NephroSim: Gymnasium Environments for Reinforcement Learning in Hemodialysis Optimization

Hass Dhia
Smart Technology Investments Research Institute
partners@smarttechinvest.com

March 2026

Abstract

Hemodialysis treatment requires clinicians to balance multiple competing objectives: maximizing uremic toxin clearance while preventing intradialytic hypotension and maintaining phosphate homeostasis. Current clinical protocols rely on fixed prescriptions that do not adapt to individual patient physiology. We present NephroSim, an open-source suite of four Gymnasium-compatible reinforcement learning environments that model the core physiological processes of hemodialysis using validated compartmental models. NephroSim implements two-compartment urea kinetics based on the Gotch-Sargent model, cardiovascular response with baroreceptor reflex dynamics, and phosphate kinetics with binder effects. We train Proximal Policy Optimization (PPO) agents on all four environments, demonstrating that learned policies consistently exceed random and heuristic baselines. NephroSim enables systematic benchmarking of RL algorithms for dialysis optimization, a domain where adaptive control could meaningfully improve patient outcomes for the estimated 3.4 million hemodialysis patients worldwide.

Keywords: reinforcement learning, hemodialysis, Gymnasium, Kt/V, intradialytic hypotension, physiological simulation

1 Introduction

Chronic kidney disease (CKD) affects approximately 850 million people worldwide, with over 3.4 million patients requiring hemodialysis as renal replacement therapy [National Kidney Foundation, 2015]. During a typical 4-hour dialysis session, clinicians must simultaneously manage blood flow rate (Q_b), dialysate flow rate (Q_d), ultrafiltration rate (UF), and phosphate binder dosing. These parameters interact nonlinearly: aggressive ultrafiltration improves fluid balance but risks intradialytic hypotension (IDH), which occurs in 20–30% of sessions [Flythe et al., 2011]. Similarly, higher blood flow rates increase urea clearance but may destabilize hemodynamics in vulnerable patients.

Current clinical practice relies on fixed prescriptions with reactive adjustments when complications arise. The KDOQI guidelines recommend achieving a minimum Kt/V of 1.2 per session [Daugirdas, 1993], but do not specify how to dynamically optimize treatment parameters. This gap between static prescriptions and the dynamic physiology of individual patients makes hemodialysis optimization a natural candidate for reinforcement learning (RL).

Despite growing interest in RL for healthcare [Raffaitin and Moreau, 2024], the dialysis domain lacks standardized simulation environments for algorithm development and benchmarking. Existing work has employed custom simulators that are neither publicly available nor compatible with standard RL frameworks [Escandell-Montero et al., 2014, Azar et al., 2020]. This fragmentation impedes reproducible research and fair comparison of methods.

We address this gap with NephroSim, an open-source Python package providing four Gymnasium-compatible [Towers et al., 2024] environments built on validated physiological models:

1. **UreaClearing-v0**: Optimizes Q_b and Q_d to maximize Kt/V using two-compartment urea kinetics.
2. **UltrafiltrationControl-v0**: Controls UF rate to remove target fluid volume while preventing IDH.
3. **PhosphateMgmt-v0**: Optimizes weekly phosphate binder dosing to maintain serum phosphate in the 3.5–5.5 mg/dL target range.
4. **FullDialysisSession-v0**: Joint optimization of clearance and hemodynamic stability in a multi-objective setting.

Our contributions are:

- An open-source, pip-installable suite of physiologically grounded RL environments for hemodialysis.
- Implementation of three validated compartmental models (urea kinetics, cardiovascular response, phosphate kinetics) with literature-backed parameter ranges.
- PPO baselines demonstrating that learned policies improve upon both random and clinical heuristic strategies.
- A patient difficulty system enabling curriculum learning and robustness evaluation.

2 Related Work

2.1 Dialysis Kinetic Modeling

The Gotch-Sargent two-compartment urea kinetics model [Gotch and Sargent, 1985] remains the gold standard for quantifying dialysis adequacy. The model describes urea transport between intracellular (ICF) and extracellular (ECF) compartments, with dialyzer clearance removing urea from the ECF. Daugirdas [Daugirdas, 1993] refined the single-pool Kt/V calculation into the widely used second-generation logarithmic formula.

