

Article

Rapid Screening of CO₂ Injection Schedules Using Activity-Based Reservoir Partitioning and Slow-Region Derivative ML Proxies

Eirini Maria Kanakaki ¹, Sofianos Panagiotis Fotias ¹ and Vassilis Gaganis ^{1,2,*}

¹ School of Mining and Metallurgical Engineering, National Technical University of Athens, 15780 Athens, Greece; ekanakaki@metal.ntua.gr (E.M.K.); sfofias@metal.ntua.gr (S.P.F.)

² Institute of Geoenergy, Foundation for Research and Technology-Hellas, 73100 Chania, Greece

* Correspondence: vgaganis@metal.ntua.gr

Abstract

Full-physics reservoir simulation for CO₂ storage becomes computationally expensive when many operational schedules must be screened, motivating machine-learning (ML) surrogates that reduce simulation burden while preserving the essential physics-driven response. We propose an activity-based partitioning methodology that produces an interpretable applicability map, identifying regions where surrogate substitution is expected to be reliable and regions where highly active dynamics make it unsafe. In this work, we focus exclusively on the slow-varying region and develop proxy models for pressure and saturation time derivatives in that domain. The fast-varying region is intentionally excluded, and no fully coupled hybrid simulator is claimed at this stage. The partition is constructed from temporal changes in derivative signals and aggregated across multiple schedules to obtain a conservative, scenario-robust delineation. For slow cells, local stencil-based neural proxies leverage overlapping time windows and features describing the local state, schedule forcing, and injector influence. Because saturation derivatives in the slow region are strongly zero-inflated, with many cells remaining outside the advancing CO₂ plume for long periods, a two-stage strategy is adopted: first detecting whether meaningful change occurs and then predicting the derivative magnitude only when active, with additional smoothing to suppress near-zero artifacts. The framework also supports selective surrogate deployment over user-selected time windows. The objective is therefore to establish a conservative zone of applicability for derivative-based ML updates, rather than to demonstrate full simulator replacement or end-to-end coupled acceleration. In the case study, 5914 of the 8243 grid blocks evaluated by the proxy workflow were classified as slow-varying, corresponding to 71.7% of the evaluated proxy-analysis domain. For the blind schedule, full-rollout pressure reconstruction produced mean absolute errors of 5.34, 3.69, and 2.80 psi over early, middle, and late time-window groups, respectively. In a future coupled implementation using the same partition, these 5914 cells could be advanced by the ML proxy, while the remaining dynamically active or unsupported cells would remain under full-physics treatment. This would reduce the full-physics active-cell count from 9212 to 3298 in the future coupled setting, although direct wall-clock acceleration remains to be quantified after simulator integration.



Academic Editors: Guoqing Han, Xingyuan Liang and Xiaojun Wu

Received: 4 May 2026

Revised: 9 June 2026

Accepted: 23 June 2026

Published: 27 June 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: CO₂ geological storage; deep saline aquifers; numerical simulation; machine learning; grid-based proxy models; classification; acceleration

1. Introduction

Deep saline aquifers are considered one of the most promising options for mitigating the climate crisis through large-scale geological storage of CO₂ [1]. By providing long-term subsurface storage capacity [2], they can help offset residual emissions from hard-to-abate sectors [3] and support pathways towards net-zero targets [4–9]. The viability of such projects depends on the ability to design injection strategies that store large amounts of CO₂ while maintaining containment integrity and respecting operational and regulatory limits [10,11]. Achieving this requires high-fidelity numerical reservoir simulations that can predict pressure build-up, CO₂ plume migration and trapping behaviour over the lifetime of a CO₂ storage project [12].

In practice, CO₂ storage projects must be assessed under multiple operating strategies, including different well locations, injection-rate profiles, ramp-up and shut-in schedules, and pressure limits motivated by regulatory requirements. These strategies are further constrained by commercial intake commitments, surface-facility capacities and economic considerations, so that many alternative scenarios must be examined before a robust development plan can be defined [13–15]. When full-physics simulators are used, the resulting number of model evaluations can quickly become a bottleneck for scenario screening, optimization and uncertainty quantification [16]. This has motivated the development of surrogate and proxy models that approximate the full-physics response at reduced cost, aiming to preserve the essential flow behaviour while alleviating the computational burden [17].

Recent CO₂ storage acceleration studies have demonstrated the value of surrogate and proxy models for rapid prediction of storage responses. Reduced-order, smart-proxy, deep-learning and neural-operator models have been used to approximate pressure buildup, CO₂ saturation, plume migration, storage performance, and monitoring-related quantities of interest in saline aquifers and related storage settings [18–25]. In particular, recent CO₂-storage surrogate studies have shown that ML models can predict spatial pressure and saturation fields, plume evolution, and surface displacement, providing useful tools for field-scale decision support [26–28]. In parallel, Design-of-Experiments-based workflows have been used to reduce the number of full-physics simulations required for screening, calibration and scenario analysis [29,30], while reinforcement-learning and surrogate-assisted optimization approaches have been explored for injection-control and pressure-management decisions [31].

However, many of these approaches are designed to approximate the reservoir response over the full model domain, or to support global scenario screening, history matching, or optimization directly. The methodological question addressed in the present work is different: before constructing the surrogate, we first ask where in the reservoir grid ML treatment is appropriate. CO₂ storage dynamics are spatially heterogeneous. Cells near injectors, along high-transmissibility pathways, and around the advancing plume front may experience rapid pressure and saturation changes, whereas large portions of the reservoir evolve much more smoothly in space and time, as illustrated in Figure 1. A single global surrogate must therefore represent both highly transient and slowly varying behavior simultaneously, which can increase model complexity, reduce computational advantage, and create risk in dynamically sensitive regions where local errors are least acceptable.

This motivates a complementary acceleration strategy in which dynamically sensitive cells are excluded from proxy treatment and retained for full-physics simulation, while only the smoother slow-varying region is considered proxy-eligible. In this way, the objective is not to shift the computational burden from the numerical simulator to an overly complex ML model, but to define a conservative grid-block-level zone of applicability where lightweight local ML updates can be used safely in a future hybrid simulator.

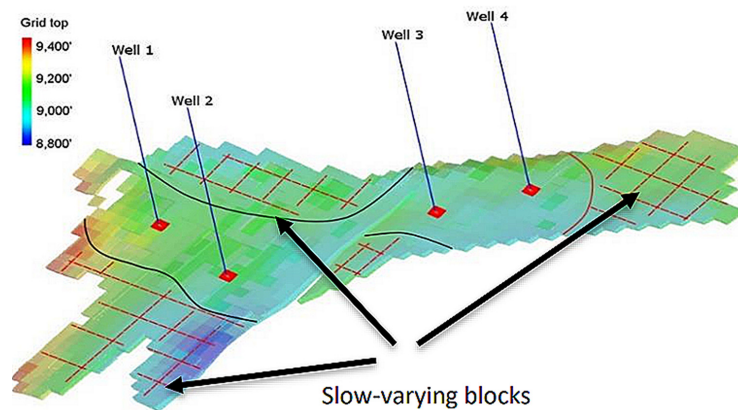


Figure 1. Example distribution of fast-varying near-well cells and smoothly evolving slow-varying regions.

In earlier work [32,33], we proposed a grid-block-level machine learning (ML) framework that implements this perspective for CO₂ storage simulations in deep saline aquifers. Rather than building a single global surrogate, the framework operates at the scale of individual grid blocks and their immediate neighborhoods: for slow-varying blocks, an ML proxy predicts the future pressure and CO₂ saturation of a cell from its own recent history and that of its face-adjacent neighbors. In that first study [32], we introduced a fully automated methodology that used data from a single injection scenario to classify grid blocks as fast or slow based on their dynamic behavior, applying an interquartile-range (IQR)-based criterion to identify cells with unusually large variations. We proposed restricting proxy prediction to slow-varying blocks and treating fast-varying blocks as outside the surrogate validity domain. On a simplified model and a single schedule, this proof-of-concept study indicated that selective use of ML at the grid-block level could support future reductions in the number of equations advanced by the simulator, while preserving the accuracy of key flow responses such as pressure evolution and CO₂ saturation.

A second study [33] extended the same IQR-based classification proxy concept to a geologically realistic deep saline aquifer model with a more complex well configuration. In this case, the automated grid-block classification framework was transferred from the idealized setting to a more complex one, again using a single representative operating scenario. The results confirmed that the workflow can be implemented in a fully automated manner and that the underlying philosophy, excluding fast-varying regions from proxy treatment and targeting slow-varying regions as candidates for proxy modelling, remains effective in a more complex geological setting. At the same time, both studies were explicitly framed as proof-of-concept: they relied on one injection schedule to define fast and slow regions, and the proxy inputs were deliberately kept relatively simple to focus on validating the combination of an IQR-based grid-block classifier with ML proxies, rather than on achieving robust generalization across operating conditions.

The main contribution of the present work is the definition and validation of a conservative zone-of-applicability framework for local ML updates under multiple CO₂ injection schedules, rather than the construction of a global reservoir surrogate. Building on our earlier proof-of-concept works, this paper presents an offline, multi-scenario framework for validating surrogate updates on a conservatively defined slow-varying domain in CO₂ storage simulations. We construct a fast/slow partition by aggregating pressure- and saturation-based activity measures across multiple injection schedules, producing an interpretable proxy-eligibility map that excludes any grid block that becomes highly transient under any schedule. On this slow domain, we train derivative proxies for pressure and saturation using five schedules and assess schedule-level generalization using a fully blind

schedule that is excluded from both training and partitioning, with performance reported over user-defined time windows. Because slow-domain saturation dynamics are strongly zero-inflated, saturation is modeled with a two-stage proxy that first detects activity and then predicts derivative magnitude conditionally on active samples, with stabilization via gating/deadband logic to preserve quiescence while capturing localized increments. Proxy stability is quantified using anchored one-step reconstruction (to measure local fidelity when reference states are available) and full-rollout reconstruction (as a conservative stress test for accumulated integration drift).

The scope of this paper is therefore to establish where and when slow-domain surrogate updates can be applied reliably, and to assess how these updates generalize across different injection schedules. The contribution should be interpreted as an offline, pre-coupling validation of a selective proxy strategy, rather than as a completed demonstration of end-to-end simulator acceleration. In the proposed future hybrid workflow, the fast-varying cells would remain under full-physics numerical treatment, while only the validated slow-varying cells would be advanced by the ML proxy. Accordingly, simulator-level and measured wall-clock speedups are reserved for future work focused on numerical coupling and runtime deployment.

