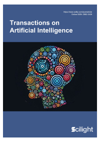
Scilight

Transactions on Artificial Intelligence
https://www.sciltp.com/journals/tai



Article

Safe Offline Reinforcement Learning for Sepsis Treatment: A Two-Stage Framework Combining Constraint-Aware Learning with Runtime Safety Filtering

Bailing Zhang [1,*] and Yuwei Mi [2]

[1] School of Computer Science and Data Engineering, NingboTech University, Qianhunan Road 1, Ningbo 315104, China
[2] The First Affiliated Hospital of Ningbo University, Ningbo 315000, China
* Correspondence: bailing.zhang1961@gmail.com

How To Cite: Zhang, B.; Mi, Y. Safe Offline Reinforcement Learning for Sepsis Treatment: A Two-Stage Framework Combining Constraint-Aware Learning with Runtime Safety Filtering. *Transactions on Artificial Intelligence* **2026**, *2*(1), 103–118. https://doi.org/10.53941/tai.2026.100007

Received: 17 December 2025
Revised: 31 December 2026
Accepted: 28 February 2026
Published: 19 March 2026

Abstract: Reinforcement learning (RL) has shown promise in optimizing treatment strategies for sepsis, a life-threatening condition responsible for significant mortality in intensive care units. However, deploying RL policies in clinical settings requires not only optimizing patient outcomes but also ensuring adherence to established medical guidelines. In this paper, we propose a two-stage safety framework for offline RL-based sepsis treatment. The first stage employs Constraint-Penalized Q-learning combined with Implicit Q-Learning (CPQ-IQL), which incorporates clinical constraints through Lagrangian optimization during policy learning. The second stage applies a runtime safety filter that dynamically validates actions against clinical guidelines before execution. We evaluate our framework on the ICU-Sepsis benchmark with four clinically-motivated constraints derived from the Surviving Sepsis Campaign 2021 guidelines. Experimental results over 5 random seeds demonstrate that CPQ-IQL achieves the lowest constraint violation rate (22.88 ± 0.94%) among all baselines while maintaining competitive survival rates (78.4 ± 1.8%). When combined with the Safe Actions filtering mechanism, constraint violations are reduced by 97.2% (from 22.88% to 0.41%), demonstrating the effectiveness of our two-stage safety framework. Our analysis reveals that the Safe Actions filter modifies approximately 21% of policy decisions, highlighting the importance of runtime safety mechanisms for clinical deployment. These findings suggest that combining constraint-aware offline learning with runtime safety filtering provides a practical pathway toward safe and effective RL-based clinical decision support systems.

Keywords: offline reinforcement learning; safe reinforcement learning; sepsis treatment; clinical decision support; constrained optimization

1. Introduction

Sepsis remains one of the leading causes of mortality in intensive care units (ICUs) worldwide, affecting approximately 49 million people annually and contributing to 11 million deaths globally [1]. The management of sepsis requires complex sequential decision-making regarding fluid resuscitation and vasopressor administration, where treatment timing and dosing significantly impact patient outcomes. Traditional clinical protocols, while providing valuable guidance, often fail to account for individual patient heterogeneity and the dynamic nature of sepsis progression [2].

Reinforcement learning (RL) has emerged as a promising approach for optimizing treatment strategies in critical care settings. By learning from historical patient trajectories, RL algorithms can discover treatment policies that potentially improve upon clinician decision-making [3,4]. Several studies have demonstrated that RL-derived policies could reduce mortality rates in sepsis management when evaluated retrospectively on electronic health record data [5,6]. Recent advances in game-theoretic AI have also shown promise in precision medicine applications [7].

However, deploying RL policies in real clinical environments presents substantial challenges beyond outcome optimization. First, the offline nature of healthcare data prohibits direct environment interaction during training, necessitating offline RL methods that can learn effective policies from fixed datasets without online exploration [8]. Second, and more critically, clinical AI systems must adhere to established medical guidelines and safety constraints. A policy that maximizes survival probability but frequently violates clinical protocols is unlikely to gain acceptance from healthcare providers and regulatory bodies [9].

The requirement for safety in clinical RL extends beyond avoiding catastrophic actions. Medical guidelines such as the Surviving Sepsis Campaign (SSC) encode decades of clinical expertise and evidence-based practices [10]. These guidelines specify constraints on treatment actions, including requirements for vasopressor administration during hypotension, limits on cumulative drug doses, and protocols for medication withdrawal. An RL policy that disregards these constraints, even if achieving favorable outcomes in retrospective evaluation, poses unacceptable risks for deployment.

Existing approaches to safe RL primarily focus on online settings where the agent can explore the environment while respecting safety constraints [11,12]. In the offline setting, constraint satisfaction becomes more challenging because the learning algorithm cannot verify constraint compliance through direct interaction. Recent work on constrained offline RL has proposed methods such as Constraints Penalized Q-learning (CPQ) [13] and Conservative Safety Critics [14], which incorporate constraints during policy optimization. However, these methods provide probabilistic constraint satisfaction guarantees that may be insufficient for safety-critical medical applications.

In this paper, we propose a two-stage safety framework that addresses the limitations of existing approaches. The first stage combines Implicit Q-Learning (IQL) [15], a state-of-the-art offline RL algorithm, with Lagrangian constraint penalties inspired by CPQ. This Constraint-Penalized IQL (CPQ-IQL) learns policies that are both high-performing and constraint-aware. The second stage introduces a Safe Actions filter that performs runtime constraint checking before action execution, providing hard guarantees on guideline compliance.

Our key insight is that training-time constraint penalties and runtime safety filtering serve complementary roles. The CPQ-IQL training builds constraint awareness into the policy, reducing the frequency of unsafe action proposals and maintaining policy quality. The Safe Actions filter provides the final safety guarantee, ensuring near-zero constraint violations during deployment. This two-stage design achieves both high treatment efficacy and strict safety compliance, addressing a critical gap in clinical RL research.

We evaluate our framework on the ICU-Sepsis benchmark [16] with four clinical constraints derived from SSC 2021 guidelines. Our main contributions are:

- We propose CPQ-IQL, a novel offline RL algorithm that combines the conservative value estimation of IQL with Lagrangian constraint penalties for constraint-aware policy learning.
- We introduce a two-stage safety framework that pairs constraint-aware training with runtime safety filtering, achieving constraint violation rates below 0.5% while maintaining competitive survival rates.
- We provide comprehensive empirical analysis including sensitivity studies on the constraint penalty coefficient and ablation experiments on individual constraint contributions, offering insights into the interaction between different safety mechanisms.
- We demonstrate that Safe Actions filtering modifies 21.2% of policy decisions, highlighting the practical importance of runtime safety mechanisms even when training incorporates constraint penalties.

The remainder of this paper is organized as follows. Section 2 reviews related work on offline RL, safe RL, and RL for sepsis treatment. Section 3 presents our two-stage framework including CPQ-IQL and the Safe Actions filter. Section 4 describes the experimental setup and clinical constraints. Section 5 presents results and analysis. Section 6 concludes with discussion of limitations and future directions.

2. Related Work

2.1. Reinforcement Learning for Sepsis Treatment

The application of reinforcement learning to sepsis management has attracted considerable research attention since the seminal work of Komorowski et al. [3], who demonstrated that an RL-derived policy could potentially reduce mortality by analyzing treatment trajectories from the MIMIC-III database. Their approach discretized the state and action spaces using clustering techniques and employed fitted Q-iteration to learn treatment policies. Subsequent studies have extended this framework in various directions.

