

A APPENDIX

A.1 FIGURES AND TABLES

Figure 5: Performance of RL in different hand-crafted reward designs to optimize clinical efficacy.

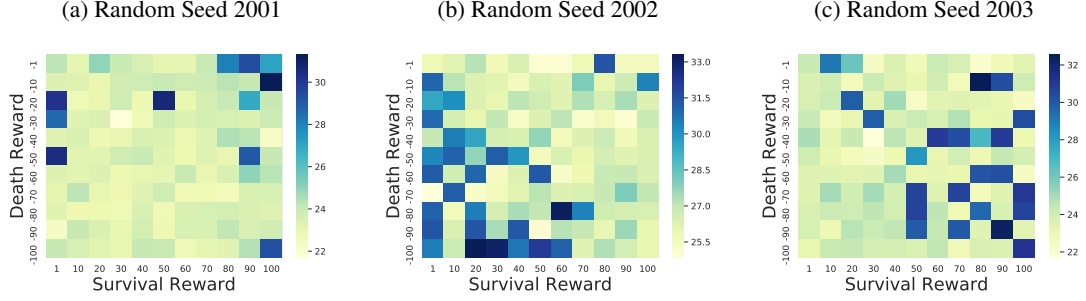


Figure 6: Performance of Multi-objective RL with linear scalarization on three reward factors: survival rate, last tumor size, and maximum toxicity levels. The linear weight assigned to each factor is one of four values: $\{1, 2, 4, 8\}$.

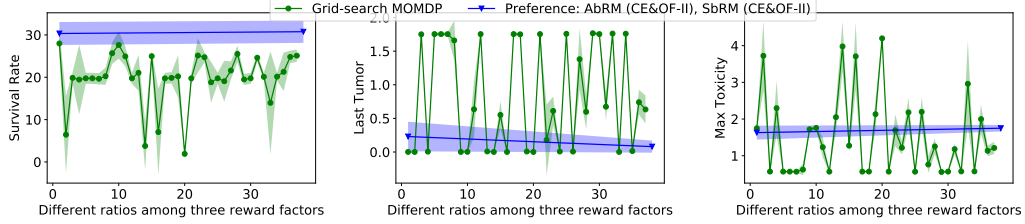


Figure 7: Cancer treatment strategies recommended by agent AbRM: (a) clinical efficacy and expected return during training, and (b) expected return of policies ending with different negative impacts during testing.

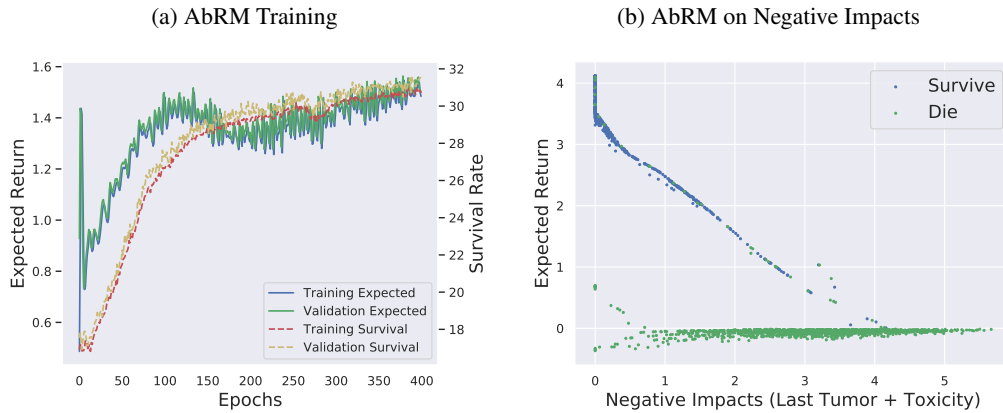


Figure 8: Effects of different agent designs on performance.

