Leveraging Fuzzy Logic Towards More Explainable Reinforcement Learning-Induced Pedagogical Policies on Intelligent Tutoring Systems

John Wesley Hostetter, Mark Abdelshiheed, Tiffany Barnes and Min Chi
College of Engineering Department of Computer Science
North Carolina State University, Raleigh, NC, USA

***Abstract*—Deep Reinforcement Learning (Deep RL) has revolutionized the field of Intelligent Tutoring Systems by providing effective pedagogical policies. However, the "black box" nature of Deep RL models makes it challenging to understand these policies. This study tackles this challenge by applying fuzzy logic to distill knowledge from Deep RL-induced policies into interpretable IF-THEN Fuzzy Logic Controller (FLC) rules. Our experiments show that these FLC policies significantly outperform expert policy and student decisions, demonstrating the effectiveness of our approach. We propose a Temporal Granule Pattern (TGP) mining algorithm to increase the FLC rules' interpretability further. This work highlights the potential of fuzzy logic and TGP analysis to enhance understanding of Deep RL-induced pedagogical policies.**

I. Introduction

Intelligent Tutoring Systems (ITSs) are e-learning systems that leverage artificial intelligence and cognitive theory to enhance students' learning experience by providing personalized guidance through problem-solving and offering adaptive hints and feedback. ITSs are a sequential decision process where pedagogical strategies or policies determine the following system action. Although giving students the power to make their own decisions can lead to increased motivation, engagement, and persistence [1]–[5], not all students have the necessary meta-cognitive skills for effective decision-making [6]–[11]. Therefore, most ITSs are tutor-driven and rely on hand-coded pedagogical rules from domain experts, but the efficacy of these predefined strategies is often uncertain [12].

Recently, Deep Reinforcement Learning (Deep RL) has proven to be highly effective in inducing pedagogical policies from student-system interaction data [13], [14]. Although traditional cause-and-effect approaches provide clear insights, Deep RL is occasionally criticized for its "black box" nature, which makes it challenging to comprehend the derived policies [15], [16]. To address this limitation, knowledge distillation is used to learn a simpler model from a more complex one [17].

We propose knowledge distillation of a Deep RL policy into a Fuzzy Logic Controller (FLC). By representing the policy as a collection of IF-THEN rules, each with a clear linguistic meaning, we aim to achieve the same function approximation power as a neural network while also maintaining interpretability [18]. FLCs are a proven technology that can solve a wide range of complex tasks, from surgical planning [19] to controlling chaotic systems [20]. For example, FLCs have controlled unmanned aerial vehicles, where the agent's rules were intuitive and easily understandable by humans [21]. By using FLCs for knowledge distillation, we can not only distill complex policies into simpler, more interpretable forms but also retain their effectiveness in solving real-world problems.

We aim to overcome the FLCs' challenges when applied to complex tasks, such as the need for domain experts for initial design and the curse of dimensionality [22]. We leverage an innovative approach called the All-Permutations Fuzzy Rule Base (APFRB) [23] to directly extract fuzzy logic rules from the pedagogical policies induced by a Deep Q-Network (DQN), a classic Deep RL model successfully applied to complex tasks in robotics [24], video game playing [25], and pedagogical policy induction [10]. Here, we focus on the pedagogical decisions related to presenting the next problem as a Worked Example (WE), a Problem Solving (PS), or a Faded Worked Example (FWE) in an ITS. WE show a detailed example of how the tutor solves a problem, PS requires students to solve the same problem independently, and FWE allows students and the tutor to *co-construct* the solution.

To evaluate the effectiveness of the FLC-derived rules, we compare their performance with policies designed by domain experts (Expert) and policies allowing students to make their own decisions (StuChoice). Our results show that the FLC approach significantly outperforms both Expert and StuChoice policies, demonstrating the effectiveness of our approach in personalizing the learning experience for students.

We sought to gain further insights into the FLC-derived rules by proposing ***Temporal Granule Pattern (TGP) mining***. Given that humans tend to think in approximate fuzzy logic [19] and f-granulation can aid our understanding of human thinking [26], we developed a TGP algorithm that combines fuzzy temporal association rule mining with f-granulation theory. The extracted temporal granule patterns shed light on the pedagogical behaviors exhibited by the FLC-derived rules, Expert rules, and StuChoice. Our results showed that the three conditions displayed distinct temporal granule patterns, and students who learned more from our tutor showed different ways than those who learned less. Our TGP algorithm provides a valuable tool for understanding the pedagogical strategies used in ITSs and could be applied to other domains. To our knowledge, this is the first attempt to identify meaningful temporal patterns in FLCs using f-granulation.

II. Background & Related Work

A. Fuzzy Logic

Historically, FLCs were manually crafted by human experts, but can now be automatically derived [22], [23], [27]–[29]. Deriving FLCs is typically problem-dependent [27], or greatly hindered by the curse of dimensionality [22], [30]. The difficulty in designing FLCs (e.g., multivariable systems), has led to the study of their equivalence with Artificial Neural Network (ANN) [31], [32]. Specifically, APFRB directly translates an ANN to a FLC —bypassing the above challenges [23]. For example, APFRB has shown great success in Iris classification [23] and LED display recognition [33]. Closely related to this work, Gevaert et al. applied FLC to DQN-induced polices [34]: using the 4-feature state Cart Pole task, a DQN policy was induced *online* —while interacting with the environment; once the DQN was trained, a FLC was also induced *online* by using the DQN's knowledge [34]. However, many real-world tasks, such as e-learning and healthcare, are *offline learning* in that both our DQN and APFRB are induced and evaluated using limited pre-collected data. Building accurate simulations or simulated students is especially challenging as human learning and disease progression are complex, poorly understood processes. *In our case, it remains unclear whether APFRB can extract effective fuzzy logic rules from DQN-induced policies with high-dimensional state representations in an offline learning manner.*