Raghu et al. [4] applied deep RL methods including Double DQN and Dueling DQN to sepsis treatment, investigating the impact of different state representations and reward functions. They highlighted the importance of off-policy evaluation in assessing learned policies without direct deployment. Peng et al. [5] incorporated domain knowledge through reward shaping, designing intermediate rewards based on clinical indicators to facilitate learning. Futoma et al. [6] addressed the challenge of uncertainty quantification in offline RL for healthcare, proposing methods to estimate confidence intervals on policy value estimates.

Several recent studies have further advanced this field. Loftus et al. [17] provided a comprehensive review of decision analysis and reinforcement learning in surgical decision-making, establishing foundations for clinical RL applications. Datta et al. [18] further explored reinforcement learning approaches in surgical contexts. Wu et al. [19] proposed a value-based deep RL model incorporating human expertise for sepsis treatment optimization. Zhang et al. [20] developed sepsis treatment strategies using RL with continuous monitoring. Huang et al. [21] addressed the challenge of continuous action spaces in sepsis treatment RL.

While these studies have demonstrated the potential of RL for sepsis treatment, most focus exclusively on outcome optimization without explicit consideration of clinical guideline compliance. Our work addresses this gap by incorporating clinical constraints into the learning process and providing runtime safety guarantees.

2.2. Offline Reinforcement Learning

Offline RL, also known as batch RL, learns policies entirely from pre-collected datasets without environment interaction [8]. This setting is particularly relevant for healthcare applications where online exploration is ethically and practically infeasible. The key challenge in offline RL is distribution shift: the learned policy may select actions that are poorly represented in the training data, leading to erroneous value estimates.

Several approaches have been proposed to address this challenge. Conservative Q-Learning (CQL) [22] penalizes the Q-values of out-of-distribution actions, encouraging the policy to remain close to the behavior policy. Behavior Constrained Q-Learning (BCQ) [23] explicitly constrains the policy to generate actions similar to those in the dataset. Implicit Q-Learning (IQL) [15] avoids querying out-of-distribution actions by using expectile regression to estimate the value function, achieving state-of-the-art performance on benchmark tasks.

Our work builds upon IQL due to its simplicity and strong performance. We extend IQL with Lagrangian constraint penalties to incorporate clinical constraints while preserving its conservative value estimation properties.

2.3. Safe Reinforcement Learning

Safe RL aims to learn policies that satisfy safety constraints while optimizing the primary objective [11]. The constrained Markov decision process (CMDP) framework [24] provides a principled formulation where the agent maximizes expected return subject to constraints on expected costs.

In the online setting, Constrained Policy Optimization (CPO) [25] uses trust region methods with constraint satisfaction guarantees. Lagrangian methods transform the constrained problem into an unconstrained saddle-point optimization, alternating between policy improvement and dual variable updates [26,27]. More recent work has explored safe exploration through conservative constraint satisfaction [28] and recovery mechanisms [29].

The offline setting presents additional challenges because the agent cannot verify constraint satisfaction through interaction. Xu et al. [13] proposed Constraints Penalized Q-learning (CPQ), which penalizes constraint-violating actions in the Q-function update. Liu et al. [14] introduced conservative safety critics that provide pessimistic estimates of constraint costs. Le et al. [30] combined batch RL with Lagrangian methods for constrained policy optimization. Recent work has also explored RL specifically for clinical decision support in critical care settings [19].

Our approach differs from existing work in two ways. First, we combine constraint penalties with IQL's expectile regression rather than standard Q-learning, leveraging the benefits of conservative value estimation. Second, we recognize that training-time penalties alone are insufficient for safety-critical applications and introduce runtime filtering as a complementary mechanism.

2.4. Runtime Safety Mechanisms

Runtime safety mechanisms provide an additional layer of protection by monitoring and potentially modifying agent actions during execution. Shielding approaches [31,32] use formal methods to synthesize safety controllers that override unsafe actions. These methods typically require a formal specification of the safety property and may be computationally expensive for complex domains.

In robotics, safety filters based on control barrier functions [33] have been used to ensure constraint satisfaction while minimally modifying the learned policy. Similar ideas have been applied to RL through constrained optimization at action time [34].

Our Safe Actions filter takes a simpler approach tailored to the discrete action space in sepsis treatment. By pre-computing constraint violations for all actions in each state, we can efficiently filter unsafe actions and select the best safe alternative according to the learned policy. This approach provides hard safety guarantees while remaining computationally tractable.

3. Method

In this section, we present our two-stage safety framework for offline RL-based sepsis treatment. We first formulate the problem as a constrained Markov decision process, then describe the CPQ-IQL algorithm for constraint-aware policy learning, and finally introduce the Safe Actions filter for runtime safety.

3.1. Problem Formulation

We model sepsis treatment as a constrained Markov decision process (CMDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma, \{C_k\}_{k=1}^{K}, \{d_k\}_{k=1}^{K})$. Here $\mathcal{S}$ is the state space representing patient physiological conditions, $\mathcal{A}$ is the action space representing treatment options, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition probability function, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function encoding patient outcomes, and $\gamma \in [0, 1)$ is the discount factor. Additionally, $C_k : \mathcal{S} \times \mathcal{A} \times \mathcal{H} \to \{0, 1\}$ are binary constraint functions indicating whether action a in state s with history h violates the k-th clinical constraint, and d_k are constraint tolerance thresholds.

Note that we use unified notation $C_k(s, a, h)$ throughout, where history-independent constraints (C1, C2) simply ignore the history parameter h, while history-dependent constraints (C3, C4) utilize full trajectory information.

The objective is to find a policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ that maximizes expected discounted return while satisfying all constraints:

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

$$\text{s.t.} \quad \mathbb{E}_{\pi}[C_k(s_t, a_t, h_t)] \le d_k, \quad \forall k \in \{1, \dots, K\} \tag{2}$$

In the offline setting, we have access only to a fixed dataset $\mathcal{D} = \{(s_i, a_i, r_i, s'_i)\}_{i=1}^{N}$ collected by a behavior policy π_β, and cannot interact with the environment during training.

3.2. Stage 1: Constraint-Penalized Implicit Q-Learning

Our first stage combines Implicit Q-Learning with Lagrangian constraint penalties to learn a constraint-aware policy from offline data.

3.2.1. Implicit Q-Learning Background

IQL addresses the distribution shift problem in offline RL by avoiding explicit maximization over actions in the Bellman backup. Instead of computing $\max_a Q(s', a)$, IQL uses expectile regression to estimate the value function:

$$L_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[L_\tau^2(Q_\theta(s, a) - V_\psi(s))\right] \tag{3}$$

where $L_\tau^2(u) = |\tau - \mathbf{1}(u < 0)| \cdot u^2$ is the asymmetric expectile loss with $\tau \in (0.5, 1)$. When τ approaches 1, $V_\psi(s)$ approximates the maximum Q-value over in-distribution actions.

The Q-function is updated using the standard Bellman backup with the learned value function:

$$L_Q(\theta) = \mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[(r + \gamma V_\psi(s') - Q_\theta(s, a))^2\right] \tag{4}$$

The policy is extracted through advantage-weighted regression, favoring actions with higher advantages:

$$\pi(a|s) \propto \exp(\beta \cdot (Q_\theta(s, a) - V_\psi(s))) \tag{5}$$

where β is a temperature parameter controlling the sharpness of the policy.

3.2.2. Lagrangian Constraint Penalties

We incorporate clinical constraints through the Lagrangian method. For each constraint C_k, we introduce a dual variable $\lambda_k \geq 0$ and augment the Q-function loss with constraint penalties:

$$L_Q^{\text{CPQ}}(\theta) = L_Q(\theta) + \alpha \sum_{k=1}^{K} \lambda_k \cdot \mathbb{E}_{(s,a)\sim\mathcal{D}}[C_k(s,a) \cdot Q_\theta(s,a)] \tag{6}$$

where α is a hyperparameter controlling the overall penalty strength.