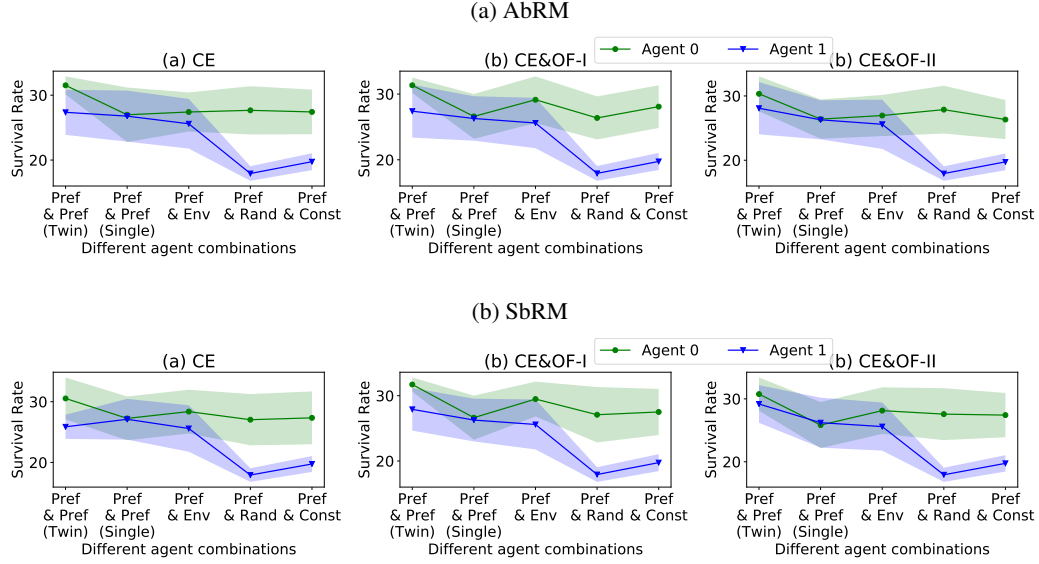
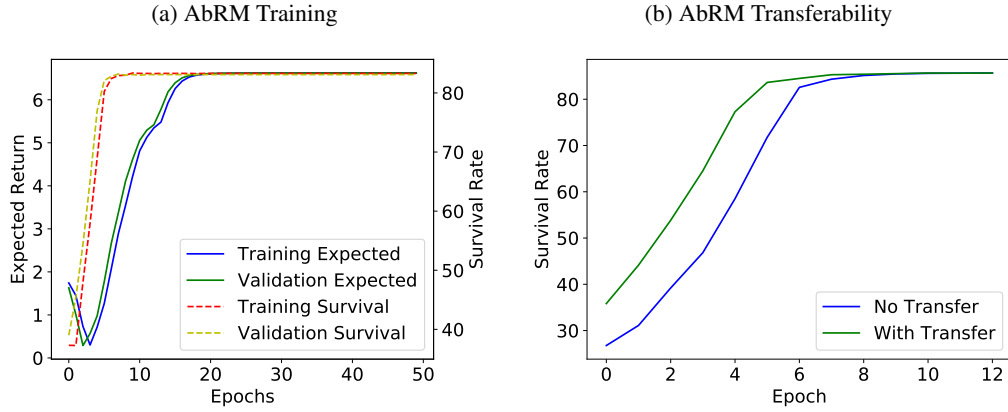
Figure 9: Sepsis treatment strategies recommended by the *AbRM* agent: (a) clinical efficacy and expected return during training, (b) reward transferability among different configurations.

Figure 10: For Cancer experiments, true expected return of DQN learning from behavioral policies of Policy Gradient and its estimations from different off-policy evaluation methods.

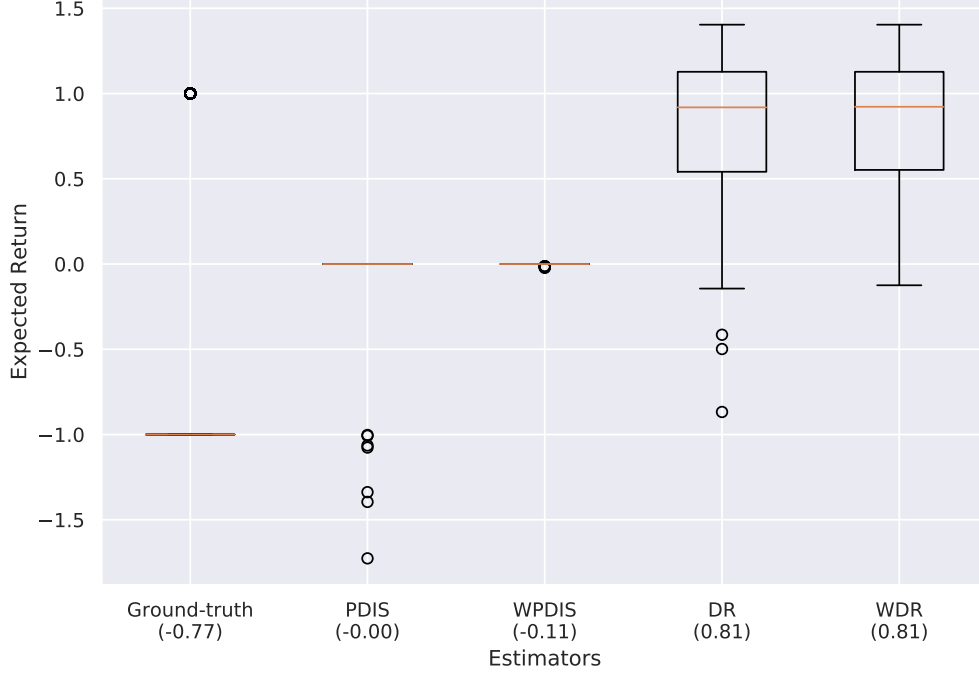


Table 2: Performance for Cancer medication recommendation considering negative impacts from two factors: the tumor size in the end and the ever experienced maximum toxicity.