While prior work has focused heavily on deriving effective fuzzy logic rules, it has largely neglected to examine how to interpret these rules despite the claim that humans think with fuzzy logic and f-granulation plays an important role in understanding human thinking [19], [26]. A method combining fuzzy temporal association rule mining with f-granulation has been proposed before [35]. However, their approach has limitations when analyzing dynamic and adaptive fuzzy rule bases since it relies on a fuzzy information granule (FIG) called Linear Gaussian FIG, which has a core that depends on time but cannot handle items' lifespans. In contrast, TGP Mining is designed to analyze temporal granule patterns of fuzzy rule bases that change over time while remaining intuitive and computationally affordable. Prior work on **applying fuzzy logic to ITSs** relied upon expert-designed FLCs and mainly involved student modeling [36]–[39].

B. Applying Reinforcement Learning to ITSs

ITSs rely on student-agent interactions that can be modeled as sequential decision-making problems under uncertainty. These problems can be tackled using Reinforcement Learning (RL), a learning paradigm aiming to maximize long-term rewards without knowing the "correct" decisions at each immediate time step [40]. Recent research has explored the application of RL and Deep RL to develop effective pedagogical policies for ITSs [14], [41], [42]. For instance, Wang et al. employed various Deep RL techniques to develop pedagogical policies that enhance students' normalized learning gain in an educational game [42]. Similarly, Sanz Ausin et al. utilized offline Deep RL to design pedagogical policies that boosted student learning, which was more effective than expert-designed baseline policies [43]. However, to our knowledge, no prior work has attempted to interpret the pedagogical policies induced by Deep RL.

C. Student Making Pedagogical Decisions

While students can benefit from making their own decisions during learning [1], [8]–[10], [14], [44], they are not always effective at making pedagogical decisions. Studies have shown that students with low prior knowledge exhibit more ineffective help-seeking behaviors than those with high prior knowledge [45]. They often do not use hints effectively as they wait too long before asking for hints [6]. In addition, even college students often make poor problem selections [46]. In contrast, our results show that the FLC significantly outperforms StuChoice. By applying TGP mining, we gain a deeper understanding of students' rules when making pedagogical decisions.

III. Methodologies

We will first review f-granulation and Cartesian granules. Then, we will describe our FLC rule induction procedure which consisted of three stages: 1) applying Deep RL by using a Deep Q-Network [47] to induce the original effective pedagogical policy; 2) transforming the Deep Q-Network into a shallow neural network with one layer using Knowledge Distillation [17], and 3) using the APFRB algorithm to derive FLC rules. Finally, we will describe the TGP algorithm.

A. F-Granulation & Cartesian Granules

Granulation means breaking down a whole into its parts, leading to *granules* [26]. Zadeh's example of the human head illustrates granulation, where the nose, cheeks, and eyes are the granules. *Information granulation* applies this concept to information using crisp set theory, but imprecise boundaries between granules limit this approach. Zadeh proposed an extension called ***the theory of fuzzy information granulation (f-granulation)*** [26], which outlines a general framework that can accommodate these imprecise boundaries.

Within f-granulation, Cartesian granules can be generalized to incorporate a variety of constraints contingent on equality, probability, fuzzy graphs, etc. In this work, they are based only on possibilistic constraints (i.e., fuzzy sets). A Cartesian granule, denoted as $\hat{G}$, has a *core*. *The core of* $\hat{G}$ is all elements that *definitively belong* to $\hat{G}$. Upon receiving a new input, $\mathbf{x} \in \mathbb{R}^m$, we calculate the distance between $\mathbf{x}$ and the $\hat{G}$'s core by its membership function, $\hat{G}(\mathbf{x})$. Thus, $\hat{G}(\mathbf{x})$ describes the *visitation* of $\hat{G}$.

B. Reinforcement Learning with Deep Q-Network

In the conventional RL, an agent interacts with an environment $\mathcal{E}$ over a series of decision-making steps, which can be framed as a Markov Decision Process. The set notation $\mathcal{S}$ and $\mathcal{A}$ denote the state and action space of a Markov decision process, respectively. At each timestep t, the agent observes

$\mathcal{E}$ in-state s_t; it chooses an action a_t from a discrete set of possible actions; and $\mathcal{E}$ provides a scalar reward r_t and evolves into next state s_{t+1}. Future rewards are discounted by a factor $\gamma \in (0, 1]$, and the return at time-step t is defined as $R_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'}$, where T is the last time-step in the episode. The agent's goal is to maximize the expected discounted sum of future rewards, also known as the return, which is equivalent to finding the optimal action-value function $Q^*(s, a)$ for all states. Formally, $Q^*(s, a)$ is the highest possible expected return starting from state s, taking action a, and following the optimal policy π^* after that. It can be calculated as $Q^*(s, a) = \max_\pi \mathbb{E}[R_t | s_t = s, a_t = a, \pi]$ and $Q^*(s, a)$ must follow the *Bellman Equation*. For any state-action pair, the optimal action-value function must follow the Bellman optimality equation in that:

$$Q^*(s, a) = r + \gamma \max_{a'} Q^*(s', a') \tag{1}$$

Here r is the expected immediate reward for taking action a at state s, γ is the discount factor, and $Q^*(s', a')$ is the optimal action-value function for taking action a' at the subsequent state s' when following policy π^* thereafter. ***Deep Q-Network (DQN)*** [47] is fundamentally a version of Q-learning that uses a deep neural network to approximate the true Q-values. The DQN calculates the current state and action $Q(s, a)$ and the Q-value for the next state and action $Q(s', a')$. The weights of the deep neural network are denoted by $\boldsymbol{\theta}$ and Eq. 2 shows its corresponding *Bellman Equation*. It is trained through gradient descent to minimize the squared difference.