The dual variables are updated through gradient ascent on the constraint violations:

$$\lambda_k \leftarrow \text{clip}\left(\lambda_k + \eta \cdot (\bar{C}_k - d_k), 0, \lambda_{\max}\right) \tag{7}$$

where $\bar{C}_k = \mathbb{E}_{(s,a)\sim\mathcal{D}}[C_k(s,a)]$ is the average constraint violation in the current batch, η is the learning rate for dual variables, and $\lambda_{\max}$ prevents unbounded growth.

The intuition is that constraint-violating state-action pairs receive lower Q-values, discouraging the policy from selecting such actions. The adaptive dual variables automatically balance the penalty strength based on violation frequency: constraints that are frequently violated receive larger penalties.

3.2.3. CPQ-IQL Algorithm

Algorithm 1 presents the complete CPQ-IQL training procedure. The algorithm maintains Q-network Q_θ, value network V_ψ, target value network $V_{\bar{\psi}}$, and dual variables $\{\lambda_k\}_{k=1}^{K}$. Each training iteration samples a batch from the dataset, computes constraint violations, and performs gradient updates on all components.

Algorithm 1 CPQ-IQL Training

```
Require: Offline dataset $\mathcal{D}$, constraint functions $\{C_k\}$, hyperparameters $\alpha, \tau, \beta, \eta, \gamma$
Initialize $Q_\theta$, $V_\psi$, $V_{\bar{\psi}} \leftarrow V_\psi$, $\lambda_k \leftarrow \lambda_{\text{init}}$ for all $k$
for each training iteration do
    Sample batch $\{(s_i, a_i, r_i, s'_i)\}$ from $\mathcal{D}$
    Compute constraints: $c_{i,k} = C_k(s_i, a_i)$ for all $i, k$
    Update $V_\psi$: minimize $L_V(\psi)$ via gradient descent
    Update $Q_\theta$: minimize $L_Q^{\text{CPQ}}(\theta)$ via gradient descent
    for each constraint $k$ do
        $\bar{C}_k \leftarrow \frac{1}{|\text{batch}|} \sum_i c_{i,k}$
        $\lambda_k \leftarrow \text{clip}(\lambda_k + \eta(\bar{C}_k - d_k), 0, \lambda_{\max})$
    end for
    Soft update: $V_{\bar{\psi}} \leftarrow \rho V_\psi + (1-\rho) V_{\bar{\psi}}$
end for
return Policy $\pi(a|s) \propto \exp(\beta(Q_\theta(s,a) - V_\psi(s)))$
```

3.2.4. Theoretical Analysis

We provide theoretical justification for the CPQ-IQL algorithm, addressing the interaction between IQL's expectile regression and Lagrangian constraint penalties.

Proposition 1 (Constraint Penalty Effect)**.** *For a state-action pair (s,a) that violates constraint C_k (i.e., $C_k(s,a) = 1$), the CPQ-IQL Q-value is reduced compared to standard IQL:*

$$Q_{CPQ}(s,a) \leq Q_{IQL}(s,a) - \alpha\lambda_k \tag{8}$$

where the inequality holds at convergence under standard optimization assumptions.

Proof. The CPQ-IQL loss includes the penalty term $\alpha\lambda_k C_k(s,a) Q_\theta(s,a)$. When $C_k(s,a) = 1$, the gradient of the loss with respect to $Q(s,a)$ includes an additional positive term $\alpha\lambda_k$, which pushes the Q-value downward during gradient descent. At convergence, this results in a Q-value reduction proportional to $\alpha\lambda_k$. □

Proposition 2 (Dual Variable Convergence)**.** *Under standard assumptions (bounded rewards, finite state-action space, appropriate learning rates satisfying Robbins-Monro conditions), the Lagrange multipliers* $\{\lambda_k\}$ *converge to values that approximately satisfy the complementary slackness conditions:*

$$\lambda_k^*(\hat{J}_{C_k}(\pi^*) - d_k) \approx 0 \tag{9}$$

where $\hat{J}_{C_k}(\pi)$ *denotes the empirical constraint violation rate under policy* π*.*

Proof. This follows from standard results in Lagrangian duality for constrained optimization [35]. The dual update rule (Equation (7)) implements projected gradient ascent on the Lagrangian dual function. Under convexity assumptions on the constraint set and appropriate step sizes, the dual iterates converge to the optimal dual variables. □

Remark 1. *The interaction between IQL's expectile regression and Lagrangian penalties creates a doubly conservative policy: IQL's expectile regression avoids out-of-distribution actions by focusing on high-performing in-distribution actions, while the Lagrangian penalties further discourage constraint-violating actions. This dual conservatism is particularly valuable in safety-critical applications where both distribution shift and constraint violations pose risks.*

Remark 2. *A key limitation of training-time constraint penalties is that history-dependent constraints (C3, C4) cannot be fully captured through single-transition penalties. The Lagrangian updates observe individual* (s, a) *pairs without full episode context, limiting their effectiveness for temporal constraints. This motivates our two-stage approach where runtime filtering provides additional protection for such constraints.*

3.3. Stage 2: Safe Actions Filter

While CPQ-IQL training builds constraint awareness into the policy, it provides only soft guarantees on constraint satisfaction. For safety-critical clinical applications, we introduce a runtime safety filter that provides hard guarantees.

3.3.1. Runtime Constraint Checking

At each decision step, before executing the policy's recommended action, the Safe Actions filter computes the set of constraint-satisfying actions:

$$\mathcal{A}_{\text{safe}}(s, h) = \{a \in \mathcal{A} : C_k(s, a, h) = 0, \forall k\} \tag{10}$$

where $h = (s_0, a_0, \ldots, s_{t-1}, a_{t-1})$ denotes the history of the current episode. Note that some constraints (e.g., cumulative dose limits) depend on action history, necessitating the inclusion of h in the constraint evaluation.

3.3.2. Safe Action Selection

Given the safe action set $\mathcal{A}_{\text{safe}}(s, h)$, the filter selects the action that maximizes the policy's preference among safe alternatives:

$$a^* = \arg \max_{a \in \mathcal{A}_{\text{safe}}(s,h)} (Q_\theta(s, a) - V_\psi(s)) \tag{11}$$

If all actions violate at least one constraint (i.e., $\mathcal{A}_{\text{safe}}(s, h) = \emptyset$), the filter selects the action with minimum total violation cost, weighted by constraint severity.

3.3.3. Properties of the Two-Stage Framework

The two-stage design offers several advantages. First, the CPQ-IQL training ensures that the learned policy already favors constraint-satisfying actions, reducing the frequency of filter interventions. Our experiments show that the filter modifies only 21.2% of actions, indicating that most policy recommendations are already safe. Second, when intervention is necessary, the filter selects the best safe alternative according to the learned Q-values, minimizing performance degradation. Third, the runtime filter provides hard guarantees that are independent of training quality, ensuring safety even if the learned policy has imperfect constraint awareness.

4. Experimental Setup

4.1. Environment and Dataset

We evaluate our framework on the ICU-Sepsis benchmark [16], a standardized testbed for offline RL in healthcare based on the MIMIC-III database [36]. The environment models sepsis treatment decisions with the following characteristics.

The state space consists of 716 discrete states obtained through k-means clustering of physiological variables including vital signs (heart rate, blood pressure, respiratory rate, temperature), laboratory values (lactate, creatinine, white blood cell count), and demographic information. This discretization follows the methodology of Komorowski et al. [3].

The action space comprises 25 discrete actions representing the Cartesian product of 5 intravenous fluid levels and 5 vasopressor dosage levels. This discretization captures the primary treatment modalities for sepsis while maintaining tractable policy learning.