Method Type	Method Name	Clinical Efficacy	Other Factors	
		Survival Rate	Last Tumor	Max Toxicity
Non-learning	Constant Best (0.4)	19.91%±0.58%	1.76±0.02	0.57±0.01
	Constant Worst (0.1)	4.89%±0.68%	3.72±0.03	0.48±0.04
	Random	17.81%±0.91%	0.82±0.02	1.64±0.04
Preference Learning	PBPI	20.80%±0.56%	1.38±0.12	1.01±0.12
Reinforcement Learning (Hand-crafted reward)	Single-objective RL	26.96%±3.02%	0.48±0.28	1.37±0.28
	Single-objective RL (<i>Ensemble</i>)	27.38%±3.32%	0.47±0.29	1.38±0.29
	Existing Multi-objective RL	18.84%±5.77%	0.25±0.10	2.16±0.60
	Grid-search Multi-objective RL	25.10%±1.44%	0.63±0.20	1.21±0.16
Reinforcement Learning (Preference-based reward)	AbRM (<i>CE</i>)	31.52%±1.38%	0.11±0.10	1.74±0.06
	AbRM (<i>CE&OF-II</i>)	30.34%±2.71%	0.23±0.22	1.63±0.19
	SbRM(<i>CE</i>)	30.54%±3.46%	0.18±0.25	1.67±0.22
	SbRM(<i>CE&OF-II</i>)	30.76%±2.64%	0.08±0.09	1.75±0.09

Table 3: Evaluating the clinical efficacy (survival rate) achieved by the proposed preference-based RL framework when hand-crafted and preference-based rewards are linear combined with different ratios.

Ratios	AbRM (CE)	AbRM (CE&OF-I)	AbRM (CE&OF-II)	SbRM (CE)	SbRM (CE&OF-I)	SbRM (CE&OF-II)
8:1	28.36%±3.09%	28.07%±3.50%	28.23%±3.82%	29.10%±3.19%	26.51%±2.47%	28.94%±3.17%
4:1	28.92%±2.82%	27.41%±3.66%	27.93%±3.61%	27.77%±3.16%	28.45%±3.18%	27.54%±3.09%
2:1	26.30%±3.19%	27.50%±3.64%	27.71%±3.47%	27.55%±3.40%	27.81%±3.15%	28.43%±3.12%
1:1	27.42%±3.24%	27.70%±3.73%	27.62%±2.88%	27.99%±2.9%	28.49%±3.23%	28.71%±3.50%
1:2	26.96%±2.95%	27.55%±2.90%	28.66%±3.19%	28.59%±2.94%	28.79%±3.21%	28.30%±3.29%
1:4	27.46%±3.53%	26.59%±3.40%	28.25%±3.47%	29.91%±1.93%	28.75%±3.06%	27.08%±3.64%
1:8	27.15%±3.00%	29.21%±2.42%	27.67%±3.38%	29.18%±2.45%	28.55%±2.85%	28.48%±3.28%
0:1 (Ours)	31.52%±1.38%	31.33%±1.18%	30.34%±2.71%	30.54%±3.46%	31.72%±1.08%	30.76%±2.64%

A.2 PREFERENCE-BASED REINFORCEMENT LEARNING ALGORITHMS

The preference-based Reinforcement Learning framework is composed of two main modules, *Preference-based Reward Learning* and *Preference-guided Agent Learning*. In *Preference-based Reward Learning*, the reward estimator parameterized by θ_P delivers step-wise rewards to the two agents parameterized by θ_A^1 and θ_A^2 based on their policy preference. In *Preference-guided Agent Learning*, the agents update their parameters so as to optimize the clinicians' objectives. The pair of policies performed by the two agents on the sampled subject is stored in the policy pool and leveraged for parameter update in reward estimator, with the aim to ensure higher expected return for the preferred policy. We list the pseudo codes for collaborative learning in Algorithm 1, *Preference-based Reward Learning* in Algorithm 2, and *Preference-guided Agent Learning* in Algorithm 3, respectively.

