A Safety Aware Deep Reinforcement Learning Technique for Automated Insulin Delivery

Giada Lops[1], Gioacchino Manfredi[2], Vito Andrea Racanelli[2], Luca De Cicco[2], Saverio Mascolo[2]

***Abstract*— Automated Insulin Delivery (AID) systems have shown great promise in managing diabetes by automating insulin administration. However, a significant challenge remains: preventing hypoglycemia or hyperglycemia during dynamic glucose fluctuations while minimizing the daily insulin dosage, or control effort. This study explores ways to enhance AID systems using a Reinforcement Learning (RL) algorithm called Maskable Proximal Policy Optimization (Maskable PPO), based on invalid action masking. Our findings demonstrate that this approach leads to a safety-aware framework for AIDs, providing highly realistic simulation scenarios for individual adult, adolescent, and child patients. The results show improved Time In Range (TIR) metrics (96.39%, 96.85%, and 54.43%), prevention of emergency bolus administration in adolescent and adult patients, and a reduction in Total Injected Insulin (TII) (17.32 U, 13.52 U, and 4.35 U per day).**

I. Introduction

Diabetes mellitus is a chronic metabolic disorder characterized by abnormally elevated blood glucose concentrations, which progressively damage vital organs such as the heart, blood vessels, eyes, kidneys, and peripheral nerves. Currently, over 800 million adults worldwide are living with diabetes, a number that has more than quadrupled since 1990. By 2030, the World Health Organization (WHO) aims for 80% of individuals with diagnosed diabetes to achieve good glycemic control, as part of the five global diabetes coverage targets established in 2022 [1]. The most prevalent form, type 2 diabetes, arises when the body's cells exhibit insulin resistance or when insulin secretion is insufficient to maintain normal glucose levels, particularly following daily food intake. In contrast, type 1 diabetes, though less common, results from autoimmune destruction of the insulin-producing beta cells in the pancreas, leading to minimal or no insulin production and necessitating lifelong insulin therapy. Effective management of diabetes involves maintaining blood glucose levels within a target range while preventing both hypoglycemia and hyperglycemia. It is important to note that a significant performance metric in diabetes management is the *Time In Range* (TIR), which is defined as the amount of time a person affected by diabetes spends within a specified glucose target range. Currently, the American Diabetes Association (ADA) recommends maintaining a TIR greater than 70% (i.e., at least 17 hours of a 24-hour day within the target range). For most adults with Type 1 or Type 2 diabetes, the recommended targets are: $\geq$ 70% between 70–180 mg/dL, $<$ 1% below 54 mg/dL, $<$ 4% from 54–69 mg/dL, $<$ 25% from 181–250 mg/dL, and $<$ 5% above 250 mg/dL. However, this general guideline should be tailored to the patient's lifetime and individual circumstances, as glucose targets vary based on factors such as age, life expectancy, kidney dysfunction, gastroparesis, cardiovascular disease, hypoglycemia unawareness, pregnancy, or trying to conceive [2].

Automated Insulin Delivery devices (AIDs), primarily designed for the management of Type 1 Diabetes, track blood sugar levels (G) using a *Continuous Glucose Monitor* (CGM), based on the number of glucose molecules in the interstitial fluid (N_{int}), as shown in the control scheme in Fig. 1. Given the *patient model*, representing the biological system (plant) being regulated and receiving the meal input (M), such systems administer a properly computed insulin dosage (I_c) using the *glucose control algorithm*, which processes glucose information, including reference glucose (G_r) and current glucose reading (G). The algorithm is actuated by the *insulin pump*. Depending on the system, doctors may also configure individual settings, such as basal rates, insulin sensitivity, and carbohydrate-to-insulin ratios, to personalize therapy. The overall goal is to maintain glucose levels within a predefined safe range (i.e., 70–180 mg/dL).

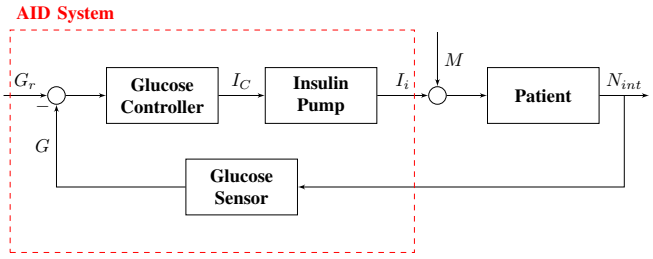


Fig. 1. Closed-loop Artificial Insulin Delivery (AID) system.

