Probabilistic Digital Twin for Data-driven Smart Weaning of Mechanical Circulatory Devices

Sophia Sun[1*] **Aysin Tumay**[1*] **Sonia Fereidooni**[1] **Aaron Dumas**[2] **Elise Jortberg**[3]

Rose Yu[1]
[1]University of California, San Diego
[2]California Institute of Technology
[3]Abiomed

Abstract

In this paper, we study the sequential decision-making for smart weaning of mechanical circulatory (MCS) devices. MCS devices are percutaneous micro-axial flow pumps for the treatment of cardiogenic shock patients, by providing left ventricular unloading and forward flow of blood into the aorta. While clinical recommendations for the weaning of MCS devices exist, the strategy varies by care team and data-driven approaches are limited. Offline reinforcement learning (RL) has proven to be successful in sequential decision-making tasks, but the prohibition of interactions with the patient as the environment constrains evaluating RL policies. This motivates the development of probabilistic digital twin models to simulate the environment. We propose a formulation for offline RL training and a probabilistic Transformer-based digital twin to model the noisy circulatory dynamics and evaluate offline RL policies. We show that our Transformer-based digital twin (TDT) achieves 35% lower error compared to baseline models. We also present a comprehensive benchmark on offline RL methods using TDT with clinically relevant metrics.

1 Introduction

This paper focuses on the problem of data-driven automatic weaning of mechanical circulatory support (MCS) devices. MCS devices assist the heart by pumping oxygen-rich blood from the left ventricle into the ascending aorta, supporting patients with compromised cardiac function. Weaning from MCS is a series of flow controls over a period of time in which the clinician aims to reduce flow support while maintaining stable hemodynamics, prior to explanting MCS [1]. Reducing the pump flow level (P-Level) is entirely at the discretion of the clinician: the manufacturer's instructions for use suggest reducing by levels of 2, and evaluating at each reduction for evidence of deterioration.

Deep reinforcement learning (RL) has shown great promise in automating sequential decision making in medical treatments, with works exploring clinical conditions such as sepsis [13, 7, 23] and cancer [18, 3]. With RL's ability to learn sequential decisions from real-world datasets, a data-driven automated policy can reduce decision fatigue for clinicians and offer richer guidance compared to rule-based guidelines. However, in the medical domain, offline RL introduces two challenges: in training, the stochasticity of clinician decisions and limited data hinder learning; in evaluation, because online interaction with patients is infeasible, assessment must rely solely on simulators.

Although there exist some physics-informed models and PDE simulators [8, 10] of patient hemodynamics, they are often deterministic and not suitable for long time-horizon simulation. Existing

*These authors contributed equally.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

solutions fail to account for noise in the real-life patient data and partial observation, due to the fact that patients on MCS often receive additional treatments (e.g., surgery, medications) that the models do not observe. To realistically evaluate a weaning strategy, a digital twin model that can faithfully quantify uncertainty over a significant time-horizon is integral.

To this end, we develop a digital twin-supported medical environment to evaluate offline RL policies for MCS weaning. To simulate the circulatory dynamics and answer "what-if" questions on patient data, we leverage a Transformer-based digital twin model. Our digital twin captures uncertainty from the variability of learning and stochasticity in medical data, and can act as a practical surrogate to model real-time interactions. We then use the digital twin to create a medical environment, where researchers can evaluate and develop offline RL policies. In summary, our contributions are:

1. We develop a transformer-based probabilistic digital twin for modeling MCS circulatory dynamics, outperforming baselines on both accuracy and uncertainty quantification metrics.
2. We present a Markov Decision Process (MDP) formulation for learning MCS weaning, and show preliminary offline RL results with domain-specific physiological and medical metrics.

We refer the readers to Appendix A for a review on safe medical decision making. In contrast to existing methods, our work employs a probabilistic transformer-based forecaster trained on real data, and leverage domain-specific medical metrics to facilitate evaluation of the offline RL models.

2 Background and Problem Formulation

Offline Reinforcement Learning. In this work, we formulate our setting as a Markov decision process (MDP), defined by the tuple $M = (\mathcal{S}, \mathcal{A}, T, r, \mu_0, \gamma)$, with state space $\mathcal{S}$, action space $\mathcal{A}$, transition dynamics $T(s'|s,a)$, reward function $r(s,a)$, initial state distribution μ_0, and discount factor, γ. Reinforcement Learning algorithms aim to find a policy $\pi := S \to A$ that maximizes the expected cumulative reward, $\mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]$. The optimal policy is defined as,

$$\pi^* = \arg\max_\pi \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]. \tag{1}$$

The *Offline RL setting* is when the algorithm only has access to a dataset sampled from the environment $\mathcal{D}_{\text{env}} = \{(s_i, a_i, r_i, s_i')\}_i$ collected by a behavior policy π^{B} and cannot interact with the environment.

Mechanical Circulatory Support (MCS). Left sided forward flow MCS devices are medical devices designed to assist the heart in pumping blood from the left ventricle into the ascending aorta to deliver oxygenated blood to the body. Cardiogenic Shock (CGS) is a syndrome characterized by cardiac output insufficient for end organ perfusion. Hemodynamically, patients in CGS exhibit low systolic blood pressures, low mean aortic blood pressures, and high heart rates. CGS's mortality rate is historically 50-80% [19, 14]. For patients in severe CGS, MCS plays an integral role in improving blood pressure, maintaining organ perfusion, and aiding heart muscle recovery.

As the patient shows signs of improvement, the care team begins to wean the patient from MCS support. The weaning process includes step-wise reduction in MCS performance pump level (P-Level) with regular assessment of patient response, see Figure 6 for examples. However, to observe patient response, the clinician must reduce P-level and induce a change in patient state. In order to learn and evaluate weaning strategies, we need an environment that predicts the patient response to the proposed change in P-level and evaluates the quality of that P-level choice.

3 Methodology

MDP Design for MCS. We first formulate the MCS weaning problem as an MDP. We define each *state* in the MDP to consist of 6 time-steps of 12 physiological features over 1 hour, calculated from the aortic and left ventricular pressure signals coming from the MCS device, i.e. $\mathcal{S} \subseteq \mathbb{R}^{72}$. The *action* space is $\mathcal{A} = \{2, 3, \cdots, 9\}$, corresponding to pump level P2 to P9 on the MCS device. The objective is to optimize patient outcome with a clinically appropriate weaning strategy. For the offline RL problem, we organize the patient data into a replay buffer dataset of $\mathcal{D} = \{(s_i, a_i, s_i', r_i)\}_i$ according to the formulation. The state space, action space, reward, and MDP design is informed by expert recommendation and empirical results as presented in Appendix C. Under the MDP formulation, we will then describe our digital twin for evaluation, followed by the setup for offline RL training.