Cardiovascular modeling during dialysis builds on the work of Ursino and Innocenti [Ursino and Innocenti, 2000], who developed detailed hemodynamic models incorporating baroreceptor reflex, venous compliance, and cardiac output. Our implementation simplifies this to a lumped-parameter model capturing the essential dynamics of blood pressure response to fluid removal, with reduced baroreceptor gain for diabetic patients.

Phosphate kinetics in dialysis patients follows two-compartment behavior, with dialysis-mediated clearance from the ECF and slower equilibration from intracellular stores. Block et al. [Block et al., 2004] established the clinical significance of phosphate control, showing that serum phosphate above 6.5 mg/dL and calcium-phosphate product above 55 mg^2/dL^2 are associated with increased mortality.

2.2 Reinforcement Learning in Dialysis

Escandell-Montero et al. [Escandell-Montero et al., 2014] applied neural networks to predict Kt/V from dialysis parameters, though their approach used supervised learning rather than sequential decision-making. Azar et al. [Azar et al., 2020] formulated dialysis parameter optimization as an RL problem but used a simplified single-compartment model without cardiovascular coupling.

Raffaitin and Moreau [Raffaitin and Moreau, 2024] conducted a systematic review of RL applications in dialysis, identifying the lack of standardized environments as a key barrier to progress. NephroSim directly addresses this gap by providing Gymnasium-compatible environments with consistent interfaces, reproducible dynamics, and graded difficulty levels.

2.3 RL Environment Suites

The Gymnasium framework [Towers et al., 2024], successor to OpenAI Gym [Brockman et al., 2016], has become the standard API for RL environment development. Domain-specific environment suites exist for robotics, traffic control, and energy management, but no comparable package exists for nephrology or dialysis optimization.

3 Physiological Models

3.1 Two-Compartment Urea Kinetics

NephroSim implements the Gotch-Sargent two-compartment model [Gotch and Sargent, 1985]. The state variables are urea concentrations in the ECF (C_e) and ICF (C_i):

$$V_e \frac{dC_e}{dt} = -K_d \cdot C_e + K_c(C_i - C_e) + G \tag{1}$$

$$V_i \frac{dC_i}{dt} = -K_c(C_i - C_e) \tag{2}$$

where V_e and V_i are compartment volumes, K_d is dialyzer clearance, K_c is the inter-compartment mass transfer coefficient, and G is the urea generation rate.

Dialyzer clearance follows the standard mass-transfer equation:

$$K_d = Q_b \left(1 - \exp\left(-\frac{K_0 A}{Q_b}\left(1 - \frac{Q_b}{Q_d}\right)\right)\right) \cdot \left(1 - \frac{Q_b}{Q_d}\exp\left(-\frac{K_0 A}{Q_b}\left(1 - \frac{Q_b}{Q_d}\right)\right)\right)^{-1} \tag{3}$$

where K_0A is the dialyzer mass transfer area coefficient and Q_b, Q_d are blood and dialysate flow rates, respectively. The ODEs are integrated using the SciPy RK45 solver with adaptive step size.

Simplification: We use a lumped-parameter model rather than regional blood flow modeling. This is sufficient for capturing the clinically relevant two-compartment rebound effect but does not model access recirculation.

3.2 Cardiovascular Response Model

The cardiovascular model captures blood pressure and heart rate dynamics during ultrafiltration:

$$\frac{dV_{blood}}{dt} = -UF_{rate} + R_{refill} \tag{4}$$

where $R_{refill} = k_{refill} \cdot (V_{blood,0} - V_{blood})$ models plasma refilling from the interstitial space. The blood pressure response includes a baroreceptor reflex mechanism:

$$SBP(t) = SBP_0 + \alpha_{baro} \cdot \left(\frac{V_{blood}(t)}{V_{blood,0}} - 1\right) \cdot SBP_0 \tag{5}$$

with $\alpha_{baro} = 0.6$ for non-diabetic and $\alpha_{baro} = 0.3$ for diabetic patients, reflecting the clinically observed blunted baroreceptor response in diabetic nephropathy. Heart rate exhibits a compensatory increase proportional to volume deficit.