In the field of control theory, significant advancements have been made towards the development of effective glucose control algorithms, ranging from Proportional-Integral-Derivative (PID) controller to Model Predictive Control (MPC) approaches [3], the latter of which are known for their remarkable ability to predict glucose level dynamic over time. Additionally, Artificial Neural Network (ANN), Fuzzy Logic and Hypoglycemic Predictive Algorithms (HPA), have shown great potential in effectively handling the glucose readings over time [4]. Nowadays, the market offers several

*This work has been co-funded by the European Union and the Swiss State Secretariat for Education, Research and Innovation (SERI), in the frame of the SHIELD (grant N° 101156751) project. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or SERI. Neither the European Union nor the granting authorities can be held responsible for them.

All the authors are with Politecnico di Bari, Bari, Italy
[1]g.lops@phd.poliba.it
[2]name.surname@poliba.it

possible solutions: some systems are fully automated (e.g., Fully Closed Loop, FCL [5]), whereas others require user intervention (e.g., Predictive Low-Glucose Suspend, PLGS [6], for meal announcements). Most of the approaches described require accurate reporting of the meals consumed by the user to estimate the correct amount of insulin.
On the other hand, solutions that do not rely on such reporting face difficulties in ensuring promising TIR values, since they lack generalizability to different patient conditions. As a result, existing risk metrics tend to rise sharply, signaling ongoing crises and compromising the performance of AID systems. Moreover, another perspective to consider is the increasing number of patients needing treatment and monitoring, for non-communicable diseases (NCDs) and chronic illnesses in general. As of 2021, non-communicable diseases (NCDs) were responsible for over 43 million deaths globally, accounting for approximately 75% of all non-pandemic-related deaths [7]. This trend requires a high availability of resources and medical staff, who, despite the recent introduction of the telemedicine paradigm, struggle to provide continuous 24-hour patient care. Currently, medical devices are designed to achieve a high level of autonomy, where the reliance on healthcare professionals should decrease, helping alleviate the heavy burden on them. The highest level of autonomy according to the definition by Beal et al. [8] is Level 5, where doctors set goals and ideally receive the expected results on patients. In the context of glycemic management, this means maintaining optimal blood glucose levels, between 70-80 mg/dL, as defined by the TIR metric, ultimately leading to improved risk metrics.

This is where our approach fits well, leveraging Reinforcement Learning in a way that ensures safe exploration of the action space through *Maskable Proximal Policy Optimization* (Maskable PPO) control algorithm, which extends the standard PPO framework by incorporating *invalid action masking*. This technique prevents the policy from selecting infeasible actions by dynamically modifying the action space at each decision step, improving sample efficiency and stability in environments with constrained action sets [9]. The proposed approach is compared against the standard PPO, Recurrent PPO and a Standard Basal Bolus controller, representing current clinical practice, through *in silico* simulations within a cohort of three patients: an adult, an adolescent, and a child. This cohort is chosen to represent the metabolic diversity of each patient, which may require highly different control approaches. The structure of the paper is as follows. Section II presents the relevant literature and theoretical foundations, Section III describes the control methodologies considered and the simulation framework, Section IV discusses the results of the comparison, including standard insulin dosage methods, leading to the conclusion.

II. Related Work

In this section, we present the current state-of-the-art in applying Deep RL to AID systems.

Our investigation models Deep RL agents using a Markov Decision Process (MDP), a framework that captures decision-making with both stochastic and deterministic rewards and penalties. A generic MDP is a tuple $M(S, A, T, R, \pi)$, where S is a set of *states* describing the environment, A is the set of *actions* that an agent can take, T is a *transition* function $T : S \times A \to S$ which maps a state s_t to a new state s_{t+1} in response to the selected action a_t, $R(S, A)$ is the immediate reward, also known as reinforcement, and lastly, the *policy* $\pi(a \mid s) : A \times S \to [0, 1]$ describes the behavior of the agent, which is a probabilistic distribution over the possible actions.

