Bootstrapping Personalized Insulin Therapy via Model-Based Reinforcement Learning: An In Silico Study

Sumana Basu[1,2], **Flemming Kondrup**[1,2], **Adriana Romero-Soriano**[1,2], **Doina Precup**[1,2]

[1] McGill University
[2] Mila

sumana.basu@mail.mcgill.ca, flemming.kondrup@mail.mcgill.ca, adriana.romerosoriano@mcgill.ca, dprecup@cs.mcgill.ca

Abstract

Personalized insulin therapy for individuals with Type 1 Diabetes via closed-loop artificial pancreas systems requires rapid adaptation of dosing strategies to each patient's unique insulin response. However, learning patient-specific policies from scratch demands extensive exploration, which is often impractical. In this work, we study a framework that integrates insulin-response-informed transfer learning with model-based reinforcement learning for insulin dosing. We first train an LSTM-based insulin responsiveness predictor on virtual patients, using their glucose, insulin, and meal history to forecast future glucose levels. Analysis of insulin responsiveness of in-silico patients uncovers natural insulin-response groups characterized by similar sensitivity and dynamics profiles. For a new patient, we identify a representative model from their response group and use it to generate synthetic trajectories. These trajectories are integrated into an enhanced H-step Deep Dyna-Q algorithm, enabling accelerated policy optimization through model-based planning. The dynamics model trained entirely in simulation achieves 91.31% accuracy in predicting blood glucose ranges on the Ohio Type 1 Diabetes dataset, indicating strong zero-shot generalization. Additionally, we find that bootstrapping a new patient with a physiologically-matched reference model accelerates convergence of effective dosing policies across in-silico cohorts of children, adolescents, and adults. These findings suggest that leveraging response-group-specific synthetic experience can expedite personalized insulin therapy, offering a promising pathway towards clinical validation.

1 Introduction

Type 1 diabetes (T1D) is a chronic condition affecting millions of patients globally, which is characterized by the body's inability to produce sufficient insulin, a crucial hormone for blood glucose regulation (Ogrotis, Koufakis, and Kotsa 2023). Managing T1D requires meticulous insulin administration to maintain blood glucose levels within an acceptable range, which is necessary to prevent both acute and chronic complications. These include hyperglycemia, which can cause long-term damage to organs such as the eyes, kidneys, nerves, heart, and blood vessels (American Diabetes Association 2009), as well as hypoglycemia, which can lead to severe outcomes like loss of consciousness, coma, and even death (Carroll, Burge, and Schade 2003). Type 1 diabetes patients traditionally manage blood glucose through continuous monitoring and manual insulin dosing, which can be burdensome and prone to error. Finding an optimal treatment is also time-consuming due to the highly individualized nature of the condition, with factors like diet, metabolism, exercise, sleep, and stress affecting insulin needs. Correct dosing is critical, particularly in conditions like T1D, where the therapeutic window is narrow (Maxfield and Zineh 2021).

Technological advancements such as Continuous Subcutaneous Insulin Infusion (CSII) and Continuous Glucose Monitoring (CGM) systems enhance treatment efficacy and reduce patient burden (Peters et al. 2016). These systems can be integrated into what is known as sensor-augmented pump (SAP) therapy, further refined into an "Artificial Pancreas" (AP) system that automates insulin delivery based on near-constant glucose monitoring (Bruttomesso et al. 2009; Bergenstal et al. 2011). This approach can significantly improve the accuracy of treatment and prevent potential hyperglycemic or hypoglycemic events, improving patient safety and quality of life. (Peters et al. 2016).

Recent progress in AI has enabled the development of model-free RL approaches for chronic-disease management, including T1D. Such systems are more dynamic than traditional approaches, as they can continuously evolve their strategies in response to the changing patterns of the disease in a specific patient. However, they require large datasets to learn an optimal policy. To address this, model-based reinforcement Learning (MBRL) (Moerland et al. 2022) offers a promising avenue, as it provides a learning approach that can require significantly less data from a patient, generating instead synthetic data from a learned model. MBRL methods can also limit risky real-world exploration through the use of samples from learned models. However, constructing an accurate dynamics model often demands extensive data, defeating the purpose of using MBRL.

In this work, we propose leveraging historical patient data to accelerate the development of personalized insulin therapies for new patients. We begin by identifying a prior patient whose physiological characteristics would be most relevant and use their data to initialize the glucose-dynamics model for a new patient. We validate the model quality on both simulated data and real data from the Ohio T1D database.

We show that initializing a new patient's insulin–response model with one trained on data from a clinically matched prior patient enables faster learning of effective insulin dosing policies. In the next stage, we enhance the H-step Dyna-DQN algorithm with model-generated synthetic rollouts, facilitating efficient optimization of individualized policies, as shown through evaluation on the FDA-approved Simglucose simulator.

2 Related Work

Continuous Glucose Monitoring (CGM) systems have enabled sophisticated algorithmic interventions by collecting subcutaneous blood glucose measurements 24 hours a day. The vast amount of CGM data has facilitated the significant advancement of research in Automated Insulin Delivery (AID) algorithms to enhance therapeutic outcomes through insulin pumps in Artificial Pancreas (AP) systems. The most common algorithm in the commercial Artificial Pancreas (AP) systems is the Proportional-Integral-Derivative (PID) control (Eg, Medtronic 670G) (Trevitt, Simpson, and Wood 2015; Garg et al. 2017), which is a linear controller to manage blood glucose by continuously adjusting insulin doses based on the deviation between the measured glucose level and a predefined target. Although they are suitable for hybrid controllers, where a human delivers the preprandial (pre-meal) insulin doses (bolus), and the PID controller only manages the inter-prandial doses (basal), they are not fit for a fully automated closed-loop insulin delivery system responsible for both types of doses (Bergenstal et al. 2016). Their relatively slow response time limits their ability to react fast enough to the rapid postprandial (post-meal) glucose spikes, increasing the risk of hyperglycemia. Moreover, without built-in safety constraints, PID controllers can administer excessive insulin, heightening the risk of hypoglycemia(Ruiz et al. 2012).

The most recent commercial Artificial Pancreas (AP) systems (Eg, FDA-approved Medtronic's Minimed 780G) have upgraded their Automated Insulin Delivery (AID) algorithms with Model Predictive Control (MPC). In contrast to the reactive PID controllers, Model Predictive Control (MPC) is a proactive method that uses mathematical glucose prediction models based on CGM history and adjusts insulin infusion rate to reduce the predicted glucose level against a pre-set target glucose level over the next 1.5 to 4 hours (DiabetesNet 2025a). However, the mathematical models are often fixed and do not adjust to patient responses and their evolving dynamics (See (DiabetesNet 2025b) for a list of algorithms used in commercial AIDs).