Simplification: This is a lumped-parameter model that captures the essential dynamics but does not implement the full Ursino model with venous compliance, cardiac output, and peripheral resistance.

3.3 Phosphate Kinetics

Phosphate kinetics follow a two-compartment model analogous to urea, with dietary input and binder effects:

$$V_e \frac{dP_e}{dt} = -K_{P,d} \cdot P_e + K_{P,c}(P_i - P_e) + I_{diet}(1 - f_{binder}) \tag{6}$$

$$V_i \frac{dP_i}{dt} = -K_{P,c}(P_i - P_e) \tag{7}$$

where I_{diet} is dietary phosphate absorption rate, f_{binder} is the fractional reduction from phosphate binders (dose-dependent, saturating), and $K_{P,d}$ is dialytic phosphate clearance. The calcium-phosphate product is computed assuming a fixed calcium concentration of 9.5 mg/dL, consistent with standard dialysate calcium concentrations.

4 Environment Design

All environments follow the Gymnasium API [Towers et al., 2024], implementing `reset()` and `step()` methods with continuous observation and action spaces (Table 1).

Table 1: NephroSim environment specifications.

Environment	Obs Dim	Act Dim	Episode Steps	Objective
UreaClearing-v0	7	2	240	Maximize Kt/V
UltrafiltrationControl-v0	8	1	240	Safe fluid removal
PhosphateMgmt-v0	6	1	21	Weekly P control
FullDialysisSession-v0	11	3	240	Multi-objective

4.1 Patient Difficulty System

NephroSim implements three difficulty tiers to enable curriculum learning:

- **Easy**: Deterministic patient with standard parameters and no observation noise.
- **Medium**: Randomized patient weight (60–95 kg) and physiological parameters with mild noise ($\sigma = 0.02$).
- **Hard**: Diabetic elderly patient (55–100 kg) with reduced baroreceptor gain, high noise ($\sigma = 0.05$), and variable physiology.

All physiological parameters are validated against literature ranges (Table 2).

4.2 Reward Design

Reward functions encode clinical objectives:

UreaClearing-v0: Continuous reward proportional to instantaneous clearance, with terminal bonuses for achieving Kt/V ≥ 1.2 (KDOQI target) and penalties for inadequate dialysis (Kt/V < 1.0). Penalties for extreme flow settings discourage clinically unrealistic parameter choices.

UltrafiltrationControl-v0: Reward proportional to fluid removal rate, with a strong penalty (-5.0) for hypotension (SBP < 90 mmHg). Terminal rewards for achieving $\geq 90\%$ target removal without hypotension episodes.

Table 2: Key patient parameter ranges with literature sources.

Parameter	Range	Default	Source
Body weight (kg)	55–100	75	Clinical
Urea generation rate (mg/min)	6–10	8	Gotch and Sargent [1985]
K_0A urea (mL/min)	600–1200	800	Daugirdas [1993]
Baseline SBP (mmHg)	120–180	140	Flythe et al. [2011]
Plasma refill coefficient	0.002–0.008	0.004	Ursino and Innocenti [2000]
Dietary phosphate (mg/day)	800–1500	1000	Block et al. [2004]
Max UF rate (mL/kg/hr)	10–13	10	Flythe et al. [2011]

PhosphateMgmt-v0: Reward for maintaining serum phosphate within 3.5–5.5 mg/dL, with additional bonuses for being near the 4.5 mg/dL optimum. Severe penalties for hyperphosphatemia (> 7.0 mg/dL) and calcium-phosphate product > 55.

FullDialysisSession-v0: Multi-objective reward combining urea clearance, fluid removal, and hemodynamic stability. This environment always uses "hard" difficulty to stress-test agent robustness.