Collaborative Learning Algorithm 1 illustrates the collaborative learning process between the two modules in order to estimate reward and learn policies in personalized treatment recommendation. In the beginning, the model parameters are randomly initialized (line 1), and the policy pools for the reward estimator and the two agents are created as empty sets (line 2). In each iteration, one subject is sampled from the training set for agent learning (line 3 to 5). At each simulation step, the two agents are asked to make decisions based on the current state and the reward estimator generates corresponding step-wise reward for each of them (line 6 to 10). The subject's internal state keeps on updating until the simulation time has reached or the subject dies intermediately according to the underlying mathematical modeling. The policy pools of the two agents are augmented with the trajectories on the newest sampled subject (line 9 and 11), while the policy pool for the reward estimator is also updated (line 13) after computing the ground-truth preference label (line 12). After all the samples have been utilized for policy generation, the reward estimator minimizes the classification loss during policy preference inference with Algorithm 2 (line 16), while the RL agents optimize the expected return with Algorithm 3 (line 17).

Algorithm 2 PREFERENCE-BASED REWARD LEARNING

Require:

\mathcal{D}_n : sampled policy pairs in n-th iteration
 θ_P : parameters to update in reward function
 γ_P : discounted factor on reward
 β : step size for parameter update

```

1:  $L \leftarrow 0$ 
2: for all  $(\tau^1, \tau^2, pre(\tau^1, \tau^2)) \in \mathcal{D}_n$  do
3:    $R(\tau^1; \theta_P) \leftarrow 0, R(\tau^2; \theta_P) \leftarrow 0$ 
4:   for all  $(s_t^1, a_t^1, r_{\theta_P, t}^1, s_{t+1}^1) \in \tau^1$  do
5:      $R(\tau^1; \theta_P) \leftarrow R(\tau^1; \theta_P) + \gamma_P^t r_{\theta_P, t}^1$ 
6:   end for
7:   for all  $(s_t^2, a_t^2, r_{\theta_P, t}^2, s_{t+1}^2) \in \tau^2$  do
8:      $R(\tau^2; \theta_P) \leftarrow R(\tau^2; \theta_P) + \gamma_P^t r_{\theta_P, t}^2$ 
9:   end for
10:  Compute  $p(\tau^1 \succ \tau^2)$ 
11:  if  $\tau^1 \succ \tau^2$  then
12:     $L \leftarrow L + \log p(\tau^1 \succ \tau^2)$ 
13:  else if  $\tau^2 \succ \tau^1$  then
14:     $L \leftarrow L + \log (1 - p(\tau^1 \succ \tau^2))$ 
15:  else if  $\tau^1 \sim \tau^2$  then
16:     $L \leftarrow L + 0.5 \log p(\tau^1 \succ \tau^2) + 0.5 \log (1 - p(\tau^1 \succ \tau^2))$ 
17:  end if
18: end for
19: Update  $\theta_P \leftarrow \theta_P - \beta \Delta_{\theta_P} L$ 
20: return  $\theta_P$ 

```

Preference-based Reward Learning Given pairs of policies with corresponding preferences, the reward estimator updates its parameters to maximize the probability that the preferred policy achieves

higher expected return than the other. As shown in Algorithm 2, the discounted expected returns achieved by each agent are firstly calculated respectively for each sampled policy pair (line 3 to 9). Then the probability that policy τ^1 is preferred to τ^2 is positively correlated to the expected return of τ^1 , and is computed as (Agresti & Kateri, 2011) introduced (line 10). Hence $p(\tau^2 \succ \tau^1)$ is equal to $1 - p(\tau^1 \succ \tau^2)$. Then the loss value is computed considering different kinds of preference relationships between the two policies (line 11 to 17). Incomparable policy pairs are also leveraged in reward learning for better preference space exploration (line 15 to 16).

Algorithm 3 PREFERENCE-GUIDED AGENT LEARNING

Require:

Γ_n : sampled policies from one agent in n-th iteration
 α : step size for parameter update
 \mathcal{M} : one of the two reward assignment methods
 L_{θ_A} : loss function in any deep RL approach parameterized by agent parameters θ_A
 1: $\varepsilon = \emptyset$
 2: **for all** $(s_t, a_t, r_{\theta_P, t}, s_{t+1}) \in \Gamma_n$ **do**
 3: **if** \mathcal{M} is *Action-based Reward Modification* **then**
 4: $r_t \leftarrow r_{\theta_P}(s_t, a_t)$
 5: **else if** \mathcal{M} is *State-based Reward Modification* **then**
 6: $r_t \leftarrow h_{\theta_P}(s_t) - h_{\theta_P}(s_{t-1})$
 7: **end if**
 8: $\varepsilon \leftarrow \varepsilon \cup \{(s_t, a_t, r_t, s_{t+1})\}$
 9: **end for**
 10: Update $\theta_A \leftarrow \theta_A - \alpha \Delta_{\theta_A} \sum_{x \in \varepsilon} L_{\theta_A}(x)$
 11: **return** θ_A

Preference-guided Agent Learning Each agent updates their parameters individually as Algorithm 3 depicts. The agent receives rewards computed by either *Action-based Reward Modification* (line 3 to 4) or *State-based Reward Modification* (line 5 to 6). Then we leverage (s_t, a_t, r_t, s_{t+1}) to update the agent model implemented by any deep Reinforcement Learning approach.