The goal of an MDP is to find an optimal policy, denoted π^*, which maximizes cumulative rewards over time, also known as the *expected discounted sum of rewards*. In the case of an infinite horizon, it is expressed as: $\mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{a_k}(s_k, s_{k+1})\right]$, while for a finite horizon H, it focuses on a fixed and shorter time frame, given by: $\mathbb{E}\left[\sum_{k=0}^{H-1} \gamma^k R_{a_k}(s_k, s_{k+1})\right]$. Where γ is a parameter, $0 \leq \gamma \leq 1$, called *discount factor*. A value of γ close to 1 mainly focuses on long-term rewards, whereas a value close to 0 prioritizes immediate rewards and largely ignores future rewards. RL enables the development of an intelligent agent that interacts with a given environment to learn an optimal policy. Classical RL methods typically rely on tabular representations or linear function approximations, which become impractical for high-dimensional state and action spaces. Deep RL (DRL) methods overcome this limitation by leveraging deep neural networks to approximate value functions, policies, allowing learning in complex, high-dimensional environments [10].

Several DRL algorithms have been applied to AID, including Deep Q-Networks (DQN) [11], Deep Deterministic Policy Gradient (DDPG) [12], Twin Delayed DDPG (TD3) [13] and Soft Actor-Critic (SAC) [14]. Notably, the Proximal Policy Optimization (PPO) algorithm [15] has demonstrated promising results, advancing the adoption of RL-based controllers for insulin dosage. PPO was introduced as a more computationally efficient alternative to its predecessor, Trust Region Policy Optimization (TRPO). Differently from TRPO, which enforces a trust region by solving a constrained optimization problem that involves computing the Hessian matrix, PPO simplifies this process by using a surrogate objective to limit the Kullback-Leibler (KL) divergence between the old and new policies. This first-order optimization approach makes PPO more scalable and practical for large-scale problems.

A modular Deep RL PPO algorithm, whose aim is to reach full autonomy avoiding meal announcement or estimation, was introduced in [16]. It was tested over 20 patients (10 adults and 10 adolescents), through *in silico* simulations with a meal schedule with a daily carbohydrate intake of 180g, reporting a TIR of 73% and 64% for the two different cohorts, respectively.

As the most recent published result, Dénes-Fazakas et al. [17] designed a PPO framework for the AID system, relying on a mathematical model tailored for only adults patients with type IVP (i.e., Identifiable Virtual Patient) diabetes. This

specific framework included random carbohydrate intake and CGM noise to replicate real-life conditions, and the best reported TIR metric was again 73%.

III. Methodologies

This section presents the preliminaries of the chosen Deep RL algorithm, which served as the foundational knowledge for our work. We then introduce the proposed scenario, along with the chosen environment settings and design considerations.

A. Maskable Proximal Policy Optimization (Maskable PPO)

Invalid action masking was introduced by Huang et al. [9] to address certain PPO's limitations. The authors stated that when playing games as Real-time Strategy (RTS) ones, they require complex rules, meaning that each specific state might have a set of valid actions that differ in size. The standard PPO framework requires sampling each action from a full action space; however, not all sampled actions are valid for every game state, necessitating their exclusion. In the aforementioned paper, the authors used a state-dependent differentiable function, namely a *mask*, which showed better performance than invalid action penalization, the latter often struggling with scalability and reward acquisition.

Mask Designing: In this proposed pipeline, we implement invalid action masking using a custom function. Following the same notation and approach employed in [9], in this work we consider a continuous state space and a discrete action space. As a result, the agent interacts with the environment by selecting insulin doses as actions just by sampling from the set of valid actions possible in the given state. The discrete action set is defined as $A = \{a_1, \ldots, a_{n_T}\}$, where each action a_i corresponds to a different insulin dosage level, with $a_1 = 0 < \cdots < a_{n_T} = a_{\max}$. The step size between consecutive insulin dosage is given by $\Delta a = a_{n_T}/n_T$, where n_T represents the total number of actions defining the full discrete action space. The state $s \in S$ represents the patient's current glucose level G_t. The custom masking function, defined in Eq. 1 as mask : $\mathbb{R} \to \mathbb{R}$, ensures that only safe actions are available to the agent. Specifically, this function sets the probability of selecting invalid actions to zero, preventing the agent from choosing unsafe insulin doses, even if the probability of doing so would otherwise be extremely low. This guarantees patient safety by eliminating the risk of selecting harmful actions.

