# Extended Abstract

**Motivation** Stimulated geologic hydrogen generates hydrogen through the serpentinization reaction $2FeO + H_2O \rightarrow Fe_2O_3 + H_2$, which is most active within a narrow temperature window of approximately 200–260°C. Many accessible ultramafic reservoirs are cooler than this range, motivating the use of hot-water circulation to externally heat the reservoir. However, substantial heating costs are incurred long before hydrogen production begins, making project economics highly sensitive to how water is injected over time. This naturally creates a long-horizon sequential decision problem. We investigate whether modern reinforcement learning and imitation-learning methods can learn effective control policies under this extreme reward delay.

**Method** We develop a simplified physics-based 2D reservoir simulator for stimulated geologic hydrogen production. The simulator couples fluid flow, heat transport, temperature-dependent hydrogen generation, and FeO depletion. At each decision step, an agent controls the injection rate and temperature of 10 wells using a compact reservoir-state representation. Rewards are defined as net energy production, with an additional thermal-progress bonus to provide dense learning signals before hydrogen generation begins.

Episodes span three years (1,095 days), while meaningful hydrogen rewards emerge only after approximately 220 days. We compare five control strategies: (1) heuristic policies, (2) Soft Actor-Critic (SAC), (3) Behavioral Cloning (BC), (4) Implicit Q-Learning (IQL), and (5) BC-initialized SAC.

**Implementation** All methods use a shared neural network architecture consisting of two hidden layers with 256 units each. SAC employs automatic entropy tuning and experience replay, while BC is trained on expert trajectories generated by the thermostat controller. IQL is trained offline on positive-return trajectories collected from heuristic policies. Hyperparameters were selected through preliminary tuning and held fixed across experiments.

**Results** The long reward delay proved challenging for standard online RL. SAC frequently collapsed to 'lazy policy', i.e. no injection to avoid heating cost, or became unstable during training. In contrast, BC achieved the best performance, producing 59.5M mol $H_2$ and a net energy return of +39.1M MJ, outperforming the thermostat expert by 4.1% and 6.9%, respectively. IQL improved upon raw SAC but remained limited by the quality and coverage of the offline dataset. Initializing SAC from a pretrained BC policy yielded positive returns but failed to consistently improve upon the BC baseline. Overall, simple imitation learning outperformed both online and offline reinforcement learning approaches in this highly delayed-reward control problem.

**Discussion and Conclusion** Our results suggest that delayed credit assignment, rather than reward scaling or algorithmic design choices, is the dominant challenge in this problem. The approximately 220-day lag between heating actions and positive hydrogen rewards makes it difficult for online RL methods to propagate useful learning signals across hundreds of decision steps. In contrast, BC learned a smoother version of the thermostat controller and unexpectedly surpassed its demonstrator, highlighting how simple imitation learning can benefit from the inductive bias of neural function approximation. Overall, expert-guided imitation learning proved more effective than both online and offline RL in this highly delayed-reward setting. Future work will investigate methods specifically designed for long-horizon credit assignment, including multi-step returns, PPO with high-$\lambda$ GAE, and model-based planning approaches.

# Sequential Injection Control for Optimal Stimulated Geologic Hydrogen Production through Deep Reinforcement Learning

**Zhihao (Spencer) Zhang**
Energy Science & Engineering
Stanford University
zhang99@stanford.edu

## Abstract

Many scientific control problems are characterized by sparse rewards, long physical delays, and expensive experimentation, making them challenging for reinforcement learning (RL). We study this setting in the context of stimulated geologic hydrogen production, where heat injected into ultramafic reservoirs may require months before generating economically valuable hydrogen. To investigate this challenge, we develop a physics-based benchmark that couples fluid flow, heat transport, temperature-dependent hydrogen generation, and FeO depletion within a reservoir-scale simulator. The resulting control problem exhibits severe delayed rewards, with meaningful hydrogen production emerging approximately 220 days after heating decisions are made.

We compare heuristic control, Behavioral Cloning (BC), Soft Actor-Critic (SAC), Implicit Q-Learning (IQL), and BC-initialized SAC. Despite multiple stabilization strategies, SAC frequently converged to low-injection policies or exhibited unstable training dynamics. In contrast, BC trained on a small number of expert demonstrations achieved the strongest performance, exceeding its thermostat-based teacher by 4.1% in cumulative hydrogen production and 6.9% in net energy return. IQL remained stable and achieved near-expert performance but did not surpass BC.

Our results suggest that delayed credit assignment, rather than exploration or reward scaling, is the dominant challenge in this benchmark. More broadly, they demonstrate that simple imitation learning can outperform both online and offline RL methods when rewards are separated from actions by long physical timescales.

## 1 Introduction

Geologic hydrogen has recently emerged as a potential low-carbon energy resource generated through water-rock reactions in iron-bearing ultramafic formations. Hydrogen generation through serpentinization is strongly temperature dependent, with the most favorable reaction window occurring between approximately 200–260°C. However, many accessible subsurface reservoirs are substantially cooler, limiting hydrogen production rates.

One proposed solution is **stimulated geologic hydrogen production**, in which hot water is circulated through ultramafic reservoirs to raise temperatures into the reactive window. Prior thermo-reactive reservoir studies suggest that reservoir heating can be energetically favorable over long timescales, but project performance depends critically on how injection temperatures and flow rates are controlled over time.

This naturally motivates a sequential decision-making framework. At each timestep, an operator must balance competing objectives: accelerating thermal breakthrough, maximizing cumulative hydrogen

production, and minimizing heating and pumping costs. The resulting dynamics are governed by coupled fluid flow, heat transport, and temperature-dependent reaction kinetics, producing a high-dimensional nonlinear control problem.

Reinforcement learning (RL) has demonstrated success in a variety of complex control tasks and has increasingly been applied to subsurface energy systems. However, stimulated geologic hydrogen production introduces a particularly challenging feature: severe delayed rewards. In our environment, positive hydrogen-production rewards do not emerge until approximately 220 days after heating begins, requiring learning algorithms to assign credit across hundreds of decision steps.