A.3 SIMULATION PLATFORM DESIGN

A.3.1 MEDICATION RECOMMENDATION FOR GENERAL CANCER

Survival Analysis Within time interval $(t-1, t]$, where $(1 \leq t \leq 6)$, the survival status is assumed to depend on both the current tumor size y_t and the toxicity level x_t . The probability of a patient's death is modeled as follows:

Hazard function: $\lambda(t) = \exp(-4 + y_t + x_t)$, Cumulative hazard function: $\Delta\Delta(t) = \int_{t-1}^t \lambda(s)ds$,
 Survival function: $\Delta F(t) = \exp(-\Delta\Delta(t))$, Death probability: $p_{\text{death}} = 1 - \Delta F(t)$.

Implementation Details The action space is discrete and the dosage amount decisions are selected among 4 options: 0.1, 0.4, 0.7, 1.0 (Fürnkranz et al., 2012). For state initialization, the tumor size and the toxicity level in the 0th month are generated independently from the uniform distribution $\mathcal{U}(0, 2)$. The simulation terminates after $t = 6^{\text{th}}$ month or if the patient dies intermediately.

Model Implementation and Training For 6-month simulation, we randomly sample 10,000 subjects for training, 2,000 for validation, and 2,000 for testing. The neural networks for all deep learning approaches including *preference learning* and *reinforcement learning* share the similar network structure and hyper-parameters: 2 fully-connected layers, the first followed by ReLU activation and the second followed by different activation functions for different approaches. In one epoch, the agent gets updated after seeing all the training samples. The learning rate is set to 0.01 and all the networks converge after 400 epochs. For deep RL methods, we set the discount factor γ to 1.

A.3.2 BLOOD PURIFICATION RECOMMENDATION FOR SEPSIS

Mathematical Modeling in Simulation Sepsis is initiated by spillover of pathogens into blood, where the pathogen is allowed to spread throughout the organism in which systemic inflammation takes place (Stojkovic et al., 2016). Motivated by the promising results of blood purification in other critical illness conditions like acute kidney failure (Ronco et al., 2000), blood purification has gained attention as a potentially effective solution for septic subjects (Rimmelé & Kellum, 2011). In blood purification treatment, the patient is connected to an extracorporeal hemoabsorption device that removes harmful particles from the blood and leads the patient towards a healthy state.

We employ the mathematical model derived by Song *et al.* to simulate the acute inflammation process in response to an infection (Song et al., 2012). Both heuristic knowledge about the mechanism underlying infection and real measurements from experiments on CLP-induced septic rats were leveraged for the model design. The distribution of initial physiological features and their interactions are derived from domain knowledge. The initial physiological features that characterize a subject accords with the probability distributions based on real experimental measurements for septic rats. The parameters in transition functions are calibrated so that the generated trajectories closely follow experimentally observed temporal patterns in septic rats.

Figure 12 demonstrates the feature interaction network. There are 19 physiological features that govern sepsis dynamics, 8 of which are observable (features above the horizontal dashed line) while the remaining 11 are conceptual variables (features below the horizontal dashed line). When a blood purification operation is made, three components in the circulation are eliminated (features marked by red dashed ring), i.e., activated neutrophils N_a and the pro- and anti-inflammatory mediators PI and AI . Besides effects from the blood purification operation, the variables influence each others' progression through Ordinary differential equations (ODEs).

State Transition There are 18 ODEs to describe feature interactions and 3 ODEs for the hypothetical mechanism of blood purification. The hypothetical mechanisms of action of the blood purification are implemented by assuming the hemoabsorption device eliminates only three components in the circulation: activated neutrophils (N_a), pro-inflammatory mediators (PI), and anti-inflammatory mediators (AI) during the treatment period. We here only show the transition equation of these three key features with and without operation, ODEs concerning other features can be found in (Song et al., 2012).