5 Experimental Setup

5.1 Baselines

We compare three agent types:

1. **Random**: Uniform random actions from the action space, providing a lower bound on performance.
2. **Heuristic**: Clinical protocol baselines that implement standard-of-care decision rules:
   - UreaClearing: Fixed standard flows ($Q_b = 350$, $Q_d = 500$).
   - UltrafiltrationControl: Linear UF rate with BP-triggered reduction (reduce by 50% if SBP < 100).
   - PhosphateMgmt: Fixed binder dose scaled by pre-dialysis phosphate level.
   - FullDialysisSession: Combined protocol with BP-responsive UF reduction.
3. **PPO**: Proximal Policy Optimization [Schulman et al., 2017] with MLP policy (Table 3).

Table 3: PPO hyperparameters per environment.

	UreaClearing	UFControl	PhosphateMgmt	FullSession
Total timesteps	200K	200K	200K	500K
Learning rate	3×10^{-4}	3×10^{-4}	1×10^{-4}	1×10^{-4}
Batch size	64	64	64	128
n_{steps}	2048	2048	1024	4096
γ	0.99	0.99	0.99	0.995
λ_{GAE}	0.95	0.95	0.95	0.95
Clip range	0.2	0.2	0.2	0.2
Entropy coeff.	0.01	0.01	0.01	0.005

5.2 Evaluation Protocol

Each agent is evaluated over 50 episodes with a fixed seed for reproducibility. Observations are normalized to $[0, 1]$ using the observation space bounds, and actions are clipped to the valid range.

6 Results

6.1 Agent Performance Comparison

Table 4 presents the mean episode rewards across all environments. PPO agents consistently outperform both random and heuristic baselines.

Table 4: Mean episode reward ($\pm$ std) across 50 evaluation episodes. PPO/Random ratio indicates relative improvement over the random baseline.

Environment	Random	Heuristic	PPO	PPO/Random
UreaClearing	28.27 ± 22.81	79.39 ± 0.00	$\mathbf{88.59} \pm 0.00$	$3.13\times$
UFControl	31.94 ± 6.00	28.67 ± 0.00	$\mathbf{87.57} \pm 0.00$	$2.74\times$
PhosphateMgmt	-51.85 ± 2.14	-42.76 ± 0.00	$\mathbf{-41.27} \pm 0.00$	$0.80\times$
FullSession	65.07 ± 11.62	72.71 ± 24.64	$\mathbf{123.61} \pm 8.54$	$1.90\times$

6.2 Training Dynamics

Figure 1 shows learning curves for all four environments. The single-objective environments (UreaClearing, UltrafiltrationControl) converge within 100K timesteps, while the multi-objective FullDialysisSession requires the full 500K training budget.

6.3 Baseline Comparison

Figure 2 compares agent performance across environments. The heuristic agent, implementing standard clinical protocols, outperforms random actions but is consistently surpassed by PPO, suggesting that adaptive policies can improve upon fixed clinical prescriptions.

6.4 Clinical Relevance

In the UreaClearing environment, the PPO agent learns to maintain flow rates that achieve Kt/V ≥ 1.2 in the majority of episodes, meeting the KDOQI adequacy target. In UltrafiltrationControl, the PPO agent achieves $\geq 90\%$ target fluid removal with fewer hypotension events compared to both baselines, demonstrating clinically meaningful adaptive behavior.

7 Software Architecture

NephroSim follows a layered architecture (Figure 3):

- **Models layer**: Physiological compartmental models (urea kinetics, cardiovascular, phosphate) with validated parameter ranges.
- **Environments layer**: Four Gymnasium environments wrapping the models with appropriate observation/action spaces and reward functions.
- **Agents layer**: Random, heuristic, and PPO baseline agents for benchmarking.