In this work, we develop the first RL benchmark for stimulated geologic hydrogen production. We construct a physics-based thermo-reactive reservoir simulator that couples heat transport, fluid flow, hydrogen generation, and resource depletion. Using this benchmark, we evaluate heuristic control, Behavioral Cloning (BC), Soft Actor-Critic (SAC), and Implicit Q-Learning (IQL). Surprisingly, BC outperforms both online and offline RL methods. Our results suggest that delayed credit assignment, rather than exploration or reward scaling, is the primary challenge in this domain and highlight the need for RL methods capable of reasoning over long physical timescales.

## 2 Related Work

Reinforcement learning for reservoir control has been explored across several problem classes. Hourfar et al. (Hourfar et al., 2019) applied tabular Q-learning to waterflooding optimization on the Egg benchmark model, controlling injection rates and producer pressures to maximize NPV; rewards arrive at each monthly control step with no structural delay. Dixit and ElSheikh (Dixit and ElSheikh, 2022, 2023) used PPO and A2C for stochastic well control under geologic uncertainty, formulating the problem as a POMDP over saturation and pressure observations — the closest structural analogy to our work in terms of partial observability, though their physics is isothermal two-phase flow with no thermal coupling. Nasir and Durlofsky (Nasir and Durlofsky, 2023) applied PPO with a temporal-convolution transformer policy to life-cycle BHP optimization across uncertain geologic models, demonstrating that on-policy methods can handle multi-year horizons when control steps are coarsened to monthly or annual intervals. Zhang et al. (Zhang et al., 2022) trained SAC — the same algorithm we adopt — on continuous BHP control for NPV maximization; their reward accumulates incrementally at each control step rather than being structurally delayed.

A common pattern across this body of work is that rewards are *immediately informative*: oil production and pressure responses follow injection decisions within the current control interval (days to months), so the effective credit-assignment gap is short regardless of discount factor. Stimulated geologic hydrogen production breaks this assumption fundamentally. Serpentinisation is mediated by a temperature window (200–260°C) that cannot be reached until thermal energy has propagated through rock at roughly $13\times$ the fluid velocity, creating a structural $\sim$220-day gap between injection decisions and any positive reward. At our daily timestep resolution and $\gamma = 0.995$, this delay sits at the boundary of what one-step TD bootstrapping can resolve — a regime that, to our knowledge, has not been studied in the reservoir RL literature. We benchmark SAC, behavioral cloning, and offline IQL (Kostrikov et al., 2022) against hand-designed heuristics to characterize which approaches survive this credit-assignment gap.

## 3 Environment

### 3.1 Physics-Based Thermo-Reactive Reservoir Simulator

We develop a physics-based reservoir simulator for stimulated geologic hydrogen production. The environment models heat injection into an ultramafic reservoir, thermal transport through porous rock, temperature-dependent $H_2$ generation, and depletion of reactive FeO. This creates a long delay between heating actions and hydrogen production, making the environment a challenging benchmark for reinforcement learning.

The reservoir is represented as a 2-D vertical cross-section ($x$ horizontal, $z$ vertical) discretized on a $500 \times 100$ Cartesian grid with cell dimensions $\Delta x = 10$ m and $\Delta z = 1$ m, giving a total domain size of $5000 \times 100$ m. Ten injector–producer well pairs are placed at equal horizontal spacing. Injectors

are located at the reservoir base ($z = 0$), while producers are located at the top boundary ($z = 100$ m).

At each $\Delta t = 1$ day timestep, the simulator couples four processes: (i) steady-state Darcy flow, (ii) advective-diffusive heat transport, (iii) temperature-dependent $H_2$ generation, and (iv) FeO depletion. Key physical parameters are listed in Table 1.

Table 1: Simulation parameters.

| Symbol | Description | Value | Units |
|---|---|---|---|
| *Ultramafic rock* | | | |
| $T_0$ | Initial temperature | 140 | °C |
| $k$ | Permeability | $10^{-14}$ | $\mathrm{m}^2$ |
| $\phi$ | Porosity | 0.05 | – |
| $\lambda_r$ | Thermal conductivity | 2.9 | $\mathrm{W\,m^{-1}K^{-1}}$ |
| $\rho_r$, $c_{p,r}$ | Density, heat capacity | 3000, 850 | $\mathrm{kg\,m^{-3}}$; $\mathrm{J\,kg^{-1}K^{-1}}$ |
| $w_{\mathrm{FeO}}$ | FeO mass fraction | 0.10 | – |
| *Injected fluid* | | | |
| $\mu$, $\rho_f$ | Viscosity, density | $1.4 \times 10^{-4}$, 870 | Pa s; $\mathrm{kg\,m^{-3}}$ |
| $c_{p,f}$ | Heat capacity | 4500 | $\mathrm{J\,kg^{-1}K^{-1}}$ |
| *$H_2$ kinetics* | | | |
| $[T_{\mathrm{lo}}, T_{\mathrm{peak}}, T_{\mathrm{hi}}]$ | Reaction window | [200, 230, 260] | °C |
| $r_{\max}$ | Peak generation rate | $10^{-5}$ | $\mathrm{mol\,m^{-3}\,s^{-1}}$ |

**Fluid flow.** We solve a steady-state incompressible Darcy pressure equation:

$$\frac{k}{\mu}\nabla^2 P = -q_{\mathrm{vol}}, \qquad q_{\mathrm{vol}}^{(i)} = \frac{Q_{\mathrm{inj}}^{(i)}}{\Delta x\,\Delta z}, \tag{1}$$

where $Q_{\mathrm{inj}}^{(i)}$ is the volumetric injection rate per unit reservoir width at well $i$. Producer cells are assigned fixed pressure conditions, while domain boundaries use no-flow conditions.