The variable PI stands for the extent of the systemic inflammation and progresses as follows:

$$\begin{aligned} \frac{dPI}{dt} = & \left(\frac{B/B_\infty}{h_{PI_B} + B/B_\infty} \left(1 - \frac{D^n}{h_{PI_D}^n + D^n} \right) \left(1 - \frac{AI^n(1-PI)}{h_{PI_AI}^n + AI^n} \right) \right. \\ & + \left(1 - \frac{B/B_\infty}{h_{PI_B} + B/B_\infty} \right) \frac{D^n}{h_{PI_D}^n + D^n} \left(1 - \frac{AI^n(1-PI)}{h_{PI_AI}^n + AI^n} \right) \\ & \left. + \frac{B/B_\infty}{h_{PI_B} + B/B_\infty} \frac{D^n}{h_{PI_D}^n + D^n} \left(1 - \frac{AI^n(1-PI)}{h_{PI_AI}^n + AI^n} \right) - PI \right) \frac{1}{\tau_{PI}}, \end{aligned} \quad (3)$$

$$PI(t' + 1) = \begin{cases} PI(t') + \frac{dPI}{dt}(t') & \text{If no operation is performed} \\ PI(t') + \frac{dPI}{dt}(t') - \frac{PI}{h_{PIHA} + PI} & \text{Otherwise} \end{cases}, \quad (4)$$

where B is the population of bacteria in the peritoneum, D is a coarse-grained representation of integrated tissue damage, variables $h_{PI_B}, h_{PI_D}, h_{PI_AI}, \tau_{PI}$ are subject-specific parameters, B_∞ is a predefined upper bound of B , $h_{PIHA} = 0.3$ and $n = 3$.

Table 4: Configurations for Sepsis treatment simulation. h represents hour in the simulation platform.

Step Size τ	Horizon Length T	Operation Time Interval L	Decision-making Frequency f	Duration per Operation l
$\tau = 0.1$ h	$T = 100$ h	5^{th} to 18^{th} h	$f = 2$ h or 4h	$l = 2$ h or 4h

The variable AI describes the level of the anti-inflammation corresponding to systemically acting anti-inflammatory mediators and gets updated as follows:

$$\frac{dAI}{dt} = \left(\frac{PI^{n_1}}{h_{AI_PI}^{n_1} + PI^{n_1}} \left(1 - \frac{N_a/N_\infty}{h_{AI_N_a} + N_a/N_\infty} \right) + \left(1 - \frac{PI^{n_1}}{h_{AI_PI}^{n_1} + PI^{n_1}} \right) \frac{N_a/N_\infty}{h_{AI_N_a} + N_a/N_\infty} \right. \quad (5)$$

$$\left. + \frac{PI^{n_2}}{h_{AI_PI}^{n_2} + PI^{n_2}} \frac{N_a/N_\infty}{h_{AI_N_a} + N_a/N_\infty} - AI \right) \frac{1}{\tau_{AI}},$$

$$AI(t' + 1) = \begin{cases} AI(t') + \frac{dAI}{dt}(t') & \text{If no operation is performed} \\ AI(t') + \frac{dAI}{dt}(t') - \frac{AI}{h_{AI_HA} + AI} & \text{Otherwise} \end{cases}, \quad (6)$$

where variables $h_{AI_PI}, h_{AI_N_a}, \tau_{AI}$ are subject-specific parameters, N_∞ is a predefined upper bound of neutrophils, $h_{AI_HA} = 0.3$, $n_1 = 1$ and $n_2 = 3$.

The variable N_a represents the activated blood neutrophils and transits in each simulation step as follows:

$$\frac{dN_a}{dt} = \underbrace{\frac{N_r PI^n}{h_{N_r_N_a}^n + PI^n} \frac{1}{\tau_{N_r_N_a}}}_{\text{transmission from } N_r \text{ to } N_a} + \underbrace{\frac{N_p PI^n}{h_{N_p_N_a}^n + PI^n} \frac{1}{\tau_{N_p_N_a}}}_{\text{transmission from } N_p \text{ to } N_a} - \frac{N_a}{\tau_{N_a}} - \underbrace{\frac{N_a PI^n}{h_{N_a_N_s}^n + PI^n} \frac{1}{\tau_{N_a_N_s}}}_{\text{transmission from } N_a \text{ to } N_s}, \quad (7)$$

$$N_a(t' + 1) = \begin{cases} N_a(t') + \frac{dN_a}{dt}(t') & \text{If no operation is performed} \\ N_a(t') + \frac{dN_a}{dt}(t') - \frac{N_a/N_\infty}{h_{N_a_HA} + N_a/N_\infty}(t') & \text{Otherwise} \end{cases}, \quad (8)$$

where N_r is resting blood neutrophils, N_p is blood neutrophils, N_s is neutrophils sequestered in the lung capillaries, variables $h_{N_r_N_a}, h_{N_p_N_a}, h_{N_a_N_s}, \tau_{N_r_N_a}, \tau_{N_p_N_a}, \tau_{N_a_N_s}$ are subject-specific parameters, $h_{N_a_HA} = 0.3$, and $n = 3$.