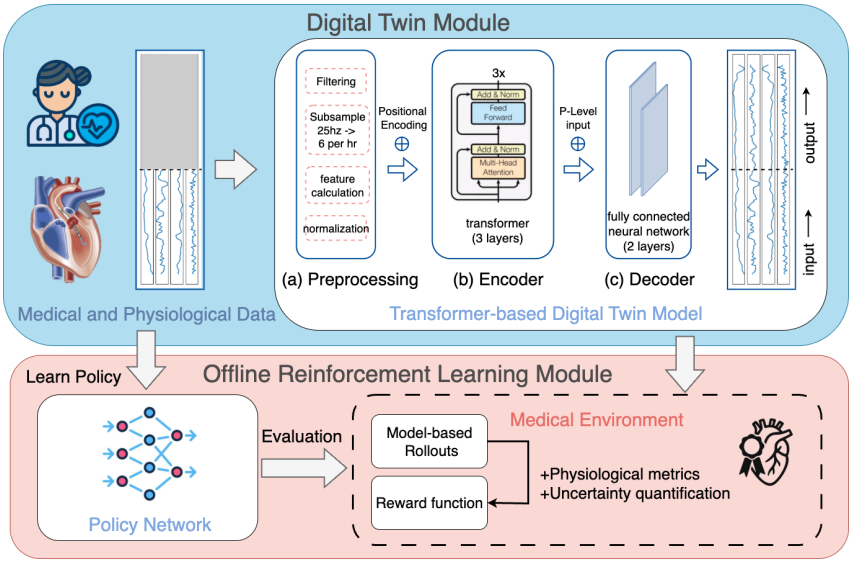

Figure 1: System diagram of the proposed framework. **Digital twin module:** We use a 3-layer transformer architecture as an encoder to learn a latent representation of the patient's history, concatenate the representation with the P-Level input, and then decode the output using a fully connected neural network. **Offline RL module:** The digital twin can then be deployed in our medical environment to evaluate RL policies trained fully *offline*, with rich medical and physiological metrics.

Transformer-based Digital Twin Design. To simulate patient trajectories during weaning, we develop a Transformer-based digital twin (TDT) that models patient hemodynamic signals under MCS, denoted as $\mathcal{F} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$, serving as a proxy of the stochastic transition function for the RL task, i.e. $\mathcal{F}(s, a) = p(s'|s, a)$. We use the DT terminology to indicate that the environment can be used to evaluate learned policies, not to be confused with a complete replication of the system.

The digital twin's model architecture is shown in Figure 1. The encoder with three multi-head self-attention layers captures long-range temporal dependencies in the multivariate physiological time series. The future actions (pump motor speed represented by discrete P-Level) are concatenated to the latent representation and passed to the decoder. The decoder (2 fully connected perceptron layers) predicts the next physiological state, enabling safe synthetic "what-if" scenarios by simulating patient responses to candidate weaning actions. Probabilistic prediction is achieved by retaining dropout ($p = 0.1$) in the decoder layers. We train the model to minimize the MSE between the predicted and observed future states, using historical patient data.

Offline Reinforcement Learning. Using the TDT model, we simulate the next hemodynamic state of a patient given the current state and the recommended P-level to evaluate with clinical metrics. Before training, we first collect $\mathcal{D}_{\text{env}}$ by organizing patient trajectories into a replay buffer according to our state and reward design. Then, a policy $\hat{\pi}$ can be trained on $\mathcal{D}_{\text{env}}$ with the offline RL objective (Eq. 1). The policy's performance is evaluated in the digital twin environment by the following metrics: Physiological Reward, reflecting well-being from MAP, heart rate, and pulsatility over the past hour; Action Change Penalty (ACP) [23], accumulating the magnitude of P-level changes; and Weaning Score (WS), capturing the decrease in P-level and penalizing when P-level is increased every hour conditioned on observed hemodynamic stability (see Appendix B for definitions).

4 Experiments

Digital Twin Training and Performance. Training and evaluation of the digital twin environment is performed on a proprietary dataset of 379 patients, with an average length of record of 65.5 hours. We split the patients by ratio 65-15-20 into training, validation, and testing sets. TDT's performance is measured by predictions accuracy of (over different aspects and subsets of the data) and uncertainty calibration (CRPS) in comparison with common dynamics modeling architectures. **Please see appendix D for detailed introduction of metrics and baselines.** Table 1 shows that our TDT model with sinusoidal encoding consistently outperforms baselines across accuracy and calibration metrics, indicating robustness under both static and dynamic conditions. The strong performance of

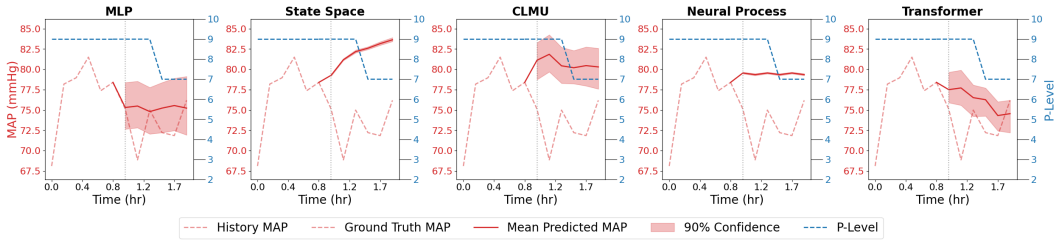

Figure 2: Digital twin prediction visualization compared with baselines. The Transformer model is more accurate in reflecting response to P-level change and more expressive when capturing large changes in patient state, resulting in its higher accuracy. More qualitative results Appendix D.

the Transformer-based architecture is demonstrated through higher prediction variability, and more accurate modeling of P-Level change response, as demonstrated in figures 2 and 7.

	MAE	MAE (MAP only)	MAE Static PL	MAE changing PL	Trend Acc.	CRPS
MLP	9.85 $\pm$ 0.44	4.11 $\pm$ 0.01	8.88 $\pm$ 0.45	13.76 $\pm$ 0.40	0.83 $\pm$ 0.03	7.43 $\pm$ 0.22
Neural Process	8.32 $\pm$ 0.18	4.63 $\pm$ 0.06	6.83 $\pm$ 0.26	14.31 $\pm$ 0.22	0.89 $\pm$ 0.00	4.92 $\pm$ 0.66
CLMU	7.61 $\pm$ 0.12	4.31 $\pm$ 0.04	7.00 $\pm$ 0.11	10.06 $\pm$ 0.17	0.89 $\pm$ 0.00	5.48 $\pm$ 0.09
SSM	8.12 $\pm$ 0.46	4.12 $\pm$ 0.11	7.49 $\pm$ 0.55	10.65 $\pm$ 0.11	0.88 $\pm$ 0.00	4.43 $\pm$ 0.29
TDT (rot.)	6.04 $\pm$ 0.72	4.04 $\pm$ 0.10	5.55 $\pm$ 0.70	8.02 $\pm$ 0.79	**0.90** $\pm$ **0.01**	3.92 $\pm$ 0.54
TDT (sin.)	**5.41** $\pm$ **0.05**	**3.88** $\pm$ **0.12**	**4.90** $\pm$ **0.05**	**7.47** $\pm$ **0.08**	0.88 $\pm$ 0.01	**3.45** $\pm$ **0.12**

Table 1: Digital twin model evaluation; See appendix D for detailed explanation of metrics and baselines. Transformer outperforms baselines in all metrics.