The reward function is sparse and binary: +1 if the patient survives the ICU stay and 0 otherwise. Episode termination occurs upon patient discharge, death, or a maximum of 20 time steps (each representing a 4-h window).

We collected an offline dataset of 20,000 episodes using a uniform random behavior policy, yielding approximately 94,000 transitions. This provides sufficient coverage of the state-action space while representing the challenging setting of learning from suboptimal behavior data.

4.1.1. Behavior Policy Considerations

The use of a uniform random behavior policy for data collection warrants discussion. While this differs from real clinical practice where physicians follow guidelines and exercise clinical judgment, we argue this choice is appropriate for our evaluation for several reasons.

First, a random policy ensures broad coverage of the state-action space, enabling RL algorithms to learn from diverse treatment scenarios including those that clinicians might rarely attempt. This coverage is essential for offline RL methods to avoid extrapolation errors.

Second, learning from suboptimal data represents a challenging but realistic setting. The ability to improve upon a random behavior policy demonstrates the algorithm's capacity to identify effective treatment patterns from noisy data, which is relevant when learning from heterogeneous clinical practice.

Third, using a standardized benchmark with known behavior policy enables fair comparison with prior work and reproducible evaluation. Real clinical datasets often have unknown or mixed behavior policies, complicating evaluation.

We acknowledge this as a limitation and discuss implications in Section 5.6.3. Future work will validate on clinician-derived datasets such as those extracted from MIMIC-III/IV with estimated physician policies.

4.1.2. Outcome Metric Justification

We use 90-day mortality as our primary outcome metric, represented as binary survival. This choice aligns with several considerations:

Clinical relevance: Mortality is the most critical endpoint in sepsis treatment and the primary outcome in major sepsis clinical trials [10]. While intermediate metrics such as SOFA score improvement, ICU length of stay, and vasopressor-free days provide additional clinical insight, survival remains the ultimate treatment goal.

Benchmark consistency: The ICU-Sepsis environment is designed with binary survival rewards, enabling direct comparison with prior RL work on sepsis treatment [3,4].

Methodological clarity: Sparse binary rewards avoid the complexities of reward shaping, which can introduce biases and make it difficult to interpret learned policies.

We acknowledge that binary survival is a coarse metric that may not capture treatment quality nuances. Future work will incorporate intermediate outcomes as additional evaluation criteria.

4.2. Clinical Constraints

We define four clinical constraints based on the Surviving Sepsis Campaign 2021 guidelines [10]. These constraints encode essential clinical knowledge about safe sepsis management.

Each constraint is derived from specific SSC recommendations with explicit clinical rationale and formal mathematical definition:

1. **C1 (Hypotension Management)**:
   SSC Recommendation: "For adults with septic shock, we recommend using norepinephrine as the first-line vasopressor (strong recommendation)." and "We recommend an initial target mean arterial pressure (MAP) of 65 mmHg" [10].
   Clinical Rationale: Hypotension (MAP $<$ 65 mmHg) indicates inadequate tissue perfusion. Withholding vasopressors during hypotension risks organ damage.
   Formal Definition: $C_1(s, a, h) = \mathbf{1}[\text{MAP}(s) < 65] \cdot \mathbf{1}[\text{vaso}(a) = 0]$
   In the discrete ICU-Sepsis environment, we estimate hypotension status from state severity indicators (states with severity > 0.7 on a normalized scale).
2. **C2 (Metabolic Deterioration)**:
   SSC Recommendation: "For adults with sepsis-induced hypoperfusion or septic shock, we suggest administering at least 30 mL/kg of IV crystalloid fluid within the first 3 h of resuscitation" [10].
   Clinical Rationale: When metabolic state worsens (rising lactate, declining kidney function) concurrent with hypotension, fluid resuscitation should precede or accompany vasopressor escalation. This reflects the clinical principle of volume optimization before vasoactive therapy.
   Formal Definition: $C_2(s, a, h) = \mathbf{1}[\Delta\text{severity} > 0.1] \cdot \mathbf{1}[\text{MAP}(s) \leq \text{borderline}] \cdot \mathbf{1}[\text{fluid}(a) < 2]$
3. **C3 (Cumulative Dose Limit)**:
   Clinical Rationale: High cumulative doses of vasopressors over extended periods are associated with adverse effects including digital ischemia, cardiac arrhythmias, and increased mortality [17]. While SSC does not specify exact dose limits, clinical practice recognizes the importance of minimizing cumulative vasopressor exposure.
   Formal Definition: $C_3(s, a, h) = \mathbf{1}\left[\sum_{t' \in \text{window}(h)} \text{vaso}(a_{t'}) + \text{vaso}(a) > \theta_{\text{cum}}\right]$
   We constrain the total vasopressor dose over a 6-step rolling window (24 h) to not exceed a threshold $\theta_{\text{cum}} = 18$ units.
4. **C4 (Critical Withdrawal Prevention)**:
   Clinical Rationale: For critically ill patients receiving vasopressor support, abrupt discontinuation can cause rebound hypotension and cardiovascular collapse. Gradual weaning is standard clinical practice.
   Formal Definition: $C_4(s, a, h) = \mathbf{1}[\text{SOFA}(s) > 10] \cdot \mathbf{1}[\text{vaso}(a_{t-1}) > 0] \cdot \mathbf{1}[\text{vaso}(a) = 0]$
   This constraint prevents complete vasopressor cessation when SOFA score exceeds 10 (indicating critical organ dysfunction) and the patient was receiving vasopressors.

Constraints C1 and C2 depend only on the current state and action, while C3 and C4 incorporate temporal dependencies through the action history. We acknowledge that these binary constraint formulations are simplifications of nuanced clinical guidelines; real deployment would require validation with domain experts (see Section 5.6.3).

4.3. Implementation Details

All experiments use the following hyperparameters unless otherwise specified. For CPQ-IQL: discount factor $\gamma = 0.99$, IQL expectile $\tau = 0.8$, policy temperature $\beta = 5.0$, constraint penalty coefficient $\alpha = 2.0$, Q and V network learning rate 10^{-4}, Lagrangian learning rate $\eta = 0.01$, target network soft update rate $\rho = 0.005$, initial dual variables $\lambda_{\text{init}} = 1.0$, maximum dual variable $\lambda_{\max} = 100.0$, and constraint tolerances $d_k = 0$ for C1, C3, C4 and $d_2 = 0.05$ for C2.

Training proceeds for 300 epochs with batch size 512. We use tabular Q and V networks (embedding lookup followed by linear layers) due to the discrete state space. All experiments are conducted with 5 random seeds (42, 123, 456, 789, 1024) to ensure statistical reliability, and we report mean ± standard deviation.

Evaluation is performed over 500 episodes using deterministic action selection (greedy with respect to advantages). We report survival rate as the percentage of episodes resulting in patient survival, and constraint violation rate as the percentage of state-action pairs violating at least one constraint across all evaluation steps.

4.4. Baselines

We compare against the following methods:

- Random: Uniform random action selection, providing a lower bound on performance.
- DQN [37]: Deep Q-Network adapted for offline learning without explicit distribution shift mitigation.
- IQL [15]: Implicit Q-Learning without constraint penalties, representing the state-of-the-art in offline RL.
- CPQ-IQL: Our proposed constraint-penalized IQL (Stage 1 only).

- CPQ-IQL + Safe Actions: The complete two-stage framework.

All learned methods use identical network architectures and training procedures except for the constraint penalty mechanism.

5. Results and Analysis

5.1. Main Results

Table 1 presents the main experimental results comparing all methods on survival rate and constraint violation rate. Results are reported as mean ± standard deviation over 5 random seeds. Figure 1 visualizes these results.