The remainder of the paper is organized as follows: Section 2 describes the multi-scenario activity-based partitioning, Section 3 presents the grid-based pressure- and saturation-derivative proxy models, Section 4 introduces the case-study reservoir model, injection schedules and scenario-similarity assessment, Section 5 presents the results, and Sections 6 and 7 discuss implications and conclude.

2. Multi-Scenario Activity-Based Partitioning into Fast- and Slow-Varying Grid Blocks

This section defines the activity-based classifier used to partition the grid into fast- and slow-varying cells across multiple injection scenarios. Fast cells are those that exhibit strong temporal transients in pressure or CO₂ saturation in at least one scenario. Cells that remain smooth in all scenarios are designated as slow-varying and are later used as targets for proxy prediction, while fast cells are excluded from proxy prediction and serve only to define a conservative proxy-eligibility mask.

Formally, let $s \in \{1, \dots, N_{scen}\}$ index injection scenarios, $c \in \{1, \dots, N_c\}$ index active grid blocks, and $t_k, k = 1, \dots, N_t$, denote the stored simulator output times, where N_t is the total number of stored time levels. For each pair (s, c) , the simulator provides time series of pressure and saturation,

$$\left(P_c^{(s)}(t_k), S_c^{(s)}(t_k) \right)_{k=1}^{N_t} \quad (1)$$

which are processed in two stages

$$\left(P_c^{(s)}(t_k), S_c^{(s)}(t_k) \right)_{k=1}^{N_t} \xrightarrow{\Phi} \left(A_{p,c}^{(s)}, A_{s,c}^{(s)} \right) \xrightarrow{C_\tau^{(s)}} f_c^{(s)} \in \{0, 1\} \quad (2)$$

Here, Φ denotes the feature-extraction operator used in the first stage. It maps the pressure and saturation time series of cell c in scenario s to scalar activity measures $\left(A_{p,c}^{(s)}, A_{s,c}^{(s)} \right)$, which summarize the temporal behavior of that cell over all N_t stored simulator time steps. The classifier C_τ , used in the second stage, then compares the scenario-specific thresholds $\tau_p^{(s)}(\alpha)$ and $\tau_s^{(s)}(\alpha)$, computed from the empirical activity distributions in scenario s . If either activity measure exceeds its corresponding threshold, the cell is assigned a fast label for that scenario; otherwise, it is assigned a slow label. In a final step, the scenario-level labels $f_c^{(s)}$ are aggregated across scenarios to obtain a global indicator f_c^{union} . This two-stage

processing and the subsequent multi-scenario aggregation are summarized schematically in Figure 2.

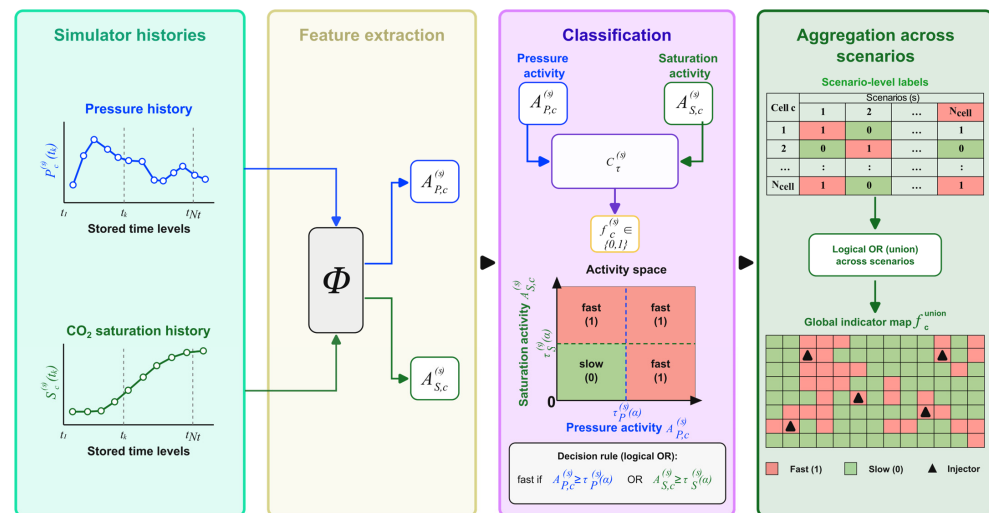


Figure 2. Schematic workflow of the multi-scenario activity-based fast/slow grid-block classification framework.

2.1. Activity Measures for Pressure and Saturation

This subsection specifies the feature-extraction operator Φ in Equation (2). For each scenario s and cell c , Φ maps the pressure and CO₂ saturation histories in Equation (1) to scalar activity measures $(A_{p,c}^{(s)}, A_{s,c}^{(s)})$ that quantify the temporal variability of pressure and saturation. This is done in three steps: computing time derivatives, forming slope changes of these derivatives, and averaging their absolute values over time.

First, node-based time derivatives are estimated from the stored simulator outputs. One-sided finite differences are used at the first and last stored time levels, while a three-point finite-difference stencil is used at interior time levels to account for non-uniform time spacing. The resulting derivative signals are collected as

$$\left(\dot{P}_c^{(s)}(t_k), \dot{S}_c^{(s)}(t_k) \right)_{k=1}^{N_t} \quad (3)$$

evaluated at the same times t_k .

Next, changes in the node-based derivative signals are computed between consecutive stored time levels and collected as

$$\left(\Delta \dot{P}_c^{(s)}(t_k), \Delta \dot{S}_c^{(s)}(t_k) \right)_{k=2}^{N_t} \quad (4)$$

with $\Delta \dot{P}_c^{(s)}(t_k) = \dot{P}_c^{(s)}(t_k) - \dot{P}_c^{(s)}(t_{k-1})$, $\Delta \dot{S}_c^{(s)}(t_k) = \dot{S}_c^{(s)}(t_k) - \dot{S}_c^{(s)}(t_{k-1})$, $k = 2, \dots, N_t$

The scalar activity measures used in Equation (2) are then defined as step-averaged absolute slope changes,

$$A_{p,c}^{(s)} = \frac{1}{N_t - 1} \sum_{k=2}^{N_t} \left| \Delta \dot{P}_c^{(s)}(t_k) \right|, \quad (5)$$

$$A_{s,c}^{(s)} = \frac{1}{N_t - 1} \sum_{k=2}^{N_t} \left| \Delta \dot{S}_c^{(s)}(t_k) \right| \quad (6)$$

High values of $A_{p,c}^{(s)}$ or $A_{s,c}^{(s)}$ indicate that the corresponding cell undergoes frequent or intense changes in the rate of pressure or saturation evolution, whereas low val-

ues are characteristic of cells whose responses remain close to quasi-monotonic, slowly varying behaviour.

From a reservoir-engineering standpoint, the pair $(A_{p,c}^{(s)}, A_{s,c}^{(s)})$ provides a compact, physically interpretable summary of the dynamic response of each cell in each scenario. From a classification standpoint, these two scalars define a two-dimensional activity space in which scenario-level thresholds will next be applied to obtain binary fast/slow labels per scenario.

2.2. Scenario-Level Thresholding and Multi-Scenario Fast/Slow Partition

In the second step, the activity measures $A_{p,c}^{(s)}$ and $A_{s,c}^{(s)}$ from Section 2.1 are converted into binary fast/slow labels by comparing them to empirical percentile thresholds. The outcome is a scenario-level classification for each cell, which is then combined across scenarios to form a single multi-scenario partition.

For a fixed scenario s , the sets $\{A_{p,c}^{(s)}\}_{c=1}^{N_c}$ and $\{A_{s,c}^{(s)}\}_{c=1}^{N_c}$ collect the pressure and saturation activity values over all active grid blocks. Given a chosen percentile level $\alpha \in (0, 100)$ (e.g., $\alpha = 90\%$), scenario-specific thresholds

$$\tau_p^{(s)}(\alpha) = \text{perc}_\alpha \left(\{A_{p,c}^{(s)}\}_{c=1}^{N_c} \right), \quad (7)$$

$$\tau_s^{(s)}(\alpha) = \text{perc}_\alpha \left(\{A_{s,c}^{(s)}\}_{c=1}^{N_c} \right) \quad (8)$$

are defined as the empirical α -th percentiles, where the empirical percentile operator can be written as

$$\tau_p^{(s)} = \inf \left\{ x \in \mathbb{R} : \frac{1}{N_c} \sum_{c=1}^{N_c} 1_{\{A_{p,c}^{(s)} \leq x\}} \geq \frac{\alpha}{100} \right\} \quad (9)$$

and analogously for $\tau_s^{(s)}$. Here $1_{\{\cdot\}}$ denotes the indicator function. Thus $\tau_p^{(s)}(\alpha)$ is the smallest value for which at least an $\alpha\%$ fraction of cells in scenario s satisfies $A_{p,c}^{(s)} \leq \tau_p^{(s)}(\alpha)$. Cells with $A_{p,c}^{(s)} > \tau_p^{(s)}(\alpha)$ belong to the upper $(100 - \alpha)\%$ tail of the pressure-activity distribution. The same interpretation holds for $\tau_s^{(s)}(\alpha)$ and the saturation-activity values.

Activity is then mapped to binary indicators via

$$f_{p,c}^{(s)} = 1_{\{A_{p,c}^{(s)} > \tau_p^{(s)}(\alpha)\}}, \quad (10)$$

$$f_{s,c}^{(s)} = 1_{\{A_{s,c}^{(s)} > \tau_s^{(s)}(\alpha)\}} \quad (11)$$

and combined into a scenario-level fast label

$$f_c^{(s)} = \max(f_{p,c}^{(s)}, f_{s,c}^{(s)}) \in \{0, 1\} \quad (12)$$

Cell c is therefore classified as fast in scenario s if its pressure activity exceeds $\tau_p^{(s)}(\alpha)$ or its saturation activity exceeds $\tau_s^{(s)}(\alpha)$. Otherwise, it is classified as slow in that scenario. Note that the classification outcome depends monotonically on the choice of activity thresholds $\tau_p^{(s)}$ and $\tau_s^{(s)}$. Lower thresholds (e.g., using the 90th percentile of the activity distribution instead of the 95th percentile) expand the fast set and lead to a more conservative partition, in the sense that a larger fraction of the grid is excluded from the proxy-eligible domain. Conversely, higher thresholds shrink the excluded set and increase the proportion of slow-varying blocks that are deemed proxy-eligible. This trade-off directly controls the

balance between computational savings and the risk of exposing the surrogate to strongly transient flow regimes.