Offline RL Performance Benchmark. The offline RL algorithms are trained on the replay buffer created from our dataset of size 17865, and evaluated using the digital twin, autoregressively extended 1-hour to 6-hour horizon. Our digital twin serves as a model-based offline policy (OPE) evaluation method [25, 22]. We present the offline dataset P-levels as the expert policy alongside three offline RL policies: Behavioral Cloning (BC) for supervised training without exploration; Model-based Policy Optimization (MBPO) [6] modeling the transition function, Model-based Offline Policy Optimization (MOPO) [24] penalizing reward with transition uncertainty; and Support Value Regularization (SVR) [11] for out-of-distribution (OOD) regularization. See Appendix E for details.

	Expert (DT)	Expert (Real)	BC	MBPO	MOPO	SVR
Phys. Reward (↑)	0.557	0.371	0.175 ± 0.118	0.420 ± 0.139	0.373 ± 0.129	0.530 ± 0.152
ACP (↓)	1.79	1.78	0.068 ± 0.012	0.459 ± 0.032	0.984 ± 0.020	0.599 ± 0.057
WS (↑)	0.042	0.076	0.321 ± 0.009	0.120 ± 0.008	0.035 ± 0.005	0.140 ± 0.005

Table 2: Different offline RL models evaluated with 6-hour roll-outs in the TDT environment for 1000 episodes in terms of physiological reward, action change penalty, and weaning score. Results are averaged over 5 seeds. We include expert policy evaluation with digital twin rollouts (DT) and offline data (Real) for 744 episodes. (↑) and (↓) mean higher and lower is better, respectively. Low ACP indicates an overly stationary policy; high ACP indicates excessive stochasticity. High physiological reward suggests patient stability, while high WS shows successful weaning of the MCS device.

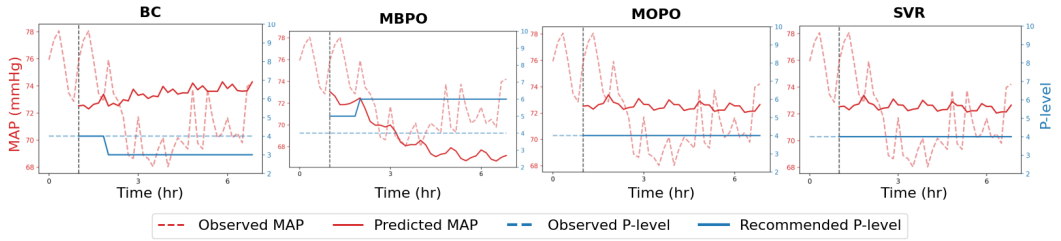

Figure 3: We depict P-level recommendations of offline RL policies (solid blue) and the digital twin mean arterial pressure (MAP) 6-hour roll-outs (solid red) compared to the expert (clinician) P-level and observed MAP. Left of the vertical gray line shows the input MAP to the digital twin.

We define a clinically salient policy as one that excels across all metrics, showing high-quality policy training with high physiological reward, low stochasticity with low ACP, and successful weaning with high WS. Note that we do not aggregate the metrics, as their relative importance should first be determined through adoption and feedback from intensive care unit clinicians. In Table 2, we emphasize that no single offline RL model performs best across all metrics for our task. We expect the expert (Real) physiological reward to be close to 0 as we normalized it to stabilize policy training. All expert evaluations depict high ACP and low WS, suggesting frequent P-level changes despite stable physiology. BC demonstrates good performance in ACP and WS, learning from the successful weaning cases. MOPO and MBPO achieve similar physiological rewards, although they show limited weaning and high P-level variability based on high ACP and low WS, as in Figure 3, indicating a conservative policy. SVR is promising for WS and physiological reward, though it results in higher ACP than BC and MBPO, underscoring over-regularization. These results indicate the further need for RL models with OOD regularization and uncertainty quantification.

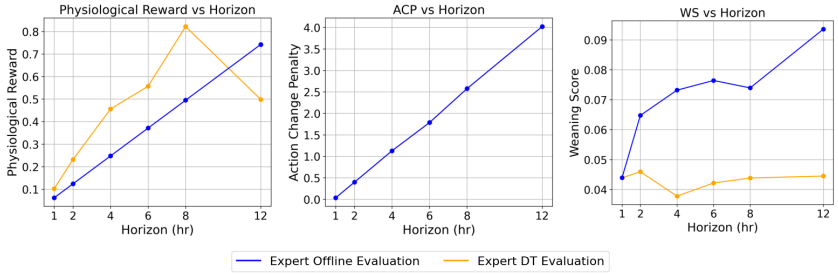

Figure 4: Comparison of change in physiological reward, ACP, and WS metrics with respect to various evaluation horizons in hours. This experiment is done on the clinician P-level decisions evaluated by our digital twin (orange) and on the offline patient outcomes (blue).

Simulation-Real Gap of Digital Twin on Expert Policy. In Figure 4, we observe that our digital twin performs similarly to the offline dataset within bounds. Noting that digital twin results in mostly higher physiological rewards, the gap between offline and digital twin evaluation increases within a tolerable bound, where the largest increase is 60%. The overall digital twin ACP remains the same since the compared actions are the real P-levels, while only the next state differs. In WS, the gap increases as horizon increases with offline dataset always having higher WS. In light of these, we select 6 hours for the evaluation horizon as it results in a tolerable gap between digital twin and offline data across physiological reward and WS.

5 Discussion

We presented a probabilistic transformer-based digital twin to model MCS weaning dynamics and evaluate offline RL policies. Our TDT consistently outperformed baselines in modeling patient hemodynamics. Using our TDT as model-based offline policy evaluation, we evaluated offline RL algorithms on our task. We found that no existing method excelled across all metrics, highlighting the task complexity and the importance of developing and incorporating medical metrics for learning. This paper represents a step toward data-driven clinical decision support in critical care, and we hope to contribute insights on how to design and verify an offline RL-based medical decision-making system from scratch. Limitations of this work include high-dimensional data and the inherent constraints of offline RL approaches. Our future work includes developing uncertainty-aware offline RL algorithms, incorporating medical metrics into RL training, and evaluations for real-world safety and efficacy before clinical deployment.