$$Q_{\boldsymbol{\theta}}(s, a) = \mathop{\mathbb{E}}_{s' \sim \mathcal{E}}[r + \gamma \max_{a'} Q_{\boldsymbol{\theta}}(s', a')] \tag{2}$$

C. Knowledge Distillation of a Deep Q-Network

Let $Q_{\boldsymbol{\theta}} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ represent a Deep Q-Network that has been trained offline using the Deep Q-Learning algorithm with experience replay, where $\boldsymbol{\theta}$ are the parameters. The greedy policy can then be expressed as $\pi_{\boldsymbol{\theta}}(s) = \arg\max_a Q_{\boldsymbol{\theta}}(s, a)$ for some $s \in \mathcal{S}$ and $a \in \mathcal{A}$. The original Deep Q-Network (DQN) is not well-suited for use with All-Permutations Fuzzy Rule Base (APFRB) due to several reasons. Firstly, the transformation via APFRB does not scale effectively with artificial neural networks that have multiple hidden layers, as this can impact the interpretability of the resulting fuzzy logic rules. Secondly, the activation functions used in DQN, ReLU for the hidden neurons and linear for the output neurons, are not compatible with APFRB which requires the use of only hyperbolic tangent (*tanh*) activation function. Finally, the transformation through APFRB results in an exponential growth of the knowledge base, with the number of fuzzy logic rules produced guaranteed to be 2^m where m is the number of neurons **A.3**, making it intractable.

To meet the prerequisites of APFRB, we convert the Deep Q-Network to a shallower artificial neural network through knowledge distillation. Specifically, we used a single-layer artificial neural network with n inputs s_i, $i = 1, \ldots, n$, a hidden layer of m neurons with activation function $h : \mathbb{R} \to \mathbb{R}$, and a single output unit. The output is then $f = \sum_{j=1}^{m} c_j h(x_j + b_j)$ where $x_j = \sum_{i=1}^{n} w_{ji} s_i$ with $1 \leq j \leq m$. The weight, w_{ji}, is between the i^{th} input node and the the j^{th} hidden neuron. Furthermore, b_j is the bias for the j^{th} hidden neuron and c_j is the weight between the j^{th} neuron and the output node. By offline knowledge distillation, let $f \approx \pi_{\boldsymbol{\theta}}$.[1]

D. All-Permutations Fuzzy Rule Base

F-granulation underlies fuzzy logic control [26], where a Cartesian granule $\hat{G}$ determines a FLC's rule activation by measuring the truth of its antecedents. Rules with partially true antecedents are still applied. A FLC maps $\hat{G}$s to actions, and a $\hat{G}$ is also the IF-part of a fuzzy logic rule. Definition 1 outlines the conditions for a FLC to be an APFRB, which leverages the m hidden units of an artificial neural network as input $\mathbf{x}$. $\hat{\mathbb{G}}$ is the set of all possible $\hat{G}$s, which APFRB discovers automatically due to condition **A.3**. APFRB produces fuzzy logic rules conditioned on the presence or absence of each x_j under condition **A.2**. Here is a sample fuzzy logic rule for $m = 6$ that has been extracted for our FLC's pedagogical policy:

IF *(x_1 is absent)* $\wedge$ *(x_2 is absent)* $\wedge$ *(x_3 is present)* (3)
$\wedge$ *(x_4 is absent)* $\wedge$ *(x_5 is absent)* $\wedge$ *(x_6 is present)*
THEN *Give A Faded Worked Example (FWE)*

where the entire IF-part is a Cartesian granule and the THEN-part is the recommended consequent for that Cartesian granule; as the IF-part becomes more true (i.e., approaches 1), or the greater the input "visits" the Cartesian granule (i.e., the input gets closer to the *core*), then the more the FLC will "vote" to recommend an FWE (by fuzzy inference). Upon receiving input $\mathbf{x}$ at any given time-step, the FLC consults these mappings and produces action by measuring $\mathbf{x}$'s distance to each $\hat{G}$. This calculation is a fuzzy set where $\hat{\mathbb{G}}(\mathbf{x}) = \left\{(\hat{G}, \hat{G}(\mathbf{x})) \mid \hat{G} \in \hat{\mathbb{G}}\right\}$. Given a trained feed-forward artificial neural network with the *tanh* as its activation function (end of Section III-C), we derive a FLC using Definition 1 [23].

Definition 1: A fuzzy rule base with input $\mathbf{x} \in \mathbb{R}^m$ and output $\tilde{f}$ is called an APFRB if the following conditions hold:

A.1 Every input variable x_j is characterized by two linguistic terms: term_-^j and term_+^j. The linguistic terms $\mu_-^j(x_j)$ and $\mu_+^j(x_j)$ satisfy the following constraint: there exists a $b_j \in \mathbb{R}$ such that

$$\frac{\mu_+^j(x_j) - \mu_-^j(x_j)}{\mu_+^j(x_j) + \mu_-^j(x_j)} = tanh(x_j + b_j), \quad \forall x_j \in \mathbb{R} \tag{4}$$

A.2 The form of every rule is

$$\begin{aligned} &\textbf{\textit{IF}}\ x_1 \texttt{ is } \texttt{term}_\pm^1 \wedge \ \ldots \ \wedge x_m \texttt{ is } \texttt{term}_\pm^m \\ &\textbf{\textit{THEN}}\ \tilde{f} = c_1 \pm c_2 \ldots \pm c_m \end{aligned} \tag{5}$$

[1]Replication of $\pi_{\boldsymbol{\theta}}$ with f was $\sim 86.7\%$.

where $c_j \in \mathbb{R}$. The signs in the THEN-part (consequents) are determined in the following manner: if the term characterizing x_j in the IF-part (antecedents) is term^j_+, then in the consequents, c_j appears with a plus sign; otherwise, c_j appears with a minus sign.

