Adaptive PID Control for Setpoint Tracking Using Reinforcement Learning: A Case Study for Blood-Glucose Control

Anna Hakhverdyan[1,2,⋆], **Golnaz Mesbahi**[1,2,⋆], **Martha White**[1,2,3]
{hakhverd,mesbahi,whitem}@ualberta.ca

[1]**Department of Computing Science, University of Alberta, Canada**
[2]**Alberta Machine Intelligence Institute (Amii)**
[3]**CIFAR AI Chair**

⋆Equal contribution

Abstract

Blood-glucose control is a classic example of setpoint tracking, where the controller must continuously adjust insulin delivery to maintain a desired glucose level. While simple feedback controllers, like proportional-integral-derivative (PID), are commonly used, they can not leverage contextual information that could lead to better performance. Reinforcement learning (RL) has shown promise for such control problems, but its use in continual setpoint tracking—where learning happens online during deployment—remains underexplored. In this work, we study how the on-policy RL algorithm PPO performs in blood-glucose control under different observability conditions. We build a continuing blood-glucose control environment based on the Bergman model and evaluate PPO in a series of increasingly difficult scenarios: starting with a deterministic case, then introducing stochasticity, and finally testing how well learned policies transfer across different patients. Our results show that standard PPO struggles even in relatively simple settings, underscoring the need for further research to make RL more reliable for setpoint tracking. However, we find that modifying PPO's policy to output PID gains—effectively using PPO to tune a PID controller—significantly improves stability and performance, demonstrating a promising direction for RL in process control.

1 Introduction

Reinforcement learning (RL) has been extensively applied to various process control problems (de Rezende Faria et al., 2022; Yoo et al., 2021; Spielberg et al., 2017; Martinez-Piazuelo et al., 2020), ranging from wastewater treatment (Chen et al., 2020) to nonlinear processes in continuous stirred tank reactors (Shah & Gopal, 2016). RL offers potential advantages over traditional control methods when the environment is stochastic, particularly when there are no explicit presumptions about the process (Mowbray et al., 2021). However, deploying RL agents for online control in such settings presents several challenges, including ensuring stability and adaptability in dynamic environments and achieving robust generalization to new conditions.

Blood glucose control for diabetic patients is an example of a chemical process control problem and a crucial challenge in healthcare (Bergman et al., 1979). The complexity arises from individual variability in lifestyle, diet, and physiological responses, making glucose regulation a highly dynamic task, requiring continuous and online control.

A well-established category of solutions for process control problems consists of classic feedback control methods, such as proportional-integral-derivative (PID) controllers. These controllers con-

tinuously adjust their actions according to the difference between the process variable and the desired setpoint. In a blood glucose control setting, a PID controller controls insulin levels in response to fluctuations in glucose levels, maintaining glucose level to a particular setpoint (Khaqan et al., 2022). While PID controllers are widely used because of their simplicity and stability, they often struggle with handling time delays and highly dynamic and nonlinear systems (Lee et al., 1999).

Given these challenges in both PID and the process control systems, RL presents a promising approach (Fox & Wiens, 2019), as it enables adaptive decision-making in uncertain environments (Tejedor et al., 2020). Previous research has explored various RL approaches for blood glucose control, including model-free, actor-critic algorithms combined with recurrent neural networks (Fox et al., 2020). Other works have integrated model-based learning with deep RL techniques to improve performance and stability (Hettiarachchi et al., 2024). However, many of these solutions still rely on manual human feedback (Emerson et al., 2025), extensive hyperparameter tuning, and carefully designed safety-informed reward functions (Zhao et al., 2025).

A promising approach integrates PID with RL, where the RL agent dynamically tunes PID gains while the PID controller regulates the process (Qin et al., 2018; Dogru et al., 2022; Shuprajhaa et al., 2022a;b; Adesanya et al., 2024). Several variations of this approach have been explored: one method parameterizes an RL agent as a PID controller and uses it to adjust PID gains in an offline manner (McClement et al., 2022); another method augments the RL agent with a PID controller while leveraging evolutionary learning techniques to enhance performance (Bloor et al., 2024); a third approach modifies the learning framework to include episodic updates and rollback mechanisms during early training stage to mitigate instability (Lakhani et al., 2022). Other solutions incorporated attention networks with PID actions to guide the RL agent's decisions in the early stages of training (Lim et al., 2021).

However, most of these methods operate in an offline setting (Emerson et al., 2023) and do not account for online learning, potentially limiting adaptation to changing environments. Moreover, real-world process control applications, such as blood glucose regulation, often involve partial observability, where the agent lacks access to all environment variables, further complicating adaptation (Li et al., 2015; Ni et al., 2021; Hausknecht & Stone, 2015). Therefore, an effective online agent must continually adapt its learned policies (Zhu et al., 2023) to the new conditions.

This work explores the use of RL algorithms, specifically Proximal Policy Optimization (PPO) (Schulman et al., 2017), for blood-glucose regulation in an online learning setting. Despite the environment's simplicity, standard PPO—and even its recurrent variant—struggle, particularly under partial observability and stochasticity. To address this, we allow PPO agents to output PID controller gains instead of direct insulin dosages. Our experiments show that PID-based agents outperform standard PPO, providing greater stability, lower variability, and improved transfer between patients, highlighting the robustness of PID-based approaches with on-policy methods like PPO.

2 Blood-Glucose environment

In this paper, we focus on insulin-dependent diabetes, where blood glucose levels are regulated through insulin injections, diet, and exercise. It is a compelling environment to study, as poor action choices can have serious consequences: too much insulin can lead to dangerously low blood sugar, while too little can result in long-term complications like nerve damage, cardiovascular disease, and kidney failure. An effective controller must, therefore, continuously monitor glucose levels and maintain them within a safe range around the *basal* level G_b. Real-world challenges–such as stochasticity in meal and exercise timings, with delays and noise in glucose measurements–make this a rich and realistic testbed for RL-based approaches[1].

To model blood glucose control as a Markov Decision Process (MDP), we use the **Bergman Minimal Model** (Bergman, 1989) and its extended variant, the **exercise-augmented Bergman Minimal Model** (Roy & Parker, 2007) to define the environment's dynamics. An MDP is defined by the tuple

[1] Code is available at https://github.com/anna-ssi/bg-rlpid

$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function, and $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ specifies the transition dynamics. At each time step t, the agent observes state s_t, selects an action a_t, transitions to the next state s_{t+1} and receives a reward r_{t+1} according to the distribution $P(s_{t+1}, r_{t+1} \mid s_t, a_t)$. The agent's goal is to learn a policy $\pi(a \mid s)$ that maximizes the expected return, defined as the cumulative discounted reward, over time.

The Bergman model is a system of three differential equations that describe the dynamics of glucose and insulin. It has three core variables: glucose level G_t, insulin concentration I_t, and insulin action variable X_t. The exercise-augmented model expands this framework by introducing six additional equations, detailed in Appendix C. When we set the added variables to zero, we recover the original Bergman model. The equations common to both models are:

$$G_{t+1} = G_t + \Delta t \left(-P_1 G_t - X_t(G_t + G_b) + D_t + \frac{W}{V_G}(G_{\text{prod},t} - G_{\text{gly},t} - G_{\text{up},t}) \right),$$
$$I_{t+1} = I_t + \Delta t \left(-n(I_t + I_b) + \frac{U_t}{V_1} - I_{e,t} \right),$$
$$X_{t+1} = X_t + \Delta t \left(-P_2 X_t + P_3 I_t \right),$$

where the parts highlighted in blue correspond to the three equations used in the original Bergman Minimal Model. The parameters without the subscript t, such as P_1, P_2, P_3, W, along with basal glucose G_b and basal insulin I_b, among others, are patient-specific and are detailed in Appendix H.