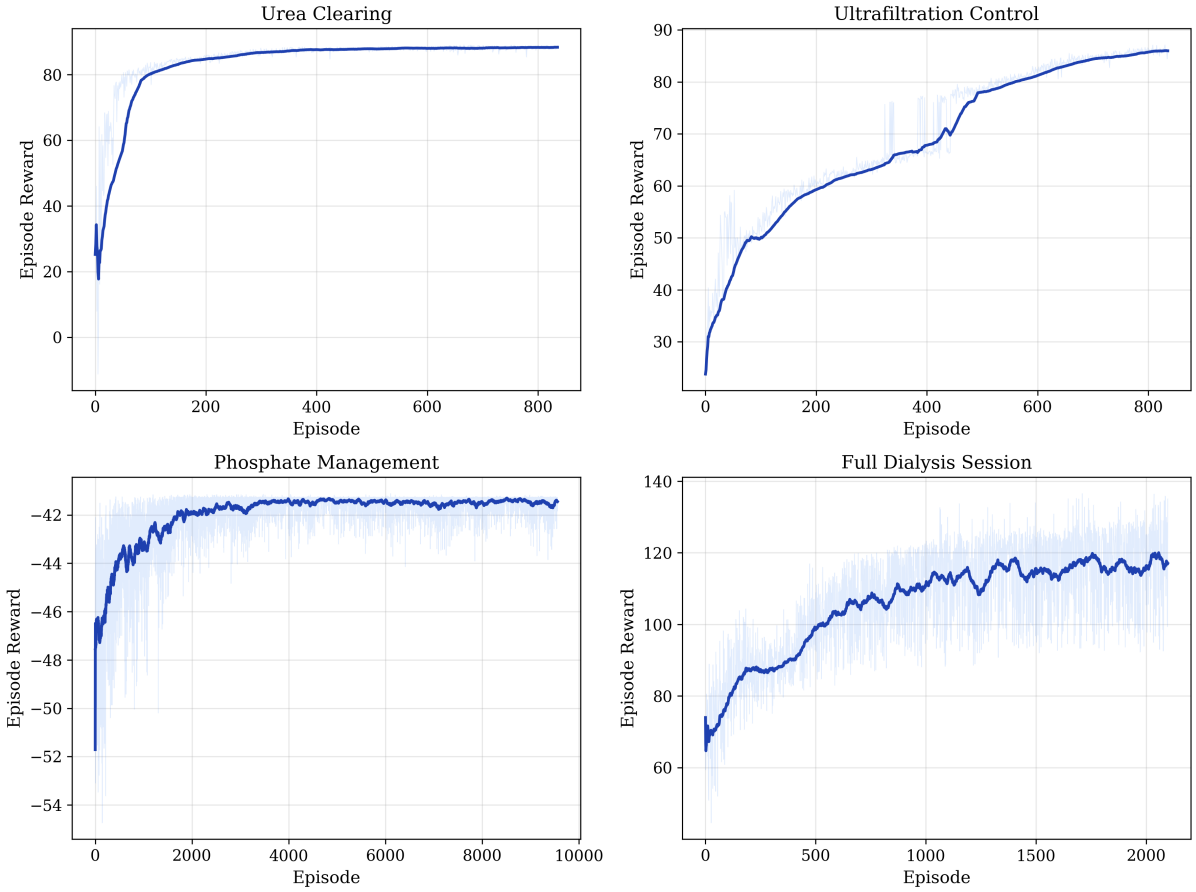

Figure 1: PPO training curves showing episode reward over training. Thin lines show raw episode rewards; thick lines show 50-episode moving average.