RL based AID algorithms have gained traction due to their ability to learn personalized dosing strategies through sequential interactions with the environment, enabling adaptation to a patient's changing physiological state. A systematic review (Tejedor, Woldaregay, and Godtliebsen 2020) of RL strategies for managing blood glucose reported 347 studies published from 1990 to 2019 on this subject. RL has enhanced glucose management by enabling faster responses to abrupt glycemic fluctuations, adaptation to lifestyle changes, and safer dosing recommendations (Daskalaki, Diem, and Mougiakakou 2016; Fox et al. 2020). Glucose forecasting in T1D has also been studied extensively (see Nemat et al. (2024) for a survey), using feed-forward networks (Ben Ali et al. 2018), RNNs (Fox et al. 2020), and models that integrate pharmacokinetic priors (Potosnak et al. 2025).

Although these advances promise personalized treatment, existing methods learn policies exclusively from the target patient's data. Collecting the large datasets required can take months, during which the patient remains on suboptimal therapy and may experience side effects. Despite growing interest in MBRL as a sample-efficient and safer alternative (Yamagata et al. 2020; Lim et al. 2021), current glycemic-forecasting models are still trained solely on data from the target individual. While personalization tailors therapy to each individual, the treatment policy does not need to be learned entirely from scratch for every new patient. Reusing data from previous patients can accelerate personalization, but identifying an appropriate source patient is non-trivial. We study a clinically motivated nearest-neighbor selection method that bootstraps each new patient's glucose-dynamics model from the most relevant prior patient, yielding sample-efficient personalization. We evaluate the studied model-based approach against PID and basal–bolus controllers as well as model-free RL baselines.

3 Background

Markov Decision Process (MDP). In sequential decision-making problems, an agent interacts with its environment and receives feedback in the form of a numerical reward. This interaction is typically modelled as a Markov Decision Process (MDP), $\mathcal{M} = (\mathcal{S}, \mathcal{A}, r, \mathcal{P}, \gamma)$, where $\mathcal{S}$ is a (finite) set of states, $\mathcal{A}$ is a (finite) set of actions, $\mathcal{P}$ is the state transition probability $\mathcal{P}(s_{t+1} = s' | s_t = s, a_t = a)$, $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the reward function and $\gamma \in [0, 1)$ is the discount factor. The goal of an RL agent is to find a probability distribution over actions, conditioned on states, called a policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ that maximizes the expected cumulative discounted return, $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1})\right]$. For simplicity we denote $r_t = r(s_t, a_t, s_{t+1})$.

Model-Based Reinforcement Learning. MBRL estimates an environment model to forecast forthcoming states and/or rewards, conditioned on the current state and action (Moerland et al. 2022). This capability enables agents to simulate and assess potential actions without acting directly in the environment, which enhances both the safety and interpretability of the resulting solution. In its simplest form, a model predicts the next state $\hat{s}_{t+1}$ (and optionally the reward $\hat{r}_{t+1}$) from the current state s_t and action a_t. Variants include using a context of k steps $(s_{t-k+1:t-1})$ as input or predicting several steps ahead $(s_{t+1:t+k})$. The model is usually approximated, for example, by using a neural network.

Dyna-Q. Dyna-Q (Sutton 1991) is an MBRL algorithm that accelerates policy learning by combining Q-learning (Watkins and Dayan 1992) and transitions generated using a learned environment model. The dynamics model $\mathcal{M}_\phi$ is learned using supervised learning on transitions collected from the environment, and predicts the next state $\hat{s}_{t+1}$ (and optionally the reward $\hat{r}_{t+1}$). These imagined transitions are added to the replay buffer $\mathcal{D}_{sim} \leftarrow$

$(\hat{s}_t, \hat{a}_t, \hat{r}_t, \hat{s}_{t+1})$, and are used to train the Q-network along with the real transitions by optimizing the DQN (Mnih et al. 2015) loss, as shown in Equation 1.

$$L_i(\theta) = \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{D}_{real}\cup\mathcal{D}_{sim}} \left[(r_t + \gamma \max_a Q(s_{t+1}, a; \theta^-) - Q(s_t, a_t; \theta)^2\right] \quad (1)$$

where, θ^- are the parameters of the target network, and γ is the discount factor.

H-step Reinforcement Learning. In H-step RL, the Q-function is not updated immediately after observing the next state and reward, but only after H steps, using the H-step return:

$$G_t^H = \sum_{k=0}^{H-1} \gamma^k r_{t+k} + \gamma^H \max_a Q(s_{t+H}, a) \quad (2)$$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[G_t^H - Q(s_t, a_t)] \quad (3)$$

In case of H-step Dyna-Q, the model $\mathcal{M}_\phi$ rolls out H-step trajectories $\hat{s}_{t+k}, \hat{r}_{t+k} = \mathcal{M}_\phi(\hat{s}_{t+k-1}, \hat{a}_{t+k-1})$, where $k = 1, 2, ..., H$ and $\hat{a}_{t+k-1} \sim \pi_\theta(\hat{s}_{t+k-1})$, which are used to estimate the Q-functions.

4 Method

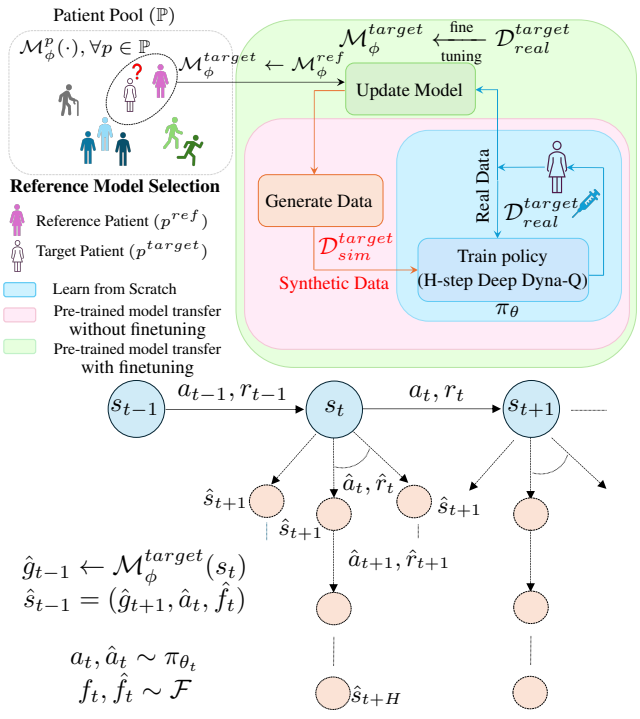


Figure 1: Overview. *Top*: A new patient is matched to a physiologically similar reference patient, whose dynamics model is used to initialize the target patient's model. Treatment policies are trained using both real and model-generated data. *Bottom:* H-step counterfactual rollouts generated at each step along the real trajectory.