A.3 Contains exactly 2^m rules spanning, in their IF-part, all the possible assignment combinations of $x_1, ..., x_m$.

Upon doing so, the FLC that has been extracted is equivalent to the artificial neural network that produced it.

E. Temporal Granule Pattern Mining

We propose a method for identifying the most critical or important fuzzy logic rules, called TGP mining. To obtain a TGP candidate, we restrict the fuzzy set $\hat{\mathbb{G}}$ only to include $\hat{G}$s where $\hat{G}(\mathbf{x}) \geq \epsilon$, with ϵ in the range $(0, 1]$. This is based on the ϵ-completeness property [48], which requires a FLC to have at least one fuzzy set with membership degree at least ϵ for every possible input. By varying the value of ϵ, we can control the amount of information captured in the TGP candidate, with smaller values giving more detail but being more computationally expensive. On the other hand, ϵ values closer to 1 are more myopic, and TGP mining will only produce patterns that occur extremely close to the cores of the $\hat{G}$s. [2] We also aim to identify cases where the FLC is uncertain about the appropriate action and how often this occurs.

Since our approach is general, *we are able to apply this TGP mining on student-tutor interactions that were assigned to the StuChoice or Expert conditions.* This is because $\hat{G}$s are merely an alternative form of a space-partitioning technique (see Figure 1). Thus, we can examine the StuChoice and Expert students' interactions with the ITS, and how those interactions differ from those assigned to the FLC.

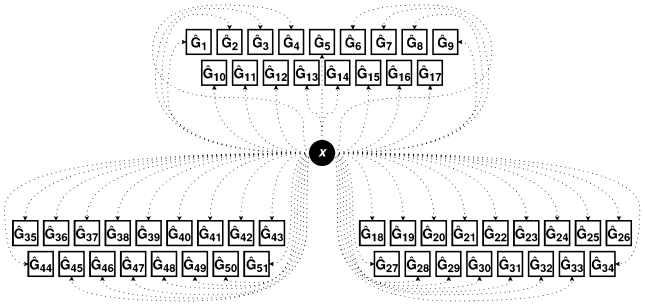


Fig. 1. x interacts with $\hat{\mathbb{G}}$ through Cartesian granules, where membership of x to each granule is calculated to determine its proximity to "fuzzy regions". $\mathbb{G}(x)$ represents the strength of x's visit to every region.

More formally, the process of obtaining the TGP candidate, $\hat{\mathbb{G}}_{\geq\epsilon}(\mathbf{x})$, is well-grounded in fuzzy set theory and can be expressed precisely. First, the α-cut operator from fuzzy set theory is applied to obtain a non-fuzzy set, ${}^{\alpha}\hat{\mathbb{G}}(\mathbf{x})$, in order to reduce the Cartesian granules in the transaction. However, ${}^{\alpha}\hat{\mathbb{G}}(\mathbf{x})$, does not describe how strongly Cartesian granules were visited (i.e., $\hat{G}(\mathbf{x})$ where $\hat{G} \in \hat{\mathbb{G}}$). To remedy this, the *level set* of $\hat{\mathbb{G}}(\mathbf{x})$, $\Lambda(\hat{\mathbb{G}}(\mathbf{x}))$, is consulted for the distinct α values and α-cuts but is further constrained as

$$\Lambda(\hat{\mathbb{G}}(\mathbf{x}))_{\geq\epsilon} = \left\{\hat{G}(\mathbf{x}) \mid \hat{G}(\mathbf{x}) \geq \epsilon \ \texttt{for} \ \hat{G} \in \hat{\mathbb{G}} \wedge \mathbf{x} \in \mathbb{R}^m\right\}. \tag{6}$$

Finally, we obtain the ***TGP candidate***, $\hat{\mathbb{G}}_{\geq\epsilon}(\mathbf{x})$, by modifying the *Third Decomposition Theorem of a fuzzy set* [49] so

$$\hat{\mathbb{G}}_{\geq\epsilon}(\mathbf{x}) = \bigcup_{\alpha\in\Lambda(\hat{\mathbb{G}}(\mathbf{x}))_{\geq\epsilon}} {}_{\alpha}\hat{\mathbb{G}}(\mathbf{x}) \tag{7}$$

where $\bigcup$ is fuzzy union, and the *special fuzzy set*, ${}_{\alpha}\hat{\mathbb{G}}(\mathbf{x})$, is

$${}_{\alpha}\hat{\mathbb{G}}(\mathbf{x}) = \alpha \ \cdot^{\alpha} \hat{\mathbb{G}}(\mathbf{x}). \tag{8}$$

A special fuzzy set assigns a degree of membership of α to each element. The resulting set of $\hat{\mathbb{G}}_{\geq\epsilon}(\mathbf{x})$ forms our *temporal quantitative transaction database* [50], which includes transactions involving items with both quantity and time components. These items correspond to Cartesian granules, or fuzzy regions in space, where the quantity component represents the degree of visitation and the time component corresponds to the moment that degree of visitation was recorded (e.g., first pedagogical decision, second decision, etc.). However, not all possible temporal quantitative items are included in the database if specific granules are never visited.