**Heat transport.** Assuming local thermal equilibrium between fluid and rock, temperature evolves according to:

$$\frac{\partial T}{\partial t} = \alpha_T \nabla^2 T - \eta\,\mathbf{u}\cdot\nabla T, \qquad \alpha_T = \frac{\lambda_{\mathrm{eff}}}{\rho c_{\mathrm{eff}}}, \qquad \eta = \frac{\rho_f c_{p,f}}{\rho c_{\mathrm{eff}}}, \tag{2}$$

where

$$\lambda_{\mathrm{eff}} = \phi\lambda_f + (1-\phi)\lambda_r, \qquad \rho c_{\mathrm{eff}} = \phi\rho_f c_{p,f} + (1-\phi)\rho_r c_{p,r}.$$

The effective thermal retardation factor is:

$$\mathcal{R} = \frac{\rho c_{\mathrm{eff}}}{\phi\rho_f c_{p,f}} \approx 13. \tag{3}$$

This means the thermal front propagates roughly 13 times more slowly than the injected fluid. As a result, heating actions can take approximately 220 days before producing meaningful $H_2$ rewards at the producer wells. This physically induced delay is the main credit-assignment challenge studied in this work.

Heat transport is solved using an explicit finite-volume scheme with CFL-based sub-stepping.

**$H_2$ generation and FeO depletion.** Hydrogen generation is modeled using a simplified temperature-gated serpentinization reaction:

$$2\,\mathrm{FeO} + \mathrm{H_2O} \rightarrow \mathrm{Fe_2O_3} + \mathrm{H_2}.$$

The local reaction rate follows a bell-shaped temperature dependence:

$$r(T,\xi) = r_{\max}\max\left[1 - \left(\frac{T - T_{\mathrm{peak}}}{w_{1/2}}\right)^2, 0\right]\xi, \qquad w_{1/2} = \frac{T_{\mathrm{hi}} - T_{\mathrm{lo}}}{2} = 30\,^{\circ}\mathrm{C}, \tag{4}$$

where $\xi \in [0, 1]$ is the local fraction of unreacted FeO.

Each timestep updates the FeO potential:

$$\xi \leftarrow \xi - \frac{r(T, \xi)\Delta t}{\rho_{H_2}^{\max}},$$

where the maximum $H_2$ density follows from stoichiometry:

$$\rho_{H_2}^{\max} = \frac{w_{\text{FeO}}\rho_r}{2M_{\text{FeO}}} = \frac{0.10 \times 3000}{2 \times 0.07185} \approx 2087 \text{ mol m}^{-3}.$$

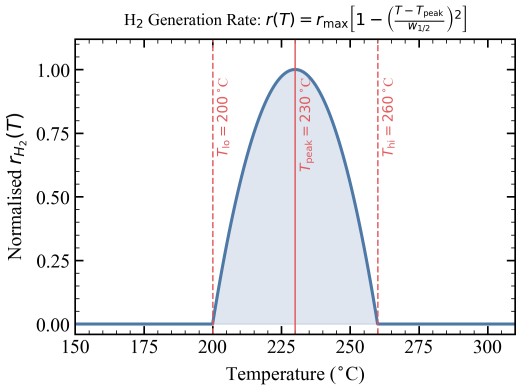



Figure 1: Simplified temperature-dependent reaction-rate scaling used for $H_2$ generation. Reaction is active between 200–260°C and peaks at 230°C.

### 3.2 MDP Formulation

**State and observation.** The full simulator state consists of temperature, pressure, and FeO potential fields:

$$\{T, P, \xi\} \in \mathbb{R}^{3\times 500\times 100},$$

corresponding to 150,000 state variables. Directly exposing this full field would make learning difficult and would include many cells far from active well regions. We therefore provide the agent with a compact 206-dimensional observation focused on well-adjacent temperatures and global reservoir statistics:

$$\mathbf{o}_t = \Bigg[ \underbrace{\tilde{T}^{(1)}_{w,0:19}, \ldots, \tilde{T}^{(10)}_{w,0:19}}_{\text{well-column temperature profiles, 200 dims}}, \bar{T}_{\text{norm}}, T_{\text{max,norm}}, f_{\text{rxn}}, \bar{\xi}, \tilde{H}_2, \frac{t}{T} \Bigg] \in \mathbb{R}^{206}. \tag{5}$$

Here $\tilde{T}^{(i)}_{w,j} = (T_{x^{(i)}_w, j} - T_0)/(T_{\max} - T_0)$ is the normalized temperature at depth $j$ beneath well $i$, $f_{\text{rxn}}$ is the fraction of cells in the 200–260°C reaction window, $\bar{\xi}$ is the mean remaining FeO potential, $\tilde{H}_2$ is the normalized cumulative $H_2$ yield, and $t/T$ is episode progress.

Because the agent cannot directly observe inter-well temperatures, the full pressure field, or the spatial distribution of FeO depletion, the task is partially observable.

**Action space.** At each timestep, the agent independently controls the injection rate and temperature of each of the 10 wells:

$$\mathbf{a}_t = \left[ Q^{(1)}_{\text{inj}}, \ldots, Q^{(10)}_{\text{inj}}, T^{(1)}_{\text{inj}}, \ldots, T^{(10)}_{\text{inj}} \right] \in \mathbb{R}^{20}. \tag{6}$$

The physical action bounds are:

$$Q^{(i)}_{\text{inj}} \in [0,\ 2 \times 10^{-4}]\ \text{m}^2\,\text{s}^{-1}, \qquad T^{(i)}_{\text{inj}} \in [140,\ 400]\ ^\circ\text{C}.$$

Policy outputs are normalized to $(-1, 1)$ and rescaled to physical units before environment interaction.

**Reward function.** The reward approximates net energy production in MJ. It rewards $H_2$ generation and penalizes heating and pumping costs:

$$r_t = \underbrace{n_{H_2,t}\Delta H_{H_2}}_{\text{H}_2\text{ energy}} - \underbrace{\frac{E_{\text{heat},t}}{10^6}}_{\text{heating cost}} - \underbrace{w_p \frac{E_{\text{pump},t}}{10^6}}_{\text{pumping cost}} + \underbrace{\lambda \bar{\phi}_t}_{\text{thermal shaping bonus}}, \tag{7}$$

where $n_{H_2,t}$ is the amount of $H_2$ generated at timestep $t$, $\Delta H_{H_2} = 0.2418$ MJ mol$^{-1}$ is the higher heating value of $H_2$, and $w_p = 1.0$.