Acknowledgment

This work was supported in part by a research grant from Abiomed, Inc., a Johnson & Johnson company, the U.S. Army Research Office under Army-ECASE award W911NF-07-R-0003-03, the U.S. Department Of Energy, Office of Science, IARPA HAYSTAC Program, and NSF Grants #2205093, #2146343, #2134274, CDC-RFA-FT-23-0069, DARPA AIE FoundSci and DARPA YFA.

References

[1] V. Atti, M. A. Narayanan, B. Patel, S. Balla, A. Siddique, S. Lundgren, and P. Velagapudi. A comprehensive review of mechanical circulatory support devices. *Heart International*, 16(1):37–48, Mar. 2022.

[2] E. Buitenwerf, M. F. Boekel, M. I. van der Velde, M. F. Voogd, M. N. Kerstens, G. J. Wietasch, and T. W. Scheeren. The haemodynamic instability score: Development and internal validation of a new rating method of intra-operative haemodynamic instability. *European Journal of Anaesthesiology| EJA*, 36(4):290–296, 2019.

[3] J.-N. Eckardt, K. Wendt, M. Bornhaeuser, and J. M. Middeke. Reinforcement learning for precision oncology. *Cancers*, 13(18):4624, 2021.

[4] T. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

[5] J. D. Hamilton. State-space models. *Handbook of econometrics*, 4:3039–3080, 1994.

[6] M. Janner, J. Fu, M. Zhang, and S. Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

[7] M. Komorowski, L. A. Celi, O. Badawi, A. C. Gordon, and A. A. Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature medicine*, 24(11):1716–1720, 2018.

[8] K. Kuang, F. Dean, J. B. Jedlicki, D. Ouyang, A. Philippakis, D. Sontag, and A. M. Alaa. Med-real2sim: Non-invasive medical digital twins using physics-informed self-supervised learning. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024.

[9] A. Li, Z. Zhou, E. Jortberg, and R. Yu. Forecasting aortic pressure cross-cohort with deep sequence models. In *2022 Computing in Cardiology (CinC)*, volume 498, pages 1–4. IEEE, 2022.

[10] L. E. Lingsch, D. Grund, S. Mishra, and G. Kissas. Fuse: Fast unified simulation and estimation for PDEs. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024.

[11] X. Ma and N. Kallus. Supported value regularization for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 32878–32890, 2022.

[12] N. Prasad, L. F. Cheng, C. Chivers, M. Draugelis, and B. Engelhardt. A reinforcement learning approach to weaning of mechanical ventilation in intensive care units. In *Proceedings of the 3rd Machine Learning for Healthcare Conference (MLHC)*, pages 282–299, 2018.

[13] A. Raghu, M. Komorowski, L. A. Celi, P. Szolovits, and M. Ghassemi. Continuous state-space models for optimal sepsis treatment: a deep reinforcement learning approach. In *Machine Learning for Healthcare Conference*, pages 147–163. PMLR, 2017.

[14] J.-T. Sieweke, D. Berliner, J. Tongers, L. C. Napp, U. Flierl, F. Zauner, J. Bauersachs, and A. Schäfer. Mortality in patients with cardiogenic shock treated with the impella cp microaxial pump for isolated left ventricular failure. *European Heart Journal: Acute Cardiovascular Care*, 9(2):138–148, 2020.

[15] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[16] S. Sun, W. Chen, Z. Zhou, S. Fereidooni, E. Jortberg, and R. Yu. Data-driven simulator for mechanical circulatory support with domain adversarial neural process. In *6th Annual Learning for Dynamics & Control Conference*, pages 1513–1525. PMLR, 2024.

[17] X. Tang, Y. Jia, J. Sun, and Y. Fan. Supervised reinforcement learning with recurrent neural network for dynamic treatment recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pages 2447–2456, 2018.

[18] H.-H. Tseng, Y. Luo, S. Cui, J.-T. Chien, R. K. Ten Haken, and I. E. Naqa. Deep reinforcement learning for automated radiation adaptation in lung cancer. *Medical physics*, 44(12):6690–6705, 2017.

[19] C. Vahdatpour, D. Collins, and S. Goldberg. Cardiogenic shock. *Journal of the American Heart Association*, 8(8):e011991, 2019.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[21] A. Voelker, I. Kajić, and C. Eliasmith. Legendre memory units: Continuous-time representation in recurrent neural networks. *Advances in neural information processing systems*, 32, 2019.

[22] C. Voloshin, H. M. Le, N. Jiang, and Y. Yue. Empirical study of off-policy policy evaluation for reinforcement learning. In *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS Track on Datasets and Benchmarks)*, 2021.

[23] R. Yan, X. Shen, A. Wachi, S. Gros, A. Zhao, and X. Hu. Offline guarded safe reinforcement learning for medical treatment optimization strategies, 2025.

[24] T. Yu, S. Kumar, A. Gupta, et al. Mopo: Model-based offline policy optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 14129–14142, 2020.

[25] M. R. Zhang, T. Le Paine, O. Nachum, C. Paduraru, G. Tucker, Z. Wang, and M. Norouzi. Autoregressive dynamics models for offline policy evaluation and optimization. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, 2021.

A Related Works

Prasad et al. [12] pioneered the application of RL for weaning mechanical ventilation, yet their fitted Q-iteration approach struggled with suboptimal clinical data. SRL-RNN [17] then leveraged recurrent neural networks to capture temporal dependencies for dynamic treatment recommendations, although imitation learning limits performance to clinician-level decisions. Peng et al. OGSRL [23] applied a guarded offline RL for Sepsis treatment. Kuang et al. [8] built patient-specific cardiac hemodynamic twins via physics-informed self supervised learning. Lingsch et al. [10] proposed neural surrogates for PDE forward simulation and inverse parameter estimation on simulated data. In contrast to these methods, our work employs a domain-specific probabilistic transformer-based forecaster trained on real-life data and medical metrics to facilitate the evaluation and development of the offline RL models.

B Medically-informed Metrics

Action Change Penalty (ACP) (similar to [23]): Abrupt and extreme changes in P-level may maximize rewards; however, they can induce physiological instability in a real-world setting. ACP gauges policy volatility and is given by:

$$\text{ACP} = \sum_{i=1}^{T} ||a_{i-1} - a_i||_2, \text{ if } ||a_{i-1} - a_i||_2 > 2.$$

where a_{i-1} is an action at state $i-1$, a_i is a subsequent action, and T is the episode length. Lower ACP values indicate stable physiology and safe weaning, but note that a value of 0 is undesirable as the P-level must be lowered for weaning.