To obtain a single partition valid across all scenarios, the scenario-level labels are aggregated by a union in the scenarios space,

$$f_c^{union} = \max_{1 \leq s \leq N_{scen}} f_c^{(s)} \in \{0, 1\} \quad (13)$$

Equivalently, $f_c^{union} = 1$ if and only if cell c is classified as fast in at least one scenario. The complementary slow indicator

$$s_c = 1 - f_c^{union} \quad (14)$$

equals one precisely for cells that satisfy both activity thresholds in all scenarios. The slow-varying set $S_{slow} = \{i : s_c = 1\}$ defines the subset of the grid on which the machine-learning proxies are trained, whereas cells with $f_c^{union} = 1$ are treated as outside the proxy-eligible domain and reserved for the full-physics model.

This final fast/slow partition is scenario-agnostic yet conservative with respect to all tested injection strategies. It provides a well-defined spatial mask that restricts proxy targets to the slow-varying cells and conservatively excludes grid blocks with potentially sharp or schedule-dependent dynamics from the proxy-eligible domain.

From a reservoir engineering perspective, the slow-varying set should not be interpreted as a region that is completely unaffected by the CO₂ injection and storage workflow. Rather, it represents grid blocks whose pressure and saturation responses remain sufficiently smooth over the simulation interval used for classification. In an active-injection interval, these cells are typically located outside the strongest near-well pressure-transient zone and outside the most active CO₂ plume body or advancing front. Pressure may still propagate into them, but in a smoother and more diffusive manner, while saturation is commonly zero, near-zero, or only gradually changing. If the same classification procedure is applied to a post-injection monitoring interval, the slow-varying set is expected to expand because the imposed well-rate forcing is removed, near-well pressure transients relax, pressure gradients dissipate across the reservoir, and plume motion becomes dominated by gradual redistribution processes.

3. Grid-Based Proxy Models for Pressure and CO₂ Saturation Dynamics

This section introduces the proxy models used to predict local reservoir dynamics under multiple CO₂ injection schedules. The proxies are formulated to predict pressure and CO₂ saturation time derivatives rather than raw pressure and saturation values. Predicting derivatives rather than state increments makes the learned target less directly tied to a specific time-step length, which is useful for compatibility with the simulator's adaptive time-stepping strategy over the range of time-step sizes represented in the training data.

The high-level structure of the proxy-modelling workflow is summarized in Figure 3. Each proxy sample is associated with a focal grid block c and a time window w . The focal cell and its face-adjacent neighboring cells provide the local spatial context, while the time window provides the temporal context from which state and schedule-related information are extracted. For each (c, w) sample, these quantities are assembled into an input vector and supplied to separate proxy models for pressure and CO₂ saturation dynamics. The pressure proxy predicts the pressure derivative of the focal cell, whereas the saturation proxy predicts the CO₂ saturation derivative and is implemented with a separate two-stage structure to account for the strongly zero-inflated nature of saturation changes in the slow-varying region. The exact definition of the cells treated by the proxy

models, the construction of the time windows, the detailed input structures, and the specific learning logic of the pressure- and saturation-derivative proxies are presented in the following subsections.

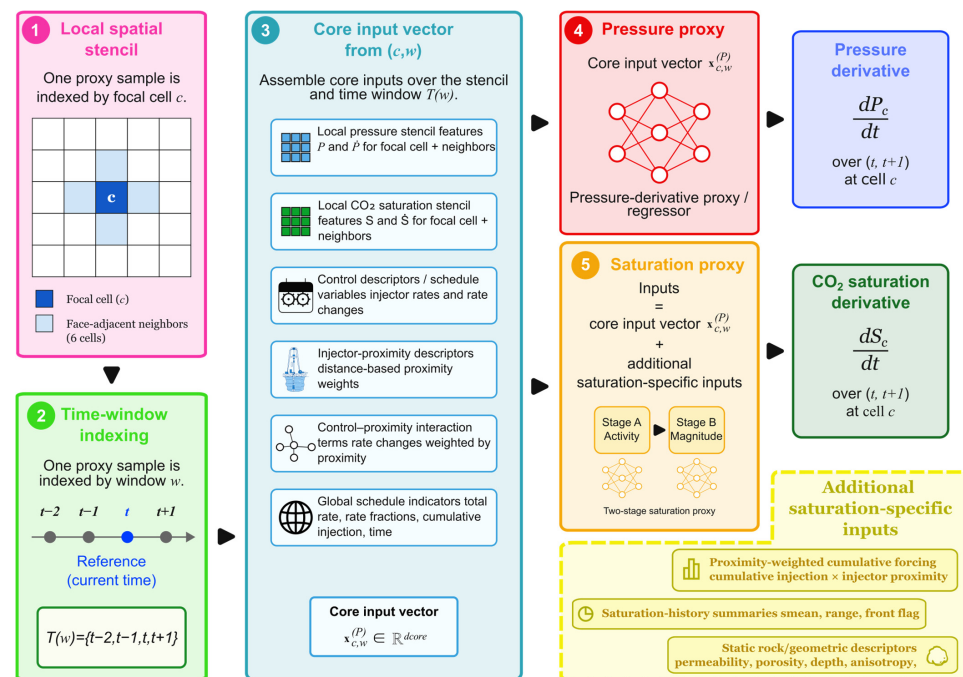


Figure 3. Schematic workflow of the grid-based derivative proxy models.

3.1. Grid Blocks and Local Spatial Stencil Treated by the Proxy Models

The first step in constructing a proxy sample is to define the local spatial stencil around the focal grid block. In this study, the proxy models are applied at the level of individual cells. Let $A(c)$ denote the set of actual face-adjacent neighbors of cell c .

The primary prediction set consists of interior cells, defined as cells that do not touch the external boundary of the reservoir model and are surrounded by six physical face-adjacent neighbors in the three-dimensional Cartesian grid, i.e., $|A(c)| = 6$. For such cells, the local proxy stencil is defined as $L(c) = \{c\} \cup A(c)$, so that each interior-cell sample is represented by a uniform seven-block stencil consisting of the focal cell and its six face-adjacent neighbors.

Boundary cells are grid blocks located on the external faces, edges, or corners of the reservoir model. Unlike interior cells, which are surrounded by neighboring grid blocks on all six Cartesian faces in a three-dimensional grid, boundary cells touch the outer boundary of the model and therefore have fewer physical face-adjacent neighbors. These cells satisfy $|A(c)| < 6$ and therefore introduce additional connectivity cases that would reduce the uniformity of the proxy formulation. To preserve broad spatial coverage while maintaining a single treatment, the dominant boundary class in the model is additionally included: boundary cells with five face-adjacent neighbors, i.e., $|A(c)| = 5$.

These cells are handled through an imaginary-cell construction that restores an interior-equivalent configuration by augmenting the neighbor set with one imaginary neighbor c^*

$$A^*(c) = A(c) \cup \{c^*\}, |A^*(c)| = 6 \quad (15)$$

The completed local proxy stencil is then defined as

$$L^*(c) = \{c\} \cup A^*(c), |L^*(c)| = 7 \quad (16)$$

The imaginary cell is assigned the same state as the boundary focal cell

$$P(c^*) = P(c), \quad (17)$$

$$S(c^*) = S(c) \quad (18)$$

where P denotes pressure and S denotes saturation. This choice encodes a no-flow (zero-gradient) condition on the external boundary face: setting $P(c^*) = P(c)$ imposes zero pressure drop across that face, while setting $S(c^*) = S(c)$ is consistent with no advective transport across a closed boundary. Figure 4 schematically illustrates the imaginary-cell adjustment for a five-neighbor boundary cell. The left panel shows a focal boundary cell (green) that has five face-adjacent neighbors (wireframe cells), while the right panel shows the same case after adding an imaginary neighbor (purple) across the missing face to recover an interior-equivalent. This imaginary-cell completion allows every cell to be represented with the same 7-block stencil (focal cell + six face-adjacent neighbors), independent of boundary proximity.

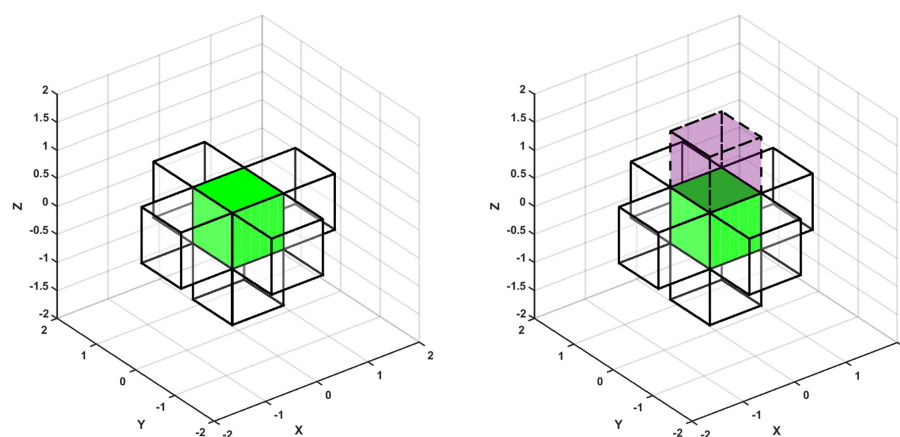


Figure 4. Imaginary cell completion of a five-neighbor boundary cell.

3.2. Time-Window Formulation for Proxy Modelling

After the local stencil has been defined, the temporal dimension of the learning problem is organized through a sequence of overlapping time windows. Let the simulator output states at discrete time steps $t = 1, 2, \dots, T$. Each training sample is indexed by a pair (w, c) , where c denotes a focal cell and w indexes a time window organized around a reference, or current, simulator time step.

Each time window is defined by four consecutive simulator time steps. Using w as the window index, the set of simulator time steps belonging to window w is

$$T(w) = \{t_w, t_{w+1}, t_{w+2}, t_{w+3}\}, \quad w = 1, \dots, T-3 \quad (19)$$

Within this set, the third element t_{w+2} is treated as the current simulator time step denoted as $t = t_{w+2}$, which allows the same window to be expressed equivalently as

$$T(w) = \{t-2, t-1, t, t+1\} \quad (20)$$

Consecutive windows are generated by advancing the window index by one simulator time step, yielding a strongly overlapping sequence of time-step sets. This sliding construction provides a uniform temporal indexing of the simulated dynamics that is applied consistently across injection schedules. A schematic illustration of the time-window indexing is provided in Figure 5.

