 323–349, 2022.
- [8] R. Schmucker, N. Pachapurkar, S. Bala, M. Shah, and T. Mitchell, "Learning to give useful hints: Assistance action evaluation and policy improvements," in *European Conference on Technology Enhanced Learning*. Springer, 2023, pp. 383–398.
- [9] J. Chevalère, R. Lazarides, H. S. Yun, A. Henke, C. Lazarides, N. Pinkwart, and V. V. Hafner, "Do instructional strategies considering activity emotions reduce students' boredom in a computerized open-ended learning environment?" *Computers & Education*, vol. 196, p. 104741, 2023.
- [10] G. Gao, X. Yang, and M. Chi, "Get a head start: On-demand pedagogical policy selection in intelligent tutoring," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 11, 2024, pp. 12 136–12 144.
- [11] F. M. Fahid, J. Rowe, Y. Kim, S. Srivastava, and J. Lester, "Online reinforcement learning-based pedagogical planning for narrative-centered learning environments," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 21, 2024, pp. 23 191–23 199.
- [12] Y. Ren, K. Liang, Y. Shang, and X. Zhang, "Fully adaptive recommendation paradigm: top-enhanced recommender distillation for intelligent education systems," *Complex & Intelligent Systems*, vol. 9, no. 2, pp. 2159–2176, 2023.
- [13] Y. Yun, H. Dai, R. An, Y. Zhang, and X. Shang, "Doubly constrained offline reinforcement learning for learning path recommendation," *Knowledge-Based Systems*, vol. 284, p. 111242, 2024.
- [14] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.
- [15] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.