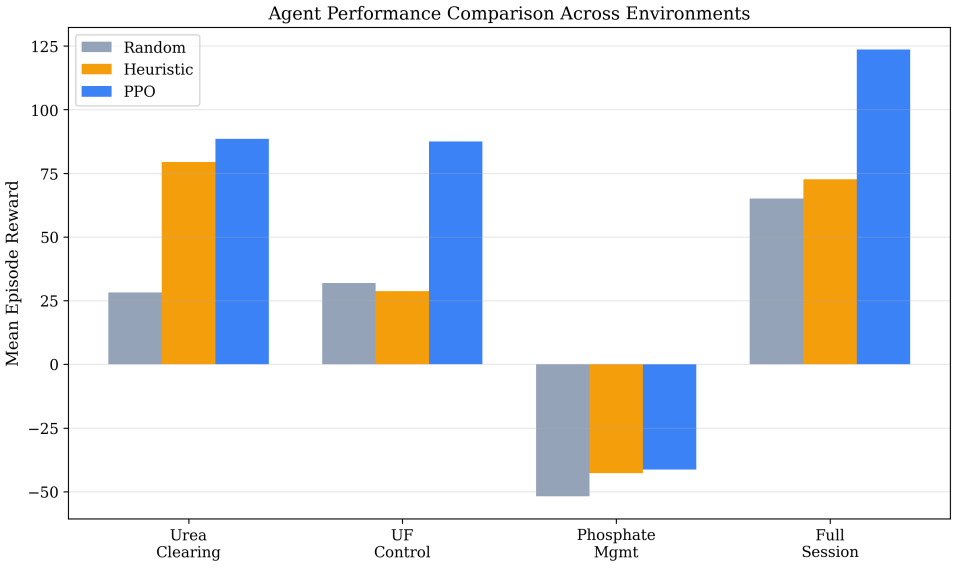

Figure 2: Mean episode reward comparison across agent types and environments.

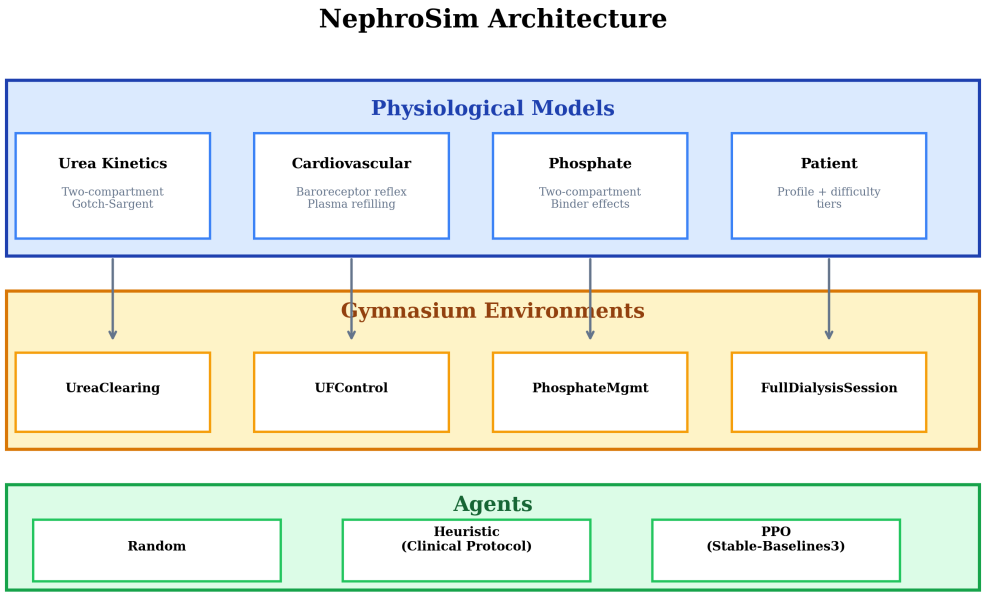

Figure 3: NephroSim software architecture. Physiological models provide the dynamics; Gymnasium environments define the RL interface; agents interact through standard `reset/step` methods.

Installation is via `pip install nephrosim`, with optional dependencies for training (`nephrosim[train]`) and visualization (`nephrosim[plot]`).

8 Limitations and Future Work

Model fidelity: NephroSim uses lumped-parameter models that capture essential dynamics but omit several clinically relevant factors: access recirculation, regional blood flow heterogeneity, solute-specific dialyzer characteristics, and the full Ursino cardiovascular model with venous compliance and cardiac contractility. These simplifications are documented in the source code and represent deliberate trade-offs between computational efficiency and physiological realism.