4.1 Problem Formulation

In this section, we present our formulation for precision insulin dosing as an RL problem:

States. We define state at time t as $s_t = (g_{c_t}, i_{c_t}, f_{c_t})$, where g_{c_t}, i_{c_t} and f_{c_t} denote sequences of recent blood glucose measurements, insulin doses, and carbohydrate intakes over a variable length context window c_t (bounded by a maximum length c). As insulin effect diminishes over time, past doses beyond this window are assumed to have negligible influence on future glucose levels (Basu et al. 2023).

Actions. We define action a_t as the insulin dose i_t administered at time t, selected from a discerte action set $\mathcal{A}$. To reflect age-dependent insulin sensitivity, we define separate discrete action spaces consisting of 11 discrete actions for each cohort:

- Adults: [0, 2.5] IU (1 unit $\simeq$ 0.23 IU)
- Adolescents: [0, 2] IU (1 unit $\simeq$ 0.18 IU)
- Children: [0, 1] IU (1 unit $\simeq$ 0.09 IU)

These fine-grained steps are substantially smaller than typical basal-bolus doses, because we want to ensure safety. Since in a closed-loop system, the agent is capable of administering insulin more frequently, access to high doses of insulin could be fatal. Instead, the agent can achieve equivalent glycemic control by learning sequences of multiple smaller doses, especially around meals.

Reward. We adopt the reward function from (Basu et al. 2023), which optimizes the duration spent within a target safe zone (100 - 150 mg/dL), while minimizing insulin usage, to reduce the risk of insulin resistance, a state in which the body becomes less responsive to low doses of insulin. Formally, this reward is defined as:

$$r_t\left(s_{t-1}, a_t, s_t\right) = r_{t,\text{ state}}\left(g_{t-1}, g_t\right) - 0.1 * a_t^2 \quad (4)$$

Where the second term penalizes overdosing and:

$$r_{t,state}(s_{t-1}, s_t) = \begin{cases} -100 & s_t < 70 \text{ or } s_t > 200 \text{ (episode termination)} \\ -1 & s_t < 100 \text{ and } s_t - s_{t-1} < 0.5 \text{ (hypoglycemia)} \\ -1 & s_t > 150 \text{ and } s_t - s_{t-1} > 0.5 \text{ (hyperglycemia)} \\ 10 & 100 <= s_t <= 150 \text{ (target blood glucose)} \end{cases} \quad (5)$$

We further normalize the reward to [0, 1] to improve optimization stability. We define an episode as the period in which blood glucose measurements stay in the range of (70, 200) mg/dL. An episode starts uniformly from any glucose value in [100, 170]. Each step in an episode is equivalent to 9 minutes of real time.

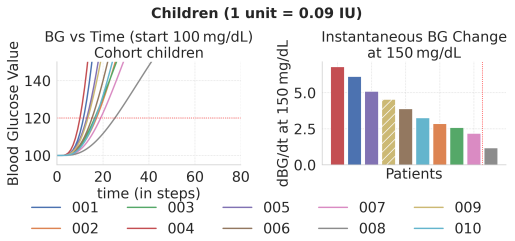
Transition Dynamics. We treat the simulator as a black-box, and assume the transition dynamics $\mathcal{P}$ are unknown, reflecting real-world scenarios where an individual's insulin response model is unknown.

Discount factor. We use a discount factor $\gamma = 0.99$.

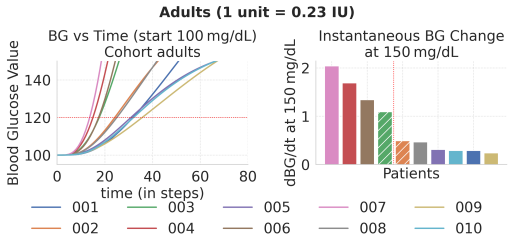
4.2 Model in the Context of Insulin Therapy

We train a 1-step prediction model $\hat{g}_{t+1} \leftarrow \mathcal{M}_\phi(s_t)$, that forecasts the next blood glucose g_{t+1} given the current state $s_t = (g_{c_t}, i_{c_t}, f_{c_t})$ on the data of the reference patients (described in section 4.3). We implement the model with an LSTM for simplicity, but a model with better generalization capabilities could further improve performance.

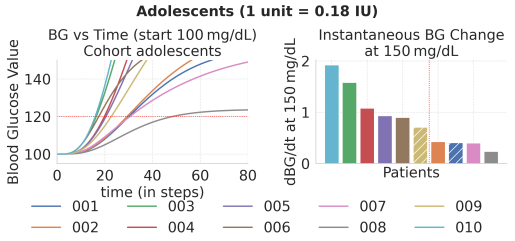
4.3 Reference Patient Selection



(a) Children: Nearly all patients form a single group, with Child#008 as the sole outlier.



(b) Adults: Three distinct groups emerge.



(c) Adolescents: The cohort splits into two groups of roughly equal size.

Figure 2: Insulin-Response Clustering. Effect of 1 unit of insulin on blood glucose in the absence of food across cohorts. For each patient, we initialized blood glucose at 100 mg/dL (with no residual insulin or food), administered the cohort's smallest insulin increment, and then observed glucose changes without further insulin or carbohydrate intake. Groups are separated by vertical red dotted lines in the bar plots. For group sizes of 2 or more, we randomly select a reference patient from that group (marked with stripes in the bar plots).

We cluster patients within each cohort based on their glycemic response to identical insulin doses under controlled conditions. Each virtual subject's blood glucose is reset to an initial level g_{init}, with no active insulin or recent food intake. The smallest nonzero insulin dose $i_{\min} = \min\{a \in \mathcal{A} \mid a > 0\}$ for the cohort is administered, and the glucose level is monitored until it reaches a final value, g_{final}. The insulin sensitivity index κ_p for patient p is defined as the rate at which blood glucose rises after a unit dose of insulin (Eq. 6). Lower κ_p means higher insulin sensitivity, as insulin is preventing the rise in blood glucose for a longer period. Patients with similar κ_p are grouped, as they respond to insulin similarly.

$$\kappa_p = \frac{g_{\text{final}} - g_{\text{init}}}{i_{\min} \cdot (t_{g_{\text{final}}} - t_{g_{\text{init}}})} \tag{6}$$

We used $g_{init} = 100$ mg/dL and $g_{final} = 120$ mg/dL for our experiments. Fig. 2 summarizes the clustering results. The children cohort is largely homogeneous, with Child#008 as the sole outlier (see Fig. 2a). In contrast, the adult cohort shows greater heterogeneity in terms of age, weight, and insulin sensitivity, forming two adult clusters: {Adult#003, Adult#004, Adult#006, Adult#007}, and rest (see Fig. 2b). The adolescent cohort forms two clusters. As illustrated in the *Reference Model Selection* block of Figure 1, at the beginning of treatment for a new(target) patient p^{target}, we identify the individual with the most similar insulin sensitivity gradient (referred to as reference patient p^{ref}) from the existing patient pool $\mathbb{P}$. We train a glucose-dynamics model $\mathcal{M}_\phi^{ref}$ on the reference patient's data $\mathcal{D}_{real}^{ref}$, and initialize the target patient's model $\mathcal{M}_\phi^{target}$ with it ($\mathcal{M}_\phi^{target} \leftarrow \mathcal{M}_\phi^{ref}$). From each cohort, we randomly select one reference patient (marked with stripes in Figure 2) and assess the effectiveness of model bootstrapping when the target and reference patients belong to the same versus different clusters.