We define the state $s_t = [G_t, I_t, X_t]$ to contain the three key variables the Bergman model. The transition from state s_t to s_{t+1} in the original Bergman model can be expressed as:

$$s_{t+1} = \begin{bmatrix} G_{t+1} \\ I_{t+1} \\ X_{t+1} \end{bmatrix} = s_t + \Delta t \cdot \begin{bmatrix} -P_1 G_t - X_t(G_t + G_b) + D_t \\ -n(I_t + I_b) + \frac{U_t}{V_1} \\ -P_2 X_t + P_3 I_t \end{bmatrix}.$$

For the exercise-augmented model, the state s_t is expanded to include the additional variables listed in Equation (2):

$$s_{t+1} = \left[G_{t+1}, I_{t+1}, X_{t+1}, G_{\text{prod},t+1}, G_{\text{up},t+1}, I_{e,t+1}, PVO^{\max}_{2,t+1}, G_{\text{gly},t+1}, A_{t+1} \right].$$

The action a_t represents the insulin dose administered at each time step t: for RL agents directly controlling the process, $a_t = U_t$, while for PID-based agents, $a_t = [K_p, K_i, K_d]$, where K_p, K_i, and K_d are the coefficients of the PID controller, which we discuss further in the next section. The reward is defined as $r_t = -|G_t - G_b|$, penalizing deviations of the glucose level from the basal G_b.

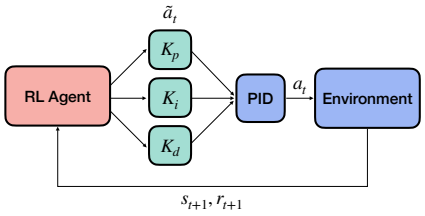
A key variable in this model is D_t, which represents system disturbances. It allows us to adjust the difficulty of the environment by simulating more complex or irregular glucose intake patterns, which we discuss further in Appendix B.

3 Controlling PID with RL

The PID controller has three components: proportional, integral, and derivative. The proportional term corresponds to the current error in the system, defined as $e_t = x_t - x_t^*$, where x_t is the system's current state and x_t^* is the desired target. In the case of blood glucose regulation, this corresponds to the difference between the current blood glucose level G_t and the basal level G_b. The integral term accumulates the error over time, helping the system reach x^* by correcting for past deviations. The derivative term anticipates future errors with the rate of change of the process variable, reducing the output if the system is changing too rapidly. Each term is weighted by its corresponding *gain* value – K_p, K_i, and K_d – serving as its adjustable parameters, as shown below:

$$u_t = K_p e_t + K_i \sum_{i=0}^{t} e_i \Delta t + K_d \frac{e_t - e_{t-1}}{\Delta t} \tag{1}$$

where Δt is the time step between successive control actions (set to 1 in our experiments). At the initial time step $t = 0$, since e_{t-1} is undefined, the derivative term is set to zero.



To enable the agent to adjust the gain variables, we treat the PID controller as part of the environment, as shown on the left. The RL agent receives the current state s_t from the environment and outputs the gain values $\tilde{a}_t = \{K_p, K_i, K_d\}$, which are then used by the PID controller to produce the action a_t. To accommodate this pipeline, we augment the state space to include the three key components of the PID controller: the error term e_t, cumulative error $\sum_{i=0}^{t} e_i$, and derivative error $e_t - e_{t-1}$. The different environment configurations, along with their corresponding state and action spaces, are detailed in Appendix F. The algorithmic implementation is presented in Algorithm 1 below:

Algorithm 1 On-policy RL for the PID control

Input: RL Algorithm `Alg`, total steps T, trajectory length τ
Initialize: system state x_0, setpoint x_0^*, start state s_0
for t in T **do**
  $\tilde{a}_t = (K_p, K_i, K_d) \leftarrow \texttt{Alg}(s_t)$
  $e_t \leftarrow x_t^* - x_t$
  $a_t \leftarrow$ Equation (1)(K_p, K_i, K_d, e_t, t)
  Take a step in the environment with a_t, get r_{t+1}, s_{t+1}, and the system states x_{t+1}, x_{t+1}^*
  Add $(s_t, \tilde{a}_t, r_{t+1}, s_{t+1})$ to the trajectory
  if $t \mod \tau = 0$ **then**
    Update `Alg` with the trajectory
  end if
end for

4 Experiments and results

We conducted a series of experiments in the Blood-Glucose environment with various configurations, including different meal and exercise schedules, levels of observation noise, and observability. Although the environment is periodic by nature, where each episode represents a single day, we focus on the continuing setting, where the state persists without resetting back to the start state, meaning each new day starts where the previous one left off.

We utilize the PPO algorithm under two observability settings: (1) *full observability*, where agents receive all relevant variables from the Bergman model and the PID controller, and (2) *partial observability*, where agents observe only the current blood glucose level. We evaluate four agent variants: **PPO**, **RecurrentPPO**, **PID-PPO**, and **PID-RecurrentPPO**, utilizing recurrence to improve performance under partial observability (Mnih et al., 2016). PPO and RecurrentPPO agents directly select the insulin action, while the PID-based agents determine the gains of the PID controller. As a baseline, we use a PID controller with default values of K_p, K_i, and K_d provided in (Hansen et al., 2019). All plots include bootstrapped confidence intervals of 95% calculated over 20 random seeds.

4.1 Deterministic environment

In this section, we evaluate the performance of the four PPO agents in a deterministic Blood-Glucose environment where the patient consumes three fixed meals per day. These meals produce three predictable glucose spikes, with meal times and amounts kept constant. Patient-specific data can be found in Appendix H.

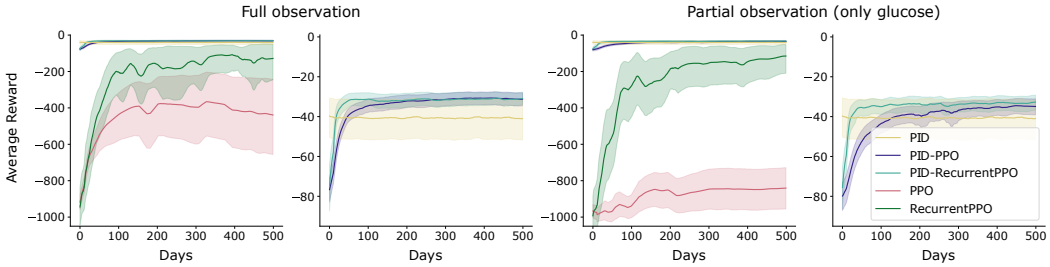


Figure 1: Online performance – measured as the average undiscounted reward – of the agents in the **exercise-augmented Blood-Glucose environment** with obese disturbance function, evaluated over 500 days. The first two plots show the performance of PPO agents under full observability, with the second plot showing a zoomed-in view of the PID-based agents. The last two plots present the agents' performance under partial observability.

As expected, in Figure 1, RecurrentPPO consistently outperforms the standard PPO agent in partially observable settings – a trend that also holds in fully observable settings. Notably, RecurrentPPO maintains similar performance across both observability settings, while standard PPO experiences a significant decline under partial observability.

Both PID-based agents perform similarly across the fully and partially observable settings. However, under partial observability, a slight performance gap emerges: PID-RecurrentPPO converges to baseline-level performance more quickly than PID-PPO. A significant advantage of the PID-based PPO agents – particularly evident in the zoomed-in view of Figure 1 – is their stability and improved sample efficiency: even within the first 100 episodes, these agents exhibit notable performance gains. Ultimately, both PID-based agents reach levels of performance comparable to or better than baseline PID in all configurations tested, as shown in Appendix H.