The heating cost accounts for heat recycling by comparing injection temperature to producer outlet temperature:

$$E_{\text{heat},t} = \sum_{i=1}^{N_w} Q_{\text{prod}}^{(i)} \rho_f c_{p,f} \max\left(T_{\text{inj}}^{(i)} - T_{\text{prod}}^{(i)}, 0\right) \Delta t. \tag{8}$$

Once thermal breakthrough occurs and $T_{\text{prod}} \to T_{\text{inj}}$, reheating cost decreases substantially and sustained $H_2$ generation becomes energy-positive.

To provide a dense learning signal before $H_2$ production begins, we add a thermal progress shaping bonus:

$$\bar{\phi}_t = \frac{1}{N_w} \sum_{w=1}^{N_w} \text{clip}\left(\frac{T_{\text{prod},w} - T_{\text{natural}}}{T_{\text{peak}} - T_{\text{natural}}}, 0, 1\right), \qquad \lambda = 50{,}000 \text{ MJ}. \tag{9}$$

Here $T_{\text{natural}} = 140\,^\circ$C and $T_{\text{peak}} = 230\,^\circ$C. The shaping bonus is zero when producers remain at natural temperature and reaches its maximum when producers reach the peak reaction temperature. The value $\lambda = 50{,}000$ MJ was chosen so that full thermal breakthrough gives a bonus comparable to the dominant phase-1 heating cost, ensuring that early heating receives a meaningful learning signal.

Each episode spans $T = 1095$ timesteps, corresponding to approximately three years of operation. We use discount factor $\gamma = 0.995$, giving an effective horizon of approximately $1/(1-\gamma) = 200$ steps.

# 4 Experimental Setup

## 4.1 Heuristic Baseline Policies

Prior to learning-based experiments, we evaluate five hand-designed injection strategies to establish baseline performance and generate offline demonstration data.

`random` Samples injection rates and temperatures uniformly from the action bounds at every step.

`high_temp` Injects at maximum temperature ($400\,^\circ$C) with randomly sampled rates, rapidly heating the reservoir but incurring high energy cost.

`sweep_rate` Linearly ramps injection rate from $Q_{\min}$ to $Q_{\max}$ over the episode at a fixed mid-range temperature.

`greedy_heat` Allocates higher injection rates to the coldest well columns at each step, with injection temperature fixed at $0.85T_{\max} = 340\,^\circ$C.

`thermostat` A physics-inspired two-phase controller. During startup, all wells inject at $230\,^\circ$C to drive the reservoir toward the reaction window. Once producer temperature exceeds $180\,^\circ$C, injection temperature is reduced to $T_{\text{prod}} + 5\,^\circ$C to maintain reaction conditions while reducing reheating cost:

$$T_{\text{inj},w} = \begin{cases} 230\,^\circ\text{C}, & T_{\text{prod},w} < 180\,^\circ\text{C} \\ T_{\text{prod},w} + 5\,^\circ\text{C}, & T_{\text{prod},w} \geq 180\,^\circ\text{C} \end{cases} \qquad Q_w = Q_{\max}.$$

This policy serves as the BC teacher and primary performance target.

Each heuristic policy is rolled out for 5 episodes, producing 5,475 transitions per policy. Results are shown in Fig. 2 and Table 2.

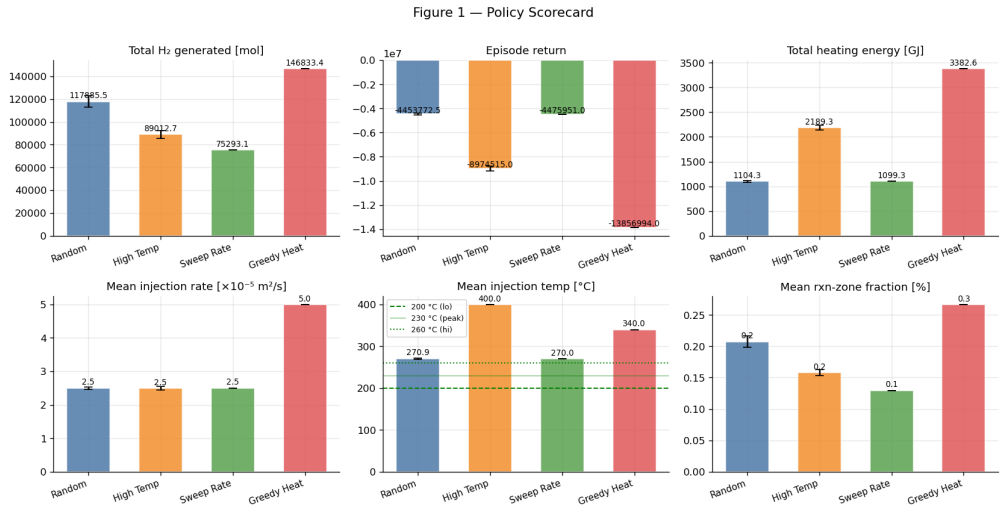



Figure 2: **Scorecard of heuristic reservoir-control policies.** Aggressive heating increases hydrogen production but can incur large energy penalties. The thermostat policy provides the strongest baseline by balancing early heating with later reheating-cost reduction.