Why is the population mean not used as the reference model? Training a glycemic dynamics model from pooled cohort data implicitly targets the global population mean, mirroring a one-size-fits-all therapy. This can lead to suboptimal performance, as the mean is sensitive to outliers and biases the policy toward patients with average dynamics. Instead, we initialize the target patient's model with the most physiologically similar individual to ensure relevance. When multiple similar candidates exist, a local mean may be used to smooth variability. This personalized warm start accelerates learning by providing informative prior experience, while unmatched patients must rely solely on their own data.

4.4 Model Learning

Training Data Generation. We generate synthetic trajectories using an expert policy, specifically, an Action-specific Deep Recurrent Q-Network (ADRQN) (Zhu, Li, and Poupart 2017), trained on a single reference patient selected as described in Sec. 4.3. The expert policies serve as surrogates for clinician dosing strategies, and could be replaced by any other clinically plausible policy. For each reference patient, we generate 10,000 trajectories and train the LSTM-based 1-step glucose-forecast model $\mathcal{M}_\phi^{ref}$. Details of the model architecture and hyperparameters are provided in Sec. A.1 of the Appendix.

How is the model used? As illustrated in the bottom panel of Fig. 1, at every step t, the learned dynamics model ${\mathcal{M}_\phi}^{target}$, initialized with a model trained on the data from the selected reference patient, is unrolled recursively to generate an H-step look-ahead (45 mins in our experiments) using the current dosing policy π_{θ_t}, producing forecasts $\hat{g}_{t+1:t+H}$. Since the LSTM supports variable-length contexts, synthetic roll-outs can be initiated from the start of each episode. From every state s_t, the agent samples k imagined trajectories with high exploration, and appends the resulting transitions $\mathcal{D}_{sim}^{target}$ to the replay buffer alongside the real data $\mathcal{D}_{real}^{target}$. This real-synthetic hybrid buffer $\mathcal{D}_{real}^{target} \cup \mathcal{D}_{sim}^{target}$ is used to train the policy π_{θ_t}, and enables the agent to evaluate potentially risky actions in silico, facilitating safer and more efficient policy learning.

Effect of Cumulative Error in Model Prediction. Recursively unrolling a one-step prediction model introduces compounding error, which can cause imagined trajectories to diverge from the true physiology. However, with sufficient one-step accuracy, these deviations remain physiologically plausible, enriching the replay buffer with diverse and informative states. We further leverage this effect by using a high exploration rate during model rollouts, allowing the agent to learn from counterfactual scenarios that are not observed in the real trajectory.

4.5 Policy Learning

H-step Deep DynaQ. We adapt H-step Deep Dyna-Q for precision insulin dosing (Algorithm 1 in Section A.2), where the agent forecasts various ways the recent insulin doses and meals can influence glucose levels over the next H steps. Unlike classical Dyna-Q, which imagines single-step synthetic transitions, this variant generates multi-step rollouts of length H. After each real interaction at time t, the agent rolls out its dynamics model $\mathcal{M}_\phi^{target}$ under the current policy π_{θ_t} to predict glucose values $\hat{g}_{t+1:t+H}$. Using a known reward function, it computes the corresponding H-step return $\hat{G}_t$ and updates the Q-network using both real and imagined transitions, accelerating learning while maintaining safety.

5 Experiments

[1] In this section, we conduct experiments to assess when transferring a glucose-dynamics model improves policy performance over learning from scratch. We train the models on randomly selected reference patients from the in-silico population, and evaluate on target patients from within and outside the reference cluster to assess the impact of patient matching. We evaluate the policies on in-silico patients, and the models on real-world data.

5.1 Data

Simulated Data. We use Simglucose (Xie 2018), a Gym-compatible implementation of the FDA-approved UVA/-Padova T1D Mellitus Simulator (T1DMS) (Man et al. 2014). In its Premarket Approval (PMA) guidelines, FDA acknowledges the use of in-silico models and virtual patients as non-clinical alternatives to animal trials in Artificial Pancreas (AP), and authorizes their use as an "assessment tool to justify and support initiation and expansion of human clinical trials" (U.S. Food and Drug Administration 2012). Only simulators that capture the nuances of real-world scenarios, such as "variability in human glucose metabolism, performance characteristics of the CGM and insulin pump, pharmacokinetics of insulin, and diffusion of glucose between the blood and interstitial fluid," (U.S. Food and Drug Administration 2012) meet the rigorous approval standards. Therefore, experimental results obtained using Simglucose may be considered equivalent to those from animal studies. Simglucose has an in-silico population of 30 patients, 10 in each of the following cohorts: children, adolescents, and adults. It takes carbohydrate intake and insulin doses as inputs, and provides blood glucose level measurements as outputs. The underlying mathematical models (Man et al. 2014) define the gastrointestinal tract, glucose kinetics, and insulin kinetics by differential equations with parameters tailored to simulate individual variability. We perform all the experiments presented in this paper on the simulated data.

Real Data. The Ohio T1D Dataset (Marling and Bunescu 2020) is a publicly available set of CGM data collected from 12 adults in 5-minute intervals over a period of 8 weeks,also including basal, temporary basal, and bolus insulin doses, carbohydrate amount, and additional physiological and activity-related data. For our analysis, we use the test data of patient 540 (male in the age group of 20-40). We discard the missing entries and retain only CGM readings in the range of [70, 200], to match the Blood Glucose region used in simulation. We only consider the basal and temporary basal insulin doses, ignoring the bolus doses, despite their effect on the glucose measurements, which may contribute to higher validation error. As filtering for this glucose range leads to discontinuities in measurements, we segment the data into contiguous blocks and treat each of them as an episode, retaining only those with at least 15 measurements. We then generate data points with variable-length context windows to mirror the pre-processing used in simulation. The final dataset of 1,911 samples is used to validate a model trained on simulated data collected from Adult#004. We do not use this dataset to validate the learned policies due to its observational nature, which lacks counterfactual outcomes and thus only supports the evaluation of the original treatment policy (Gottesman et al. 2018). Assessing alternative policies requires interventional data, which is only available through clinical trials.