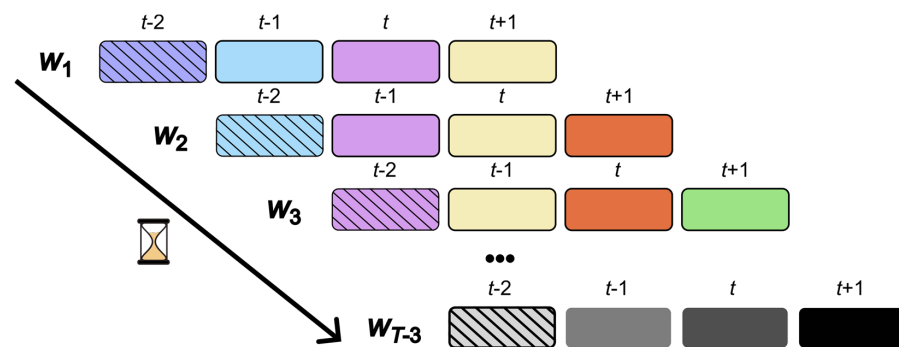


Figure 5. Time-window stencil used in proxy modelling.

The four-step window was selected as a practical compromise between temporal context and proxy compactness. It provides sufficient recent history to represent short-term pressure/saturation trends and rate-switching effects before the forward interval being predicted, while avoiding the larger input dimension and stronger redundancy that would result from longer overlapping windows. Shorter windows would provide less information about the local temporal trend, whereas longer windows would increase training complexity and the risk of overfitting.

3.3. Pressure-Derivative Proxy Model

With the spatial stencil and time-window indexing defined, the pressure proxy is formulated as a regression model that maps each window–cell input vector to the pressure derivative of the focal cell. The pressure proxy approximates the grid-block pressure derivative through the regression map

$$\hat{P}_c(t) = M_P(\mathbf{x}_{w,c}) \quad (21)$$

where c is the focal grid block and w is the time-window index, and $\mathbf{x}_{w,c}$ is the feature vector assembled for that window–cell sample. The predictor M_P is implemented as a feedforward artificial neural network (ANN) with one hidden layer (10 units) and a linear output. This compact architecture was selected because the proxy is applied only in the slow-varying region, where pressure dynamics are smooth after the fast-varying cells have been excluded by the activity-based mask. In this setting, the purpose of the ANN is not to emulate the full reservoir response globally, but to learn a local derivative correction from structured, physics-informed inputs. A small network therefore provides sufficient flexibility while limiting overfitting risk and keeping the proxy computationally lightweight. More complex architectures, such as graph neural networks, transformers, or physics-informed neural networks, may be advantageous for global reservoir-field surrogates, unstructured-grid learning, long-range spatial or temporal dependency modelling, or direct incorporation of governing-equation residuals into the training objective. However, these capabilities are not required for the present objective, which is a local derivative model applied only after the fast-varying cells have been excluded. Since the final goal of the framework is simulation acceleration, the ML component should not introduce an online computational cost that offsets the savings obtained by reducing the full-physics simulator domain. The local stencil-based ANN was therefore selected as a deliberately lightweight and interpretable architecture that matches the slow-region prediction task while keeping inference cost low.

The network is trained by minimizing the mean squared error (MSE) using the Levenberg–Marquardt algorithm. Inputs and targets are scaled by min–max normalization using statistics computed from the training set only. The same scaling is then

applied unchanged to validation, test, and blind-scenario samples, and predictions are finally mapped back to physical units [psi/month].

Each training sample corresponds to a window-cell pair (w, c) . The four-step window is indexed as $\{t_w, t_{w+1}, t_{w+2}, t_{w+3}\}$, and the reference time is taken as the third node, so that the window can also be written as $\{t-2, t-1, t, t+1\}$. The target $\dot{P}_c(t)$ is an interval-based derivative associated with the forward time interval $[t, t+1] = [t_{w+2}, t_{w+3}]$. The feature vector $\mathbf{x}_{w,c} \in \mathbb{R}^d$ is constructed from the same window and provides the local state context and the control evolution leading into that interval.

This formulation makes the learning problem explicit: the proxy must infer how pressure changes over the upcoming interval $[t, t+1]$ from the current local flow state and its recent evolution, and the imposed well controls that drive the pressure field, modulated by spatial connectivity and distance from the injectors. For this reason, the input design is intentionally hybrid: it is grid-local, so that transmissibility-driven coupling and interference are represented through a fixed neighbor stencil, and it is control-aware, so that rate changes enter the model together with their spatial influence. Accordingly, the feature vector is assembled as

$$\mathbf{x}_{w,c} = \begin{bmatrix} \mathbf{x}_{w,c}^{base} \\ \mathbf{x}_w^{rates} \\ \mathbf{x}_c^{prox} \\ \mathbf{x}_{w,c}^{inter} \\ \mathbf{x}_w^{global} \end{bmatrix}, \dim(\mathbf{x}_{w,c}) = d \quad (22)$$

where the five feature groups provide complementary information about (i) the local flow state and its recent temporal evolution $\mathbf{x}_{w,c}^{base}$, (ii) the control forcing over the window $\mathbf{x}_w^{rates}$, (iii) the geometric influence of each injector on block c $\mathbf{x}_c^{prox}$, (iv) the physically motivated coupling between controls and spatial influence $\mathbf{x}_{w,c}^{inter}$, and (v) the global operating context at the reference time $\mathbf{x}_w^{global}$.

(i) Base stencil features $\mathbf{x}_{w,c}^{base}$

Grid locality is enforced through a fixed 7-block stencil centered at the focal block c , denoted by $L(c)$. For interior cells, $L(c)$ consists of c and its six physical face-adjacent neighbors. For the dominant boundary class with five neighbors, the completed stencil $L^*(c)$ is used, after adding one imaginary neighbor across the missing face as described in Section 3.1. Thus, the proxy always operates on a uniform seven-block stencil.

The base block stacks pressures and saturations together with their time derivatives evaluated on this stencil at the reference time t and the preceding time step $t-1$. In particular, for all $j \in L(c)$, $\mathbf{x}_{w,c}^{base}$ contains

$$\dot{P}_j(t), \dot{P}_j(t-1), \dot{S}_j(t), \dot{S}_j(t-1), P_j(t), S_j(t) \quad (23)$$

This design injects two reservoir-engineering drivers of pressure response: neighbor coupling through local connectivity (pressure interference and transmissibility-weighted gradients), and short-term temporal inertia through $\dot{P}$ and $\dot{S}$, which carry information about whether the system is accelerating or relaxing as it enters the interval $[t, t+1]$. Including saturation and $\dot{S}$ alongside pressure quantities is essential in CO₂ storage, because mobility evolution and front advancement modify effective transmissibilities and therefore feed back into the pressure derivative.

(ii) Control descriptors $\mathbf{x}_w^{rates}$

To represent the forcing imposed by the wells over the same four-step window, injector controls are summarized using a compact, control-agnostic “level-delta” representation.

Let N_I denote the number of injectors and $q_k(\cdot)$ be the injection control signal for injector $k \in \{1, \dots, N_I\}$. The CO₂ rate history is sampled at the four window nodes and augmented by three increments:

$$r_k(w) = [q_k(t-2), q_k(t-1), q_k(t), q_k(t+1), \Delta q_k(t-2 \rightarrow t-1), \Delta q_k(t-1 \rightarrow t), \Delta q_k(t \rightarrow t+1)]^T \quad (24)$$

The first four entries describe the rate levels over the window, whereas the last three entries explicitly encode rate transitions between consecutive time levels. Although these increments are mathematically recoverable from the four rate levels, they are retained as engineered descriptors so that schedule changes are provided directly to the compact ANN, rather than requiring the network to infer finite-difference relationships internally. This is useful because abrupt rate changes are among the main drivers of pressure transients and well-interference effects.

Rates are first min–max scaled independently for each injector to the interval $[0, 1]$. The normalized rate levels are then used to compute the corresponding deltas, so that the proxy captures ramp-ups, step changes, and shut-ins without being tied to absolute rate magnitudes. The inclusion of both levels and deltas is deliberate: levels convey the operating regime, whereas deltas encode schedule transitions that tend to generate strong pressure transients and interference effects. Stacking the level–delta descriptors across injectors yields

$$\mathbf{x}_w^{rates} = \begin{bmatrix} r_1(w) \\ \vdots \\ r_{N_I}(w) \end{bmatrix} \quad (25)$$

(iii) Proximity features $\mathbf{x}_c^{prox}$

Control changes do not influence all grid blocks equally. Their impact is spatially localized and depends on geometric distance and connectivity to injector completions. To provide a compact geometric surrogate for this influence, a proximity scalar is computed for each injector.

Let C_k denote the set of perforated (completion) grid blocks associated with injector I_k , and let $\mathbf{r}_c \in \mathbb{R}^3$ be the centroid of block c . For each completion block $j \in C_k$, let $\mathbf{r}_j \in \mathbb{R}^3$ denote its centroid-position vector. The injector–block distance is defined as the minimum Euclidean distance from the focal block c to the completion set of injector I_k :

$$d_k(c) = \min_{j \in C_k} \|\mathbf{r}_c - \mathbf{r}_j\|_2, \quad k = 1, \dots, N_I \quad (26)$$

Proximity is then modeled through an exponentially decaying kernel,

$$\pi_k(c) = \exp\left(-\frac{d_k(c)}{\lambda}\right), \quad k = 1, \dots, N_I \quad (27)$$

where λ is a length scale derived from typical grid spacing. To keep the range consistent across models and grids, each π_k is normalized over the proxy domain (interior and interior-equivalent cells) to lie in $[0, 1]$. The proximity block is finally assembled as

$$\mathbf{x}_c^{prox} = [\pi_1(c), \dots, \pi_{N_I}(c)]^T \in [0, 1]^{N_I} \quad (28)$$

These features provide a compact geometric summary of “how strongly” each injector can influence the pressure response at c , which is essential for capturing spatially localized near-well dynamics and injector-specific interference.