Table 2: Heuristic policy comparison. Values are deterministic evaluations averaged over 5 episodes.

| **Policy** | **$H_2$ [mol]** | **Return [MJ]** | **Notes** |
|---|---|---|---|
| `random` | 15.2M | +4.7M | High variance across seeds |
| `high_temp` | 7.7M | −13.8M | Excessive heating cost |
| `sweep_rate` | 10.2M | −1.7M | Fixed deterministic baseline |
| `greedy_heat` | 10.2M | +10.0M | Best naive policy |
| `thermostat` | **57.2M** | **+36.6M** | **BC teacher** |

### 4.2 Soft Actor-Critic

We train Soft Actor-Critic (SAC) (Haarnoja et al., 2018) with automatic entropy tuning. Both the actor and twin critics are two-layer MLPs with 256 hidden units and ReLU activations. The actor outputs a stochastic Gaussian policy over the 20-dimensional action space, while the twin critics estimate Q-values from the concatenated observation-action vector. Main hyperparameters are listed in Table 4.

Because the environment contains large reward magnitudes and long delayed rewards, we evaluate several SAC stabilization strategies:

1. **Reward normalization:** Online Welford normalization is used to maintain approximately zero-mean, unit-variance rewards.
2. **Frozen seed buffer:** The replay buffer is initialized with heuristic-policy trajectories and sampled using a 50/50 split between seed and online experience.
3. **BC regularization:** A behavioral-cloning penalty is added to the actor objective to keep SAC close to the thermostat policy.
4. **BC warm start:** The SAC actor is initialized from a pretrained BC policy before online fine-tuning.

**Reward normalization.** Raw per-step rewards span roughly $\pm 60{,}000$ MJ, producing large critic targets and unstable training. We therefore apply online Welford normalization before storing rewards in the replay buffer and clip normalized values at $\pm 5\sigma$. The stepwise reward scale is shown in Fig. 3.

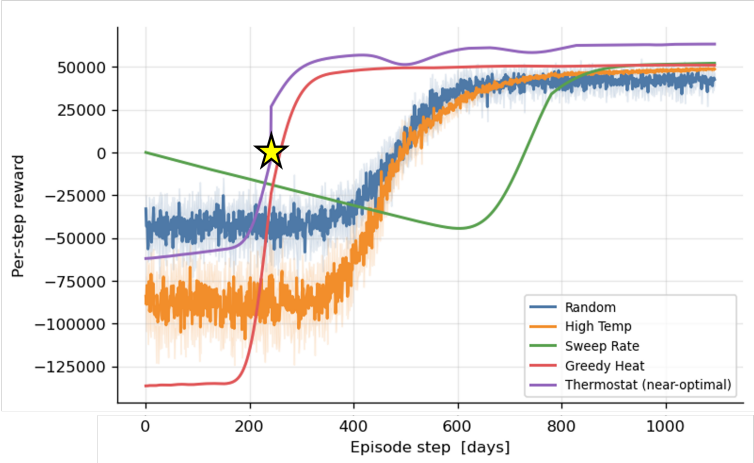



Figure 3: **Stepwise reward under heuristic policies.** Large heating penalties and delayed hydrogen-production rewards create a difficult scale for value learning.

**Frozen seed buffer.** The replay buffer is initialized with 27,375 transitions collected from all five heuristic policies. To prevent useful demonstrations from being diluted by poor early SAC trajectories, the seed region is frozen and each minibatch is sampled 50% from the seed buffer and 50% from online experience.

### 4.3 Behavioural Cloning

Behavioral Cloning (BC) provides both a supervised baseline and a warm-start initialization for SAC. Training data consists of five thermostat rollouts collected in the live environment, yielding 1,095 unique state-action pairs. The policy network shares the same two-layer MLP architecture as SAC and is trained by minimizing mean-squared error:

$$\mathcal{L}_{\text{BC}}(\theta) = \mathbb{E}_{(\mathbf{o},\mathbf{a})\sim\tau_{\text{therm}}}\left[\|\pi_\theta(\mathbf{o}) - \mathbf{a}\|^2\right].$$

Actions are normalized to the $(-1, 1)$ range to match the actor output space. BC is trained for 5,000 gradient steps using Adam with learning rate $3 \times 10^{-4}$ and batch size 256. No critic, entropy term, or replay buffer is used.

### 4.4 Implicit Q-Learning

We train Implicit Q-Learning (IQL) (Kostrikov et al., 2022) on the two positive-return heuristic policies, `thermostat` and `greedy_heat`, yielding 10 episodes and 10,950 transitions.

IQL separates value learning from policy extraction using expectile regression:

$$\mathcal{L}_V = \mathbb{E}\left[L_\tau^2(Q(\mathbf{o},\mathbf{a}) - V(\mathbf{o}))\right], \tag{10}$$

$$\mathcal{L}_Q = \mathbb{E}\left[\left(r + \gamma V(\mathbf{o}') - Q(\mathbf{o},\mathbf{a})\right)^2\right], \tag{11}$$

$$\mathcal{L}_\pi = -\mathbb{E}\left[\exp(\beta(Q(\mathbf{o},\mathbf{a}) - V(\mathbf{o})))\log\pi(\mathbf{a}|\mathbf{o})\right], \tag{12}$$

where

$$L_\tau^2(u) = |\tau - \mathbf{1}(u < 0)|u^2.$$

The value network is an additional two-layer MLP with 256 hidden units. We use expectile $\tau = 0.7$, advantage temperature $\beta = 3.0$, and train offline for 100,000 gradient steps.

### 4.5 Evaluation Protocol

All methods are evaluated using the same deterministic protocol. Every 50 training episodes for SAC, or every 5,000 gradient steps for BC and IQL, we run one deterministic rollout using the actor mean with fixed seed 0. This removes sampling variance and makes evaluation comparable across methods. Best performance is tracked across all evaluation checkpoints, and reported values correspond to the best checkpoint. All returns include the full reward with thermal shaping bonus ($\lambda = 50{,}000\,\mathrm{MJ}$).

## 5 Results

We present results across three families of methods: SAC online variants, BC, and offline IQL. All reported numbers use deterministic evaluation (actor mean, seed $= 0$). Table 3 summarizes the full experimental comparison.