In this section, we showed that in the deterministic Blood-Glucose environment, the **PID-based agents outperform standard PPO agents by a significant margin, exceeding the performance of the baseline PID**. These results highlight the stability and efficiency of PID-based PPO agents, even in partially observable settings where glucose regulation is based solely on glucose measurements at each time step.

4.2 Stochastic environment

Imposing fixed meal times, portion sizes, and daily exercise routines is an unrealistic constraint that does not accurately reflect real-world conditions. Additionally, glucose measurement devices are inherently imprecise, with readings affected by up to 10% noise (Facchinetti, 2016). To better evaluate agent performance under more realistic conditions, we extend our environment to incorporate stochastic elements, including variability in meals, exercise, and sensor noise.

We first evaluate the performance of the four PPO agents under stochastic meal intake and exercise timing. Although we still assume that the patient exercises daily, we vary the time and duration of each session by randomly choosing a time from a set range. Similarly, for meal intake, we randomly choose the timing and glucose intake of the three main meals and introduce the possibility of snacking, adding more variability to the dynamics of the system, as described in Appendix B.

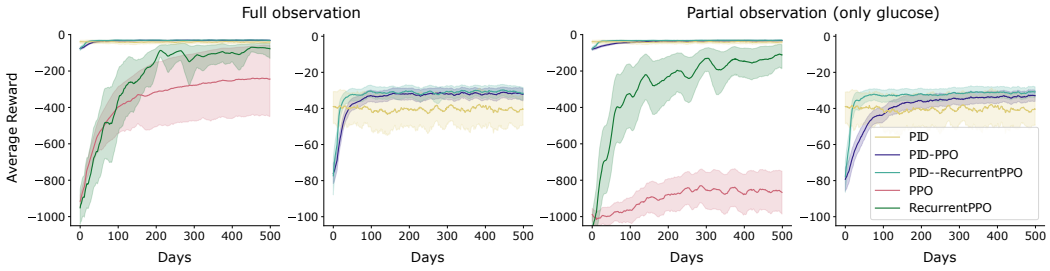


Figure 2: The online performance of the agents in the stochastic meal and exercise setting with the obese disturbance function, evaluated over 500 days. Following the format of the Figure 1, the first two plots show results under the fully observable setting, while the last two illustrate agent performance under partial observability.

As shown in Figure 2, the general trend observed in the deterministic experiments persists: standard PPO struggles in the partially observable setting, and RecurrentPPO consistently outperforms it in both settings. PID-based agents surpass standard PPO agents and the baseline PID in both fully and partially observable settings, demonstrating better stability and efficiency.

Interestingly, the added stochasticity appears to benefit the RecurrentPPO agent, resulting in improved performance compared to the deterministic case. However, in the partially observable setting, this difference is minimal. These findings suggest that the additional stochasticity does not significantly impair agent performance; on the contrary, it may promote robustness and generalization, potentially aiding learning under more realistic conditions.

To introduce additional stochasticity to the exercise routine, we allow the patient to skip workouts randomly. To accommodate this, we integrate both Bergman models: on exercise days, we use the exercise-augmented model, while on non-exercise days, we revert to the standard three-equation Bergman model. Since the exercise-augmented model is a strict superset of the standard Bergman model when the patient skips a workout, we set the intermediate variables to zero on the next day, retaining only the shared state variables: G_t, X_t, and I_t.

For this experiment, we focus on the partially observable setting, where the agent receives only glucose readings. In addition to combining the standard and exercise-augmented environments, we introduce up to 10% observational noise, reflecting real-world conditions where glucose meters typically exhibit measurement errors of this magnitude.

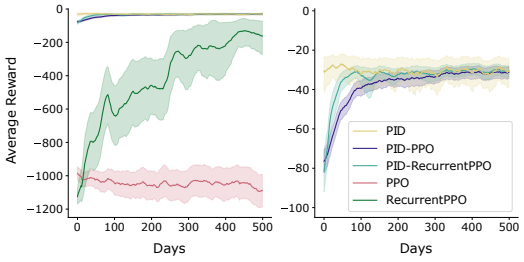


Figure 3: The online performance of the agents, evaluated over 500 days, assessed in the integrated stochastic environment with the obese disturbance function and noisy observations in the partially observable setting.

As shown in Figure 3, the performance closely mirrors the results in the previous experiments, particularly for the standard PPO agents. However, the PPO agent exhibits a clear downward trend, indicating its inability to adapt effectively to noisy glucose readings. In contrast, RecurrentPPO, though impacted by the added noise, maintains stable performance, demonstrating robustness to partial observability and sensor inaccuracies.

However, PID-based agents are noticeably affected by the added stochasticity, with their performance closely mirroring the baseline. Despite this, they exhibit lower variance and better stability than the baseline PID, suggesting that while they struggle to achieve a clear advantage in the noisy setting, they remain reliable. An-

other notable effect is the reduction in sample efficiency: unlike in previous experiments, the PID-based agents no longer reach strong performance within the first 100 episodes. Nevertheless, they outperform all non-PID agents, further reinforcing the stabilizing benefits of PID-based control in glucose regulation.

In this section, we observed that with added stochasticity, standard PPO agents struggle, especially in partially observable settings, while RecurrentPPO consistently outperforms them. PID-based agents surpass baseline PID, demonstrating more stable and efficient performance when stochasticity is introduced in meal and exercise schedules. Even with noisy glucose readings, these agents maintain performance comparable to the baseline while exhibiting lower variance. Overall, despite increased uncertainty, **PID-based agents demonstrate greater robustness and stability compared to their non-PID counterparts in the stochastic Blood-Glucose environment**, highlighting their potential reliability in real-world scenarios.

4.3 Online transfer between patients

The ability to transfer learned knowledge is essential for real-world deployment, allowing agents to adapt to new situations by leveraging prior experience without performance degradation. To assess this capability, we evaluate agents across 12 patients with varying body compositions (obese and normal-weighted). Patient order is randomized, and agents interact with each patient sequentially for 100 days. The final patient is the one used in all of the previous experiments. We consider the same two partially observable settings used in the last section. Detailed information about the 12 new patients is provided in Appendix E, while Appendix H contains the data for the patient used in all previous experiments.

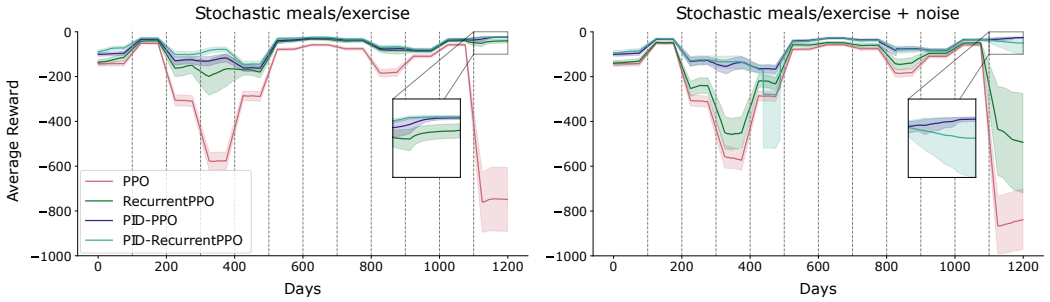


Figure 4: The online performance of the agents was evaluated across 12 patients in the partially observable stochastic environments introduced in the previous section. Each agent was given 100 days to administer insulin per patient, with transitions between patients indicated by dotted vertical lines. Additionally, both plots include a zoomed-in view of the patient used in all previous experiments.