(iv) Control–proximity interaction features $\mathbf{x}_{w,c}^{inter}$

The interaction block is not an independent source of information, but a derived feature block constructed from the rate descriptors $\mathbf{x}_w^{rates}$ and the proximity descriptors $\mathbf{x}_c^{prox}$. While $\mathbf{x}_w^{rates}$ encodes how the controls evolve in time and $\mathbf{x}_c^{prox}$ encodes where the focal block sits relative to the injectors, pressure transients depend on their combination: a schedule change at injector I_k matters most where $\pi_k(c)$ is large. This coupling is therefore introduced explicitly by multiplying each injector’s level–delta descriptor by its corresponding proximity weight.

Using $r_k(w)$ from Equation (24), define

$$\mathbf{x}_{w,c}^{inter} = \begin{bmatrix} \pi_1(c)r_1(w) \\ \vdots \\ \pi_{N_I}(c)r_{N_I}(w) \end{bmatrix} \quad (29)$$

Equivalently, $\mathbf{x}_{w,c}^{inter} = f(\mathbf{x}_w^{rates}, \mathbf{x}_c^{prox})$, where f denotes this injector-wise multiplication. Thus, no additional primary model input is introduced. The interaction block simply exposes a physically motivated product between schedule changes and spatial injector influence. In practice, these terms help the ML model represent injector-specific pressure propagation and well-interference effects without requiring it to infer all spatial weighting solely from the local stencil variables.

(v) Global schedule indicators $\mathbf{x}_w^{global}$

Finally, a small set of global features describes the overall operating context at reference time t . These features are broadcast (shared) across all grid blocks and help disambiguate similar local stencil states that arise under different global regimes (e.g., early ramp-up versus late-time sustained injection, or different allocation of rates among injectors).

Let $q_k(t)$ denote the CO₂ injection control for injector I_k at time t , for $k = 1, \dots, N_I$. The total injection rate is

$$q_{tot}(t) = \sum_{k=1}^{N_I} q_k(t) \quad (30)$$

The context block contains four components:

- (1) Normalized total injection indicator. $\bar{q}_{tot}(t)$ is obtained by min–max scaling $q_{tot}(t)$ over the scenario time series.
- (2) Instantaneous rate partitioning (injector fractions). To capture how the total rate is distributed among injectors, the per-injector rate fractions are defined as

$$s_k(t) = \frac{q_k(t)}{q_{tot}(t) + \mathcal{E}}, \text{ with } \mathcal{E} > 0 \text{ to avoid division by zero} \quad (31)$$

These fractions distinguish operating regimes that have similar $q_{tot}(t)$ but different allocation among injectors, which can lead to different interference patterns and pressure propagation.

- (3) Normalized cumulative injection (schedule stage). To encode how far the schedule has progressed, cumulative injected volumes are formed as

$$Q_k(t) = \sum_{\tau=1}^t q_k(\tau) \Delta t_\tau \quad (32)$$

where Δt_τ is the simulator time-step size. Each $Q_k(t)$ is min–max normalized over time to yield a unitless cumulative indicator in $[0, 1]$. These features provide “history” information

that is not recoverable from local stencil variables alone: two windows can exhibit similar local states and short-term rate patterns, yet correspond to early-time pressurization versus late-time sustained injection, which can imply different background pressure buildup and hence different $\bar{P}$.

- (4) Normalized time coordinate. A final scalar t/T is included, where T is the total number of simulator time steps, to provide a simple monotone index of schedule progression that complements $Q_k(t)$ and improves identifiability when cumulative profiles are similar across injectors.

Collectively, $\mathbf{x}_w^{global}$ provides global context that helps the ML model disambiguate similar local stencil states arising under different overall operating regimes (e.g., early ramp-up versus late-time sustained injection, or different injector allocations), thereby improving robustness of the pressure-derivative mapping across heterogeneous schedules.

3.4. Saturation-Derivative Proxy Model

The saturation derivative proxy is constructed to remain fully consistent with the pressure-derivative proxy in its window–cell learning setup and in the way inputs are organized around local state, schedule forcing, and injector influence. The essential difference is physical: pressure signals propagate diffusively and affect the entire domain, whereas saturation evolution is transport-controlled and remains confined to the portion of the reservoir that the CO₂ plume has reached. When the total simulated time is relatively limited (for example, in the order of ~ 20 years), the plume typically does not progress across the full reservoir. Only a subset of cells is reached by the plume body and the advancing front within the simulated period. Cells undergoing rapid plume-front passage or strong local saturation redistribution exhibit the largest saturation transients and are identified as fast-varying by the activity criterion of Section 2, so they remain in the full-physics simulator. However, CO₂ presence alone does not define a cell as fast: plume-affected cells with smooth or nearly stabilized saturation histories may remain in the slow-varying proxy domain. The saturation proxy is applied on the complementary slow-varying region, where many samples fall outside the plume footprint over the chosen horizon and therefore have saturation derivatives that are exactly zero or numerically negligible, while nonzero events cluster near the evolving boundary of plume influence. This structure is challenging for a single regressor because the learning target is strongly zero-inflated. Accordingly, dS/dt is predicted with a zero-inflated, two-stage pipeline (Figure 6): Stage A estimates the probability that a given sample corresponds to a non-negligible saturation-change event, Stage B predicts the corresponding change magnitude, and the two outputs are combined through soft gating (probability-weighted magnitude) followed by a post-gate deadband that snaps residual near-zero outputs to exactly zero, with the soft-gate strength and deadband tuned on the validation set.

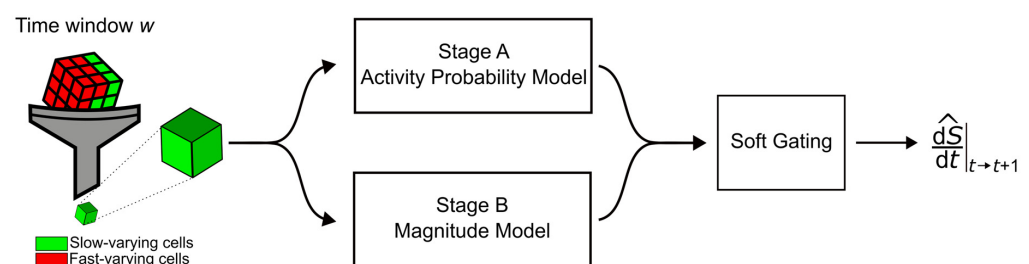


Figure 6. Two-stage zero-inflated saturation-derivative proxy with activity detection, magnitude regression, soft gating and deadband stabilization.

To formalize the zero-inflation, each window–cell sample (w, c) is assigned a binary activity label by thresholding the true derivative magnitude,

$$\alpha_{w,c} = \mathbb{I}(|dS/dt| > \epsilon_{\text{act}}) \quad (33)$$

Stage A predicts the corresponding activity probability conditioned on the feature vector

$$\hat{p}_{\text{act}} = \mathbb{P}(\alpha_{w,c} = 1 | x_{w,c}) \quad (34)$$

In this work, the activity network is trained using mean-squared error (MSE) on the probabilistic output. Under this choice, the optimized objective is equivalent to minimizing a Brier-type [34] score for probabilistic binary events. For the active class probability $\hat{p}_{\text{act}}$ and the binary activity indicator $\alpha_{w,c} \in \{0, 1\}$, the per-sample Brier score is

$$BS = (\hat{p}_{\text{act}} - \alpha_{w,c})^2 \quad (35)$$

which penalizes deviations between predicted probabilities and observed outcomes (smaller penalty when $\hat{p}_{\text{act}}$ is close to $\alpha_{w,c}$, penalty when confidently wrong). Because true active samples are sparse in the slow-varying region, class imbalance is handled explicitly during training by constructing a balanced TRAIN subset that retains all positive samples and only a controlled multiple of negative samples, and by adding extra weight to positive samples to penalize false negatives more strongly. Once trained, the classifier produces $\hat{p}_{\text{act}}$ for all TRAIN/VAL/TEST samples. An operating probability threshold p_{thr} is selected on the validation set by scanning a grid of candidate thresholds and maximizing an F_β score with $\beta = 2$,

$$p_{\text{thr}} = \underset{p \in P}{\operatorname{argmax}} F_\beta(p), \quad F_\beta = \frac{(1 + \beta^2) \text{Precision} \times \text{Recall}}{\beta^2 \text{Precision} + \text{Recall}} \quad (36)$$

with $\text{Precision} = TP / (TP + FP)$ and $\text{Recall} = TP / (TP + FN)$. This tuning deliberately prioritizes recall to reduce missed-active events, since false negatives would suppress physically meaningful saturation-change events to zero. The selected p_{thr} is used for classifier diagnostics (e.g., confusion matrices and per-window error summaries), including on the held-out blind scenario, while the final proxy prediction uses soft gating rather than hard thresholding.

Stage B then targets the magnitude of dS/dt once activity is plausible, a second network is trained only on samples with $\alpha_{w,c} = 1$, so that the regression is not dominated by the abundant inactive zeros. Denoting the magnitude prediction by $\hat{y}_{w,c}$, the regression target may be stabilized using an asinh transform (preserving sign and compressing heavy tails),

$$z = \operatorname{asinh}\left(\frac{y}{s}\right), \quad y = s \sinh(z) \quad (37)$$

where $y = (dS/dt)_{w,c}$ on active samples and s is a fixed scale. The transformed targets are scaled using training-only normalization, and predictions are mapped back to physical units through the inverse transformation. To reduce rare extrapolation spikes, the transform-space predictions are additionally clamped into training-active quantiles prior to inversion. Input normalization (and, when used, target normalization in transform space) is fit on TRAIN only and then applied unchanged to VAL/TEST and the blind scenario.