Table 3: Full experimental results. All returns include the thermal shaping bonus ($\lambda = 50{,}000\,\mathrm{MJ}$). Deterministic evaluation at the best checkpoint.

| **Method** | **Finding** | **$H_2$ [mol]** | **Return [MJ]** | **Status** |
|---|---|---|---|---|
| *Heuristic baselines* | | | | |
| *Greedy Heat* | Best naive policy; no phase awareness | 10.2M | +10.0M | Baseline |
| **Thermostat** | Two-phase heuristic; BC teacher | 57.2M | +36.6M | Baseline |
| *SAC variants (online RL)* | | | | |
| Vanilla SAC | Lazy policy collapse; 220-step credit gap | ∼10M | ∼ +5M | Degrades |
| SAC + Welford norm | Critic stable; lazy policy persists | ∼10M | ∼ +5M | Degrades |
| SAC + frozen 50/50 buffer | Best SAC eval +6.2M; still degrades | ∼10M | +6.2M | Degrades |
| SAC + BC reg ($\lambda = 0.1$) | BC/entropy arms race; critic ep ∼110 | ∼5M | ∼ +2M | Partial |
| SAC from BC warm-start | Stable until ep 350; then critic diverges | ∼10M | ∼ +2M | Partial |
| *Offline methods* | | | | |
| **BC** (5 demos) | +6.9% return vs thermostat | **59.5M** | **+39.1M** | **Best** |
| **IQL offline** ($\tau = 0.7$) | 97% of thermostat; stable convergence | ∼50M | +35.5M | Converged |

### 5.1 SAC Variants Fail to Solve Credit Assignment

All five SAC variants eventually converged to a lazy policy that stopped injecting meaningful amounts of hot water to avoid high initial heating cost. Figure 4 shows the training dynamics of three representative runs.

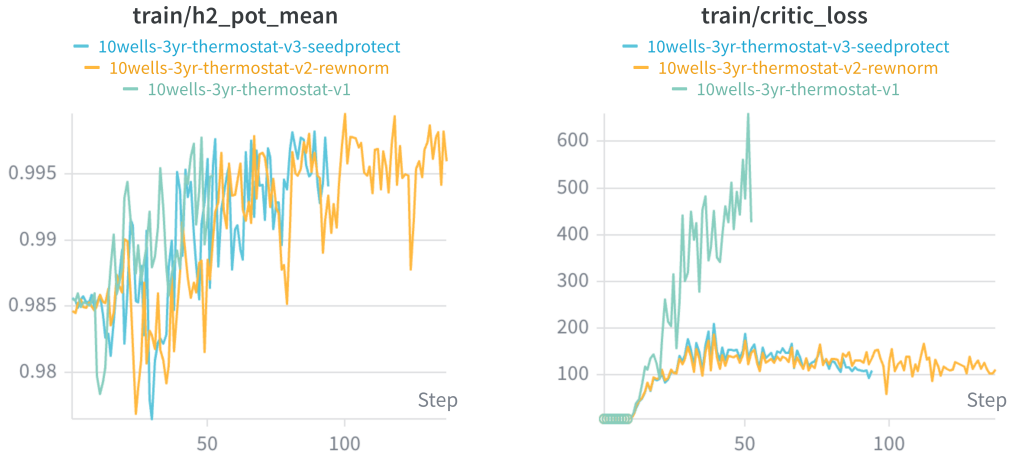
**Vanilla SAC.** The agent initially discovers reasonable injection strategies, producing roughly 10M mol $H_2$ and achieving about +5M MJ return. However, performance collapses after roughly 60–80 episodes, and the policy gradually stops injecting.

The main issue is the long delay between action and reward. Injecting hot water immediately incurs heating costs, while the resulting hydrogen production does not appear until roughly 220 days later. Early in training, the critic only sees the immediate negative cost and therefore learns that injection is bad. Propagating the positive reward signal back to the beginning of the episode requires roughly

$$N_{\text{updates}} \approx 220 \times \frac{|\mathcal{B}|}{B} \approx 172{,}000 \tag{13}$$

gradient updates, which is longer than the time needed for the policy to collapse into a lazy solution.

**SAC with reward normalization.** Reward normalization significantly improved numerical stability and reduced critic loss. However, the learned policy still converged to the same low-injection behavior. This suggests that the main problem is not unstable rewards but delayed credit assignment.




(a) Mean hydrogen potential in the replay buffer. Both reward normalization and seed-buffer protection improve coverage of high-value states compared to the vanilla SAC baseline.

(b) Critic loss during training. Reward normalization and seed-buffer protection substantially reduce critic instability relative to vanilla SAC.

Figure 4: Diagnostics for SAC stabilization strategies. Reward normalization and replay-buffer protection improve critic stability and delayed the agent's collapse towards the lazy policy. However, all variants ultimately collapsed to the lazy no-injection policies, suggesting that the primary challenge is delayed credit assignment rather than numerical instability.

**SAC with frozen 50/50 seed buffer.** To prevent expert demonstrations from being overwhelmed by the agent's own poor trajectories, we froze the thermostat data and forced each minibatch to contain 50

This improved peak performance to approximately $+6.2$M MJ, but the policy still degraded over time. Although expert transitions remained in the replay buffer, the critic continued to assign poor values to early heating actions because the delayed rewards were too far away.

**SAC with BC regularisation.** We next added a behavioral cloning loss to keep the policy close to the thermostat expert. This initially improved behavior but eventually caused training instability.

The BC term pushed the policy toward a deterministic thermostat-like behavior, while SAC's entropy objective simultaneously encouraged exploration. These two objectives conflicted with each other, causing the entropy coefficient to grow and eventually leading to critic divergence.

**SAC from BC warm-start.** Initializing SAC from a pretrained BC policy produced the best SAC results. The agent achieved positive returns of roughly $+2$M MJ and maintained stable training for much longer than the other variants.

However, performance eventually deteriorated as the policy drifted away from the BC initialization. Small value-function errors were repeatedly amplified through bootstrapping, ultimately causing the critic to diverge. This behavior is consistent with the classic "deadly triad" of function approximation, bootstrapping, and off-policy learning.