This definition requires to consider a policy parameterized by $\boldsymbol{\theta} = [l_1, \ldots, l_{n_T}]$, where each l_i represents the logit associated with action a_i. A logit is an unnormalized score that determines the relative preference for selecting a particular action. For a given state s, the function is defined as follows [9]:

$$\text{mask}(\theta_i) = \begin{cases} l_i, & \text{if } a_i \text{ is valid in } s \\ M, & \text{otherwise} \end{cases} \tag{1}$$

With this structure, the mask only updates logit values to a large negative number M whenever the action is declared invalid for that given state, and leaves them unchanged otherwise. In our application, when the current glucose reading $G_t < 90$ mg/dL, the only valid action is the first one, i.e. $a_1 = 0$ and only allows the last n_v actions when the glucose reading exceeds 160 mg/dL. The action probabilities are then computed using the softmax function: $\boldsymbol{\pi_\theta}(s) = \text{softmax}(\boldsymbol{\theta})$, where $\text{softmax}(\theta_i) = e^{\theta_i} / \sum_{j=1}^{n} e^{\theta_j}$.

In the following we make a concrete example that shows the mask role in the learning phase. Let us suppose for a given glucose state s, we compute the following action probabilities:

$$\boldsymbol{\pi_\theta}(s) = \text{softmax}\left([0.1, 0.2, 0.5, 0.2, \ldots, 0.05]\right) \tag{2}$$

Now, suppose the glucose level is 85 mg/dL. According to the proposed mask function, the valid action is to select a null insulin dosage (a_1). Therefore, the logits for the invalid actions are masked out by setting them to a large negative number M:

$$\boldsymbol{\pi_\theta}'(s) = \text{softmax}\Big(\text{mask}\big(\boldsymbol{\theta}\big)\Big) = \text{softmax}\Big([l_1, M, \ldots, M]\Big) \tag{3}$$

This results in the following action probabilities in Eq. 4, according to which the agent will always select action a_1.

$$\boldsymbol{\pi_\theta}'(s) = ([1.0, 0.0, \ldots, 0.0]) \tag{4}$$

Similarly, if the glucose reading is, f.i., 175 mg/dL, the mask function will allow only high insulin doses, and the action probabilities will be updated to reflect this:

$$\boldsymbol{\pi_\theta}'(s) = \text{softmax}\left([M, \ldots, M, l_{n_T - n_v + 1}, \ldots, l_{n_T}]\right) \tag{5}$$

where $l_{n_T - n_v + 1}, \ldots, l_{n_T}$ correspond to the valid actions in the high range. After applying the mask, the agent will choose one of these valid actions with non-zero probability.

Following action execution, the policy is updated per the observed reward r_t as follows:

$$r_t = \begin{cases} -3 & \text{if } G_t < 70 \\ 1 & \text{if } 70 \leq G_t \leq 180 \\ -1.5 & \text{if } G_t > 180 \end{cases} \tag{6}$$

where t denotes the step. Notice that, in Eq. (6) a higher penalty (-3) is assigned to the case of hypoglycemia, whereas a lower penalty is set in the case of hyperglycemia (-1.5).

The policy gradient g for invalid action masking is then computed as:

$$g = \mathbb{E}_T \left[\nabla_\theta \sum_{t=0}^{T-1} \log(\pi'_\theta(a_t|s_t)) \sum_{k=0}^{\infty} \gamma^k r_{t+k} \right] \tag{7}$$

It must be pointed out that if the current glucose reading is 105 mg/dL, no action would be specifically invalid or masked. Consequently, the agent has the flexibility to freely sample the action within the entire range space of discrete actions. This approach ensures that the agent only selects insulin doses that are valid based on the patient's current glucose levels, thereby avoiding dangerous or ineffective insulin administrations whenever the patient is potentially close to reaching hypoglycemia or hyperglycemia conditions.

B. Evaluation Framework for Simulations

The proposed control algorithm has been trained and tested through the Simglucose simulator [18], which is a Python implementation of the FDA-approved UVa/Padova Simulator (2008 version) [19]. Our evaluation framework includes three Type 1 Diabetes patients provided from the simulator. The main features of the considered cohort are listed in Table I where BW is the Body Weight of the patient in kilograms (kg), CR represents the Carbohydrate Ratio indicating how many grams of carbohydrate require 1 U of insulin, CF is the Carbohydrate Factor in mg/dL showing how much 1 U of insulin will lower the blood glucose and Total Daily Insulin (TDI) refers to the total amount of insulin that a person with diabetes needs in a 24-hour period.

In our study we assume $n_T = 20$ and $n_v = 4$. For each of the patients, we design a specific discrete action space, setting the Δa value for the adult, adolescent, and child according to their actual insulin needs, at 0.0040, 0.0036, and 0.00075 U/min, respectively. These values were determined by considering that children are characterized by a higher insulin sensitivity and lower body weight compared to adolescents and adults resulting in higher CFs and lower TDIs [19].