To mine TGPs, we need an algorithm that handles temporal quantitative transactions with elements in the form of (ℓ, v), where ℓ is an item and v is a positive real number representing its quantity, along with a time-step t. We chose FTARM [50] for its ability to handle such data, but it requires fuzzy sets to describe item quantities. We used CLIP [28] to define Gaussian fuzzy sets for $\hat{G}$ visitations and assigned linguistic terms to these sets. FTARM generates candidate rules for TGPs similarly to the Apriori algorithm. The resulting meta-knowledge from applying FTARM to TGPs describes when and how $\hat{G}$s were significantly visited for each condition. Fig. 2 shows the linguistic terms we assigned to the fuzzy sets.

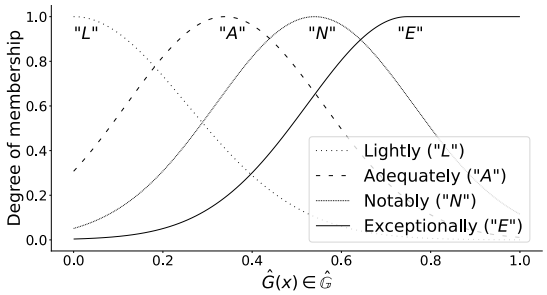


Fig. 2. The linguistic terms used to interpret $\hat{G}$ visitation.

IV. DQN-Induced Policy & Knowledge Distillation

Our DQN-policy induction was conducted offline using pre-collected training data containing 1,307 students' interaction logs over seven semesters of classroom studies (2016 Fall to 2020 Spring). During these studies, all students used the same tutor, followed the same general procedure, studied the

[2] Our experiments found that $\epsilon = 0.1$ provided a good balance between information and computational efficiency.

same training materials, and worked through the same training problems. The training corpus provides the state representation, action, and reward information for policy induction. **State:** We extracted 142 features that might impact student learning from the student-system interaction logs; these state features can be categorized into the following five groups: *Autonomy:* the amount of work done by the student; *Temporal Situation:* the time-related information about the work process; *Problem-Solving:* information about the current problem-solving context; *Performance:* information about the student's performance during problem-solving; *Student Behaviors:* the statistical measurement of student's behavior. **Actions:** are the pedagogical actions, WE/PS/FWE described above. **Reward:** There's no immediate reward during tutoring, and the delayed reward is the students' Normalized Learning Gain (NLG), which measures their learning gain irrespective of their incoming competence. NLG is defined as $\frac{posttest-pretest}{1-pretest}$, where 1 is the max score for both pre- and post-test. Knowledge Distillation [17] was then leveraged to convert the complex DQN-induced policy into a much smaller one-layer neural network with m hidden units. We explored the effectiveness of our one-layer neural network approximating the performance of the original DQN by varying m from 2 to 10 and chose the best performance when $m = 6$. Thus, we have six hidden variables $\mathbf{x} = \{x_1, x_2, ..., x_6\}$, and their biases are passed into the APFRB to induce 2^6 fuzzy logic rules due to condition **A.3**; 51 FLC rules remained after sensitivity analysis [23].

V. Experimental Setup

Participants: In the spring of 2021, a Computer Science class at North Carolina State University participated in our study. *IRB is obtained from the institution. The use/test of the ITS is overseen by a departmental committee, ensuring it does not risk the academic performance and privacy of the participants.* The study was designed to be completed within one week, and students were told they would be graded based on effort rather than learning performance. A total of 196 students were randomly assigned to one of three conditions: FLC ($N = 58$), StuChoice ($N = 59$), and Expert ($N = 79$). The difference in size between the conditions was due to additional students being assigned to the Expert condition as a baseline for monitoring population shifts between semesters. Of the 196 students, 178 completed the study, but 8 students were excluded from subsequent statistical analysis because they achieved perfect scores on the pretest. The final sample sizes were FLC ($N = 50$), StuChoice ($N = 54$), and Expert ($N = 66$). A Chi-square test found no significant difference in completion rates between the three conditions: $\chi^2(2) = 1.7444$, $p = 0.418$.

Our ITS Tutor is a web-based system teaching 10 probability principles (e.g., the Addition Theorem and Bayes' Theorem), with a dialogue window for student messages and input. Definitions of variables and equations are displayed on the left, and the ITS provides adaptive instructions, immediate feedback, and on-demand hints. Pedagogical decisions determine whether the next problem is a PS, WE, or FWE.

The experiment procedure:

1) **Textbook:** Students study probability principles.
2) **Pretest:** Prior knowledge is evaluated with an exam consisting of 14 single- and multiple-principle problems.
3) **ITS training:** Students receive automated tutor assistance on 12 assigned problems (same order for each student).
4) **Posttest:** Students are evaluated using a 20-problem test, 14 of which are isomorphic to the pretest, and 6 are non-isomorphic multiple-principle problems.[3]

Tests were double-blind graded by 2 experts and normalized.

VI. Empirical Results

The pretest scores did not significantly differ between the three conditions ($F(2, 167) = 0.247$, $p = 0.781$), indicating a balanced level of incoming competence. Additionally, the hours spent training on the ITS did not significantly differ among the three conditions as determined by a one-way Welch ANOVA ($Welch's F(2, 106.525) = 0.479$, $p = 0.621$).

Learning by ITS Training: We evaluated the effectiveness of our ITS by comparing the pretest and isomorphic posttest scores for each condition using a repeated one-way ANOVA. FLC and Expert had significant learning, with $F(1, 49) = 33.074$, $p \leq 0.0005$, partial $\eta^2 = 0.403$ and $F(1, 65) = 8.156$, $p = 0.006$, partial $\eta^2 = 0.111$, respectively, but StuChoice did not show significant learning: $F(1, 53) = 0.079$, $p = 0.780$, partial $\eta^2 = 0.001$. This motivated us to apply TGP mining to compare their learning process.