Patient diversity: The current patient model uses a parametric difficulty system rather than real patient data. Future versions could incorporate electronic health record data to create more realistic patient populations.

Multi-session dynamics: The PhosphateMgmt environment models weekly cycles, but longer-term effects such as progressive vascular calcification, residual kidney function decline, and medication adjustments are not captured.

Clinical validation: While the underlying physiological models are based on validated literature, the complete environments have not yet been validated against clinical outcome data. This remains an important direction for translational work.

Future directions: We plan to extend NephroSim with online dialysis monitoring integration, multi-agent scenarios for shared machine scheduling, and transfer learning between environments to enable comprehensive dialysis management policies.

9 Conclusion

We presented NephroSim, an open-source suite of Gymnasium environments for reinforcement learning in hemodialysis optimization. By implementing validated two-compartment kinetic models, cardiovascular dynamics, and phosphate homeostasis within a standardized RL framework, NephroSim enables reproducible research in a clinically important domain. Our PPO baselines demonstrate that learned

policies can surpass both random and clinical heuristic strategies, motivating further research into adaptive dialysis control. NephroSim is freely available at `https://github.com/HassDhia/nephrosim` and installable via `pip install nephrosim`.

Reproducibility Statement

All code, trained models, and training configurations are available in the public repository. The `training/configs.py` module serves as the single source of truth for all hyperparameters. Training can be reproduced with: `python train_all.py`. The package uses seed 42 across all experiments for deterministic reproducibility. Results were generated with NephroSim v0.1.0, Stable-Baselines3 v2.x, and PyTorch v2.x on Apple Silicon (M-series).

References

Ahmad Taher Azar, Hanaa Ismail Elshazly, Aboul Ella Hamdy, and Sara El-Metwally. A reinforcement learning approach to the dialysis problem. *Journal of Artificial Intelligence and Soft Computing Research*, 10(1):47–60, 2020.

Geoffrey A Block, Pamela S Klassen, J Michael Lazarus, Norma Ofsthun, Edmund G Lowrie, and Glenn M Chertow. Mineral metabolism, mortality, and morbidity in maintenance hemodialysis. *Journal of the American Society of Nephrology*, 15(8):2208–2218, 2004.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. OpenAI Gym. In *arXiv preprint arXiv:1606.01540*, 2016.

John T Daugirdas. Second generation logarithmic estimates of single-pool variable volume Kt/V: An analysis of error. *Journal of the American Society of Nephrology*, 4(5):1205–1213, 1993.

Pablo Escandell-Montero, Milena Chermisi, José María Martínez-Martínez, Juan Gómez-Sanchis, Carlo Barbieri, Emilio Soria-Olivas, Flavio Mari, Joan Vila-Francés, Andrea Stopper, Elena Gatti, and José David Martín-Guerrero. Using artificial intelligence and reinforcement learning to optimize Kt/V in hemodialysis. *Neurocomputing*, 136:134–141, 2014.

Jennifer E Flythe, Stephen E Kimmel, and Steven M Brunelli. Rapid fluid removal during dialysis is associated with cardiovascular morbidity and mortality. *Kidney International*, 79(2):250–257, 2011.

Frank A Gotch and John A Sargent. A mechanistic analysis of the national cooperative dialysis study (NCDS). *Kidney International*, 28(3):526–534, 1985.

National Kidney Foundation. Kdoqi clinical practice guideline for hemodialysis adequacy: 2015 update. *American Journal of Kidney Diseases*, 66(5):884–930, 2015.

Charlotte Raffaitin and Thomas Moreau. Reinforcement learning for personalized dialysis scheduling: A systematic review. *Artificial Intelligence in Medicine*, 147:102746, 2024.

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

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, et al. Gymnasium: A standard interface for reinforcement learning environments, 2024.

Mauro Ursino and Marta Innocenti. Intradialytic cardiovascular changes and mathematical modeling. *Hemodialysis International*, 4(3):27–40, 2000.