TABLE I
PATIENT DATA WITH CR, CF, AGE, TDI AND BW

Name	CR	CF	Age (y)	TDI (U)	BW (kg)
adult#001	10	8.7731	61	50.4167	102.32
adolescent#001	12	15.0360	18	36.7339	68.71
child#001	25	42.7177	9	17.4729	34.56

Besides the TDI, we consider the Risk Index (RI) as a performance indicator defined in [20]. In a nutshell, the RI estimates the risk of blood glucose levels as a function of the clinical metrics Low Blood Glucose Risk Index (LBGI) and High Blood Glucose Risk Index (HBGI).

When constructing the Gymnasium [21] environment, it was crucial to select consistent Sensory and Insulin Pump devices for each patient as provided by the simulator. The Dexcom [22] sensor and Insulet [23] pump were chosen for this purpose. A glucose target of 112 mg/dL was set as it effectively minimizes the RI to zero when the safe glucose range is between 70 and 180 mg/dL.

For our simulation, we utilized the Maskable PPO version from the Stable-Baselines3 Contrib Python Library [24], using hyperparameter values as reported in Table II.

TABLE II
MASKABLE, RECURRENT AND STANDARD PPO MODEL HYPERPARAMETERS.

Parameter	Value
Batch Size	128
Learning Rate	5×10^{-4}
Discount Factor (γ)	0.99
Clipping Range	0.2
Value Function Coefficient	0.5
Max Gradient Norm	0.5

The selected policy type, which is the neural network structure, is a multilayer perceptron. The meal schedule for the purpose of training is designed to ensure generalizability among patients. We developed a custom scenario featuring a five-meal plan with a standard deviation of 10 g of carbohydrates per meal and a 30-minute variation in meal timing. The average meal plan consists of 40 g (breakfast), 15 g (mid-morning snack), 60 g (lunch), 10 g (mid-afternoon snack), and 55 g (dinner) of carbohydrates, respectively. This results in a daily carbohydrate intake ranging from a minimum of 130 g to a maximum of 180 g for each adult patient.

The model was trained using data from the given patients, each undergoing 14,440 simulation steps. This corresponds to a real-world time of 43,320 minutes (due to the $t_{sample} = 3$ s, of the chosen CGM sensor), or 722 hours, roughly 30 days. The truncated condition is encountered when glucose levels drop below 50 mg/dL or rise above 250 mg/dL. Moreover, to further discourage these events, a strong negative reward of -5 is applied. The trained model was evaluated on 100 unseen scenarios to assess its generalization capability under diverse conditions. Our goal is to achieve an average performance that adheres to safety guidelines [2] while also potentially overcoming current results reported in the other aforementioned studies.

C. Other evaluated Control Methodologies

For a comprehensive comparison we choose to test the performance of both the standard PPO and the Recurrent PPO. We opt to preserve the same hyperparameters of the Maskable PPO proposed control strategy. Recurrent PPO [25] allows to obtain recurrent policy, despite a standard PPO control algorithm. The Stable-Baselines3 Contrib Python library [24] supports this version of PPO, utilizing exclusively LSTMs (Long Short Term Memory), over GRUs (Gated Recurrent Units), as recurrent network architectures. These networks are particularly suitable for POMDPs (Partially Observable Markov Decision Processes) by retaining the action-observation history in their hidden layers, which compensates for the non-fully observable states.

To complete our evaluation, we include a *Standard Basal Bolus Controller* (SBBC) in our analysis. The latter represents the current clinical practice which relies on a mathematical computation of the insulin bolus and basal dosage, given an announced carbohydrate intake [18]. The basal and bolus insulin are computed as follows [18]:

$$i_{total} = \underbrace{\frac{u2ss \times BW}{6000}}_{i_{basal}} + \underbrace{\frac{\left(\frac{M}{CR} + \frac{G-G_r}{CF}\right)}{t_{sample}}}_{i_{bolus}} \tag{8}$$

where $u2ss$ is the steady-state insulin requirement in units per kilogram body weight (pmol/(L·kg)) and 6000 is a constant used to convert the steady-state insulin rate into the basal insulin rate in U/min. The other parameters have been previously defined. The computed bolus dose is adjusted to an insulin rate, converting the bolus amount into U/min for use in the simulation environment [26].