During inference, the regressor is evaluated for all samples to obtain $\hat{y}_{w,c}$, and the classifier provides $\hat{p}_{\text{act},w,c}$. The final two-stage estimate is formed using the soft gate shown in Figure 6,

$$g(\hat{p}_{act,w,c}) = (\hat{p}_{act,w,c})^\gamma, \quad \left(\frac{d\hat{S}}{dt}\right)_{w,c} = g(\hat{p}_{act,w,c})\hat{y}_{w,c} \quad (38)$$

A small post-gate deadband is then applied so that tiny spurious values are set exactly to zero,

$$\left(\frac{d\hat{S}}{dt}\right)_{w,c} \leftarrow \begin{cases} 0, & \left|\left(\frac{d\hat{S}}{dt}\right)_{w,c}\right| < \varepsilon_{\text{pred}} \\ \left(\frac{d\hat{S}}{dt}\right)_{w,c}, & \text{otherwise} \end{cases} \quad (39)$$

which specifically targets near-zero “speckle” in truly inactive regions (mitigating the vertical smear around the true $dS/dt = 0$ axis). In the implementation, the soft-gate exponent γ and the deadband $\varepsilon_{\text{pred}}$ are auto-tuned on the validation set by minimizing a composite objective that balances magnitude accuracy on true-active samples against residual nonzero predictions on true-inactive samples,

$$J = (\gamma, \varepsilon_{\text{pred}}) = w_{\text{act}} \text{MAE} \left(\left(\frac{d\hat{S}}{dt}\right) - \left(\frac{dS}{dt}\right) \middle| \alpha = 1 \right) + w_{\text{inact}} \text{MAE} \left(\left(\frac{d\hat{S}}{dt}\right) - \left(\frac{dS}{dt}\right) \middle| \alpha = 0 \right) \quad (40)$$

This tuning is designed to simultaneously reduce missed-active events that contribute to a horizontal $\frac{d\hat{S}}{dt} = 0$ band and suppress spurious nonzero outputs in inactive regions. Performance is therefore reported in two complementary ways: overall metrics that include the inactive zeros (reflecting the deployed distribution in the slow-varying region), and active-only metrics restricted to $\alpha_{w,c} = 1$ (isolating magnitude accuracy once activity is present). Finally, the complete proxy (normalizers, classifier, regressor, p_{thr} , and tuned γ and $\varepsilon_{\text{pred}}$) is applied to the blind scenario, for which the same diagnostics are generated and per-window tables of true/predicted dS/dt are exported to support downstream analysis.

To support the two-stage gated mapping, the same input vector is used in both Stage A (activity classification) and Stage B (magnitude regression). Specifically, the input vector $\mathbf{x}_{w,c}$ reuses the full core representation of Section 3.3 without modification (stencil state, control descriptors, proximity, control–proximity interactions, and global schedule context) and appends an augmentation block $\mathbf{x}_{w,c}^{aug}$ that injects additional information tailored to saturation transport and plume-front onset. The augmented vector is written as

$$\mathbf{x}_{w,c} = \begin{bmatrix} \mathbf{x}_{w,c}^{base} \\ \mathbf{x}_w^{rates} \\ \mathbf{x}_c^{prox} \\ \mathbf{x}_{w,c}^{inter} \\ \mathbf{x}_s^{global} \\ \mathbf{x}_w^{aug} \\ \mathbf{x}_{w,c}^{aug} \end{bmatrix}, \quad \dim(\mathbf{x}_{w,c}) = d_s \quad (41)$$

where the new content lies entirely in $\mathbf{x}_{w,c}^{aug}$ and the total dimension d_s depends on how many augmented descriptors are retained.

The augmentation $\mathbf{x}_{w,c}^{aug}$ is designed to address a limitation that is specific to saturation: in the slow-varying region, whether $\dot{S}$ becomes nonzero depends primarily on whether the plume has had enough time and cumulative injected mass to approach the neighborhood of c , and whether heterogeneity provides a connected pathway that allows that approach. These dependencies cannot be inferred robustly from the short four-step rate pattern alone, especially early in the simulation when local stencil states can look similar in regions that will later diverge depending on plume arrival.

First, a set of proximity-weighted cumulative forcing indicators is introduced to represent localized “dose” delivered toward c . Using the cumulative injection $Q_k(t)$ for

injector k (defined in Section 3.3) and its normalized form, the following features are formed

$$\Phi_k(w, c) = \pi_k(c) \overline{Q_k}(t), \quad k = 1, \dots, N_I \quad (42)$$

and a corresponding total exposure measure,

$$\Phi_{tot}(w, c) = \sum_{k=1}^{N_I} \Phi_k(w, c) \quad (43)$$

These terms provide an injector-resolved and spatially localized proxy for plume access: they become large only when injector k has already injected a substantial cumulative amount by time t and the focal block has strong geometric influence from that injector (large $\pi_k(c)$). In practical reservoir terms, Φ_k distinguishes a cell that is near an injector that has been injecting steadily from a cell that is similarly located but under a late-start or low-cumulative injector, and Φ_{tot} summarizes the combined “delivered potential” from all injectors. This is especially relevant under limited horizons, where plume reach is largely controlled by cumulative injected mass and travel distance.

Second, compact saturation-history descriptors are included to act as robust indicators of proximity to the plume front, computed from saturation values already available in the window and stencil. Rather than introducing additional high-dimensional saturation fields, the augmentation uses low-dimensional summaries that are interpretable in transport terms: an average saturation level within the stencil/window (capturing gradual enrichment), a saturation range or spread (capturing the emergence of local gradients consistent with an approaching front), and a binary front flag that activates when saturation exceeds a small threshold within the stencil/window. These summaries help the classifier distinguish “true inactivity” (far from the plume) from “incipient activity” (front approaching but still weak), and they provide the regressor with a coarse measure of where the sample sits relative to the sharpness of the front. Importantly, these descriptors are not intended to resolve the full saturation profile; they provide stable cues that are difficult to recover from pressure-only information and short-term rate history.

Third, static heterogeneity and geometric descriptors are appended to encode pathway controls on plume migration. For saturation transport, preferential flow paths and barriers often dominate the timing and direction of plume advance. The augmentation therefore includes cellwise descriptors such as centroid coordinates and rock properties (permeability tensor components and porosity), together with derived quantities that directly reflect expected flow behavior: a horizontal permeability scale $k_h = \sqrt{k_x k_y}$ (capturing lateral transmissibility potential), an anisotropy ratio $k_z/k_h + \epsilon$ (capturing vertical versus lateral preference, relevant under buoyancy), a normalized depth indicator (relevant to vertical segregation and structural trapping), and a local permeability-contrast measure relative to the immediate neighborhood. The permeability-contrast term, in particular, acts as a compact indicator of whether the cell lies on a locally more transmissive corridor or in a locally tighter region, which affects both the probability of plume arrival (Stage A) and the expected magnitude/sign of $\dot{S}$ once arrival occurs (Stage B). In combination with the proximity-weighted cumulative forcing indicators, these descriptors allow the proxy to represent the intuitive reservoir-engineering statement that “plume influence arrives first where cumulative injection is high, geometric influence is strong, and permeability pathways support transport”.

With this construction, the saturation proxy preserves the structure and interpretability of the pressure proxy while introducing a two-stage gated mapping that matches the physics of saturation evolution under limited plume extent, and a targeted augmentation set that supplies the additional information needed to detect and quantify the rare nonzero $\dot{S}$ events on the slow-varying regions.

4. Case-Study Reservoir Model and Multi-Scenario Injection Design

4.1. Static Reservoir Model and Well Configuration

The reservoir case used in this study represents a deep saline aquifer characterized by conditions typical of CO₂ storage targets. The formation lies at an average depth of 10,340 ft, with a temperature of 200 °F and a brine salinity of 65,000 ppm. Under these conditions, injected CO₂ is expected to remain in a dense, supercritical state throughout the simulation period. The static model corresponds to a bulk rock volume of 2.4×10^{11} ft³ and an initial water volume of 1.7×10^9 STB, indicating a storage system of sufficient size to accommodate large-scale injection (Table 1).

Table 1. Aquifer characteristics.

Parameter	Value
Average depth (ft)	10,340
Temperature (°F)	200
Salinity (ppm)	65,000
Average horizontal permeability (mD)	272
Average vertical permeability (mD)	55
Porosity	0.25
Bulk volume (ft ³)	2.4×10^{11}
Water in place (STB)	1.7×10^9

Spatially, the reservoir is discretized on a three-dimensional Cartesian grid. The simulation grid contains $120 \times 163 \times 4$ cells in the X, Y and Z directions, respectively, giving a total of 78,240 grid cells. Of these, 9212 cells are active and describe the geometry and internal structure of the reservoir, while the remaining cells correspond to zero-porosity volumes outside the reservoir or in non-flow regions and are excluded from the flow solution arrays during simulation. Porosity in all active cells is set to 0.25, so that storage capacity is spatially uniform.

Permeability is heterogeneous and anisotropic. In the horizontal plane, permeabilities in the X and Y directions are identical, with a mean of 272 mD for the active cells, indicating the presence of laterally extensive, high-quality flow pathways (Figure 7). In contrast, vertical permeability (PERMZ) is roughly an order of magnitude lower (mean = 54 mD), reflecting typical anisotropy associated with shale barriers, laminations or other depositional features that restrict vertical flow. This configuration, which can be summarized as $k_X = k_Y \gg k_Z$, is characteristic of layered aquifer systems in which flow is predominantly lateral and vertical communication is moderated by lower-permeability features.

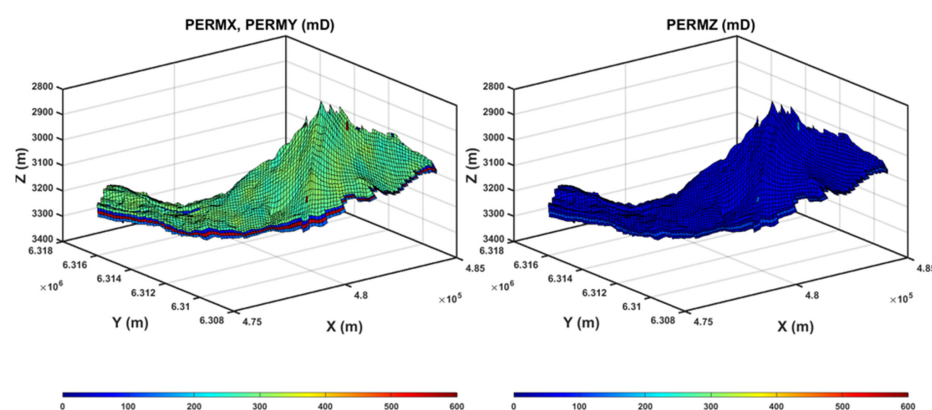


Figure 7. Aquifer grid block permeability in X, Y, and Z directions.

The well configuration adopted in this work is illustrated in Figure 8. Four vertical CO₂ injection wells (I1–I4) are completed in the central part of the aquifer and provide distributed injection along the structure. A single vertical well (P1) is located near the structurally highest region of the model and is used for pressure control and monitoring in all simulations. All wells are completed over the full thickness of the aquifer interval represented in the grid, so that vertical heterogeneity influences flow within the reservoir rather than through differences in completion depth.