In Figure 4 (left), we observe the same trend in the online transfer setting as in the previous stochastic environment experiments: standard PPO continues to struggle, while RecurrentPPO effectively adapts to new patients, achieving performance comparable to PID-based agents.

In Figure 4 (right), when we introduce noise to the agent's observations, we see an interesting shift in performance. RecurrentPPO, which previously matched the performance of PID-tuned agents, now behaves more like standard PPO rather than its PID-based counterparts. While it still exhibits some learning, as evidenced by the performance gap between the two non-PID agents, it ultimately failed in the last 100 days. It is a drastic shift from its performance in the previous experiment (Figure 3). These results suggest that the recurrent component, while beneficial for handling partial observability by maintaining a memory of past observations, becomes vulnerable in the transfer setting under noisy conditions, likely due to the accumulation of errors from different patients in the hidden state over time.

A key observation is the learning pattern of PPO agents in both the stochastic and stochastic + noisy settings: their performance patterns are almost identical. Consistent with earlier experiments, stan-

dard PPO continues to struggle in stochastic environments—this is particularly evident in the final 100 days of the plots, which correspond to the patient evaluated in prior sections. This result indicates that, despite appearing to perform well at times, the PPO agents fail to learn. The fluctuations in performance primarily reflect variability in the reward signal, driven by minor setpoint deviations in normal-weight patients, rather than genuine improvements in the agent's policy.

As for the PID-based agents, they transfer more effectively between patients, gradually improving their performance, as seen in the last 100 days. However, with added noise, PID-RecurrentPPO struggles more compared to its performance in the stochastic meal and exercise setting, where it previously outperformed the other agents. This further suggests that noise acts adversarially in the transfer setting.

In summary, we examined how agents transfer across patients in the partially observable stochastic environment with and without noisy observations. As expected, standard PPO struggled, while RecurrentPPO gradually improved, though its performance degraded significantly with noisy inputs. **PID-based agents transferred most effectively, maintaining stability and consistently outperforming standard PPO agents**. However, added noise also impacted PID-RecurrentPPO, suggesting that accumulated noise in the memory reduces generalization and learning across patients.

5 Conclusion

In this paper, we investigated the performance of reinforcement learning (RL) algorithms in the context of blood glucose regulation, where the objective is to maintain stable glucose levels through insulin administration. Despite the environment's relatively simple structure, it presented substantial challenges for PPO agents, particularly under stochastic glucose spikes, meal intake, exercise patterns, and noisy observations.

Building on prior work, we modified PPO agents to output PID controller gains instead of directly selecting insulin dosages. Our experiments in both deterministic and stochastic environments revealed that these PID-based agents consistently outperformed standard PPO agents, offering greater stability and reduced performance variability. In some cases, the learned PID controllers even surpassed the performance of a fixed baseline PID. This advantage also extended to online transfer across patients, highlighting the robustness of PID-based systems when combined with on-policy methods, like PPO. These findings underscore the potential of integrating classical control strategies with reinforcement learning. Furthermore, the consistent shortcomings of standard PPO agents emphasize the value of this environment as a benchmark for investigating partial observability and stochasticity in process control systems.

References

Misbaudeen Aderemi Adesanya, Hammed Obasekore, Anis Rabiu, Wook Ho Na, Qazeem Opeyemi Ogunlowo, Timothy Denen Akpenpuun, Min Hwi Kim, Hyeon Tae Kim, Bo Yeong Kang, and Hyun Woo Lee. Deep reinforcement learning for pid parameter tuning in greenhouse hvac system energy optimization: A trnsys-python cosimulation approach. *Expert Systems with Applications*, 2024.

Richard N Bergman. Toward Physiological Understanding of Glucose Tolerance: Minimal-Model Approach. *Diabetes*, 1989.

Richard N Bergman, Y Ziya Ider, Charles R Bowden, and Claudio Cobelli. Quantitative estimation of insulin sensitivity. *American Journal of Physiology-Endocrinology And Metabolism*, 1979.

Maximilian Bloor, Akhil Ahmed, Niki Kotecha, Mehmet Mercangöz, Calvin Tsay, and Ehecactl Antonio Del Rio Chanona. Control-informed reinforcement learning for chemical processes. *arXiv preprint arXiv:2408.13566*, 2024.

Kehua Chen, Hongcheng Wang, Borja Valverde-Perezc, Siyuan Zhai, Luca Vezzaro, and Aijie Wang. Optimal control towards sustainable wastewater treatment plants based on multi-agent reinforcement learning. *Chemosphere*, 2020.

Ruan de Rezende Faria, Bruno Didier Olivier Capron, Argimiro Resende Secchi, and Maurício B. de Souza. Where reinforcement learning meets process control: Review and guidelines. *Processes*, 2022.

Oguzhan Dogru, Kirubakaran Velswamy, Fadi Ibrahim, Yuqi Wu, Arun Senthil Sundaramoorthy, Biao Huang, Shu Xu, Mark Nixon, and Noel Bell. Reinforcement learning approach to autonomous PID tuning. *Computers & Chemical Engineering*, 2022.

Harry Emerson, Matthew Guy, and Ryan McConville. Offline reinforcement learning for safer blood glucose control in people with type 1 diabetes. *Journal of Biomedical Informatics*, 2023.

Harry Emerson, Sam Gordon James, Matthew Guy, and Ryan McConville. Flexible blood glucose control: Offline reinforcement learning from human feedback. *arXiv preprint arXiv:2501.15972*, 2025.

Andrea Facchinetti. Continuous glucose monitoring sensors: past, present and future algorithmic challenges. *Sensors*, 2016.

Ian Fox and Jenna Wiens. Reinforcement Learning for Blood Glucose Control: Challenges and Opportunities. In *Reinforcement Learning for Real Life (RL4RealLife) Workshop in International Conference on Machine Learning*, 2019.

Ian Fox, Joyce Lee, Rodica Pop-Busui, and Jenna Wiens. Deep reinforcement learning for closed-loop blood glucose control. In *Machine Learning for Healthcare Conference*, 2020.

Kevin Hansen, James Bathon, and Ramon Villafana. Diabetes: Controlling blood glucose concentrations, 2019.

Matthew Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable mdps. *AAAI Fall Symposium - Technical Report*, 2015.

Chirath Hettiarachchi, Nicolo Malagutti, Christopher J. Nolan, Hanna Suominen, and Elena 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.

Ali Khaqan, Ali Nauman, Sana Shuja, Tahir Khurshaid, and Ki Chai Kim. An intelligent model-based effective approach for glycemic control in type-1 diabetes. *Sensors*, 2022.

Ayub I. Lakhani, Myisha A. Chowdhury, and Qiugang Lu. Stability-Preserving Automatic Tuning of PID Control with Reinforcement Learning. *Complex Engineering Systems*, 2022.

Dongkwon Lee, Moonyong Lee, Suwhan Sung, and Inbeum Lee. Robust pid tuning for smith predictor in the presence of model uncertainty. *Journal of Process Control*, 1999.

Xiujun Li, Lihong Li, Jianfeng Gao, Xiaodong He, Jianshu Chen, Li Deng, and Ji He. Recurrent reinforcement learning: A hybrid approach. *arXiv preprint arXiv:1509.03044*, 2015.

Min Hyuk Lim, Woo Hyung Lee, Byoungjun Jeon, and Sungwan Kim. A Blood Glucose Control Framework Based on Reinforcement Learning With Safety and Interpretability: In Silico Validation. *IEEE Access*, 2021.