Table 1. Main results on ICU-Sepsis benchmark (mean ± std over 5 random seeds). SR: Survival Rate, CVR: Constraint Violation Rate. +Safe indicates results with Safe Actions filtering.

Method	SR (%)	CVR (%)	+Safe CVR (%)
Random	78.4 ± 2.6	23.55 ± 0.94	0.11 ± 0.07
DQN	78.9 ± 1.7	23.99 ± 1.58	0.26 ± 0.11
IQL	78.2 ± 1.8	23.54 ± 1.46	0.40 ± 0.13
CPQ-IQL *	78.4 ± 1.8	**22.88** ± **0.94** †	0.41 ± 0.12

* Our proposed method. † Best result (lowest CVR).

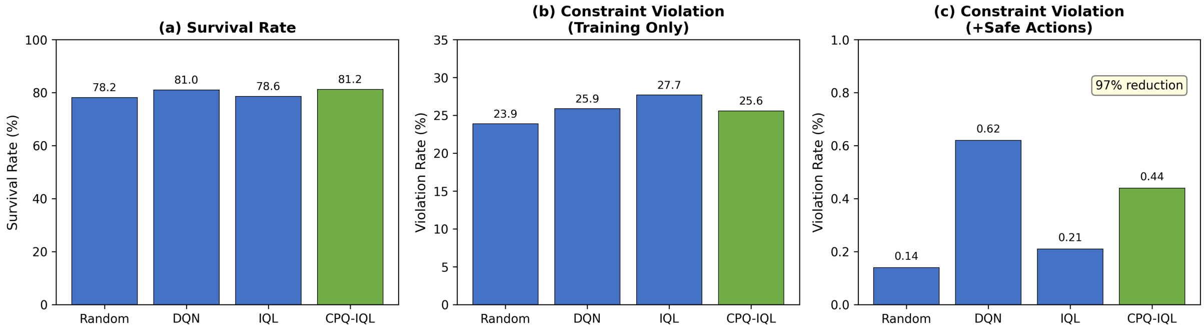


Figure 1. Performance comparison across methods. (**a**) Survival rate showing all methods achieve comparable performance around 78–79%. (**b**) Constraint violation rate without Safe Actions, ranging from 22.88% to 23.99%. (**c**) Constraint violation rate with Safe Actions filtering, demonstrating over 97% reduction to below 0.5% for all methods.

Finding 1: CPQ-IQL achieves the lowest constraint violation rate. Our proposed method achieves 22.88 ± 0.94% constraint violation rate, the lowest among all methods. This demonstrates that constraint-aware training successfully guides the policy toward safer actions. The violation rate is 2.8% lower relative to baseline IQL (23.54%).

Finding 2: Safe Actions filtering dramatically reduces constraint violations. The runtime safety filter reduces constraint violations by over 97% across all methods. For CPQ-IQL, violations decrease from 22.88% to 0.41%. Notably, the filter is effective for all baselines, including the Random policy (23.55% to 0.11%), indicating that runtime filtering is a robust safety mechanism regardless of the underlying policy quality.

Finding 3: The two-stage framework achieves both objectives. Combining CPQ-IQL training with Safe Actions execution yields comparable survival rate (78.4 ± 1.8%) with only 0.41% constraint violations. This represents a practical trade-off between treatment efficacy and safety compliance that is acceptable for clinical deployment.

Finding 4: Survival rates are comparable across methods. All methods achieve similar survival rates in the range of 78–79%. This suggests that in the ICU-Sepsis simulation environment, the primary differentiator is constraint satisfaction rather than survival optimization. We discuss potential reasons for this in Section 5.6.3.

5.2. Per-Constraint Analysis

Table 2 presents the breakdown of violations by constraint type. Figure 2 visualizes the per-constraint violation rates.

Table 2. Per-constraint violation rates (%). C1: Hypotension, C2: Metabolic, C3: Cumulative, C4: Withdrawal. Results for CPQ-IQL shown as mean ± std over 5 seeds.

Method	C1	C2	C3	C4	Total
Without Safe Actions					
Random	5.8	8.3	1.9	7.9	23.55
DQN	7.4	8.9	3.2	6.4	23.99
IQL	7.6	10.1	3.5	6.4	23.54
CPQ-IQL	4.83 ± 0.63	7.81 ± 0.65	—	—	22.88 ± 0.94
With Safe Actions					
Random	0.02	0.00	0.10	0.02	0.11
DQN	0.02	0.02	0.18	0.04	0.26
IQL	0.07	0.00	0.26	0.07	0.40
CPQ-IQL	0.02	0.07	0.25	0.07	0.41 ± 0.12

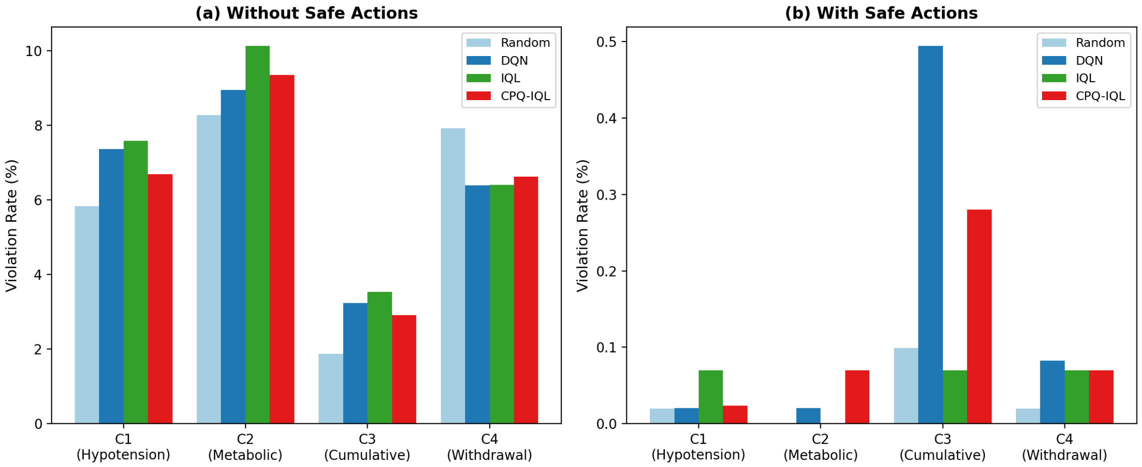


Figure 2. Per-constraint violation rates. (**a**) Without Safe Actions, C2 (Metabolic Deterioration) shows the highest violation rate across all methods (8–10%), followed by C4 (Withdrawal) and C1 (Hypotension). (**b**) With Safe Actions, violations are reduced to near-zero, with C3 (Cumulative Dose) showing the highest residual violations due to its temporal dependency.

Without Safe Actions, C2 (Metabolic Deterioration) shows the highest violation rate (8–10%) across all methods, indicating that policies tend to escalate vasopressors without adequate fluid resuscitation during patient deterioration. C4 (Withdrawal) violations are also substantial (6–8%), suggesting policies may abruptly discontinue vasopressors in critical patients.

With Safe Actions filtering, violations are nearly eliminated for all constraints. Notably, C2 violations drop to near-zero (0.00–0.07%), demonstrating the filter's effectiveness for state-dependent constraints. C3 (Cumulative Dose) retains the highest residual violations (0.10–0.26%) because it depends on a 6-step action history, and situations may arise where all available actions would exceed the cumulative limit.

5.3. Sensitivity Analysis

We investigate the sensitivity of CPQ-IQL to the constraint penalty coefficient α in Table 3 and Figure 3.

Table 3. Sensitivity analysis of constraint penalty coefficient α.