Weaning Score (WS): To capture satisfactory weaning patterns, we support P-level reductions at most every 1 hour when the patient is observed as hemodynamically stable for the past 1 hour:

$$\text{WS} = \frac{\sum_{i=0}^{T-1} \mathbb{I}(\text{MAP}(i) > \tau_{\text{MAP}} \wedge \text{HR}(i) > \tau_{\text{HR}} \wedge \text{Pulsat}(i) > \tau_{\text{Pulsat}}) \cdot \text{Weaned}(i)}{\sum_{i=0}^{T-1} \mathbb{I}\big(\text{MAP}(i) > \tau_{\text{MAP}} \wedge \text{HR}(i) > \tau_{\text{HR}} \wedge \text{Pulsat}(i) > \tau_{\text{Pulsat}}\big)},$$

$$\text{Weaned}(i) = \begin{cases} -1, & \text{if } a_{i+1} - a_i > 0, \\ |a_{i+1} - a_i|, & \text{if } a_{i+1} - a_i \in \{-1, -2\} \\ 0, & \text{otherwise}, \end{cases}$$

where $\tau_{\text{MAP}} = 60, \tau_{\text{HR}} = 50$ and $\tau_{\text{Pulsat}} = 10$, indicating limits of hemodynamic stability (see Table 3 for stability limits). Since our state design represents 1 hour in 10-minute time steps, we calculate the compared MAP value for a state as, $\text{MAP}(i) = \min_{1 \le t \le 6} \text{MAP}(i, t)$, same for HR and pulsatility. T is the episode length and $i = 0$ indicates the initial state. Higher weaning scores denote proper lowering of P-level when at a stable state, and low or negative scores imply that P-level is increased despite having healthy physiological indicators.

Physiological Reward: The reward design in Table 3 is staircase-shaped, which has two drawbacks: non-differentiability and a sparse signal. We reformulate the hemodynamic instability score in the following way.

- **Heart Rate Penalty Function** The heart rate penalty function penalizes deviations from an optimal heart rate of 75 bpm using a quadratic penalty:

$$P_{\text{hr}}(hr) = \text{ReLU}\left(\frac{(hr - 75)^2}{250} - 1\right) \tag{2}$$

  where $\text{ReLU}(x) = \max(0, x)$. This function has zero penalty for heart rates in the range $[50, 100]$ bpm and applies quadratic penalties for heart rates outside this range.

- **Minimum MAP Penalty Function** The minimum Mean Arterial Pressure (MAP) penalty function ensures MAP values remain above 60 mmHg:

$$P_{\text{minMAP}}(MAP) = \text{ReLU}\left(\frac{7(60 - MAP)}{20}\right) \tag{3}$$

	Score Component	Value	Score
Hemodynamic Variable	MAP	≥ 60	0
	50 to 59	1	
	40 to 49	3	
	< 40	7	
Minimum MAP in window	≥ 60	0	
	50 to 59	1	
	40 to 49	3	
	< 40	7	
Time Spent MAP < 60 mmHg (%)	0	0	
	2	1	
	5	3	
	> 5	7	
Pulsatility	> 20	0	
	10-20	5	
	< 10	7	
HR	> 100	3	
	< 50	3	
LVEDP	> 20	7	
	15 to 20	4	
	< 15	3	
CPO	0.6 to 1	1	
	< 0.6	3	
	< 0.5	5	

Table 3: Hemodynamic instability score table from [2]. In our MDP design, we use MAP, pulsatility, and HR score components with slight modification. Because the score indicates risk, for the reward function we multiply the score by -1.

This function applies a linear penalty when MAP falls below 60 mmHg, with the penalty increasing as MAP decreases further from this threshold.

- **Pulsatility Penalty Function** The pulsatility penalty function maintains pulsatility within the range $[20, 50]$:

$$P_{\text{pulsat}}(p) = \text{ReLU}\left(\frac{7(20-p)}{20}\right) + \text{ReLU}\left(\frac{p-50}{20}\right) \tag{4}$$

This bi-directional penalty function penalizes pulsatility values below 20 and above 50, with zero penalty for pulsatility in the range $[20, 50]$.

- **Hypertension Penalty Function** The hypertension penalty function penalizes elevated mean MAP values above 106 mmHg:

$$P_{\text{hyp}}(MAP) = \text{ReLU}\left(\frac{MAP-106}{18}\right) \tag{5}$$

This function applies a linear penalty for mean MAP values exceeding the hypertension threshold of 106 mmHg.

The overall reward function combines all penalty components and negates the sum to create a reward signal:

$$R(s) = -\left[P_{\text{minMAP}}(\min(\text{MAP})) + P_{\text{hyp}}(\overline{\text{MAP}}) + P_{\text{hr}}(\min(HR)) + P_{\text{pulsat}}(\min(\text{Pulsat}))\right] \tag{6}$$

where:

- $\min(\text{MAP})$, $\min(HR)$, $\min(\text{Pulsat})$ are the minimum values over the time horizon
- $\overline{\text{MAP}}$ is the mean MAP over the time horizon
- The negative sign converts penalties into rewards (higher rewards for lower penalties)

C Markov Decision Process (MDP) Design Details for RL

Observations. The observation space includes 12 hemodynamic features of the patient. Our inputs are the pump pressure, pump speed, and motor current 25 Hz signals recorded by the MCS device. Calculated features derived from these signals include Mean aortic pressure (MAP), mean pump speed, mean motor current, mean pump flow, left Ventricular Pressure (LVP), left ventricular end diastolic pressure (LVEDP), heart rate (HR), Systolic blood pressure (SBP), Diastolic blood pressure (DBP), Pulsatility, Relaxation Constant (Tau_LV), and elastance estimation (ESE_LV); denoted as $x_t \in \mathbb{R}^{12}$ for the tth time step for each patient. We take 246 MCS-supported CGS patients as the train set, 57 as validation set, and 76 as the test set. We down-sample patient data from 25Hz to 0.00167Hz (1 sample per 10 minutes) and process them into sliding windows of 1 hour (6 time steps) to be used as states for digital twin prediction and decision making based on expert suggestion. Therefore, the observation space is $\mathcal{S} = \mathbb{R}^{6\times 12}$, where each $s_i = x_{t:t+6}$ at some t for a patient.

In – out horizon	15min – 15min 1 sample / 30s 30ts -> 30ts	1hr – 1hr 1 sample / 5min 12ts -> 12ts	1hr – 1hr 1 sample / 10min 6ts -> 6ts	2hr – 2hr 1 sample / 5min 24ts -> 24ts	2hr – 2hr 1 sample / 10min 12ts -> 12ts
MSE	0.234	0.142 $\pm$ 0.012	**0.124** $\pm$ **0.027**	0.215 $\pm$ 0.009	0.159 $\pm$ 0.006
MAP MSE	3.03	2.711 $\pm$ 0.182	**2.59** $\pm$ **0.154**	3.583 $\pm$ 0.340	3.356 $\pm$ 0.226

Table 4: Alternative settings for the world model. Takeaway: shorter horizon and higher down-sampling produces stronger models, but need at least 1 hour of history to provide reasonable action frequency and physiological context.