Juan Martinez-Piazuelo, Daniel E. Ochoa, Nicanor Quijano, and Luis Felipe Giraldo. A multi-critic reinforcement learning method: An application to multi-tank water systems. *IEEE Access*, 2020.

Daniel G. McClement, Nathan P. Lawrence, Johan U. Backström, Philip D. Loewen, Michael G. Forbes, and R. Bhushan Gopaluni. Meta-reinforcement learning for the tuning of PI controllers: An offline approach. *Journal of Process Control*, 2022.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, 2016.

Max Mowbray, Panagiotis Petsagkourakis, Ehecatl Antonio del Rio-Chanona, Robin Smith, and Dongda Zhang. Safe chance constrained reinforcement learning for batch process control. *Computers & Chemical Engineering*, 2021.

Tianwei Ni, Benjamin Eysenbach, and Ruslan Salakhutdinov. Recurrent model-free rl can be a strong baseline for many pomdps. *Proceedings of Machine Learning Research*, 2021.

Yunxiao Qin, Weiguo Zhang, Jingping Shi, and Jinglong Liu. Improve pid controller through reinforcement learning. In *IEEE CSAA Guidance, Navigation and Control Conference*, 2018.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 2021.

Anirban Roy and Robert S Parker. Dynamic modeling of exercise effects on plasma glucose and insulin levels. *Journal of diabetes science and technology*, 2007.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arxiv:1707.06347*, 2017.

Hitesh Shah and Madan Gopal. Model-Free Predictive Control of Nonlinear Processes Based on Reinforcement Learning. *IFAC-PapersOnLine*, 2016.

T. Shuprajhaa, Shiva Kanth Sujit, and K. Srinivasan. Reinforcement learning based adaptive pid controller design for control of linear/nonlinear unstable processes. *Applied Soft Computing*, 2022a.

T. Shuprajhaa, Shiva Kanth Sujit, and K. Srinivasan. Reinforcement learning based adaptive pid controller design for control of linear/nonlinear unstable processes. *Applied Soft Computing*, 2022b.

Steven Spielberg, R. Bhushan Gopaluni, and Philip D. Loewen. Deep reinforcement learning approaches for process control. *International Symposium on Advanced Control of Industrial Processes*, 2017.

Miguel Tejedor, Ashenafi Zebene Woldaregay, and Fred Godtliebsen. Reinforcement learning application in diabetes blood glucose control: A systematic review. *Artificial Intelligence in Medicine*, 2020.

Haeun Yoo, Ha Eun Byun, Dongho Han, and Jay H. Lee. Reinforcement learning for batch process control: Review and perspectives. *Annual Reviews in Control*, 2021.

Yan Feng Zhao, Jun Kit Chaw, Mei Choo Ang, Yiqi Tew, Xiao Yang Shi, Lin Liu, and Xiang Cheng. A safe-enhanced fully closed-loop artificial pancreas controller based on deep reinforcement learning. *PLOS ONE*, 2025.

Zhuangdi Zhu, Kaixiang Lin, Anil K. Jain, and Jiayu Zhou. Transfer learning in deep reinforcement learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

A Details on experimental design

We implemented PPO and its recurrent variant using the Stable-Baselines3 package (Raffin et al., 2021). Since our focus is on online learning, all PPO agents were trained in a single-environment setting. For the standard PPO, we used the default hyperparameters. However, for the recurrent PPO, initial experiments showed limited learning with default settings, so we adopted hyperparameters from the Pendulum environment, which shares a similar reward structure, action space, and observation space with our task. We kept the PPO hyperparameters unchanged, as they are also the defaults for the Pendulum environment. For the PID controller, we used the default hyperparameters provided in Hansen et al. (2019).

PID	Values
K_p	2
K_i	0.05
K_d	50

PPO	Values
Learning rate	3e-4
Discount (γ)	0.99
Hidden layers	2
Hidden units per layer	64
Batch size	64
GAE λ	0.95
PPO clip ϵ	0.2
Value loss coefficient	0.5
Entropy coefficient	0.0
Gradient clip	0.5
Update epochs	10
Trajectory length	2048
Normalize observations	True
Normalize rewards	True

RecurrentPPO	Values
Learning rate	1e-3
Discount (γ)	0.9
Hidden layers	2
Hidden units per layer	64
Recurrent network type	LSTM
Recurrent network size	64
Batch size	128
GAE λ	0.95
PPO clip ϵ	0.2
Value loss coefficient	0.5
Entropy coefficient	0.0
Gradient clip	0.5
Update epochs	10
Trajectory length	1024
Normalize observations	True
Normalize rewards	True

Table 1: Hyperparameter values used for PID, PPO and RecurrentPPO agents.

B Details on the disturbance functions

Normal-weighted disturbance simulates the effect of meals on blood glucose levels by incorporating an exponential decay, capturing the natural reduction of glucose over time. Each meal introduces a sharp increase in blood glucose levels, after which the function decays, gradually reducing the effect over time as the glucose metabolizes. This behavior is represented mathematically as:

$$D_t = F_g \cdot e^{-0.05(t-t_m)}$$

where t_m is the time of the meal, and F_g represents the glucose contribution of the meal. The decay factor 0.05 ensures that the glucose disturbance gradually diminishes, mimicking the body's natural glucose regulation process.

Obese disturbance follows the same structure as the normal disturbance function but introduces a scaling factor of 1.2x to amplify the glucose response. This adjustment reflects physiological differences in glucose metabolism for obese individuals, where meal-induced glucose spikes tend to be more pronounced or prolonged. The disturbance function is defined as:

$$D_t = 1.2 \cdot F_g \cdot e^{-0.05(t-t_m)}$$

where F_g represents the meal's glucose contribution, and t_m is the meal time. The increased scaling factor accounts for the reduced insulin sensitivity commonly observed in obesity, leading to higher and more persistent postprandial glucose levels.

C Exercise-augmented Bergman Minimal Model

The **exercise-augmented Bergman Minimal Model** extends the original three-equation Bergman model by introducing additional variables to capture the physiological dynamics of exercise. Key parameters include G_t (glucose concentration), the glucose availability for energy, and I_t (insulin concentration), which regulates glucose uptake by tissues. The model tracks glucose dynamics through parameters like $G_{\text{prod},t}$ (glucose production) and $G_{\text{up},t}$ (glucose uptake), as well as the depletion and recovery of glycogen stores ($G_{\text{gly},t}$), which provide backup energy during exercise. Insulin, influenced by factors such as U_t (insulin injection), which is the control action, and n (insulin clearance), plays a crucial role in glucose uptake into cells, and P_1 governs the rate of glucose consumption. The intensity of exercise, A_t, influences metabolic processes such as X_t (exercise metabolism), with intense expenditure leading to increased glucose utilization and energy usage, tracked by I_e (energy expenditure). Additionally, the model incorporates $PVO_{2,t}^{\max}$ representing maximal oxygen consumption that influences both glucose production and uptake. The equations for the exercise-augmented Bergman Minimal Model are presented below, with blue highlighting the equations of the original model.