5.2 Baselines

We evaluate the proposed method with two classical (Non-RL) and three RL baselines.

Non-RL Baselines

1. **Basal-Bolus.** The basal-bolus therapy administers a fixed basal insulin that maintains a steady glucose level throughout the day, and a variable bolus dose taken

[1]Code available at https://github.com/sumanabasu/Bootstrapping-Personalized-Insulin-Therapy-via-Model-Based-RL-An-In-Silico-Study

around meal times based on carbohydrate intake. It follows a pre-set schedule, and is commonly used in clinical practice

2. **PID Controller.**The PID controller is a classical feedback control mechanism that adjusts insulin delivery based on a weighted combination of the current deviation from target glucose (proportional), the area under the curve between measured and target glucose (integral), and the rate of change of glucose (derivative). This provides a reactive control mechanism for glucose regulation. For the experiments in table 2, we use the PID coefficients tuned on the target patients.

RL Baselines

1. **Direct Policy Transfer.** This setup involves no learning. We deploy the ADRQN expert policy trained on the reference patient directly to the new target patient. This allows us to assess whether additional policy adaptation is necessary.
2. **Learn from Scratch.** The agent starts with no prior knowledge and learns the policy exclusively from its interactions with the target patient, using a model-free H-step Deep Q-Network. This is marked as the region shaded in blue in Figure 1. It is an H-step variety of the model-free RL methods used in contemporary work discussed in section 2.
3. **Pre-trained model without finetuning.** We employ the dynamics model learned from the reference patient to generate rollouts for H-step Deep DynaQ (Algorithm 1 in Appendix) training, but the model is not updated with the target patient's data. This is depicted as the pink region in Figure 1. This baseline assesses the need for fine-tuning when initializing the target patient's model.

5.3 Evaluation Metrics

Policy performance is assessed with two measures:

- **Episode length.** The number of time steps until termination. An episode ends when blood glucose falls below 70 mg/dL (hypoglycaemia) or rises above 200 mg/dL (severe hyperglycaemia).
- **Time in Target Zone (also known as Time in Range, TIR).** The number of steps during which blood glucose lies within 100–150 mg/dL.

For simulated experiments, 1 step corresponds to 9 minutes, and for real data evaluation 5 minutes. If two policies achieve similar episode lengths, the one with the greater TIR is preferred. Conversely, when TIR is equal, the policy yielding the longer episode is considered superior, as premature termination reflects unsafe glucose values.

6 Results

6.1 Glucose-Dynamics Model

Evaluation on Simulated Data. We start by validating the modeling capabilities of the glucose-dynamics models trained on the three reference patients (Adult#003, Adolescent#009, and Child#009). We evaluate the 1-step performance of the trained models on test datasets of size 5,000 and report Mean Squared Error (MSE, with 95% confidence intervals) over 5 runs in Table 1. In all cohorts, the 1-step MSE remains consistently low, indicating that the models accurately predict the short-term glucose dynamics of reference patients.

Cohort	Reference Patient	Test MSE (95% CI)
Adult	#003	0.024 (0.024–0.025)
Adolescent	#009	0.019 (0.019–0.019)
Child	#009	0.006 (0.005–0.006)

Table 1: Glucose-Dynamics Model Test Set Performance. Evaluated on 5,000 test samples from the same patient used during model training. Results are reported as the mean squared error (MSE) with 95% confidence intervals, averaged over 5 runs.

Evaluation on Real Data. We match the sampling time and action set of the simulator to train a model on 10,000 simulated trajectories from adult#004 and evaluate it on patient 540 from the Ohio T1D dataset (preprocessing detailed in Section 5.1). Figure 3 shows predicted versus actual blood glucose values for six randomly selected samples. The clinically recommended pre-prandial glucose range (80-130 mg/dL) (American Diabetes Association 2023) is shaded green, with hyperglycemic and hypoglycemic zones shaded red above and below, respectively. While predictions are not exact, they track the overall trend and lie within the correct glucose zone. Figure 4 demonstrates that 91.31% of the predicted glucose values fall within the same zone (46.21% target, 41.23% hyperglycemia, 3.87% hypoglycemia) as the corresponding ground truth values, and notably, the model almost never confuses hyperglycemia with hypoglycemia or vice versa, errors that could be clinically dangerous.

6.2 Treatment Policy

Non-RL Baselines. For each cohort, we select a target patient from the same insulin-sensitivity cluster as the reference individual. As shown in Figure 2, Child#010, Adult#006, and Adolescent#010 are clustered with reference patients Child#009, Adult#003, and Adolescent#009, respectively. We evaluate the policies over 100 episodes with identical initial glucose levels and meal schedules to ensure fairness, and report the results in Table 2. Basal-Bolus values are derived from the simulator-provided parameters of each reference patient, while the PID coefficients are tuned on the target patients. Compared to these non-RL baselines, our method consistently yields longer episodes and better time in the target glucose range across all cohorts.

RL Baselines. We evaluate the RL baselines introduced in Section 5.2 using the dynamics models pretrained on the randomly selected reference patients (marked with stripes in Figure 2) from each insulin-sensitivity cluster with more than one patient. As shown in Figure 5, policy learning improves significantly when the target patient (Adult#006) belongs to the same sensitivity cluster as the reference (Adult#003). In contrast, transferring a model

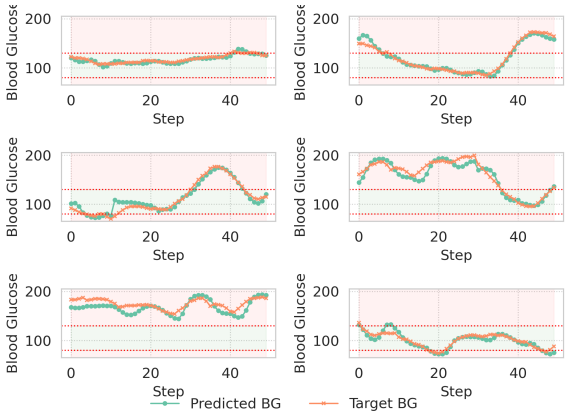


Figure 3: Model Evaluation on Real Data. Randomly selected zero-shot evaluation samples of a model trained on the simulated data from adult#004, and tested on real-world data from Ohio T1DM patient 540. Blood glucose is measured in mg/dL, with each step corresponding to a 5-minute interval. While not perfect, the predictions closely follow the overall trend and lie within the correct glucose zone.