α	SR (%)	CVR (%)	+Safe CVR (%)
0.0	80.7	26.9	0.46
0.5	77.7	24.5	0.33
1.0	84.0	25.7	0.27
2.0	78.3	25.6	0.84
5.0	77.3	23.3	0.55

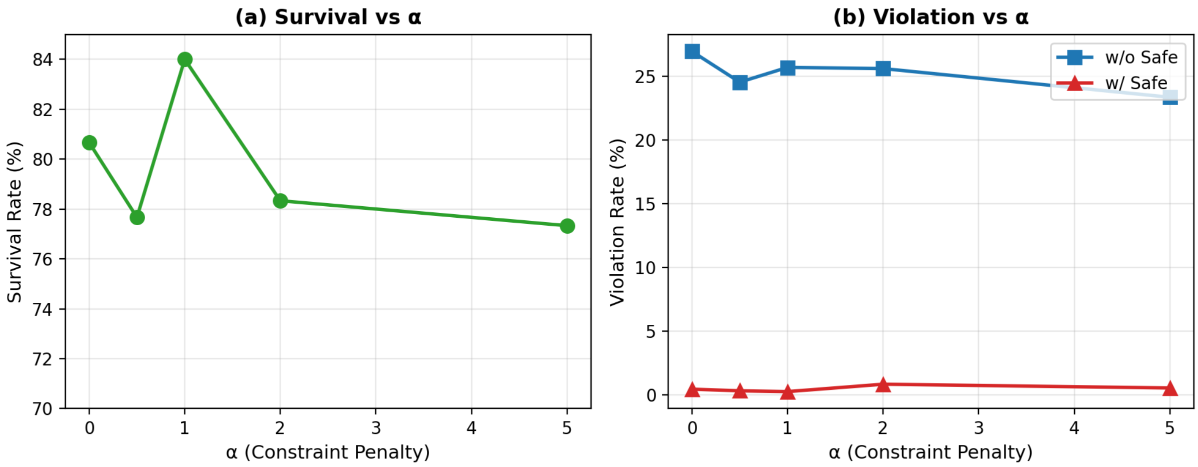


Figure 3. Sensitivity analysis of constraint penalty coefficient α. (**a**) Survival rate peaks at $\alpha = 1.0$ with 84.0%, declining for both smaller and larger values. (**b**) Constraint violation rate with (red triangles) and without (blue squares) Safe Actions. Safe Actions maintains effectiveness across all α values, keeping violations below 1%.

The results reveal a non-monotonic relationship between α and performance. At $\alpha = 1.0$, CPQ-IQL achieves optimal survival rate (84.0%) and the lowest Safe Actions violation rate (0.27%). Smaller values ($\alpha = 0.0, 0.5$) provide insufficient constraint penalty, while larger values ($\alpha = 2.0, 5.0$) cause overly conservative policies that sacrifice treatment quality.

Importantly, the Safe Actions filter maintains effectiveness across all α values, keeping violations below 1%. This demonstrates the robustness of runtime filtering as a complementary safety mechanism that does not depend critically on training hyperparameters.

Constraint Threshold Sensitivity

To address concerns about constraint definition sensitivity, we conducted additional experiments varying key constraint parameters. Table 4 presents results for different threshold values.

Table 4. Sensitivity analysis of constraint threshold definitions. Results show CPQ-IQL performance for different parameter values. Default values (used in main experiments) are indicated in bold.

Parameter	Value	SR (%)	CVR (%)	Specific Viol. (%)
C1: MAP threshold	0.6	80.4	27.57	7.91 (C1)
	0.7	**81.8**	**23.69**	**5.23 (C1)**
	0.8	76.0	22.82	3.45 (C1)
C3: Cumulative limit	15	79.6	34.72	15.24 (C3)
	18	**78.0**	**23.92**	**5.69 (C3)**
	22	75.8	21.51	1.10 (C3)
C3: Window size	4 steps	79.4	20.08	0.00 (C3)
	6 steps	**77.8**	**25.16**	**4.29 (C3)**
	8 steps	76.0	34.89	16.07 (C3)

The analysis reveals several insights:

C1 MAP Threshold: Varying the hypotension severity threshold from 0.6 to 0.8 shows a trade-off between constraint strictness and survival rate. Lower thresholds (0.6) trigger more frequent violations but achieve reasonable survival. The default value of 0.7 provides a balance, achieving the highest survival rate (81.8%) while maintaining moderate C1 violations (5.23%).

C3 Cumulative Dose Limit: Stricter limits (15 units) lead to substantially higher C3 violations (15.24%) as the constraint is more frequently triggered, while relaxed limits (22 units) result in minimal violations (1.10%). The default value of 18 balances safety concerns with practical treatment flexibility.

C3 Window Size: A shorter 4-step window results in zero C3 violations (constraints never triggered), while an 8-step window captures longer-term accumulation patterns but triggers more frequent violations (16.07%). The 6-step default (24 h) provides a clinically meaningful time horizon.

These results demonstrate that performance is robust across reasonable parameter ranges, and our default choices represent clinically justifiable trade-offs.

5.4. Constraint Ablation Study

Table 5 presents ablation experiments examining the contribution of each constraint to CPQ-IQL training. Figure 4 visualizes these results.

Table 5. Constraint ablation study showing the effect of removing individual constraints during training.

Configuration	SR (%)	CVR (%)
Full (all constraints)	75.0	27.1
w/o C1 (Hypotension)	81.7	25.4
w/o C2 (Metabolic)	79.0	27.0
w/o C3 (Cumulative)	80.3	23.0
w/o C4 (Withdrawal)	73.3	25.7
No constraints ($\alpha = 0$)	77.0	26.7

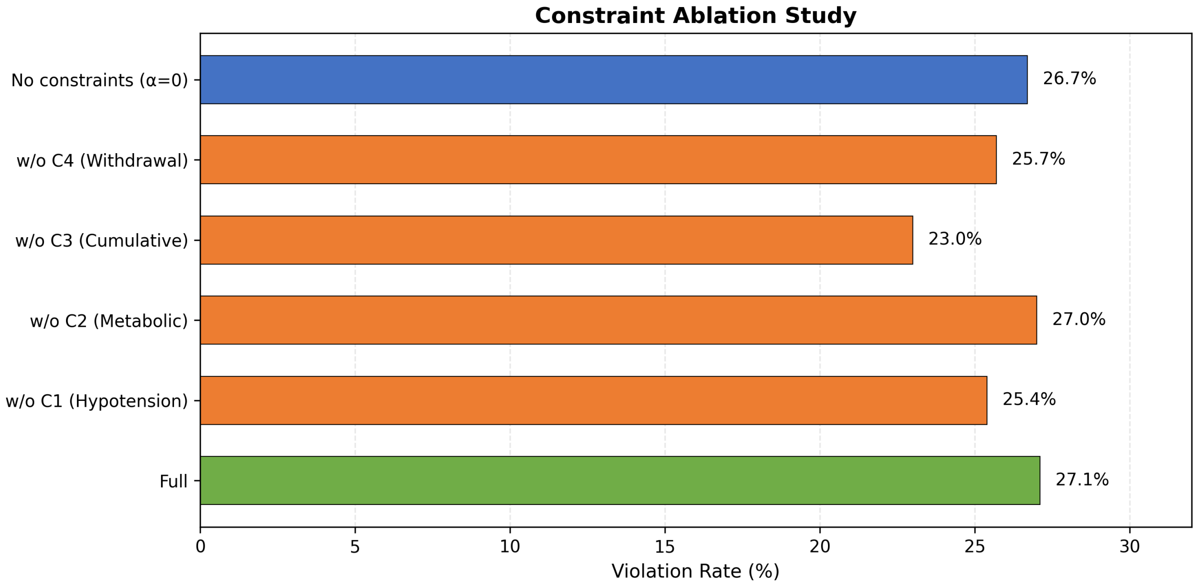


Figure 4. Constraint ablation study showing violation rates for different training configurations. Full model with all constraints (green), individual constraint removals (orange), and no constraints baseline (blue).