Posttest & NLG: Results showed a significant difference in posttest scores among the three conditions, even after controlling for pretest scores using ANCOVA ($F(2, 166) = 13.056$, $p \leq 0.0005$, partial $\eta^2 = 0.136$). Pairwise comparisons with Bonferroni adjustment revealed that the FLC outperformed both the Expert ($p = 0.021$) and StuChoice ($p \leq 0.0005$). For NLG, an ANOVA with Tukey-Kramer posthoc test found a statistically significant difference in NLG among the three conditions ($F(2, 167) = 4.567$, $p = 0.012$, partial $\eta^2 = 0.052$). Post hoc multiple comparisons show the FLC performed the best, followed by the Expert (no significant difference from FLC; $p = 0.178$) and StuChoice ($p = 0.008$). These findings suggest that the FLC is more effective in improving post-test scores and NLG compared to the other policies.

Visual Investigation: Fig. 3 displays a t-SNE visualization of the original feature space and Cartesian granules at the event level, with different colors representing the three conditions. We focus on the learning processes of two students, a high learner (blue, $NLG = 0.58$) and a low learner (red, $NLG = -0.12$). While their learning progression appears similar in the original space, their temporal progression differs significantly in the Cartesian granules representation, as expected.

Specifically, Fig. 3 (Cartesian Granules) shows that the two students start in the same region but quickly diverge, which may explain why one is a high learner, and the other is a low learner. In addition, some conditions visit certain $\hat{G}$s more than others. For example, the FLC students visit $\hat{G}_6$ more

[3]The posttest is designed to be significantly harder than the pretest.

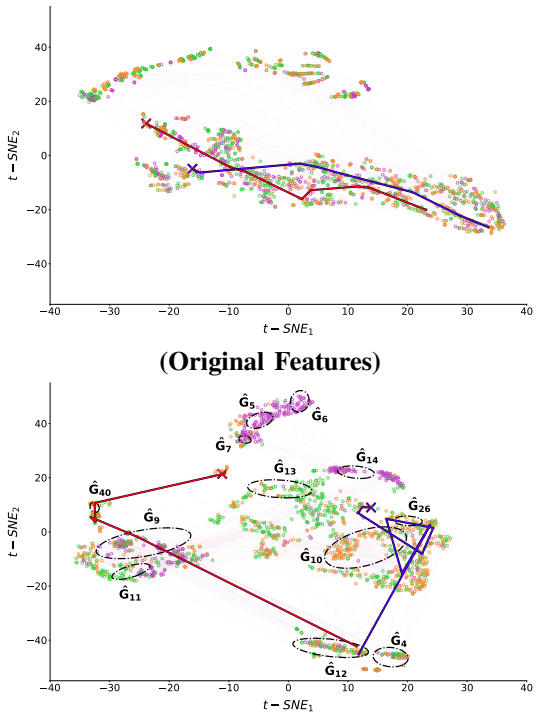
TABLE I
$NLG >$ MEDIAN

TGP (IF-THEN)	When	Con.	Sup.
FLC ($N = 31$)			
$\hat{G}_5$ is $A \rightarrow \hat{G}_6$ is A	≥ 7 steps	0.67	0.48
Expert ($N = 40$)			
$\hat{G}_{14}$ is $L \rightarrow \hat{G}_{10}$ is A	≥ 2 steps	0.77	0.24
StuChoice ($N = 14$)			
$\hat{G}_{10}$ is $N \rightarrow \hat{G}_{26}$ is A	≥ 2 steps	0.71	0.30

TABLE II
$NLG \leq$ MEDIAN

TGP (IF-THEN)	When	Con.	Sup.
FLC ($N = 19$)			
$\hat{G}_7$ is $A \rightarrow \hat{G}_5$ is L	≥ 7 steps	0.92	0.34
Expert ($N = 26$)			
$\hat{G}_{14}$ is $L \rightarrow \hat{G}_{10}$ is A	≥ 5 steps	0.61	0.20
StuChoice ($N = 40$)			
$\hat{G}_{10}$ is $E \rightarrow \hat{G}_{26}$ is A	≥ 2 steps	0.60	0.21

frequently and strongly than their peers. Given the FLC's superior performance, this suggests that adequate visitation of $\hat{G}_6$ may be a sign of a positive learning experience, which is later confirmed by the TGP results in Table I.



(Original Features)

(Cartesian Granules)

Fig. 3. Visualization of student-tutor interaction using t-SNE. The blue line is a student with $NLG = 0.58$, while the red line is the student with $NLG = -0.12$. Interactions are color-coded: FLC (purple), Expert (green), and StuChoice (orange). The end of each trajectory is marked with an "X".

TGP Results: Using TGP mining, we analyzed various temporal granule patterns in different conditions and for students with distinct NLGs. This analysis is significant as both the DQN and the induced fuzzy logic rules aim to enhance NLGs. To differentiate students with varying NLGs, we split them into High- and Low-NLG groups based on a medium split. Tables I and II display examples of TGPs discovered by FTARM for the three high-NLG groups versus those with lower NLGs that exhibit high support and confidence. Each table contains the IF-THEN patterns (TGPs), their timing (indicated as "When"), corresponding fuzzy temporal confidence (labeled as "Con."), and fuzzy temporal support (abbreviated as "Sup."). Note we only present unique patterns for High- and Low-NLG groups.

For the two FLC groups, Table I presents the results of the TGP mining for High-NLG students ($N = 31$), which shows that adequate visitation of $\hat{G}_5$ leads to the adequate visitation of $\hat{G}_6$ at time-step 7 or later. This pattern is indicative of a positive learning experience, as the adequate visitation of $\hat{G}_6$ is associated with improved learning outcomes. In contrast, the Low-NLG FLC group ($N = 19$) showed a different pattern, with the adequate visitation of $\hat{G}_7$ leading to the light visitation of $\hat{G}_5$ at time-step 7 or later.