Survival Analysis The survival status of the subject only depends on the value of the systemic pro-inflammatory response PI at the end of the simulation. When the PI value at the last time-step is smaller than the pre-defined threshold 0.5, then the subject is assumed to be alive, otherwise dead. Note that after the blood purification process, the PI value reduces as time passes, hence one cannot conclude whether the subject is alive in the intermediate time-steps. After the pre-defined simulation horizon is reached, we can confirm which subjects survive with the help of treatment. The mathematical model is quite different from the general Cancer Treatment model where subjects have a probability to die intermediately.

Implementation Details Due to phenotype differences, some subjects survive without any blood purification operation while some die. This is consistent with laboratory experiments where 30% of rats survived till seven days while the remaining died between two to five days after CLP (Zhao et al., 2009). We call the survivor group *Survival Population* and the non-survivor group *Death Population*. The survival status of the *Survival Population* gets no influence from blood purification operations. Subjects from *Death Population* have the potentials to survive if proper treatment policies are delivered. Since we are primarily concerned about the outcomes on subjects from *Death Population*, we only sample subjects from the *Death Population* in this paper to train and evaluate treatment policies.

There are a few hyper-parameters that should be set in advance: 1) Simulation step size τ : every τ time, the simulator updates the internal status of subjects by computing the ODEs with feature values from the last simulation step and the current action. 2) Simulation horizon length T : we can evaluate the performance of a policy by checking outcomes of subjects after time T . 3) Valid time

range L for patients to receive treatment: operations can take place at any time-step ($L = [0, T - 1]$) or be constrained to predefined time intervals ($L \subsetneq [0, T - 1]$). 4) Frequency of decision-making f : subjects can receive operations at each simulation step τ or less frequently. 5) Duration of each blood purification operation l : it takes some costs to turn on/off the purification device and it is also unrealistic to attach and detach the device from the subject too frequently. Therefore, there should be a pre-defined value for the purification duration to rule out the possibility of too frequent actions.

To generate testable hypotheses that guide future laboratory experiments (Song et al., 2012; Stojkovic et al., 2016), the simulation of sepsis evolution should be configured to make the generated trajectory closely follow experimentally observed temporal patterns (Song et al., 2012). Further, several constraints can be imposed on the simulation in accordance with previous blood purification studies (Song et al., 2012; Stojkovic et al., 2016). Therefore we use the configuration listed in Table. 4 for experiments.

Model Implementation and Training We randomly sample 3,000 subjects for training, 1,000 for validation, and 1,000 for testing. Implementation details of the deep RL approaches are similar to those mentioned in the Cancer task, except that the backend network is LSTM-based since this is a POMDP.

Learning efficient treatment policies for Septic subjects is more difficult for Cancer due to the larger state space and the partially observable environment. Therefore, we adopt the following methods to ensure robust learning: 1) Mini-batch gradient descent with batch size 10,000 is adopted to update parameters in reward estimator and RL agents. 2) The learning rate for RL agents is 0.01 while 0.001 for the reward estimator. 3) As discussed in Experiment Section, experience replay makes the estimated reward positively proportional to the *Survival Rate*. We randomly extract policy pairs from the latest 30,000 samples for model updates.

Figure 11: Distribution of state features (tumor size and toxicity level) from Cancer subjects without tre