### 5.2 Behavioural Cloning Surpasses the Teacher Policy

BC was trained on a single thermostat trajectory containing 1,095 state-action pairs. Despite simply imitating the expert, BC achieved the best overall performance:

- Return: $+39.1$M MJ ($+6.9\%$ vs. thermostat)
- $H_2$ production: $59.5$M mol ($+4.1\%$ vs. thermostat)
- BC loss: $2.6 \times 10^{-4}$

Figure 5 compares the learned policy against the thermostat controller. The thermostat uses a hard switching rule at $T_{\text{prod}} = 180°\text{C}$, while the neural network learns a smooth approximation. As a result, BC injects hotter water during the early stages of the episode, accelerating thermal breakthrough and increasing the time spent in the profitable hydrogen-production phase.

This behavior was not explicitly optimized for. Rather, the smooth function approximation of the neural network slightly perturbs the expert policy and inadvertently discovers a more effective heating strategy. The result highlights a surprising benefit of imitation learning: even when trained on a single expert trajectory, BC can outperform its demonstrator through beneficial approximation errors.

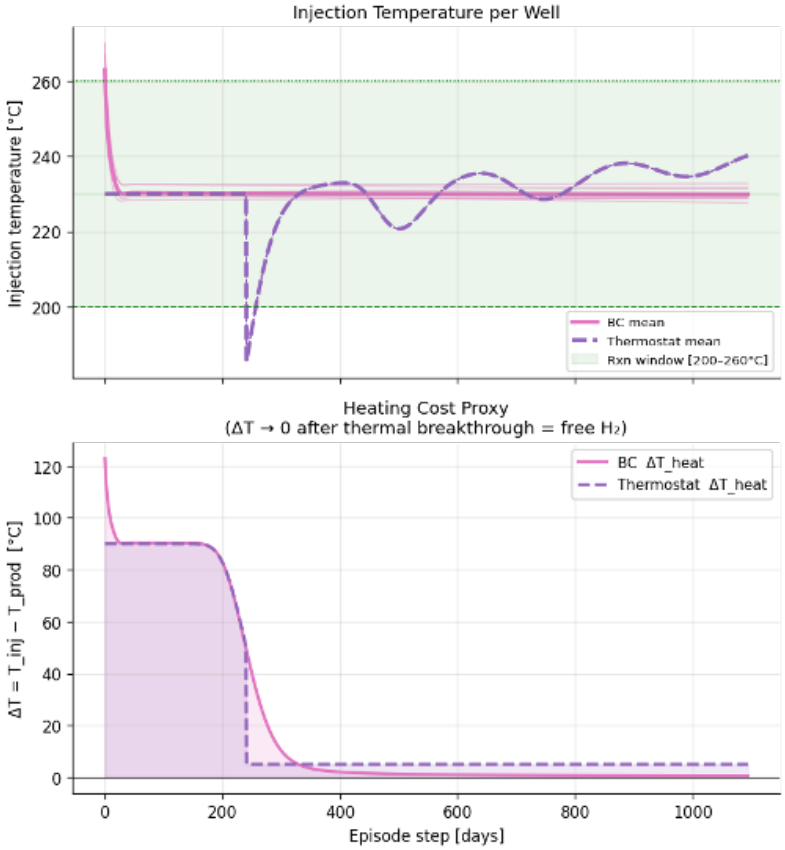



Figure 5: **BC versus thermostat policy.** BC learns a smoother injection schedule than the thermostat, resulting in more aggressive early heating and earlier thermal breakthrough.

### 5.3 IQL Provides Stable Offline Learning

IQL was trained offline on 10,950 transitions collected from thermostat and greedy-heating trajectories. Unlike SAC, IQL remained stable throughout training and converged to a return of approximately $+35.5$M MJ, corresponding to 97

The key advantage of IQL is that it avoids the instability associated with online exploration and repeated bootstrapping on out-of-distribution actions. Because the dataset already contains complete successful trajectories, the delayed reward signal can be learned through repeated passes over fixed data rather than requiring online discovery.

Although IQL nearly matches thermostat performance, it does not surpass BC. The expectile-based value function intentionally remains conservative and discourages extrapolation beyond the training distribution. In contrast, BC's unconstrained function approximation slightly modifies the thermostat strategy and accidentally discovers a more aggressive heating schedule. As a result, BC achieves the highest hydrogen production and return despite being the simplest method evaluated.

### 5.4 Summary

Online SAC fails due to the 220-step credit assignment gap — an intrinsic property of one-step temporal difference learning that no replay buffer strategy or regularisation term can overcome without fundamentally changing the backup operator. Offline methods circumvent this by learning from complete episodes containing the full reward signal. BC achieves the strongest result (+6.9% over thermostat) through smooth function approximation; IQL achieves stable convergence at 97% of thermostat through principled offline value learning. The offline-to-online gap — accurate offline critic paired with safe online exploration — remains the key open problem.

## 6 Discussion

### 6.1 Delayed Credit Assignment Is the Main Challenge

The central challenge in this benchmark is the approximately 220-day delay between negative heating costs and positive hydrogen production. Across all SAC variants, the critic initially observes only the immediate heating cost while the positive hydrogen reward appears much later in the trajectory. As a result, early injection actions are assigned overly pessimistic values and the policy gradually converges to low-injection behavior.

Several stabilization techniques improved training stability, including reward normalization, replay-buffer protection, and BC initialization. Though successfully stablized the critic learning and delayed the collapse to the lazy no-injection poliyc, none fundamentally solved the delayed credit assignment problem. These results suggest that the primary bottleneck is the difficulty of propagating reward information across hundreds of decision steps.

### 6.2 Behavioural Cloning Unexpectedly Outperforms the Expert

One of the most surprising results is that BC slightly outperforms the thermostat policy on both hydrogen production and net energy return. Inspection of the learned policy suggests that the neural network smooths the thermostat's hard switching threshold, leading to more aggressive heating during the early stages of the episode.