$$
\begin{aligned}
G_{t+1} &= G_t + \Delta t \left(-P_1 G_t - X_t(G_t + G_b) + D_t + \frac{W}{V_G}(G_{\text{prod},t} - G_{\text{gly},t} - G_{\text{up},t}) \right), \\
I_{t+1} &= I_t + \Delta t \left(-n(I_t + I_b) + \frac{U_t}{V_1} - I_{e,t} \right), \\
X_{t+1} &= X_t + \Delta t \left(-P_2 X_t + P_3 I_t \right), \\
\\
G_{\text{prod},t+1} &= G_{\text{prod},t} + \Delta t \left(a_1 PVO_{2,t}^{\max} - a_2 G_{\text{prod},t} \right), \\
G_{\text{up},t+1} &= G_{\text{up},t} + \Delta t \left(a_3 PVO_{2,t}^{\max} - a_4 G_{\text{up},t} \right), \\
I_{e,t+1} &= I_{e,t} + \Delta t \left(a_5 PVO_{2,t}^{\max} - a_6 I_{e,t} \right), \\
PVO_{2,t+1}^{\max} &= PVO_{2,t}^{\max} + \Delta t \left(0.8(u_{\text{ex},t} - PVO_{2,t}^{\max}) \right), \\
G_{\text{gly},t+1} &= G_{\text{gly},t} + \Delta t \times \begin{cases} 0, & A_t < A_{\text{TH}}, \\ k, & A_t \geq A_{\text{TH}}, \\ -\frac{G_{\text{gly},t}}{T_1}, & u_{\text{ex},t} = 0. \end{cases} \\
A_{t+1} &= A_t + \Delta t \times \begin{cases} u_{\text{ex},t}, & u_{\text{ex},t} > 0, \\ -\frac{A_t}{0.001}, & u_{\text{ex},t} = 0. \end{cases}
\end{aligned}
\tag{2}
$$

D Ablation study on state representation

In the main paper, we define the state s_t of PID-based PPO agents to include the components of the PID controller, as we explicitly model the PID as part of the environment and need to account for its existence. In this section, we analyze how each component influences agent performance and assess whether all parts of the state are necessary for effective decision-making.

We analyze agent's performance in both deterministic and stochastic environments under different state representations to assess the necessity of each component. The **Full** state includes all model variables along with PID components, while **3-variable** consists of only G_t, I_t, X_t, the shared vari-

ables across exercise and non-exercise models. The **Glucose+PID** state retains only glucose and PID components, whereas **Glucose** provides only glucose measurements. Finally, the **PID** state contains only the PID components, isolating their contribution to decision-making.

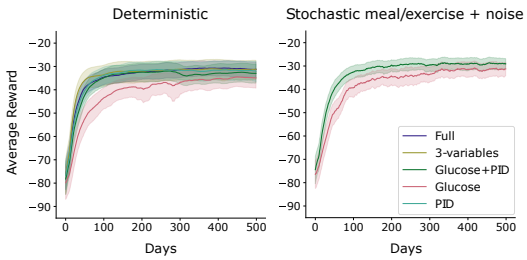


Figure 5: The performance of the PID-PPO agent in the deterministic (left) and stochastic meal/exercise + noise (right) environments with five state representations, evaluated over 500 days.

In Figure 5, we compare the performance of the PID-PPO agent across five state representations in both deterministic (left) and stochastic meal/exercise + noise (right) environments. In the deterministic setting, agents perform similarly when provided with either the full environment state or the PID components but struggle with glucose-only inputs. In particular, including only the three PID components give the agent enough information to effectively tune the gain values of the controller, even without having direct access to the glucose value. A similar pattern emerges in the stochastic setting, where augmenting glucose readings with PID components improves performance over glucose alone, suggesting that PID-based agents benefit from incorporating PID components into the state, leading to better control.

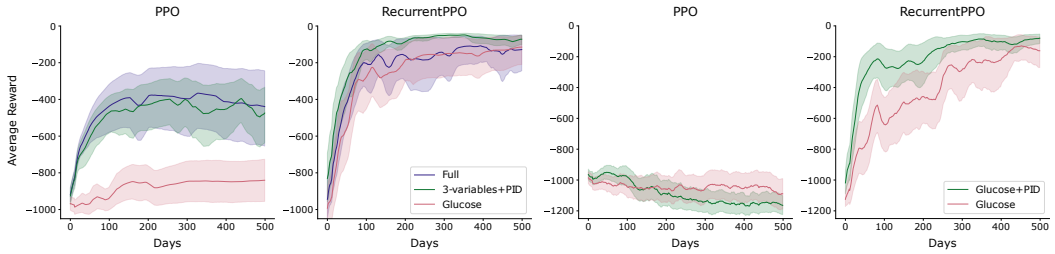


Figure 6: The performance of the PPO and RecurrentPPO agents in the deterministic (first,second) and stochastic meal/exercise + noise (third,fourth) environments with different state representations, evaluated over 500 days.

We test whether augmenting standard PPO agents with the error terms improves performance. As shown in Figure 6, adding PID components has little to no effect on standard PPO in either setting. In fact, in the stochastic environment, the additional error terms seem to negatively impact PPO's performance. However, the recurrent variant of PPO benefits significantly from this augmentation in both settings, suggesting that the error terms help to construct a more informative hidden state.

We now assess whether the additional error terms improve RecurrentPPO's performance to compete with the PID-based methods. As shown in Figure 7, while RecurrentPPO almost reaches the baseline PID with the addition of the error terms, it still falls short of PID-PPO agent in both performance and stability, exhibiting higher variability.

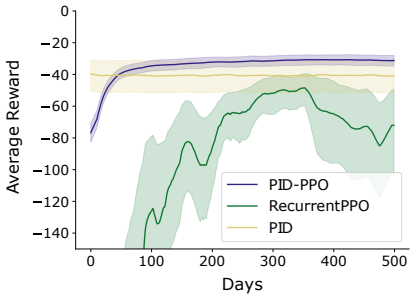


Figure 7: The performance of the PID-PPO, RecurrentPPO agents and PID baseline in the deterministic environment, evaluated over 500 days.

In summary, we demonstrate that PID-based agents benefit significantly from incorporating PID components into their state space, particularly in partially observable and stochastic settings. While adding error terms had little effect on standard PPO,

RecurrentPPO improved with the additional information, outperforming its counterpart that relied solely on environment variables. However, despite these gains, RecurrentPPO still falls short of PID-based agents in both overall performance and stability, exhibiting greater variability.

E Data for the online transfer learning

In most of our experiments – particularly in the deterministic and stochastic settings discussed in Section 4.1 and Section 4.2 – we relied on data from a single patient, originally provided in Roy & Parker (2007). To broaden the scope of our evaluation, we incorporated data for 12 additional patients from the original Bergman Minimal Model study (Bergman, 1989), as detailed below. However, the original model differs significantly from the one used in this work, so we applied additional formulas to derive the parameters required for our experimental setup.

#	G_b	I_b	G_0	I_0	n	P_1	P_2	P_3 (10^{-6})	IBW %	V_1	Sex
1	94	17	298	333	0.23	0.0296	0.0186	6.51	101	8.77	M
2	91	9	276	69	0.18	0.0192	0.0262	14.7	100	8.69	M
3	98	4	296	50	0.22	0.0136	0.0341	17.3	97	8.43	M
4	97	3	248	15	0.22	0.0151	0.0313	9.7	88	7.65	M
5	93	6	271	14	0.11	0.0217	0.0292	19.1	98	8.51	M
6	110	26	256	99	0.13	0.018	0.0108	2.29	142	9.95	F
7	99	21	217	225	0.14	0.0113	0.0034	0.97	148	10.37	F
8	92	20	224	185	0.11	0.0113	0.0240	4.89	130	9.11	F
9	102	81	258	337	0.13	0.0246	0.0069	0.55	138	9.67	F
10	109	37	267	248	0.13	0.01	0.0166	2.02	172	12.05	F
11	104	68	242	20	0.13	0.0071	0.0125	1.58	206	14.44	F
12	132	16	254	16	0.13	0.0093	0.02	7.78	153	10.72	F

Table 2: Patient data with various parameters, including body metrics, physical attributes, and health indicators such as obesity status and gender.