Action. The action for our MDP is the pump support level (P-level) of the MCS device. The device operates at 8 different speed levels, from P2-P9, each with a constant motor speed (rpm). The P-level proportionally determines the blood flow provided to the patient by the motor's speed and current. Clinicians can control the P-level while the patient is on support. The P-level generally stays unchanged in 1-hour intervals, unlike the state features, since it is manually controlled by the clinicians during the treatment. In practice, we take the mean P-level over the 1-hour interval as expert action. As a result, we define $\mathcal{A} = \{2, \ldots, 9\}$.

Rewards. The design for the reward function is elaborated in Appendix B, in line with medical consultancy. It assigns a (inverted) risk score based on acceptable intervals for hemodynamic features. The physiological reward is further normalized through Z-score normalization and clipped between $[-2, 2]$ to ensure training stability.

Challenges of Offline RL for MCS The commonly encountered issue of Offline RL is the limited access to the online environment, which results in distribution shift and large value overestimation errors to account for the shift in the real environment. While these are widely studied problems in RL, medical decision-making introduces other problems: error-prone behavioral policies, and highly imbalanced actions in the dataset.

As there is no golden recipe for weaning a patient from an MCS device, the behavioral policy and the clinician policies are naturally imperfect. To this end, we expect offline RL to reveal the true policy from the hemodynamic features. Since it is required to simulate the real environment, we largely rely on a digital twin transformer model. However, the model learns to cheat by outputting cardiac cycles copied from the observation distribution. Furthermore, the action space is by definition fully constant in a state, unlike the observation space, which challenges the model compatibility.

Example weaning. We show two examples of doctors' weaning over the course of 24 hours in Figure 6.

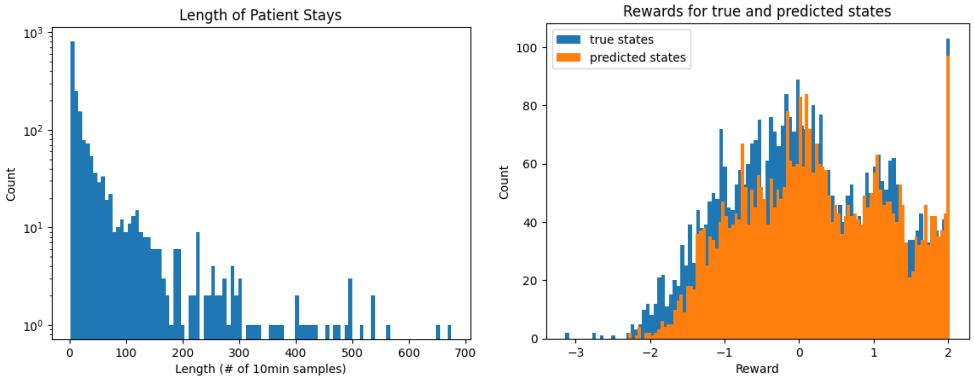

Figure 5: Length distributions of our patient data (left) and the reward score distributions of predicted states versus real patient states.

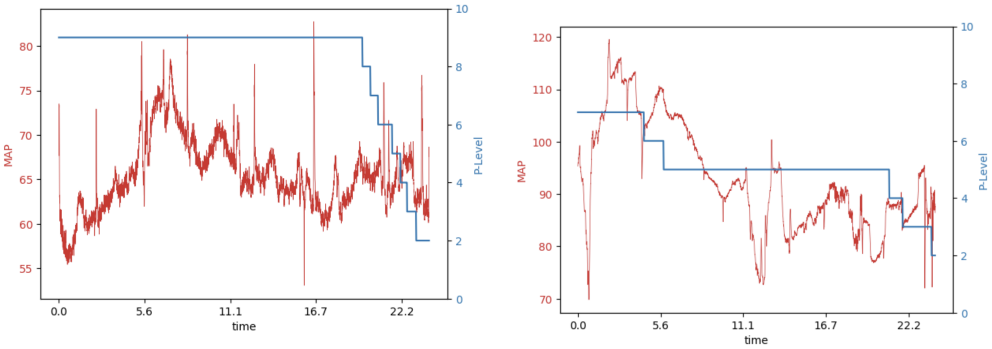

Figure 6: Example weaning of two patients over 24-hour horizon. The P-level change and the patient's MAP (Mean Arterial Pressure) is plotted. We acknowledge that MAP alone is not solely indicative of the patient's well-being; it is plotted in order to show part of the decision making process for weaning.

D Digital Twin Experiment Details

D.1 Baselines

We evaluate our approach against several established baselines for probabilistic dynamics modeling. Each baseline is configured with carefully tuned hyperparameters to ensure fair comparison:

- **Multi-Layer Perceptrons (MLPs)** with Monte Carlo dropout approximate probabilistic forecasts by treating dropout as a Bayesian approximation technique, enabling uncertainty quantification through multiple forward passes during inference. The MLP baseline employs a three-layer architecture with hidden dimensions $[512, 256, 128]$, ReLU activation functions, and a dropout rate of 0.2 applied after each hidden layer. The network flattens the input sequence and concatenates it with p-level control signals before processing through the fully connected layers.
- **Neural Processes** [16] is a meta-learning approach that conditions on context observations to predict distributions over functions, enabling few-shot adaptation to new dynamical systems while maintaining uncertainty quantification. The implementation features a latent dimension of 128, a hidden dimension of 256, and employs separate encoder networks for context processing with three-layer architectures. The context encoder processes input features augmented with time indices, while the aggregator combines encoded representations across time steps. The decoder network generates both mean and variance predictions for each feature at each forecast timestep.

- **Conditional Legendre Memory Units (CLMUs)** [9, 21] leverage orthogonal polynomial basis functions to capture long-term temporal dependencies through structured memory mechanisms. The CLMU baseline utilizes 2 layers with memory dimension 64, hidden dimension 128, and incorporates p-level conditioning through a dedicated projection layer. Each LMU layer employs Legendre polynomial transition matrices with scaling parameter $\theta = 1.0$ and applies exponential smoothing with decay rate 0.9 for stable memory updates. The output projection includes dropout with a rate 0.1 for regularization.
- **State Space Models (SSMs)** [5] represent dynamics through latent state evolution governed by linear or nonlinear transition functions, naturally incorporating temporal dependencies and enabling principled probabilistic inference over hidden states. The SSM baseline operates with state dimension 64, hidden dimension 128, and forecast horizon of 6 steps. The state transition matrix is initialized as $0.9 \cdot I + 0.1 \cdot \mathcal{N}(0, 1)$ to ensure stability, while the observation model employs a two-layer network with ReLU activation and dropout rate 0.1. Stochastic sampling is achieved by injecting Gaussian noise with standard deviation 0.01 during state transitions.
- **Transformers with sinusoidal positional embeddings (TDT sin.)** [20] and **Transformers with rotary positional embeddings (TDT rot.)** [15]. TDT (rot.) leverages self-attention mechanisms enhanced with rotary position encoding (RoPE) that captures relative positional relationships through multiplicative rotations. The transformer model's attention mechanism allows the model to attend to relevant temporal patterns and input control, improving the time series prediction. Both transformer models have the architecture outlined in our methodology section.