TABLE III

EVALUATION METRICS FOR MASKABLE PPO, STANDARD PPO, AND RECURRENT PPO ACROSS CHILD, ADOLESCENT, AND ADULT PATIENTS

Metric		Maskable PPO			Standard PPO			Recurrent PPO		
		Child	Adolescent	Adult	Child	Adolescent	Adult	Child	Adolescent	Adult
Time in Range [%]	Mean	54.43	96.85	96.39	38.52	87.42	92.65	37.70	68.25	95.60
	Median	54.17	98.33	98.33	33.41	87.60	92.71	36.21	68.85	96.46
	Min	40.00	84.79	76.25	25.78	71.46	78.96	19.10	35.00	83.96
	Max	71.25	100.00	100.00	67.20	99.17	100.00	68.90	96.04	100.00
Average Glucose Level [mg/dL]	Mean	189.06	115.05	126.09	297.62	95.69	108.28	290.40	85.58	115.39
	Median	189.91	115.17	125.70	304.14	95.11	108.54	291.50	85.33	115.76
	Min	157.36	106.51	115.42	210.69	87.11	90.85	223.77	74.66	96.49
	Max	214.54	125.65	135.01	381.95	107.06	124.84	344.97	95.75	135.63
Totale Injected Insulin [U]	Mean	4.35	13.52	17.32	2.79	16.81	21.25	1.52	18.95	19.90
	Median	4.36	13.58	17.27	3.11	16.83	21.21	1.56	18.97	19.85
	Min	2.48	12.01	15.96	1.62	15.81	19.91	1.07	17.95	18.56
	Max	4.83	15.02	19.16	3.47	17.77	22.64	2.00	19.84	21.26
Risk Index	Mean	15.89	1.47	1.55	35.17	3.05	1.97	32.83	6.12	1.38
	Median	16.16	1.42	1.38	35.97	3.02	1.73	33.11	5.91	1.31
	Min	6.91	0.55	0.63	17.09	1.33	0.62	17.81	3.10	0.53
	Max	23.03	3.67	3.56	52.41	5.33	5.24	45.37	10.68	4.62
Cumulative Rewards	Mean	-771.19	424.15	418.84	-1413.97	220.30	303.61	-734.47	-241.10	389.80
	Median	-772.00	450.00	456.75	-1636.00	232.25	332.00	-764.75	-216.50	412.00
	Min	-1262.00	159.50	111.00	-2082.50	-157.00	-227.00	-1317.00	-1088.00	17.00
	Max	-80.00	480.00	480.00	-340.00	464.00	480.00	-173.50	404.00	480.00

IV. RESULTS

This section focuses on the results achieved in the evaluation of the three given controllers. Simulations are carried out on a machine equipped with an AMD Ryzen 5 7520U processor (2.80 GHz). The training procedure takes ~ 496 s per patient per model, with a total runtime of around 1 h and 25 min.

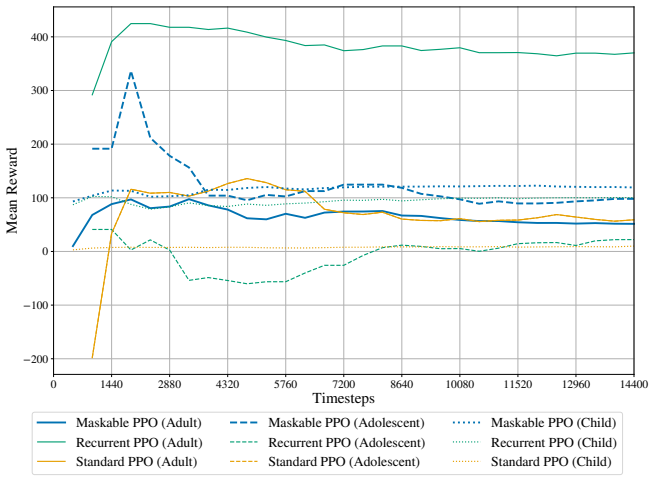


Fig. 2. Evolution of the mean reward over training steps.

Training Evolution: Fig. 2 shows mean rewards under different training conditions. Maskable PPO achieves strong performance, particularly for children (~ 120) and adolescents (~ 100), with stable convergence. Recurrent PPO excels in adults (peaking > 400) but shows instability in adolescents and poor performance in children. Standard PPO remains stable but with lower rewards, plateauing around 70 for adults, 70 for adolescents, and ~ 10 for children. Overall, Maskable PPO offers the best balance of performance and stability across groups, benefiting from invalid action masking.