Cohort	Policy	Episode Length (95% CI)	Time in Range (95% CI)
Adult#006	Basal-Bolus	23.27 (22.06–24.47)	11.54 (9.89–13.18)
	PID	23.17 (21.95–24.38)	11.53 (9.88–13.17)
	Ours	**45.56** (38.32–52.79)	**23.91** (18.78–29.04)
Adolescent#010	Basal-Bolus	20.820 (19.80–21.83)	9.040 (7.56–10.51)
	PID	20.29 (19.19–21.38)	8.97 (7.49–10.44)
	Ours	**38.64** (32.17–45.10)	**17.69** (14.05–21.33)
Child#010	Basal-Bolus	19.01 (17.75–20.26)	8.94 (7.39–10.48)
	PID	16.44 (15.70–17.17)	7.36 (6.18–8.53)
	Ours	**45.28** (38.00–52.55)	**23.13** (18.25–28.00)

Table 2: Non-RL Baselines. Evaluation of three policies over 100 episodes per cohort. All policies were initialized with identical glucose levels and meal schedules. Metrics are reported as means with 95% confidence intervals (CI).

from Adult#003 provides little benefit for Adult#008, who belongs to a different cluster. However, performance improves when switching to a model trained on a closer match (Adult#002) from the same cluster. *These results underscore the importance of model–patient similarity when bootstrapping RL policies.* The `Pre-trained Model Transfer without fine-tuning` baseline consistently underperforms compared to the fine-tuned version (our method), demonstrating that *fine-tuning is essential*. Although `direct policy transfer` performs well initially (yellow line in Figure 5), it fails to match the long-term performance of policies trained or fine-tuned on the target patient. We observe similar trends in the children and adolescent cohorts. *These results indicate that response-based clustering offers a reliable guideline for model reuse - when the target and reference patients belong to the cluster, a transferred (and fine-tuned) dynamics model substantially aids the RL agent. Otherwise, learning solely from real-world experience is the preferred strategy.*

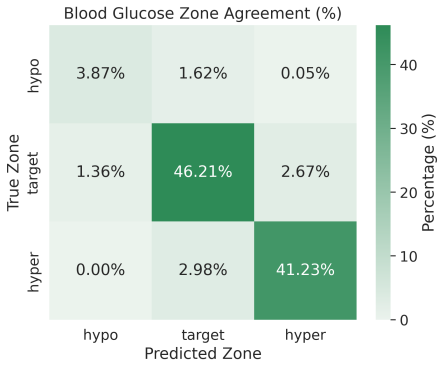


Figure 4: Glucose Zone Agreement of Predicted vs. True Blood Glucose. The confusion matrix shows zone-wise agreement between predicted and ground truth glucose values for the OhioT1DM patient 540. The model achieves 91.31% zone-level accuracy, with minimal confusion between hyperglycemia and hypoglycemia.

7 Conclusion

We present an MBRL framework that bootstraps personalized insulin therapy by transferring knowledge from physiologically similar patients. Fine-tuning the transferred model on the target patient's data proves crucial to maintain relevance and ensure effective adaptation. Our results indicate that insulin sensitivity-based transfer learning offers a clinically meaningful strategy for accelerating policy convergence. A key limitation of this study is that the learned policies are not evaluated on real-world data due to the lack of counterfactuals in observational datasets. However, the strong model generalization on real data, along with promising policy performance in simulation, motivates future exploration through small-scale interventional studies such as clinical trials.

A Implementation Details

All experiments were performed on a single A100 GPU, with smaller models trained concurrently.

A.1 LSTM Model Details

Architecture Continuous glucose values are mapped into a 16D Fourier feature(fixed). Discrete insulins are embedded into 16D embeddings(learned) and scaled with tanh. Continuous foods are linearly projected into 16D vectors (learned) and scaled with tanh. These encodings over a history of length 15 form 48×15 input sequences to the 5-layer unidirectional LSTM (hidden=1024), followed by a 2-layer MLP($1024 \rightarrow 512 \rightarrow 1$) with LeakyReLU. Output is constrained in the range [70, 200] using a scaled sigmoid activation.

Optimization Network parameters are initialized with Xavier Uniform. We train the model using Mean Squared Error between predicted and actual glucose. We use Adam optimizer(LR=0.001) with a scheduler that reduces LR by a

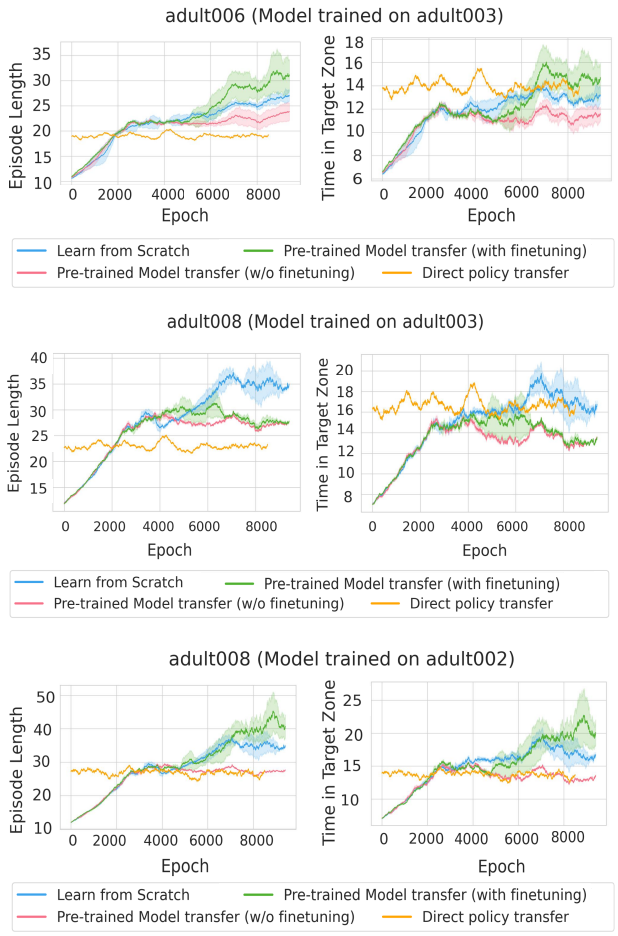


Figure 5: RL Performance on Adult Population. Impact of reference patient selection on policy performance (averaged over 3 seeds). *top:* Target (Adult006) and reference patient (Adult003) belong to the same group. *middle:* Target (Adult008) and reference patient (Adult003) belong to different groups. *bottom:* Reference patient (Adult#002) is chosen from the same group. Target-Reference similarity leads to better policy outcomes.

factor of 0.1, if the validation loss does not improve for 10 consecutive steps(min 10^{-5}).