All baselines are trained using the Adam optimizer with a learning rate 0.001 and employ Monte Carlo sampling with 50 forward passes for uncertainty quantification during inference.

D.2 Baseline Hyperparameters and Training Configuration

The training setup utilizes a batch size of 64, conducted over 50 epochs. For robust statistical analysis, the model is run three times. Additionally, uncertainty quantification is performed using 50 evaluation samples.

D.2.1 MLP with Dropout Baseline

Architecture

$$\begin{aligned} \text{Input Dimension} &= 6 \times 12 + 6 = 78 \text{ (flattened sequence + p-levels)} \\ \text{Hidden Layers} &= [512, 256, 128] \\ \text{Output Dimension} &= 66 \end{aligned}$$

Parameter	Value
Hidden Dimensions	[512, 256, 128]
Dropout Rate	0.2
Activation Function	ReLU
Learning Rate	0.001
Optimizer	Adam
Loss Function	MSE
Training Epochs	50
Batch Size	64

Table 5: MLP Dropout Baseline Hyperparameters

D.2.2 Neural Process Baseline

Architecture The Neural Process implements a variational approach with encoder-aggregator-decoder structure.

Parameter	Value
Input Dimension	12
Latent Dimension	128
Hidden Dimension	256
Learning Rate	0.001
Optimizer	Adam
Loss Function	MSE
Training Epochs	50
Batch Size	64

Table 6: Neural Process Baseline Hyperparameters

- **Context Encoder**: $(d_{input} + 1) \to 256 \to 256 \to 2 \times d_{latent}$
- **Aggregator**: $d_{latent} \to 256 \to 2 \times d_{latent}$
- **Decoder**: $(d_{latent} + d_{input} + 1) \to 256 \to 256 \to \frac{d_{output}}{6} \times 2$

Where $d_{input} = 12$, $d_{latent} = 128$, and $d_{output} = 66$.

D.2.3 Conditional Legendre Memory Unit (CLMU) Baseline

Architecture The CLMU implements a memory-based approach using Legendre polynomials for sequence modeling.

Parameter	Value
Input Dimension	12
Memory Dimension	64
Hidden Dimension	128
Number of Layers	2
Forecast Horizon	6
Dropout Rate	0.1
Learning Rate	0.001
Optimizer	Adam
Loss Function	MSE
Training Epochs	50
Batch Size	64

Table 7: CLMU Baseline Hyperparameters

- **Legendre Memory Units**: 2 layers with memory dimension 64
- **Condition Projection**: $6 \to 128$ (for p-level conditioning)
- **Hidden Layers**: $(64 + 128) \to 64$ per layer
- **Output Projection**: $64 \to 128 \to 66$ with ReLU and Dropout(0.1)

D.2.4 State-Space Model Baseline

The State-Space Model implements linear dynamical systems for time-series forecasting.

- **State Transition**: $A : 64 \to 64$ (initialized as $0.9 \times I + 0.1 \times \mathcal{N}(0, 1)$)
- **Input Mapping**: $B : 12 \to 64$
- **Control Projection**: $6 \to 64$
- **Observation Model**: $64 \to 128 \to 128 \to 66$ with ReLU and Dropout(0.1)
- **State Update**: $s_t = \tanh(As_{t-1} + Bx_t + \text{control})$

Parameter	Value
Input Dimension	12
State Dimension	64
Hidden Dimension	128
Forecast Horizon	6
Dropout Rate	0.1
Learning Rate	0.001
Optimizer	Adam
Loss Function	MSE
Training Epochs	50
Batch Size	64

Table 8: State-Space Model Baseline Hyperparameters

D.2.5 Standard Transformer

Parameter	Value
Number of Features	12
Model Dimension	256
Number of Heads	8
Encoder Layers	3
Decoder Layers	2
Encoder Dropout	0.1
Decoder Dropout	0.0
Max Sequence Length	100
Forecast Horizon	6
Learning Rate	0.001
Optimizer	Adam
Loss Function	MSE
Training Epochs	35
Batch Size	64

Table 9: Standard Transformer Hyperparameters

D.2.6 Rotary Transformer

The Rotary Transformer uses identical hyperparameters to the standard transformer, with the replacement of rotary positional embeddings as the input embedding layer.

D.3 Hyperparameter Search Configuration

The hyperparameter search was conducted using Weights & Biases with the following search space:

D.4 Metrics for evaluating digital twin

- MAE All: Mean Absolute Error across all 12 features.
- MAE MAP: Mean Absolute Error for MAP (Mean Arterial Pressure) only. Although MAP is not solely indicative of patient's well-being, it provides a guide towards the expressivity and responsiveness of the model, as other features can be more static.
- MAE Static: MAE for samples with non-changing P-Levels over the course of 2 hours. This scenario constructs 72% of our dataset.
- MAE Dynamic: MAE for samples with dynamic P-Levels. Dynamic P-Level refers to when there is at least 1 change in the P level over the 2 hour course of input-output pairs.

Parameter	Values
Model Dimension	[128, 256, 512]
Number of Heads	[4, 8, 16]
Encoder Layers	[2, 3, 4]
Decoder Layers	[1, 2, 3]
Encoder Dropout	[0.0, 0.1, 0.2]
Decoder Dropout	[0.0, 0.1, 0.2]
Model Type	[transformer, rotary_transformer]
Batch Size	[32, 64, 128]
Learning Rate	[0.0001, 0.001, 0.01]
Training Epochs	[35]

Table 10: Transformer model Hyperparameter Search Space

- Trend Acc: Predicted trend direction accuracy for MAP. Trend is classified as (1) increasing if the slope of MAP over the predicted horizon (1 hr) is ≥ 2, (2) decreasing if the slope ≤ 2, and (3) flat otherwise. Trend is an important factor that medical professionals take into consideration, being able to predict trend well shows medical saliency of the model.
- CRPS: Continuous Ranked Probability Score is a proper scoring rule [4] for uncertainty quantification, calculated from 50 samples from the probabilistic predictions (samples x, x') and the ground truth y as in equation 7.

$$\text{CRPS}(\mathcal{F}, y) = \int \left(\mathcal{F}(x) - \mathbf{1}\{x \geq y\}\right)^2 dx = \mathbb{E}_x[|x - y|] - \frac{1}{2}\mathbb{E}_{x,x'}[|x - x'|] \tag{7}$$

D.5 Additional Visualizations.