Comparison of the three PPO variants: The findings reported in Table III and Fig. 3 are computed over 100 test scenarios with unseen meal patterns during the training phase, highlighting significant variations in controller performance across different patient types. To better illustrate performance

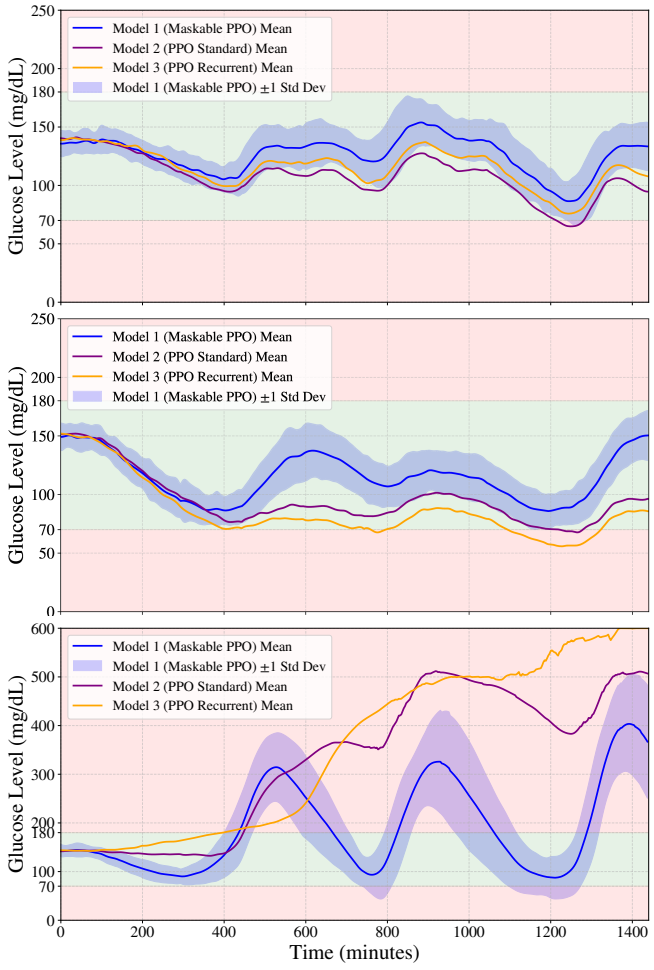


Fig. 3. Comparison of glucose levels for Adult#001 (top), Adolescent#001 (middle) and Child#001 (bottom).

variability, we report the minimum and maximum values for each metric, offering insights into both best- and worst-case scenarios across test episodes. Maskable PPO performs best overall, achieving the highest mean TIR for adult (96.39%), adolescent (96.85%), and child (54.43%) patients. While both Maskable and Recurrent PPO are effective in adults, performance declines in adolescents and even more so in children. For child patients, all controllers struggle, showing higher risk scores and lower TIR compared to adolescent and adult cases. These results highlight the challenge of achieving safe and effective glycemic control in this group. Among all controllers, only Maskable PPO exceeds the 50% TIR threshold (54.43%) recommended by the ADA for patients at higher hypoglycemia risk [2]. This suggests that further model refinement is necessary for safe glycemic management in pediatric patients. A dynamic basal infusion alone proves insufficient, and emergency bolus infusions may be required to manage glucose spikes during the day. Maskable PPO keeps average glucose levels for adolescents close to the 112 mg/dL target. The overall Risk Index (RI) indicates that it addresses the AID problem with minimal control effort, as shown by mean insulin usage (except in the child case with Recurrent PPO) and lower risk scores across all patients, without compromising glucose stability.

Comparison with Standard Basal Bolus Controller: We chose the SBBC introduced in Section III as a non-DRL baseline relying solely on reported meal intake. For this controller, key outcomes—TIR, average glucose, and insulin usage—are compared across the three patients. The adolescent achieves the highest mean TIR (97.02%), followed by the adult (81.93%) and the child (74.62%). Average glucose is highest in adults (136.01 mg/dL), lowest in children (120.39 mg/dL), with adolescents in between (129.25 mg/dL). Insulin usage averages 6.16 U (child), 18.89 U (adult), and 12.52 U (adolescent). While SBBC performs well, its effectiveness depends on accurate insulin reporting by the user. In contrast, our method seeks greater autonomy by eliminating this dependency.