LSTM Model for Real-data validation For training models on the simulator for real-world validation, we adjust the sampling interval from 9 to 5 minutes and use a discrete action set derived from the training data of patient 540:

[0.0, 0.033, 0.035, 0.037, 0.05, 0.067, 0.07, 0.078, 0.079, 0.083, 0.088, 0.092, 0.1, 0.101, 0.103, 0.132, 0.133, 0.157, 0.158, 0.167, 0.175, 0.2, 0.227, 0.238, 0.262]
This set captures all unique insulin doses (basal and temporary basal) in the training data. The test set contains a subset of these actions - [0.0, 0.033, 0.067, 0.079, 0.088, 0.133, 0.158, 0.175]. But we retain the full action set to preserve compatibility with the complete dataset in the future.

We use a random policy to generate data from Adult#004 to train the model. We apply the same preprocessing as described above, followed by input normalization. The model consists of a 2-layer unidirectional LSTM (hidden=256), followed by a 2-layer MLP($256 \rightarrow 128 \rightarrow 1$) with LeakyReLU. The output is constrained to the [0, 1] range using a softmax activation and denormalized after loss computation.

A.2 H-step Deep DynaQ Algorithm

Algorithm 1 outlines the H-step Deep DynaQ in the context of glucose control.

Algorithm 1: **H-step Deep DynaQ**

Require: Q-network $\mathcal{Q}_\theta$, policy $\pi_{\theta_t} = \epsilon$-greedy$(s_t, \mathcal{Q}_{\theta_t})$, model pre-trained on reference patient $\mathcal{M}_\phi^{ref}$, horizon $\mathcal{H}$, reward function $\mathcal{R}(\cdot)$, discount γ, batch size, rollout ratio k, model update frequency $\nu_\mathcal{M}$

Initialize buffers: $\mathcal{D}_{real}, \mathcal{D}_{sim}, \mathcal{B}_T, \mathcal{B}_\mathcal{M}$
Initialize target model: $\mathcal{M}_\phi^{target} \leftarrow \mathcal{M}_\phi^{ref}$
for episode $j = 1$ to MaxEpisodes **do**
  $g_{c_0} \leftarrow g_0$, $i_{c_0} \leftarrow 0$, $f_{c_0} \leftarrow 0$
  $s_0 \leftarrow (g_{c_0}, i_{c_0}, f_{c_0})$
  for $t = 0$ to end of episode **do**
    $i_t \sim \pi_{\theta_t}(s_t)$
    $g_{t+1}, r_t, f_t, done \leftarrow p^{target}(i_t)$
    Add $(s_t, i_t, r_t, done)$ to $\mathcal{B}_T$
    Add (s_t, i_t, f_t, g_{t+1}) to $\mathcal{B}_\mathcal{M}$
    Update $g_{c_{t+1}}, i_{c_{t+1}}, f_{c_{t+1}}$ using (g_{t+1}, i_t, f_t)
    $s_{t+1} \leftarrow (g_{c_{t+1}}, i_{c_{t+1}}, f_{c_{t+1}})$
    $(\hat{g}_{t+1:t+H}, \hat{i}^{\pi_{\theta_t}}_{t+1:t+H}) \leftarrow \mathcal{M}_{\phi_t}^{target}(s_{t+1}, \pi_{\theta_t})$
    $\hat{s}_{t+H} \leftarrow$ next simulated state using $\{(\hat{g}_k, \hat{i}^{\pi_{\theta_t}}_k, \hat{f}_k = 0)\}_{k=t+1}^{t+H}$
    $(\hat{r}_{t+1:t+H}, \hat{done}_H) \leftarrow \mathcal{R}(\hat{g}_{t+1:t+H}, \hat{i}^{\pi_{\theta_t}}_{t+1:t+H})$
    $\hat{G}_t \leftarrow r_t + \sum_{k=t+1}^{H-1} \gamma^k \hat{r}_k + (1 - \hat{done}_H)\gamma^H \mathcal{Q}_{\theta_t}(\hat{s}_{t+H}, \pi_{\theta_t}(\hat{s}_{t+H}))$
    Add $(s_t, i_t, \hat{G}_t, \hat{s}_{t+H}, \hat{done}_H)$ to $\mathcal{D}_{sim}$
    $s_t \leftarrow s_{t+1}$
    if $|\mathcal{D}_{real}| >$ batch size **then**
      Train $\mathcal{Q}_{\theta_t}$ using $\mathcal{D}_{real}$
      if $|\mathcal{D}_{sim}| >$ batch size **then**
        for $k = 1$ to K **do**
          Train $\mathcal{Q}_{\theta_t}$ using $\mathcal{D}_{sim}$
        end for
      end if
    end if
  end for
  Add $\mathcal{B}_T$ to $\mathcal{D}_{real}$; clear $\mathcal{B}_T$
  if $j \bmod \nu_\mathcal{M} = 0$ **then**
    Update $\mathcal{M}_{\phi_t}^{target}$ using $\mathcal{B}_\mathcal{M}$
  end if
end for

References

American Diabetes Association. 2009. Diagnosis and clas-

sification of diabetes mellitus. *Diabetes Care*, 32 Suppl 1(Supplement_1): S62–7.

American Diabetes Association. 2023. Glycemic Targets: Standards of Care in Diabetes—2024. https://professional.diabetes.org/sites/dpro/files/2023-12/glycemic_targets_1.pdf. Accessed: 2025-08-01.

Basu, S.; Legault, M.-A.; Romero-Soriano, A.; and Precup, D. 2023. On the Challenges of using Reinforcement Learning in Precision Drug Dosing: Delay and Prolongedness of Action Effects. arXiv:2301.00512.

Ben Ali, J.; Hamdi, T.; Fnaiech, N.; Di Costanzo, V.; Fnaiech, F.; and Ginoux, J. 2018. Continuous blood glucose level prediction of type 1 diabetes based on artificial neural network. *Biocybernetics and Biomedical Engineering*, 38(4): 828–840.

Bergenstal, R. M.; Garg, S.; Weinzimer, S. A.; Buckingham, B. A.; Bode, B. W.; Tamborlane, W. V.; and Kaufman, F. R. 2016. Safety of a Hybrid Closed-Loop Insulin Delivery System in Patients With Type 1 Diabetes. *JAMA*, 316(13): 1407–1408.

Bergenstal, R. M.; Tamborlane, W. V.; Ahmann, A.; Buse, J. B.; Dailey, G.; Davis, S. N.; Joyce, C.; Perkins, B. A.; Welsh, J. B.; Willi, S. M.; Wood, M. A.; and STAR 3 Study Group. 2011. Sensor-augmented pump therapy for A1C reduction (STAR 3) study: results from the 6-month continuation phase. *Diabetes Care*, 34(11): 2403–2405.

Bruttomesso, D.; Farret, A.; Costa, S.; Marescotti, M. C.; Vettore, M.; Avogaro, A.; Tiengo, A.; Dalla Man, C.; Place, J.; Facchinetti, A.; Guerra, S.; Magni, L.; De Nicolao, G.; Cobelli, C.; Renard, E.; and Maran, A. 2009. Closed-loop artificial pancreas using subcutaneous glucose sensing and insulin delivery and a model predictive control algorithm: preliminary studies in Padova and Montpellier. *J. Diabetes Sci. Technol.*, 3(5): 1014–1021.