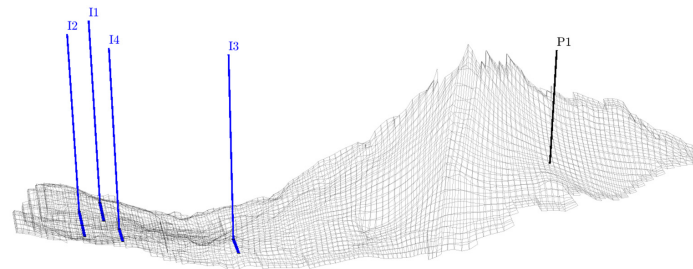


Figure 8. Injection and production well placement in the modelled aquifer.

Operational constraints are imposed through bottom-hole pressure limits. Each injector is subject to a maximum bottom-hole pressure of 9000 psi, reflecting limits associated with caprock integrity, well construction and typical regulatory bounds for deep saline storage. The pressure-control well P1 is operated with a minimum bottom-hole pressure of 3000 psi to avoid unrealistically large pressure drawdown and to keep reservoir conditions within a plausible operating window for long-term CO₂ injection.

4.2. Multi-Scenario Injection and Production Schedules

A family of CO₂ injection scenarios is generated on this static model and well configuration to provide a multi-scenario dataset for activity analysis and proxy training. Scenario design follows the splitting strategy described in [11], in which the injection controls are parameterized through a combination of spatial (injector) splits and temporal splits. The rate values assigned to the resulting space–time control regions are not randomly selected, but are inherited from the optimized schedules generated in that framework, where the control variables correspond to injection rates optimized against CO₂ storage–performance objectives. In this framework, the full injection period is first considered as a single control interval, and injectors are initially treated as one operating group. Additional complexity is introduced by recursively subdividing the injection period into shorter time intervals and by subdividing the injector set into smaller groups, so that each time interval may apply a different distribution of the total injection rate among the injectors.

The splitting procedure can be formulated as a two-dimensional hierarchy in space and time (Figure 9). Let $c_s \geq 0$ and $c_t \geq 0$ denote the spatial and temporal refinement levels, respectively. For a given pair (c_s, c_t) , the injectors are partitioned into $N_s = 2^{c_s}$ independently controlled groups, while the simulation period $[0, T]$ is partitioned into $N_t = 2^{c_t}$ control intervals. In Figure 9, blue horizontal lines indicate spatial splits: each additional blue line increases c_s by one and doubles the number of independently controlled injector groups. Green vertical lines represent temporal splits: each additional green line increases c_t by one and doubles the number of control intervals. The resulting $N_s \times N_t$ rectangles define space–time regions on which the well rates are held constant, so that the total number of independent control variables is $N_{ctrl} = 2^{c_s+c_t}$. Increasing c_t refines the time partition and creates more opportunities for rate changes, whereas increasing c_s refines the injector partition and allows different groups of wells to follow distinct control sequences. By varying (c_s, c_t) the algorithm generates operating schedules that range

from fully uniform injection ($c_s = c_t = 0$) to increasingly differentiated patterns in which individual wells follow multi-step, well-specific rate trajectories.

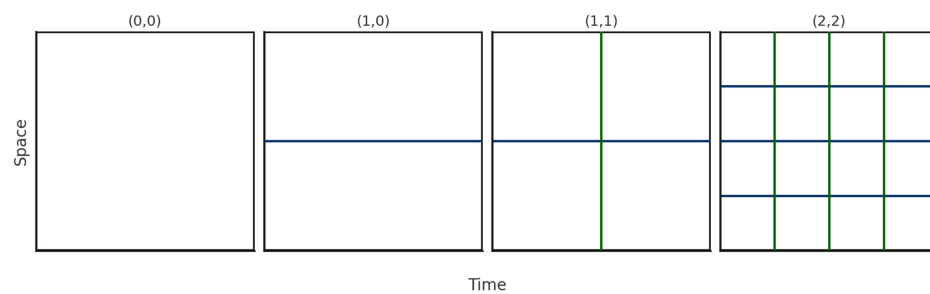


Figure 9. Conceptual space–time refinement of well controls.

Within this structured framework, a sequence of scenarios is generated for the present study, starting from the simplest possible schedule 0_0 and progressing to more complex ones. The scenario label s_t is used as compact notation, where s denotes the spatial refinement level and t denotes the temporal refinement level. Scenario 0_0 contains no temporal or spatial splits: all injectors operate under identical rate limits throughout the entire simulation period. Scenario 1_0 introduces a spatial split only: the injectors are divided into two independently controlled groups, while the rates remain constant in time. Scenario 1_1 combines this spatial split with a single temporal split, so that the two injector groups can follow different rates over two control intervals. Scenarios 2_1 – 2_5 involve multiple temporal intervals and a deeper level of spatial refinement, leading to increasingly irregular and realistic injection profiles. In these scenarios, injectors no longer share a common trajectory: the rate applied to each well may rise, fall or plateau across successive intervals, reflecting changes in group membership and rate allocation as the schedule progresses. The most intricate patterns occur in scenarios 2_4 and 2_5 , which display several distinct steps and redistributions of the injection rate. The resulting growth in control complexity is illustrated in Figures 10 and 11, which present the per-well CO₂ rate for the simplest scenario 0_0 and the most complex scenario 2_5 , respectively.

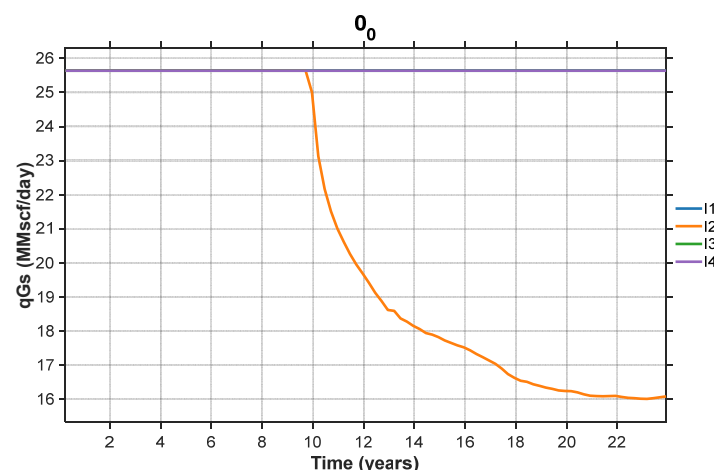


Figure 10. Per-well CO₂ rate for scenario 0_0 .

A concise summary of the structure of all scenarios is provided in Table 2. Together, these scenarios form a controlled progression from coarse, uniform injection to complex, multi-interval, multi-group schedules, providing a sufficiently diverse dataset for both activity classification and proxy-model training.

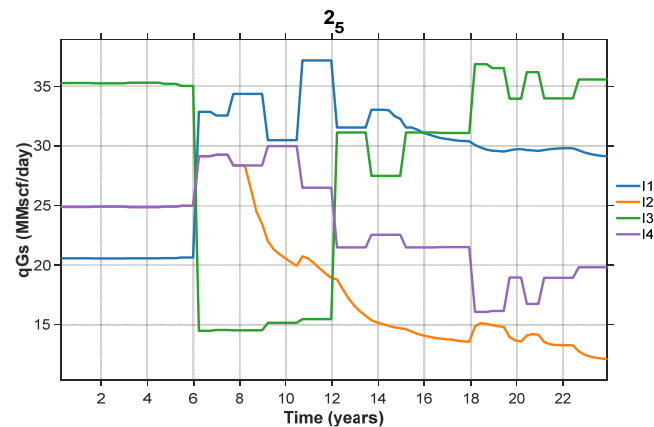


Figure 11. Per-well CO₂ rate for scenario 25.

Table 2. Refinement stages in space and time used to construct the injection scenarios.

Scenario	c_s	c_t	Description
0 ₀	0	0	1 control variable: single constant rate shared by all injectors over the full simulation period.
1 ₀	1	0	2 control variables: 2 injector groups with independent constant rates; no temporal variation.
1 ₁	1	1	4 control variables: 2 injector groups, each with independent rates in 2 time intervals.
2 ₁	2	1	8 control variables: 4 injector groups with independent rates in 2 time intervals.
2 ₂	2	2	16 control variables: 4 injector groups controlled independently in 4 time intervals.
2 ₄	2	4	64 control variables: 4 injector groups controlled independently in 16 time intervals.
2 ₅	2	5	128 control variables: 4 injector groups controlled independently in 32 time intervals.

4.3. Similarity of Pressure and Saturation Distributions Across Scenarios

The scenario comparison presented in this section provides the basis for identifying which schedules are sufficiently distinct for training and which schedule can be retained as a blind test case in Section 5. The impact of the different injection schedules on reservoir behavior is quantified by comparing pressure and saturation histories across all scenarios on a common field-wide basis. For each scenario s , two matrices are considered,

$$P_{mat}^{(s)}(t, c), \quad (44)$$

$$S_{mat}^{(s)}(t, c) \quad (45)$$

where t denotes the discrete simulation time step and c denotes the active grid cell index. Thus, each row corresponds to a specific simulation time and each column to a specific active cell in the grid. All matrices are truncated to the common set of time steps and active cells shared by every scenario so that the trajectories are directly comparable in both indices (t, c) .

Pairwise similarity between scenarios is evaluated using combined absolute/relative tolerances applied entry wise to each variable. For pressure, two entries P_i and P_j taken at the same (t, c) location in scenarios i and j are considered equal if

$$\left| P_{t,c}^{(i)} - P_{t,c}^{(j)} \right| < \max \left(3 \text{ psi}, 10^{-6} \max \left(\left| P_{t,c}^{(i)} \right|, \left| P_{t,c}^{(j)} \right| \right) \right) \quad (46)$$

For saturation, two entries S_i and S_j at the same (t, c) location are regarded as equal if

$$\left| S_{t,c}^{(i)} - S_{t,c}^{(j)} \right| < \max \left(10^{-3}, 10^{-3} \max \left(\left| S_{t,c}^{(i)} \right|, \left| S_{t,c}^{(j)} \right| \right) \right) \quad (47)$$

with saturation expressed as fraction.