Training and evaluation were conducted under controlled meal variations in timing and carbohydrate content. Yet, real-world scenarios often involve irregular or extreme deviations. Although the Maskable PPO policy showed robustness within the defined regime, its performance under out-of-distribution inputs remains to be validated. Future work will explore domain randomization and testing under irregular meal patterns to enhance robustness and safety.

Conclusion

Maskable PPO, through invalid action masking, proved effective in AID systems by ensuring valid insulin actions, reducing risks, and improving performance over other PPO variants and clinical methods. Simulations confirmed its generalizability, achieving the highest TIR with minimal control effort while maintaining near-target glucose levels.

However, results highlight the need for refinements in pediatric cases and suggest that insulin rate adjustments, when the controller does not receive meal intake input from the user, may be insufficient, potentially requiring emergency boluses. Further research could overcome these limitations and enhance its adaptability across different patient scenarios and conditions.

References

[1] WHO, "Diabetes," https://www.who.int/health- topics/diabetestab= tab1, 2024.

[2] A. D. Association, "Cgm & time in range," https://diabetes.org/about-diabetes/devices-technology/cgm-time-in-range, 2024.

[3] A. Cinar, "Automated insulin delivery algorithms," *Diabetes Spectrum*, 2019.

[4] E. Dassau, E. Atlas, and M. Phillip, "Closing the loop," *International Journal of Clinical Practice Supplement*, 2010.

[5] C. K. Boughton, "Fully closed-loop insulin delivery—are we nearly there yet?" 2021.

[6] M. Joubert, A. R. Briant, L. Kessler, F. Fall-Mostaine, S. Dubois, B. Guerci, and J.-J. Parienti, "Sensor-augmented insulin pump with predictive low-glucose suspend (plgs)," *Diabetes Therapy*, 2022.

[7] W. H. Organization, "Noncommunicable diseases," 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/noncommunicable-diseases

[8] J. Beal and M. Rogers, "Levels of autonomy in synthetic biology engineering," *Molecular Systems Biology*, 2020.

[9] S. Huang and S. Ontañón, "A closer look at invalid action masking in policy gradient algorithms," *CoRR*, 2020.

[10] Y. Li, "Deep reinforcement learning: An overview," 2018.

[11] 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*, 2015.

[12] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019.

[13] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International conference on machine learning*. PMLR, 2018.

[14] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine, "Soft actor-critic algorithms and applications," 2019.

[15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.

[16] C. Hettiarachchi, N. Malagutti, C. J. Nolan, H. Suominen, and E. Daskalaki, "G2p2c — a modular reinforcement learning algorithm for glucose control by glucose prediction and planning in type 1 diabetes," *Biomedical Signal Processing and Control*, 2024.

[17] L. Dénes-Fazakas, L. Szilágyi, L. Kovács, A. De Gaetano, and G. Eigner, "Reinforcement learning: A paradigm shift in personalized blood glucose management for diabetes," *Biomedicines*, 2024.

[18] J. Xie, "Simglucose v0.2.1," Online, 2018. [Online]. Available: https://github.com/jxx123/simglucose

[19] C. D. Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, "The uva/padova type 1 diabetes simulator: New features," *Journal of Diabetes Science and Technology*, 2014.

[20] W. Clarke and B. Kovatchev, "Statistical tools to analyze continuous glucose monitor data," *Diabetes technology amp; therapeutics*, 2009.

[21] M. Towers, A. Kwiatkowski, J. Terry, J. U. Balis, G. De Cola, T. Deleu, M. Goulão, A. Kallinteris, M. Krimmel, A. KG *et al.*, "Gymnasium: A standard interface for reinforcement learning environments," *arXiv preprint arXiv:2407.17032*, 2024.

[22] Dexcom, Inc., "Dexcom continuous glucose monitoring," 2025. [Online]. Available: https://www.dexcom.com/it-it

[23] Insulet Corporation, "Insulet: Innovative insulin delivery systems," 2025. [Online]. Available: https://www.insulet.com/

[24] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, 2021.

[25] M. Pleines, M. Pallasch, F. Zimmer, and M. Preuss, "Generalization, mayhems and limits in recurrent proximal policy optimization," 2022.

[26] P. C. Davidson, H. R. Hebblewhite, R. D. Steed, and B. W. Bode, "Analysis of guidelines for basal-bolus insulin dosing: Basal insulin, correction factor, and carbohydrate-to-insulin ratio," *Endocrine Practice*, 2008.