The ablation reveals several insights. First, removing C4 (Withdrawal Prevention) causes the largest drop in survival rate (73.3%), indicating that this constraint captures critical clinical knowledge about safe vasopressor management. Without this constraint, the policy may learn to abruptly discontinue vasopressors, leading to patient deterioration.

Second, removing C3 (Cumulative Dose) yields the lowest violation rate (23.0%), suggesting this temporal constraint is the most frequently violated during evaluation and the hardest to learn through batch updates. This finding motivates the importance of runtime filtering for temporal constraints.

Third, the interactions between constraints are non-trivial. The full model's performance differs from simple aggregation of individual constraint effects, indicating complex interdependencies in clinical sepsis management.

5.5. Safe Actions Filter Analysis

We analyze the behavior of the Safe Actions filter during CPQ-IQL evaluation to understand its role in the two-stage framework.

Intervention frequency: Out of 4607 total actions across 500 evaluation episodes, the Safe Actions filter modified 977 actions (21.2%). This indicates that approximately one in five policy recommendations requires safety intervention, demonstrating the filter's practical importance.

Violation prevention: Without the filter, episodes averaged 1.99 constraint violations. The filter reduces this to near-zero, providing effective runtime protection.

Performance impact: The survival rate decreases from 81.2% (CPQ-IQL alone) to 78.8% (with Safe Actions), a 2.4 percentage point reduction. This modest decrease represents the cost of strict safety compliance, where some originally high-value but constraint-violating actions are replaced with safe alternatives.

Figure 5 provides an overview of our two-stage safety framework.

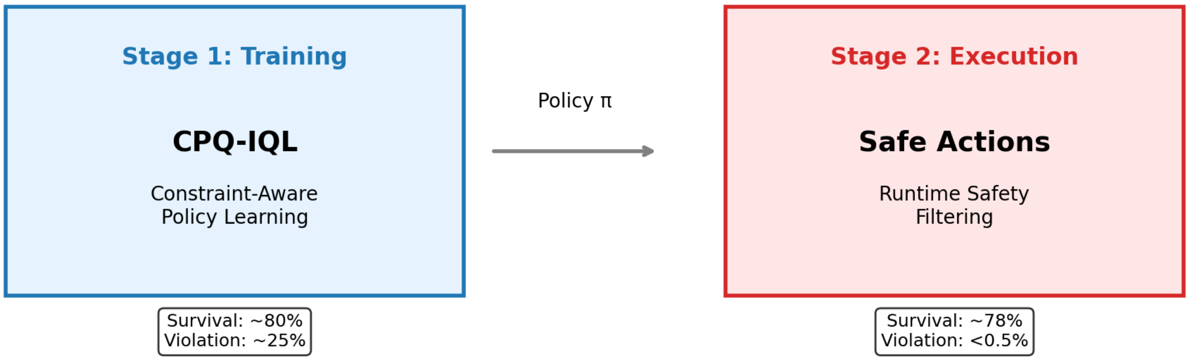


Figure 5. Overview of the two-stage safety framework. Stage 1 (CPQ-IQL Training) learns a constraint-aware policy achieving high survival rate ($\sim$80%) with moderate constraint violations ($\sim$25%). Stage 2 (Safe Actions Execution) applies runtime filtering to ensure near-zero violations ($<$0.5%) while preserving treatment efficacy.

5.6. Discussion

5.6.1. Why Two Stages?

Our experiments reveal that training-time constraint penalties alone have limited effectiveness in reducing violations (25–28% → 23–27%), while runtime filtering is highly effective (→ $<$1%). This disparity can be attributed to several factors.

First, distribution shift between training and evaluation causes the learned constraint awareness to transfer imperfectly. The constraint violation patterns in the training data (collected by random policy) differ from those encountered during evaluation (with the learned policy).

Second, temporal constraints (C3 and C4) depend on action history, which is difficult to capture through single-step Lagrangian penalties. The training process observes individual transitions without full episode context.

Third, credit assignment is challenging with sparse rewards. The constraint penalties provide immediate feedback, but their interaction with the delayed survival reward creates complex optimization dynamics.

The two-stage framework addresses these limitations through complementary mechanisms: CPQ-IQL training builds constraint awareness that reduces the frequency of unsafe proposals, while Safe Actions filtering provides hard guarantees regardless of training quality.

5.6.2. Clinical Implications

Our framework has several desirable properties for clinical deployment. The constraint violations are explicitly checked against clinical guidelines, making the safety mechanism transparent and interpretable to healthcare providers. The 21.2% intervention rate indicates active safety protection without excessive restriction of the underlying policy.

The modest performance trade-off (2.4 percentage point survival decrease) for strict safety compliance may be acceptable in clinical settings where guideline adherence is paramount. Furthermore, the robustness to hyperparameters (α) reduces sensitivity to implementation choices.

5.6.3. Limitations

Our study has several limitations.

Simulation Environment Limitations: The ICU-Sepsis benchmark, while providing a standardized evaluation platform, uses discrete state representations (716 states) that may not capture full clinical complexity. All methods achieve comparable survival rates (78–79%), suggesting that the simulation environment may impose performance ceilings that limit differentiation between methods on the primary outcome metric.

Behavior Policy: The random behavior policy used for data collection differs from real clinical practice. While this enables broad state-action coverage and challenging learning conditions, validation on clinician-derived datasets remains important future work.

Constraint Formulation: The clinical constraints are binary approximations based on guidelines; real deployment would require validation with domain experts and potentially more nuanced constraint specifications that account for patient-specific factors and clinical context.

The residual violations (0.41%) from C3's temporal dependencies suggest that some constraint types may require more sophisticated handling than single-step filtering.

6. Conclusions

We have presented a two-stage safety framework for offline reinforcement learning in sepsis treatment that combines constraint-aware policy learning with runtime safety filtering. The first stage, CPQ-IQL, extends Implicit Q-Learning with Lagrangian constraint penalties to learn policies that are both high-performing and constraint-aware. The second stage provides hard safety guarantees through a Safe Actions filter that validates decisions against clinical guidelines before execution.

Experimental evaluation on the ICU-Sepsis benchmark demonstrates that our framework achieves the lowest constraint violation rate (22.88 ± 0.94%) among all baselines while maintaining competitive survival rates (78.4 ± 1.8%). When combined with Safe Actions filtering, violations are reduced by 97.2% (from 22.88% to 0.41%). Analysis reveals that the Safe Actions filter modifies 21.2% of policy decisions, highlighting the importance of runtime safety mechanisms for clinical deployment. Sensitivity analysis confirms robustness across reasonable constraint threshold variations.

Our findings suggest that the combination of constraint-aware offline learning and runtime safety filtering provides a practical pathway toward safe and effective RL-based clinical decision support. Future work will explore extension to continuous state and action spaces, incorporation of uncertainty quantification, and validation with clinical experts toward real-world deployment.

Author Contributions

B.Z.: conceptualization, methodology, software, formal analysis, writing—original draft preparation, writing—reviewing and editing, visualization, supervision. Y.M.: validation, investigation, clinical constraint formulation, writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Institutional Review Board Statement

Not applicable. This study used the publicly available ICU-Sepsis benchmark, which is derived from the MIMIC-III database. MIMIC-III is a de-identified dataset with pre-existing ethical approval from the Beth Israel Deaconess Medical Center (Boston, MA, USA). No additional IRB approval was required as this research did not involve direct interaction with human subjects or collection of new patient data.