Carroll, M. F.; Burge, M. R.; and Schade, D. S. 2003. Severe Hypoglycemia in Adults. *Rev. Endocr. Metab. Disord.*, 4(2): 149–157.

Daskalaki, E.; Diem, P.; and Mougiakakou, S. G. 2016. Model-free machine learning in biomedicine: Feasibility study in type 1 diabetes. *PLoS One*, 11(7): e0158722.

DiabetesNet. 2025a. Comparison of Automated Insulin Delivery Systems. https://www.diabetesnet.com/diabetes-technology/comparison-of-automated-insulin-delivery-systems/. Accessed: 2025-08-01.

DiabetesNet. 2025b. Comparison of Automated Insulin Delivery Systems. https://www.diabetesnet.com/diabetes-technology/comparison-of-automated-insulin-delivery-systems/. Accessed: 2025-08-01.

Fox, I.; Lee, J.; Pop-Busui, R.; and Wiens, J. 2020. Deep Reinforcement Learning for Closed-Loop Blood Glucose Control. arXiv:2009.09051.

Garg, S. K.; Weinzimer, S. A.; Tamborlane, W. V.; Buckingham, B. A.; Bode, B. W.; Bailey, T. S.; Brazg, R. L.; Ilany, J.; Slover, R. H.; Anderson, S. M.; Bergenstal, R. M.; Grosman, B.; Roy, A.; Cordero, T. L.; Shin, J.; Lee, S. W.; and Kaufman, F. R. 2017. Glucose Outcomes with the In-Home Use of a Hybrid Closed-Loop Insulin Delivery System in Adolescents and Adults with Type 1 Diabetes. *Diabetes Technology & Therapeutics*, 19(3): 155–163.

Gottesman, O.; Johansson, F.; Meier, J.; Dent, J.; Lee, D.; Srinivasan, S.; Zhang, L.; Ding, Y.; Wihl, D.; Peng, X.; Yao, J.; Lage, I.; Mosch, C.; wei H. Lehman, L.; Komorowski, M.; Komorowski, M.; Faisal, A.; Celi, L. A.; Sontag, D.; and Doshi-Velez, F. 2018. Evaluating Reinforcement Learning Algorithms in Observational Health Settings. arXiv:1805.12298.

Lim, M. H.; Lee, W. H.; Jeon, B.; and Kim, S. 2021. A Blood Glucose Control Framework Based on Reinforcement Learning With Safety and Interpretability: In Silico Validation. *IEEE Access*, 9: 105756–105775.

Man, C. D.; Micheletto, F.; Lv, D.; Breton, M.; Kovatchev, B.; and Cobelli, C. 2014. The UVA/PADOVA type 1 Diabetes Simulator: New features. *J. Diabetes Sci. Technol.*, 8(1): 26–34.

Marling, C.; and Bunescu. 2020. The OhioT1DM Dataset for Blood Glucose Level Prediction: Update 2020. *CEUR workshop proceedings*, vol. 2675 (2020): 71–74.

Maxfield, K.; and Zineh, I. 2021. Precision dosing: A clinical and public health imperative. *JAMA*, 325(15): 1505–1506.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; Petersen, S.; Beattie, C.; Sadik, A.; Antonoglou, I.; King, H.; Kumaran, D.; Wierstra, D.; Legg, S.; and Hassabis, D. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533.

Moerland, T. M.; Broekens, J.; Plaat, A.; and Jonker, C. M. 2022. Model-based Reinforcement Learning: A Survey. arXiv:2006.16712.

Nemat, H.; Khadem, H.; Elliott, J.; and Benaissa, M. 2024. Data-driven blood glucose level prediction in type 1 diabetes: a comprehensive comparative analysis. *Scientific Reports*, 14: 21863.

Ogrotis, I.; Koufakis, T.; and Kotsa, K. 2023. Changes in the global epidemiology of type 1 diabetes in an evolving landscape of environmental factors: Causes, challenges, and opportunities. *Medicina (Kaunas)*, 59(4).

Peters, A. L.; Ahmann, A. J.; Battelino, T.; Evert, A.; Hirsch, I. B.; Murad, M. H.; Winter, W. E.; and Wolpert, H. 2016. Diabetes technology-continuous subcutaneous insulin infusion therapy and continuous glucose monitoring in adults: An Endocrine Society clinical practice guideline. *J. Clin. Endocrinol. Metab.*, 101(11): 3922–3937.

Potosnak, W.; Challu, C.; Olivares, K. G.; Dufendach, K. A.; and Dubrawski, A. 2025. Global Deep Forecasting with Patient-Specific Pharmacokinetics. arXiv:2309.13135.

Ruiz, J. L.; Sherr, J. L.; Cengiz, E.; Carria, L.; Roy, A.; Voskanyan, G.; Tamborlane, W. V.; and Weinzimer, S. A. 2012. Effect of insulin feedback on closed-loop glucose control: a crossover study. *J. Diabetes Sci. Technol.*, 6(5): 1123–1130.

Sutton, R. S. 1991. Dyna, an integrated architecture for learning, planning, and reacting. *SIGART Newsl.*, 2(4): 160–163.

Tejedor, M.; Woldaregay, A. Z.; and Godtliebsen, F. 2020. Reinforcement learning application in diabetes blood glucose control: A systematic review. *Artif. Intell. Med.*, 104(101836): 101836.

Trevitt, S.; Simpson, S.; and Wood, A. 2015. Artificial Pancreas Device Systems for the Closed-Loop Control of Type 1 Diabetes. *Journal of Diabetes Science and Technology*, 10(3): 714–723.

U.S. Food and Drug Administration. 2012. Artificial Pancreas Device System. https://www.fda.gov/medical-devices/consumer-products/artificial-pancreas-device-system. Accessed: 2025-08-01.

Watkins, C. J. C. H.; and Dayan, P. 1992. Q-learning. *Mach. Learn.*, 8(3-4): 279–292.

Xie, J. 2018. Simglucose v0.2.1 [Online]. *Avaible: https://github.com/jxx123/simglucose. Accessed on: 04-15-2024.*

Yamagata, T.; O'Kane, A.; Ayobi, A.; Katz, D.; Stawarz, K.; Marshall, P.; Flach, P.; and Santos-Rodríguez, R. 2020. Model-Based Reinforcement Learning for Type 1Diabetes Blood Glucose Control. arXiv:2010.06266.

Zhu, P.; Li, X.; and Poupart, P. 2017. On Improving Deep Reinforcement Learning for POMDPs. *CoRR*, abs/1704.07978.