As the paper only provided Ideal Body Weight (IBW) percentages along with information about the sex of the patients, we estimated the weights and V_1 (the volume of distribution of insulin) from the patients' weights. To calculate the body weight of the patients, we assumed their heights to be the average height in the US, where the data was collected: 175.26 cm (5'9 inches) for males and 162.56 cm (5'6 inches) for females. Using the Hamwi formula, we first calculated the Ideal Body Weight (IBW) for each patient.

$$\text{IBW} = \begin{cases} 48.0 + 2.7 \times (\text{Height in inches} - 60), & \text{for men} \\ 45.5 + 2.2 \times (\text{Height in inches} - 60), & \text{for women} \end{cases}$$

Then, using the following formula, we were able to approximate the weight of each patient based on their IBW and IBW percentage:

$$\text{Body Weight (kg)} = \left(\frac{\text{IBW} \times \text{IBW \%}}{100} \right)$$

This allowed us to estimate the actual weight of each patient based on the provided IBW percentage. With this data, we can now approximate the value of V1 (the volume of distribution of insulin) for

the Bergman Minimal Model, using the relationship between body weight and insulin distribution in the model.

$$V_1 \approx 0.12 * \text{Body Weight (kg)}$$

This equation is a rough approximation of the values, based on clinical and physiological data for the human body, particularly in the context of insulin dynamics or glucose metabolism (Bergman et al., 1979).

For the exercise-augmented Bergman Minimal Model, we used the confidence values introduced in Roy & Parker (2007) as a basis to randomly select values for the individuals. The values and ranges used for the exercise model variables are summarized below:

Parameter	Distribution
a_1	$\sim \mathrm{U}(0.00013, 0.0019)$
a_2	$\sim \mathrm{U}(0.0441, 0.0679)$
a_3	$\sim \mathrm{U}(0.0015, 0.0024)$
a_4	$\sim \mathrm{U}(0.0355, 0.0617)$
a_5	$\sim \mathrm{U}(0.001, 0.0015)$
a_6	$\sim \mathrm{U}(0.0588, 0.0912)$
k	$\sim \mathrm{U}(0.0085, 0.0131)$
T_1	$\sim \mathrm{U}(1.86, 10.14)$

For the online transfer section, we pre-shuffle all patients and use this shuffled order for sequential transfer. The final sequence is as follows: 12 (Obese), 1 (Not obese), 13 (Obese), 18 (Obese), 16 (Obese), 2 (Not obese), 8 (Not obese), 6 (Not obese), 14 (Obese), 15 (Obese), and 7 (Not obese), where each number corresponds to the specific patient in the Appendix E.

F Design choices for the environment

In this paper, we consider two environments: PIDBG, where the agent tunes the parameters for the PID controller, and BG, where the agent directly selects the insulin injection action. The key distinction between these environments is that, from the agent's perspective, the PID controller in PIDBG is part of the environment, fundamentally altering the dynamics the agent interacts. Additionally, we incorporate the exercise-augmented Bergman Minimal Model, as well as combined versions of BG and exercise-augmented BG: BGExercise, PIDBGExercise, BGCombined, and PIDBGCombined. The different environments, along with their action and observation spaces, are summarized in the following table:

Environment	Observation space	Action space
BG	$\mathbb{R}^3$	$\{a \in \mathbb{R}^1 \mid 0 \leq a \leq 50\}$
BGExercise	$\mathbb{R}^9$	
BGCombined	$\mathbb{R}^9$	
PIDBG	$\mathbb{R}^6$	$\{a \in \mathbb{R}^3 \mid 0 \leq a_1 \leq 100,\ 0 \leq a_2, a_3 \leq 10\}$
PIDBGExercise	$\mathbb{R}^{12}$	
PIDBGCombined	$\mathbb{R}^{12}$	

Table 3: Overview of observation and action spaces for different environments, with dimensions and value ranges. The action spaces for similar environments are grouped for clarity.

G Stochasticity in the environment

To introduce stochasticity in meal and exercise data, random perturbations are applied to meal effects, making the environment more dynamic and realistic. These perturbations influence both the magnitude of the glucose spike and the rate of decay, capturing individual differences in glucose metabolism. In stochastic cases, these variations are applied at the start of each episode or day, causing meal times and glucose intake levels to fluctuate daily, better reflecting real-world eating patterns. We use random number generators to vary glucose intake levels (F_g) and meal timing, with all values perturbed by a uniform distribution. Additionally, a snack is introduced with a 50% probability, further increasing unpredictability. For exercise, timing is also randomized uniformly, ensuring it does not overlap with meal times. In some experiments, we randomly decide whether the patient will exercise on a given day. This stochastic approach ensures a more realistic simulation of irregular human eating and activity patterns.

Meal	**Glucose (Fg)**	**Time (minutes)**	**Duration (minutes)**
Breakfast	$6.0 + \mathcal{U}(-2, 10)$	$60 + \mathcal{U}(-100, 200)$	—
Lunch	$9.0 + \mathcal{U}(-2, 10)$	$360 + \mathcal{U}(-100, 200)$	—
Dinner	$12.0 + \mathcal{U}(-2, 10)$	$660 + \mathcal{U}(-100, 200)$	—
Snack	$3.0 + \mathcal{U}(-2, 10)$	$900 + \mathcal{U}(-50, 150)$	—
Exercise	—	random meal $+ \mathcal{U}(-30, 30)$	$30 + \mathcal{U}(0, 30)$

Table 4: A detailed summary of stochastic variations in meal and exercise data, where $\mathcal{U}$ corresponds to the uniform distribution.

H Additional experiment results

Deterministic environment

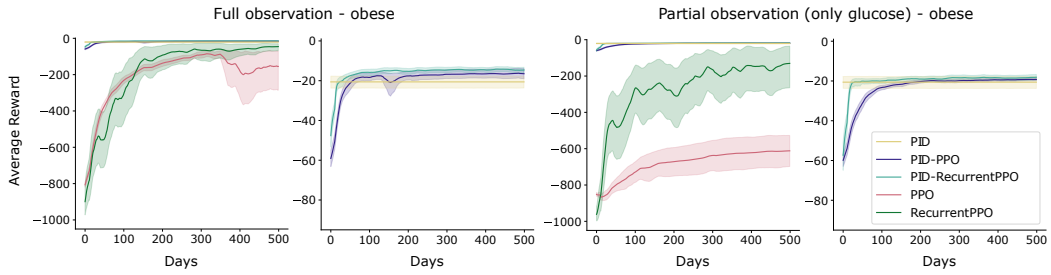


Figure 8: The online performance, the average return, of the four agents and the tuned PID baseline in the **original BGEnv** with an obese disturbance function, evaluated over 500 days.

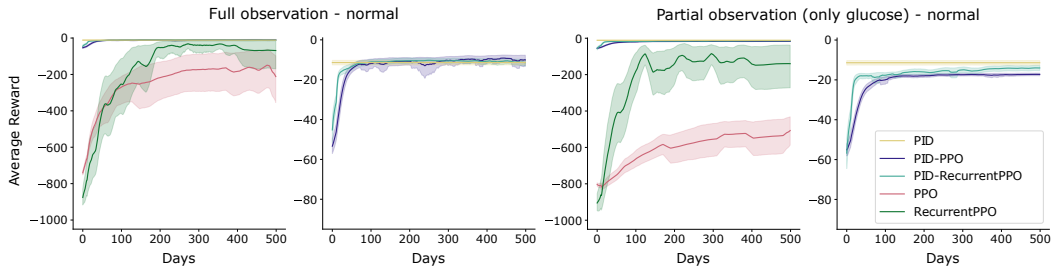


Figure 9: The online performance, measured as the average return, of the four agents and the tuned PID baseline in the **original BGEnv** with an normal disturbance function, is evaluated over 500 days.