Please see figure 7 for further qualitative examples of digital twin models.

E RL Experiment Settings

We explain the reinforcement learning baseline models and our implementation details in this section. We train all baselines on our dataset of size 17865 with the observation space of dimension 72 and action dimension of 1.

- **Behavior Cloning (BC)** uses a supervised learning model, $\pi(s, a)$, to mimic the expert (s, a) while minimizing the MSE loss as $\arg\min_\theta \ \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\|a - \pi_\theta(s)\|_2^2\right]$ without exploration unlike on Eq. 1. It ignores the dynamics of (s, a), suffering from the distribution shift in the online environment. A 3-layer MLP with hidden dimensions of [256,256,256], ReLU activation between the layers, and tanh in the output layer is our BC model. The last layer has one perceptron, indicating our action dimension as 1. We trained the model for 30000 steps with Adam optimizer using a learning rate of 1e-3 and batch size 256.
- **Model-Based Policy Optimization (MBPO)** [6] fits a dynamics model $p_\theta(s', r \mid s, a)$ and then optimizes the policy on real data augmented with short-horizon rollouts from p_θ. Our implementation of MBPO is the non-penalized version of MOPO with the same parameters in Table 11 with the reward penalty coefficient as 0.0.
- **Model-based Offline Policy Optimization (MOPO)** [24] borrows the same policy optimization by penalizing rewards with the uncertainty of the learned dynamics model during training. MOPO trains a soft actor-critic network on a replay buffer augmented with short model rollouts. While the performance depends on the precision of the uncertainty quantification method and fine-calibration of the penalty weight, it shows robust results in stochastic data. However, it might suffer from over-penalization with suboptimal hyperparameter settings. We tuned the default parameters of the MOPO implementation of the authors to produce stable and convergent training. All hyperparameters are depicted in Table 11.

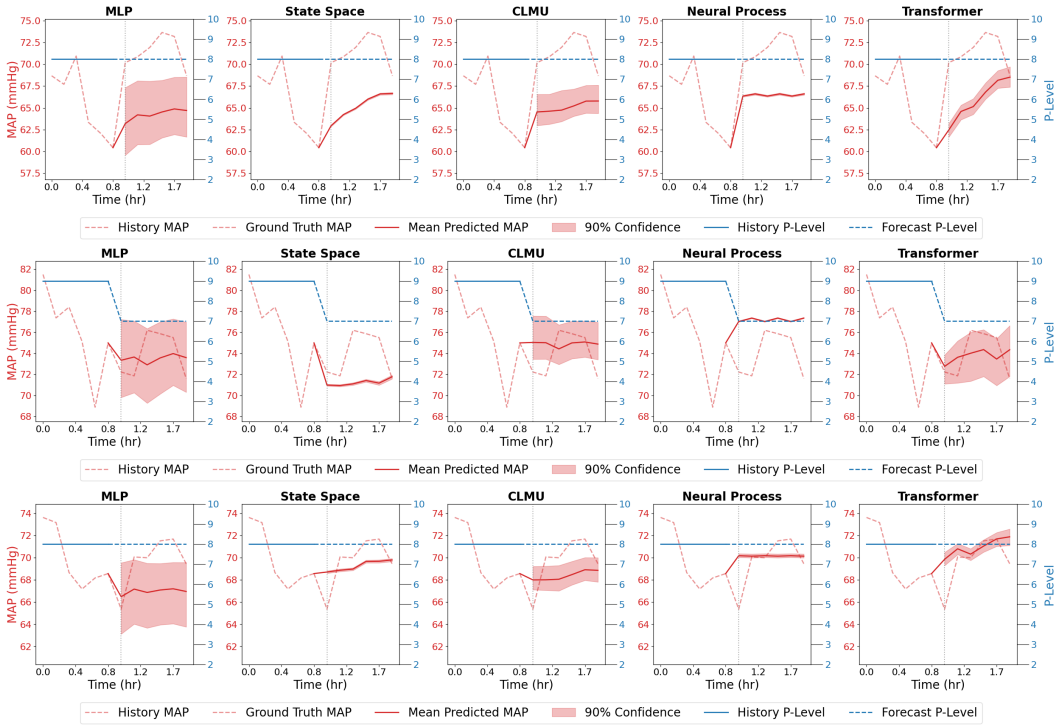

Figure 7: Digital twin prediction visualization compared with baselines. The Transformer model is more accurate in reflecting response to P-level change and more expressive when capturing large changes in patient state, resulting in its higher accuracy.

SAC includes an actor and 2 critic networks (for double clipped Q-learning) of the same architecture: 2 layers of MLP with sizes [256, 256], all trained with Adam optimizer. The transition model has 7 ensemble MLP models of 4 layers with sizes [200, 200, 200, 200] and Swish activation function.

Table 11: Hyperparameters of our MOPO implementation.

Parameters	**Value**
Actor learning rate	3×10^{-4}
Critic learning rate	3×10^{-4}
Discount factor (γ)	0.99
Target network update coefficient (τ)	0.005
Target entropy (often $-$action dimension)	-1
Temperature optimizer learning rate	3×10^{-4}
Dynamics model learning rate	1×10^{-3}
Dynamics ensemble size	7
Holdout ratio	0.2
Training epochs	100
Steps per epoch	1000
Evaluation episodes	1000
Mini-batch size	256
Reward penalty coefficient	1.0
Model rollout horizon	5
Rollout batch size	10000
Rollout frequency	1000
Real-to-model data sampling ratio	0.05

- **Support Value Regularization (SVR)** [11] adds a support-aware penalty to the Q-value objective, pushing down estimates for OOD (s, a) pairs so the policy stays within the dataset support and avoids extrapolation error. Since the regularization weights come from importance sampling based on BC, it over-regularizes the objective in the case of distribution shift. The code implementation is borrowed from the authors with some hyperparameter changes. The BC pretraining is identical to the introduced BC baseline. The SVR model includes the soft actor-critic network, the actor being in the same architecture as BC, and the critic having 4 MLP heads of size [356,256]. SVR optimizes the support value regularized temporal difference with a weight multiplier of 0.006. The regularization term suppressed the OOD state action pairs with the importance sampling weight, where we use 0.5 as the sample standard deviation. The minimum Q value for the regularization term is calculated from the replay buffer directly, resulting in -200. We train all networks for 100000 timesteps with the Adam optimizer on a learning rate of 0.0001. Actor is optimized every 2 timesteps. τ and γ are the same as in MOPO.