For the two Expert groups, both High-NLG ($N = 40$) and Low-NLG ($N = 26$) groups exhibited the same IF-THEN clause but differed in the timing of the pattern. The High-NLG group showed the pattern at time-step 2 or later, while the Low-NLG group showed it at time-step 5 or later.

For the two StuChoice groups, both High-NLG ($N = 14$) and Low-NLG ($N = 40$) groups had TGPs that involved $\hat{G}_{10}$ and $\hat{G}_{26}$ with the same timing of t being time-step 2 or later. The difference was in the level of visitation of $\hat{G}10$, with High-NLG students showing notable (N) visitation and Low-NLG students showing extreme (E) visitation. The results of the TGP mining suggest that notable visitation of $\hat{G}_{10}$ is associated with improved learning outcomes for StuChoice students. These findings are summarized in Tables I and II.

VII. Discussion, Future Work, and Limitations

Using fuzzy logic theory, we successfully distilled knowledge from Deep RL-induced pedagogical policies in ITSs, demonstrating the superiority of the FLC policy over the expert policy and students' decisions. Our TGP mining algorithm provided interpretable insights into the FLC rules, showcasing the potential of fuzzy logic and TGP analysis in enhancing the understanding of Deep RL-induced pedagogical policies. Future work can expand the experiment to more ITSs, explore other types of Deep RL-induced policies using fuzzy logic and TGP analysis, and investigate other interpretability techniques in combination with our approach for greater transparency.

VIII. Acknowledgment

This research was supported by the NSF Grants: Integrated Data-driven Technologies for Individualized Instruction in STEM Learning Environments (1726550), CAREER: Improving Adaptive Decision Making in Interactive Learning Environments (1651909), and Generalizing Data-Driven Technologies to Improve Individualized STEM Instruction by Intelligent Tutors (2013502).

REFERENCES

[1] D. I. Cordova and M. R. Lepper, "Intrinsic motivation and the process of learning: Beneficial effects of contextualization, personalization, and choice." *Journal of educational psychology*, vol. 88, no. 4, p. 715, 1996.

[2] M. B. Kinzie and H. J. Sullivan, "Continuing motivation, learner control, and cai," *Educational Technology Research and Development*, vol. 37, no. 2, pp. 5–14, 1989.

[3] T. Flowerday, G. Schraw, and J. Stevens, "The role of choice and interest in reader engagement," *The Journal of Experimental Education*, vol. 72, no. 2, pp. 93–114, 2004.

[4] M. Abdelshiheed, M. Maniktala, T. Barnes, and M. Chi, "Assessing competency using metacognition and motivation: The role of time-awareness in preparation for future learning," in *Design Recommendations for Intelligent Tutoring Systems*, 2022, vol. 9, pp. 121–131.

[5] M. Abdelshiheed, G. Zhou, M. Maniktala, T. Barnes, and M. Chi, "Metacognition and motivation: The role of time-awareness in preparation for future learning," in *CogSci*, 2020, pp. 945–951.

[6] V. Aleven and K. R. Koedinger, "Limitations of student control: Do students know when they need help?" in *ITSs*, 2000, pp. 292–303.

[7] I. Roll *et al.*, "Tutoring self-and co-regulation with intelligent tutoring systems to help students acquire better learning skills," *Design recommendations for intelligent tutoring systems*, vol. 2, pp. 169–182, 2014.

[8] M. Abdelshiheed, J. W. Hostetter, X. Yang, T. Barnes, and M. Chi, "Mixing backward-with forward-chaining for metacognitive skill acquisition and transfer," in *AIED*. Springer, 2022, pp. 546–552.

[9] M. Abdelshiheed, J. W. Hostetter, P. Shabrina, T. Barnes, and M. Chi, "The power of nudging: Exploring three interventions for metacognitive skills instruction across intelligent tutoring systems," in *CogSci*, vol. 44, 2022.

[10] M. Abdelshiheed, J. W. Hostetter, T. Barnes, and M. Chi, "Bridging declarative, procedural, and conditional metacognitive knowledge gap using deep reinforcement learning," in *Proceedings of the 45th annual conference of the cognitive science society*, 2023.

[11] M. Abdelshiheed, M. Maniktala, S. Ju, A. Jain, T. Barnes, and M. Chi, "Preparing unprepared students for future learning," in *CogSci*, 2021, pp. 2547–2553.

[12] K. Vanlehn, "The behavior of tutoring systems," *IJAIED*, vol. 16, no. 3, pp. 227–265, 2006.

[13] J. Rowe, B. Mott, and J. Lester, "Optimizing player experience in interactive narrative planning: a modular reinforcement learning approach," in *Tenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2014.

[14] M. Abdelshiheed, J. W. Hostetter, T. Barnes, and M. Chi, "Leveraging deep reinforcement learning for metacognitive interventions across intelligent tutoring systems," in *Proceedings of the 24th International Conference on Artificial Intelligence in Education*. Springer International Publishing, 2023.

[15] J. Casillas *et al.*, *Interpretability Issues in Fuzzy Modeling*, ser. Studies in Fuzziness and Soft Computing. Springer Berlin Heidelberg, 2013.

[16] B. Muslimi, M. A. M. Capretz, and J. Samarabandu, "An Efficient Technique for Extracting Fuzzy Rules from Neural Networks," *International Journal of Electrical and Computer Engineering*, vol. 2, no. 4, pp. 1231 – 1237, 2008.

[17] J. Ba and R. Caruana, "Do deep nets really need to be deep?" in *Advances in Neural Information Processing Systems*, vol. 27. Curran Associates, Inc., 2014.

[18] J. Castro, "Fuzzy logic controllers are universal approximators," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 4, pp. 629–635, 1995.