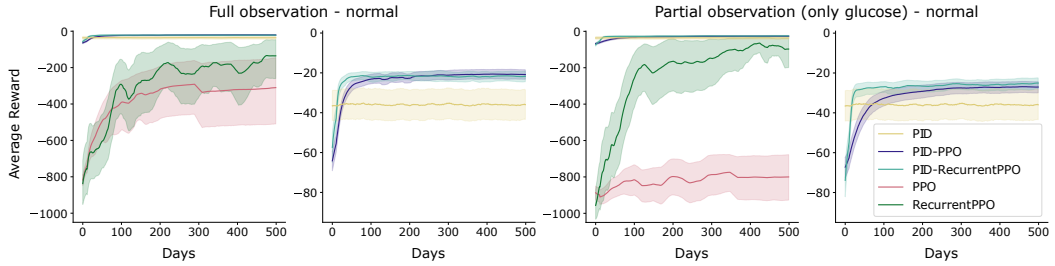


Figure 10: The online performance, measured as the average return, of the four agents and the tuned PID baseline in the **exercise-augmented BGEnv** with an normal disturbance function, is evaluated over 500 days.

Stochastic environment

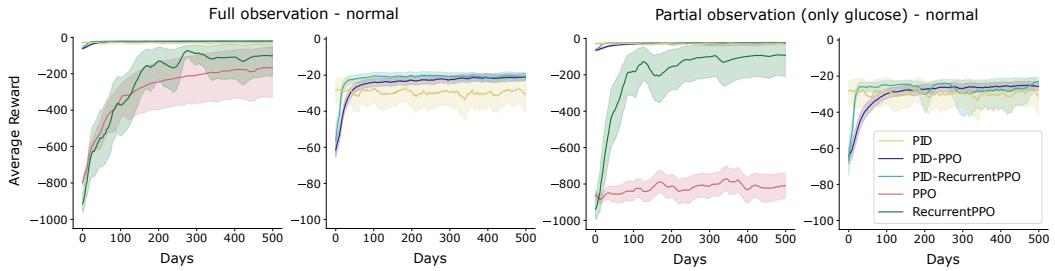


Figure 11: The online performance of the agents in the randomized meal intake and exercise setting with the normal disturbance function, evaluated over 500 days.

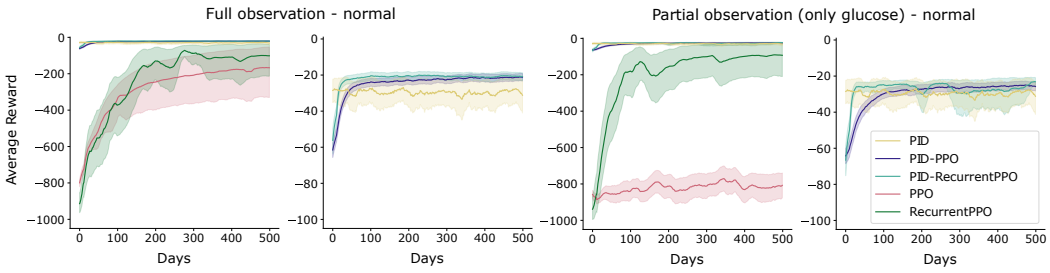


Figure 12: The online performance of the agents in the randomized meal intake and exercise setting with the normal disturbance function, evaluated over 500 days.

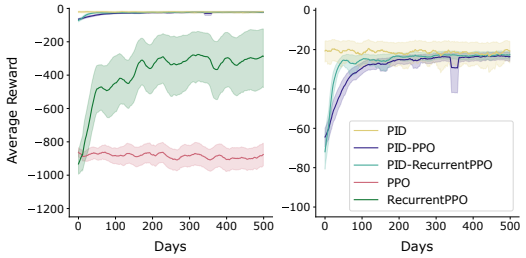


Figure 13: The online performance of the four agents, evaluated over 500 days, assessed in the integrated stochastic + noise environment with the normal disturbance function.

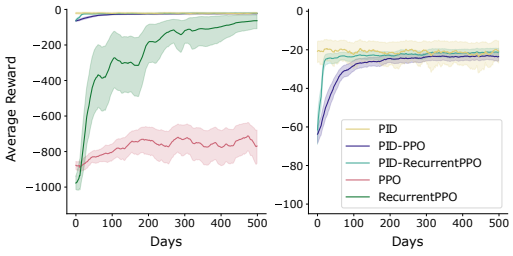


Figure 14: The online performance of the four agents with the normal disturbance function in the integrated stochastic environment, evaluated over 500 days.

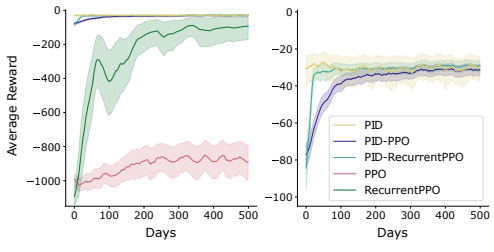


Figure 15: The online performance of the four agents with the obese disturbance function in the integrated stochastic environment, evaluated over 500 days.

Variable	Units	Description	Values
G_t	mg/dL	Plasma glucose concentration above basal value	—
I_t	mU/L	Plasma insulin concentration above basal value	—
X_t	min^{-1}	Proportional to plasma insulin concentration in the remote compartment	—
D_t	mg/(dL min)	Meal glucose disturbance	—
U_t	mU/min	Exogenous insulin infusion rate	—
G_b	mg/dL	Basal glucose concentration	81
I_b	mU/L	Basal insulin concentration	15
V_1	L	Insulin distribution volume	12
n	min^{-1}	Fractional disappearance rate of insulin	5/54
F_G	mg/min	Rate of exogenously infused glucose	—
V_G	dL	Glucose distribution space	117 - 136
P_1	min^{-1}	Glucose removal rate independent of insulin	0 or 0.028
P_2	min^{-1}	Insulin removal rate from the remote compartment	0.025
P_3	L/(mU min)	Insulin appearance rate in the remote compartment	5.3* 10^{-6}
W	kg	Patient's weight	62 - 81
$u_{ex,t}$	min	Exercise starting time	—
T_1	min	Time it takes for glycogen levels to return to basal levels	1.86 - 10.14
k	mg kg^{-1} min^{-2}	Rate of glycogen depletion when glycogen stores become close to depleted	0.0085 - 0.0131
a_1	mg kg^{-1} min^{-2}	Percentage of the maximum oxygen consumption rate	0.0013 - 0.0019
a_2	min^{-1}	Glucose release rate from the liver	0.0441 - 0.0679
a_3	mg kg^{-1} min^{-2}	The glucose absorption rate caused by exercise	0.0015 - 0.0024
a_4	min^{-1}	The removal rate of absorbed glucose	0.0355 - 0.0617
a_5	mU L^{-1} min^{-2}	Influences the insulin production rate due to exercise	0.0010 - 0.0015
a_6	min^{-1}	Insulin removal rate due to exercise	0.0588 - 0.0912

Table 5: Description of variables and parameters in the original and exercise-augmented Bergman Minimal Models used in deteremnistic and stochastic experiments.