This earlier heating accelerates thermal breakthrough and increases the time spent in the profitable hydrogen-production regime. Importantly, BC is not explicitly optimizing for improved performance; rather, the gain emerges as a consequence of approximating a threshold-based controller with a smooth neural network. While the improvement is modest, it highlights how simple imitation learning can sometimes improve upon deterministic heuristic policies.

### 6.3 Offline Learning Is Stable but Conservative

IQL achieved stable training throughout all experiments and reached approximately 97% of thermostat performance. Unlike SAC, IQL never experienced critic divergence and consistently learned useful policies from the offline dataset.

However, IQL did not surpass BC. The conservative nature of expectile-based value estimation discourages actions that deviate substantially from the training distribution. In contrast, BC's unconstrained function approximation produced a slightly different heating strategy that happened to improve performance. This highlights a trade-off between stability and policy improvement: offline RL provides robustness, while simple imitation learning may occasionally benefit from favorable approximation errors.

### 6.4 Limitations and Future Work

Several limitations remain. First, the environment uses a compact 206-dimensional observation rather than the full reservoir state, preventing the agent from directly observing temperature and reaction distributions throughout the domain. Second, the three-year horizon is short relative to the timescales of lateral thermal propagation, potentially masking benefits of adaptive multi-well control.

Future work should focus on algorithms specifically designed for long-horizon credit assignment. Multi-step returns could allow positive hydrogen-production rewards to propagate more rapidly to

early heating decisions. On-policy approaches such as PPO with high-$\lambda$ GAE may also be better suited to this environment, since Monte Carlo-style returns can capture long-delayed rewards without relying entirely on repeated bootstrapping.

Beyond algorithmic improvements, extending the simulator to multi-decade operational horizons may reveal control strategies that are not beneficial within three-year episodes. In particular, longer horizons would allow agents to exploit lateral thermal sweep, differential well-rate allocation, and other reservoir-scale effects that evolve too slowly to influence the current benchmark. Together, these extensions would provide a richer testbed for reinforcement learning in scientific control problems with long physical timescales.

## 7 Conclusion

We introduced a physics-based reinforcement learning benchmark for stimulated geologic hydrogen production, a scientific control problem characterized by severe delayed rewards. In our simulator, heating actions incur immediate costs while meaningful hydrogen production does not emerge until approximately 220 days later, creating a challenging long-horizon credit assignment problem.

Across five SAC variants, online reinforcement learning consistently struggled to discover and maintain effective heating strategies despite improvements in training stability. In contrast, Behavioral Cloning achieved the best overall performance, outperforming its thermostat demonstrator by 4.1% in hydrogen production and 6.9

These results suggest that delayed credit assignment, rather than state or action dimensionality, is the dominant challenge in this benchmark. More broadly, they demonstrate that simple imitation learning can outperform sophisticated online RL methods when expert demonstrations are available and rewards are separated from actions by long physical timescales.

Future work will investigate algorithms specifically designed for long-horizon credit assignment, including multi-step returns, on-policy methods such as PPO, model-based planning, and longer operational horizons that allow agents to exploit reservoir-scale thermal dynamics.

## 8 Team Contributions

Solo project.

## References


Atish Dixit and Ahmed H. ElSheikh. 2022. Stochastic optimal well control in subsurface reservoirs using reinforcement learning. *Engineering Applications of Artificial Intelligence* 113 (2022), 104949.

Atish Dixit and Ahmed H. ElSheikh. 2023. Robust optimal well control using an adaptive multigrid reinforcement learning framework. *Mathematical Geosciences* 55 (2023), 345–375.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. *CoRR* abs/1801.01290 (2018). arXiv:1801.01290 `http://arxiv.org/abs/1801.01290`

Farzad Hourfar, Hamed Jalaly Bidgoly, Behzad Moshiri, Karim Salahshoor, and Ali Elkamel. 2019. A reinforcement learning approach for waterflooding optimization in petroleum reservoirs. *Engineering Applications of Artificial Intelligence* 77 (2019), 98–116.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. 2022. Offline Reinforcement Learning with Implicit Q-Learning. In *International Conference on Learning Representations*. `https://arxiv.org/abs/2110.06169`

Yusuf Nasir and Louis J. Durlofsky. 2023. Deep reinforcement learning for optimal well control in subsurface systems with uncertain geology. *J. Comput. Phys.* 477 (2023), 111945.

K. Zhang, Z. Wang, G. Chen, L. Zhang, Y. Yang, C. Yao, J. Wang, and J. Yao. 2022. Training effective deep reinforcement learning agents for real-time life-cycle production optimization. *Journal of Petroleum Science and Engineering* 208 (2022), 109766.

# A Hyperparameters

Table 4: Training hyperparameters shared across SAC, BC, and IQL.

| **Hyperparameter** | **Value** | **Notes** |
|---|---|---|
| *Shared* | | |
| Hidden units | 256 | 2-layer MLP |
| Learning rate | $3 \times 10^{-4}$ | Adam optimizer |
| Batch size | 256 | |
| Gradient clipping | max-norm $= 1.0$ | |
| *SAC-specific* | | |
| Discount $\gamma$ | 0.995 | Effective horizon $\approx 200$ steps |
| Polyak $\tau$ | 0.005 | Critic target soft update |
| Target entropy $H^*$ | $-20$ | $= -\lvert\mathcal{A}\rvert$ |
| Replay buffer capacity | 200,000 | |
| Warmup steps | 10,000 | 1,000 with BC initialization |
| Seed buffer ratio | 50/50 | Frozen seed / online samples |
| Reward normalization | Welford, clip $\pm 5\sigma$ | |
| *BC-specific* | | |
| BC training steps | 5,000 | Thermostat demonstrations |
| BC dataset size | 1,095 unique pairs | 5 deterministic rollouts |
| *IQL-specific* | | |
| Expectile $\tau$ | 0.7 | Conservative value estimate |
| Advantage temperature $\beta$ | 3.0 | Policy extraction weight |
| Offline training steps | 100,000 | Fixed dataset |