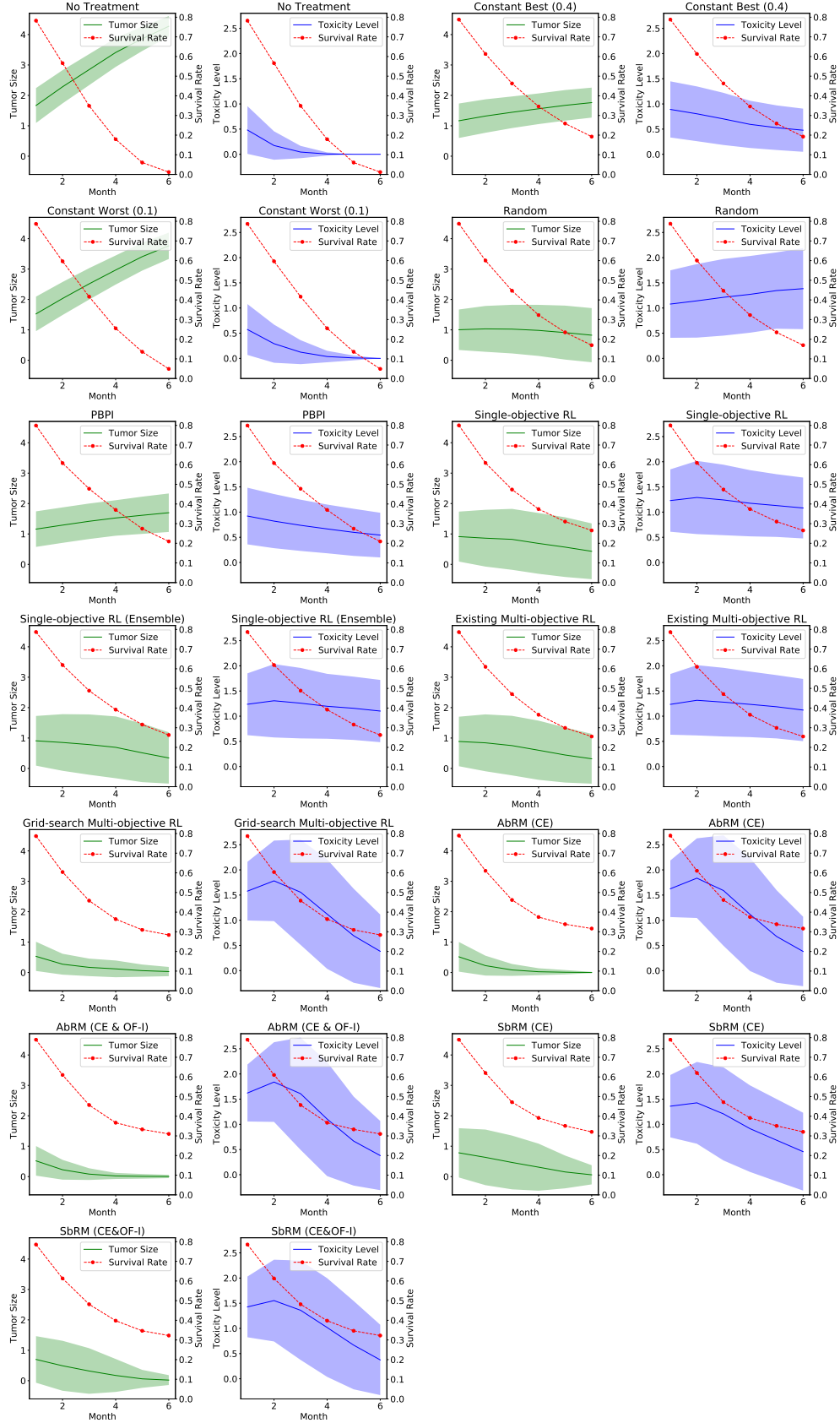


Figure 12: Interaction network of inflammatory responses and hypothetical hemoadsorption mechanisms of action in CLP-induced sepsis. Nodes in green represent components in peritoneum, nodes in orange stand for blood components, and nodes in purple stand for lung components. Edges represent network interactions under blood purification treatment compiled from literature. When blood purification is performed, only features *PI*, *AI* and *Na* are influenced (marked by red dashed rings).

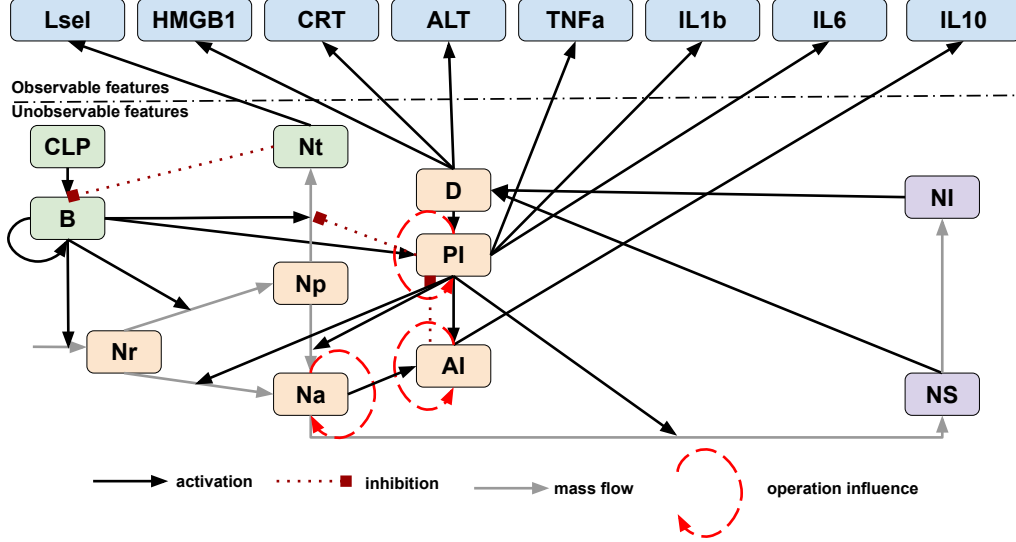


Figure 13: Distribution of 19 state features from Septic subjects without treatment or with treatment from SbRM.