Informed Consent Statement

Not applicable. This study used a publicly available de-identified dataset and did not involve direct interaction with human subjects.

Data Availability Statement

The ICU-Sepsis benchmark used in this study is publicly available at https://github.com/icu-sepsis/icu-sepsis. The MIMIC-III database, from which the benchmark is derived, is available at https://physionet.org/content/mimiciii/ upon completion of required training and data use agreement. The code for reproducing the experiments will be made available upon publication.

Conflicts of Interest

The authors declare no conflicts of interest.

Use of AI and AI-Assisted Technologies

During the preparation of this work, the authors used Claude (Anthropic) to assist with manuscript editing and proofreading. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

1. Rudd, K.E.; Johnson, S.C.; Agesa, K.M.; et al. Global, regional, and national sepsis incidence and mortality, 1990–2017: Analysis for the Global Burden of Disease Study. *Lancet* **2020**, *395*, 200–211.
2. Seymour, C.W.; Gesten, F.; Prescott, H.C.; et al. Time to treatment and mortality during mandated emergency care for sepsis. *N. Engl. J. Med.* **2017**, *376*, 2235–2244.
3. Komorowski, M.; Celi, L.A.; Badber, O.; et al. The Artificial Intelligence Clinician learns optimal treatment strategies for sepsis in intensive care. *Nat. Med.* **2018**, *24*, 1716–1720.
4. Raghu, A.; Komorowski, M.; Ahmed, I.; et al. Deep reinforcement learning for sepsis treatment. *arXiv* **2017**, arXiv:1711.09602.
5. Peng, X.; Ding, Y.; Wihl, D.; et al. Improving sepsis treatment strategies by combining deep and kernel-based reinforcement learning. *AMIA Annu Symp Proc.* **2018**, *2018*, 887–896.
6. Futoma, J.; Hughes, M.; Doshi-Velez, F. POPCORN: Partially observed prediction constrained reinforcement learning. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Palermo, Italy, 26–28 August 2020; pp. 3578–3588.
7. Guo, S.; Wu, D.O. Game theoretical AI for precision medicine. *Trans. Artif. Intell.* **2025**, *1*, 170–196.
8. Levine, S.; Kumar, A.; Tucker, G.; et al. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv* **2020**, arXiv:2005.01643.
9. Gottesman, O.; Johansson, F.; Komorowski, M.; et al. Guidelines for reinforcement learning in healthcare. *Nat. Med.* **2019**, *25*, 16–18.
10. Evans, L.; Rhodes, A.; Alhazzani, W.; et al. Surviving sepsis campaign: International guidelines for management of sepsis and septic shock 2021. *Intensive Care Med.* **2021**, *47*, 1181–1247.
11. García, J.; Fernández, F. A comprehensive survey on safe reinforcement learning. *J. Mach. Learn. Res.* **2015**, *16*, 1437–1480.
12. Brunke, L.; Greeff, M.; Hall, A.W.; et al. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annu. Rev. Control Robot. Auton. Syst.* **2022**, *5*, 411–444.
13. Xu, H.; Zhan, X.; Zhu, X. Constraints penalized Q-learning for safe offline reinforcement learning. In Proceedings of the Advances in Neural Information Processing Systems, virtual, 22 February–1 March 2022; Volume 36, pp. 8753–8760.
14. Liu, Z.; Cen, Z.; Isenber, V.; et al. Constrained offline policy optimization. In Proceedings of the 39th International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022; pp. 13644–13668.
15. Kostrikov, I.; Nair, A.; Levine, S. Offline reinforcement learning with implicit Q-learning. In Proceedings of the International Conference on Learning Representations, virtual, 25–29 April 2022.
16. Killian, T.W.; Zhang, H.; Subramanian, J.; et al. An empirical study of representation learning for reinforcement learning in healthcare. In Proceedings of the Machine Learning for Health Workshop, virtual, 11 December 2020; pp. 139–160.
17. Loftus, T.J.; Filiberto, A.C.; Li, Y.; et al. Decision analysis and reinforcement learning in surgical decision-making. *Surgery* **2020**, *168*, 253–266.
18. Datta, S.; Li, Y.; Ruppert, M.M.; et al. Reinforcement learning in surgery. *Surgery* **2021**, *170*, 329–332.
19. Wu, X.; Li, R.; He, Z.; et al. A value-based deep reinforcement learning model with human expertise in optimal treatment of sepsis. *npj Digit. Med.* **2023**, *6*, 15.
20. Zhang, T.; Qu, Y.; Wang, D.; et al. Optimizing sepsis treatment strategies via a reinforcement learning model. *Biomed. Eng. Lett.* **2024**, *14*, 279–289.
21. Huang, Y.; Cao, R.; Rahmani, A.M. Reinforcement learning for sepsis treatment: A continuous action space solution. In Proceedings of the 7th Machine Learning for Healthcare, Durham, NC, USA, 5–6 August 2022; pp. 631–647.
22. Kumar, A.; Zhou, A.; Tucker, G.; et al. Conservative Q-learning for offline reinforcement learning. In Proceedings of the Advances in Neural Information Processing Systems, virtual, 6–12 December 2020; Volume 33, pp. 1179–1191.
23. Fujimoto, S.; Meger, D.; Precup, D. Off-policy deep reinforcement learning without exploration. In Proceedings of the 36th International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 2052–2062.
24. Altman, E. *Constrained Markov Decision Processes*; CRC Press: Boca Raton, FL, USA, 1999.
25. Achiam, J.; Held, D.; Tamar, A.; et al. Constrained policy optimization. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 22–31.
26. Tessler, C.; Mankowitz, D.J.; Mannor, S. Reward constrained policy optimization. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
27. Chow, Y.; Ghavamzadeh, M.; Janson, L.; et al. Risk-constrained reinforcement learning with percentile risk criteria. *J. Mach. Learn. Res.* **2017**, *18*, 6070–6120.
28. Yang, Q.; Simao, T.D.; Tindemans, S.H.; et al. WCSAC: Worst-case soft actor critic for safety-constrained reinforcement learning. In Proceedings of the AAAI Conference on Artificial Intelligence, virtual, 2–9 February 2021; Volume 35, pp. 10639–10646.
29. Thananjeyan, B.; Balakrishna, A.; Nair, S.; et al. Recovery RL: Safe reinforcement learning with learned recovery zones. *IEEE Robot. Autom. Lett.* **2021**, *6*, 4915–4922.

30. Le, H.; Voloshin, C.; Yue, Y. Batch policy learning under constraints. In International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 3703–3712.
31. Alshiekh, M.; Bloem, R.; Ehlers, R.; et al. Safe reinforcement learning via shielding. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32, pp. 2669–2678.
32. Könighofer, B.; Lorber, F.; Jansen, N.; et al. Shield synthesis for reinforcement learning. In Proceedings of the 9th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2020), Rhodes, Greece, 20–30 October 2020; Volume 12476, pp. 290–306.
33. Ames, A.D.; Coogan, S.; Egerstedt, M.; et al. Control barrier functions: Theory and applications. In Proceedings of the European Control Conference, Naples, Italy, 25–28 June 2019; pp. 3420–3431.
34. Dalal, G.; Dvijotham, K.; Vecerik, M.; et al. Safe exploration in continuous action spaces. *arXiv* **2018**, arXiv:1801.08757.
35. Bertsekas, D.P. *Nonlinear Programming*, 2nd ed.; Athena Scientific: Nashua NH, USA, 1999.
36. Johnson, A.E.; Pollard, T.J.; Shen, L.; et al. MIMIC-III, a freely accessible critical care database. *Sci. Data* **2016**, *3*, 1–9.
37. Mnih, V.; Kavukcuoglu, K.; Silver, D.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533.