[19] M.-L. Nault *et al.*, "Fuzzy–logic–assisted Surgical Planning in Adolescent Idiopathic Scoliosis," *Clinical Spine Surgery*, vol. 22, no. 4, 2009.

[20] A. M. Harb and I. Al-Smadi, "Chaos Control Using Fuzzy Controllers (Mamdani Model)," in *Integration of Fuzzy Logic and Chaos Theory*, ser. Studies in Fuzziness and Soft Computing. Berlin, Heidelberg: Springer, 2006, vol. 187.

[21] C. Sabo and K. Cohen, "Fuzzy Logic Unmanned Air Vehicle Motion Planning," *Advances in Fuzzy Systems*, vol. 2012, p. 989051, 2012.

[22] K. Ang and C. Quek, "RSPOP: Rough Set-Based Pseudo Outer-Product Fuzzy Rule Identification Algorithm," *Neural computation*, vol. 17, pp. 205–43, 2005.

[23] E. Kolman and M. Margaliot, "Are artificial neural networks white boxes?" *IEEE Transactions on Neural Networks*, vol. 16, no. 4, pp. 844–852, 2005.

[24] C. Chen, H.-X. Li, and D. Dong, "Hybrid Control for Robot Navigation - A Hierarchical Q-Learning Algorithm," *IEEE Robotics & Automation Magazine*, vol. 15, no. 2, pp. 37–47, 2008.

[25] V. Mnih *et al.*, "Human-level control through deep reinforcement learning." *Nature*, no. 518, pp. 529–533, 2015.

[26] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems*, vol. 90, no. 2, pp. 111–127, 1997.

[27] R. T. Das *et al.*, "ieRSPOP: A novel incremental rough set-based pseudo outer-product with ensemble learning," *Applied Soft Computing*, vol. 46, pp. 170–186, 2016.

[28] S. W. Tung, C. Quek, and C. Guan, "SaFIN: A Self-Adaptive Fuzzy Inference Network," *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 1928–1940, 2011.

[29] J. W. Hostetter and M. Chi, "Latent space encoding for interpretable fuzzy logic rules in continuous and noisy high-dimensional spaces," in *2023 IEEE International Conference on Fuzzy Systems*. IEEE, 2023.

[30] J. W. Hostetter, M. Abdelshiheed, T. Barnes, and M. Chi, "A self-organizing neuro-fuzzy q-network: Systematic design with offline hybrid learning," in *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2023.

[31] O. Ciftioglu, "On the implication of equivalence of fuzzy systems to neural networks," in *FUZZ*, vol. 1, 2003, pp. 19–24 vol.1.

[32] H.-X. Li and C. Chen, "The equivalence between fuzzy logic systems and feedforward neural networks," *IEEE Transactions on Neural Networks*, vol. 11, no. 2, pp. 356–365, 2000.

[33] E. Kolman and M. Margaliot, "Knowledge Extraction From Neural Networks Using the All-Permutations Fuzzy Rule Base: The LED Display Recognition Problem," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 925–931, 2007.

[34] A. Gevaert *et al.*, "Distillatie van diepe reinforcement learning modellen," 2019.

[35] Z. Li, F. Bu, and F. Yu, "Temporal fuzzy association rules mining based on fuzzy information granulation," in *2017 13th ICNC-FSKD*, 2017, pp. 1168–1174.

[36] A. Karaci, "Intelligent tutoring system model based on fuzzy logic and constraint-based student model," *Neural Computing and Applications*, vol. 31, no. 8, pp. 3619–3628, 2019.

[37] C. Troussas *et al.*, "Machine learning and fuzzy logic techniques for personalized tutoring of foreign languages," in *AIED*, 2018, pp. 358–362.

[38] M. Machado *et al.*, "A fuzzy logic application in virtual education," *Procedia Computer Science*, vol. 91, pp. 19–26, 2016.

[39] M. Zarandi *et al.*, "A Fuzzy Expert System Architecture for Intelligent Tutoring Systems: A Cognitive Mapping Approach," *Intelligent Learning Systems and Applications*, vol. 4, no. 1, pp. 29–40, 2012.

[40] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[41] T. Mandel *et al.*, "Offline policy evaluation across representations with applications to educational games," in *AAMAS*, 2014, pp. 1077–1084.

[42] P. Wang *et al.*, "Interactive narrative personalization with deep reinforcement learning," in *IJCAI*, 2017.

[43] M. Sanz Ausin *et al.*, "Exploring the impact of simple explanations and agency on batch deep reinforcement learning induced pedagogical policies," in *Artificial Intelligence in Education*. Cham: Springer International Publishing, 2020, pp. 472–485.

[44] S. Schneider *et al.*, "The autonomy-enhancing effects of choice on cognitive load, motivation and learning with digital media," *Learning and Instruction*, vol. 58, pp. 161–172, 2018.

[45] H. Wood and D. Wood, "Help seeking, learning and contingent tutoring," *Computers & Education*, vol. 33, no. 2, pp. 153–169, 1999.

[46] A. Mitrovic and B. Martin, "Scaffolding and fading problem selection in sql-tutor," in *AIED*, 2003, pp. 479–481.

[47] V. Mnih, K. Kavukcuoglu, D. Silver *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[48] C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I & II," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 2, pp. 404–435, 1990.

[49] G. Klir and B. Yuan, *Fuzzy sets and fuzzy logic - theory and applications*. Upper Saddle River, New Jersey: Prentice-Hall Inc., 1995.

[50] C.-H. Chen, T.-P. Hong, and S.-B. Lin, "Mining fuzzy temporal knowledge from quantitative transactions," in *Proceedings 2011 International Conference on System Science and Engineering*, 2011, pp. 405–409.