These tolerances combine an absolute engineering threshold with a small relative term. For the pressure and saturation ranges encountered in this study, the absolute terms dominate the comparison; therefore, the criteria effectively treat pressure differences below 3 psi and saturation differences below 10^{-3} as practically indistinguishable. The absolute pressure limit of 3 psi is several orders of magnitude smaller than the operating pressure window of the aquifer (3000–9000 psi) and smaller than typical uncertainties arising from formation pressure estimates, fracture-gradient bounds or numerical discretization. Differences below this level have no meaningful impact on injectivity, plume migration or caprock integrity. The saturation tolerance of 10^{-3} corresponds to a 0.1 percentage point change in fluid fraction, which is below the resolution associated with grid-block averaging and geological uncertainty at the model scale. The relative terms are retained only as a standard scale-aware numerical safeguard, but they do not control the tolerance for the pressure and saturation magnitudes considered here.

Using the criteria in Equations (46) and (47), element-wise equality masks are constructed for every ordered pair of scenarios (i, j) . For pressure,

$$E_P^{(i,j)} \in \{0, 1\}^{N_t \times N_c}, \quad E_{P,t,c}^{(i,j)} = \begin{cases} 1, & \text{if } P_{t,c}^{(i)} \text{ and } P_{t,c}^{(j)} \text{ satisfy (46)} \\ 0, & \text{otherwise.} \end{cases} \quad (48)$$

and analogously for saturation,

$$E_S^{(i,j)} \in \{0, 1\}^{N_t \times N_c}, \quad E_{S,t,c}^{(i,j)} = \begin{cases} 1, & \text{if } S_{t,c}^{(i)} \text{ and } S_{t,c}^{(j)} \text{ satisfy (47)} \\ 0, & \text{otherwise.} \end{cases} \quad (49)$$

These masks are then averaged and converted into scalar equality fractions,

$$Q_P^{(i,j)} = \frac{1}{N_t N_c} \sum_{t=1}^{N_t} \sum_{c=1}^{N_c} E_{P,t,c}^{(i,j)}, \quad (50)$$

$$Q_S^{(i,j)} = \frac{1}{N_t N_c} \sum_{t=1}^{N_t} \sum_{c=1}^{N_c} E_{S,t,c}^{(i,j)} \quad (51)$$

which measure the fraction of all (t, c) positions at which two scenarios produce indistinguishable values within the prescribed tolerances. Collecting these values over all pairs (i, j) yields pressure- and saturation-based similarity matrices Q_P and Q_S with entries in $[0, 1]$. Figure 12 shows Q_P , whereas Figure 13 presents the corresponding matrix Q_S .

The pressure-based similarity (Figure 12) exhibits generally low off-diagonal values (typically 0.02–0.06), indicating that, under the tight tolerances in Equation (46), most time-cell pressure entries differ across scenarios. From an engineering standpoint, this shows that changes in how the total injection rate is distributed between wells and control intervals are sufficient to generate distinct pressure footprints in space and time. Two groups with elevated similarity can nevertheless be identified. The pair 1_0 – 1_1 shows a moderate level of agreement (≈ 0.52), meaning that roughly half of the time-cell locations have pressure differences below the engineering tolerance, while the remainder still exhibits meaningful deviations. At a deeper refinement level, scenarios 2_2 , 2_4 and 2_5 display higher equality fractions than the rest of the matrix. Within this group, the pair 2_4 – 2_5 is the most alike, with an equality fraction close to 0.98, whereas the pairs involving 2_2 remain clearly more similar

to each other than to the earlier schedules but do not reach this extreme level. Scenario 2_1 (four injector groups, two time intervals) stands out as a transitional case: its off-diagonal equality fractions remain low against both the simpler schedules ($0_0, 1_0, 1_1$) and the more refined schedules ($2_2, 2_4, 2_5$), indicating a distinct pressure evolution that reflects the first introduction of a full four-group spatial split while the temporal partition is still relatively coarse. Overall, this pattern is consistent with the hierarchical construction of the schedules: incremental refinements in space and time tend to preserve the main pressure gradients and interference structure within a cluster, whereas larger changes in the control strategy, such as the step from 1_1 to 2_1 or from 2_1 to 2_2 – 2_5 , lead to clearly different pressure responses that are relevant for pressure management and containment assessment.

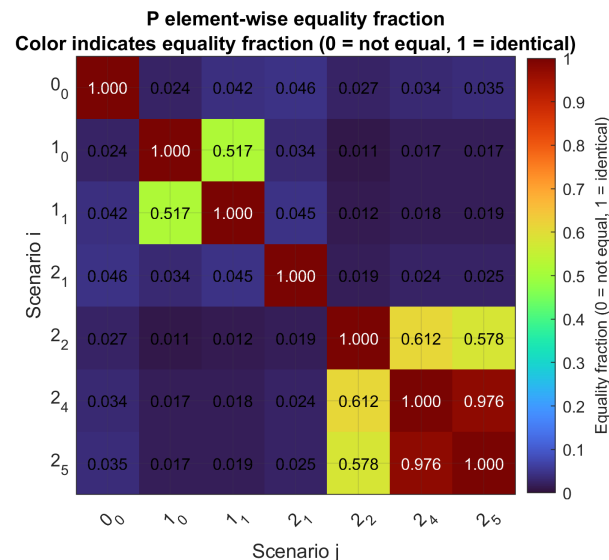


Figure 12. Pairwise element-wise pressure equality between injection scenarios.

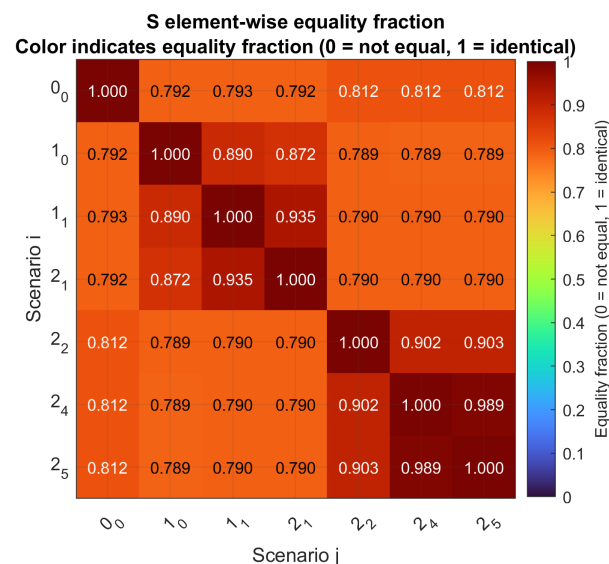


Figure 13. Pairwise element-wise saturation equality between injection scenarios.

In contrast, saturation similarity (Figure 13) is high across the entire scenario set. Most off-diagonal entries fall in the range 0.79–0.90, and closely related scenarios such as 2_2 , 2_4 and 2_5 exceed 0.90. For the aquifer and 24-year injection period considered here, this indicates that the large-scale CO_2 saturation distribution is comparatively insensitive to the detailed allocation of injection among wells and control intervals. From a reservoir-

engineering perspective, this behavior reflects the finite spatial extent of the CO₂ plume over the simulated time frame: after 24 years of injection, a substantial portion of the model remains unswept and retains essentially the initial brine saturation in all scenarios, while plume growth is confined to regions in the vicinity of the injectors and along the same high-permeability pathways. Many grid cells therefore either remain unchanged or experience very similar displacement histories across scenarios, so grid-block-averaged saturation is controlled more by static heterogeneity and total injected mass than by fine details of the rate history. Over longer injection periods or for substantially higher cumulative injected volumes, larger divergences in saturation distribution would be expected. However, within the 24-year window examined here, alternative schedules that produce appreciably different pressure signatures (and thus different implications for caprock integrity and pressure management) still lead to broadly similar plume extents and sweep patterns at the field scale.

5. Results

5.1. Activity-Based Fast/Slow Classification of Grid Blocks

Following the pressure- and saturation-similarity assessment across scenarios (Section 4.3), five injection schedules (0_0 , 1_0 , 1_1 , 2_2 and 2_4) were selected to define the set of operating conditions used for constructing the activity mask and for subsequent proxy development. Together, these schedules span the spectrum from smooth controls (0_0) to highly time-varying controls (2_4), which is essential for identifying grid blocks that repeatedly experience strong transients under realistic variations in injection strategy. Scenario 2_5 was not retained because it represents a close variant of the same refined-control tier as scenario 2_4 and would therefore over-represent that regime without meaningfully expanding the range of transient behavior captured. The blind scenario used for out-of-sample proxy evaluation is 2_1 , which is excluded from both the mask-definition set and all proxy-training datasets.

The activity-based classifier defined in Section 2 was applied separately to each of the five retained scenarios using scenario-specific 90th-percentile thresholds for both activity channels: the pressure-activity threshold $\tau_p^{(s)}$ and the saturation-activity threshold $\tau_s^{(s)}$. Thus, within each scenario, a grid block is considered highly dynamic in pressure if it lies in the upper 10% tail of the pressure-activity distribution, and highly dynamic in saturation if it lies in the upper 10% tail of the saturation-activity distribution. The 90th percentile is used here as a conservative working choice rather than as a universal threshold. Lower percentile values would classify more cells as fast and restrict proxy use to a smoother slow-varying subset, whereas higher values would increase proxy coverage but reduce the conservativeness of the partition. The activity statistics were computed only on proxy-eligible interior-equivalent grid blocks (interior cells and imaginary interior cells with a complete six-neighbor stencil). The scenario-level fast labels were then aggregated through the union operator of Equation (13), resulting in a single conservative, multi-scenario partition of the grid into fast- and slow-varying blocks. This partition is subsequently used as the operational mask that determines where the proxy may be applied.

Applying this procedure yields 8243 interior-equivalent grid blocks, of which 2329 (28.3%) are classified as fast and 5914 (71.7%) as slow. From a reservoir-engineering standpoint, this split is consistent with the localization of CO₂-storage dynamics: strong temporal gradients are expected near injectors, where pressure responds rapidly to changes in rate and well interference, and in parts of the plume body/front where saturation changes strongly. The far field remains largely brine-filled and responds more smoothly, making it a natural candidate for proxy replacement.

A physical reference for the spatial pattern is provided in Figure 14, which shows the gas saturation distribution at the final simulation time for scenario 2₄. The plume is localized and structured, with elevated saturations concentrated around injectors I1–I4 and along the dominant up-dip migration corridor toward P1, while distal regions remain at zero or near-zero CO₂ saturation.

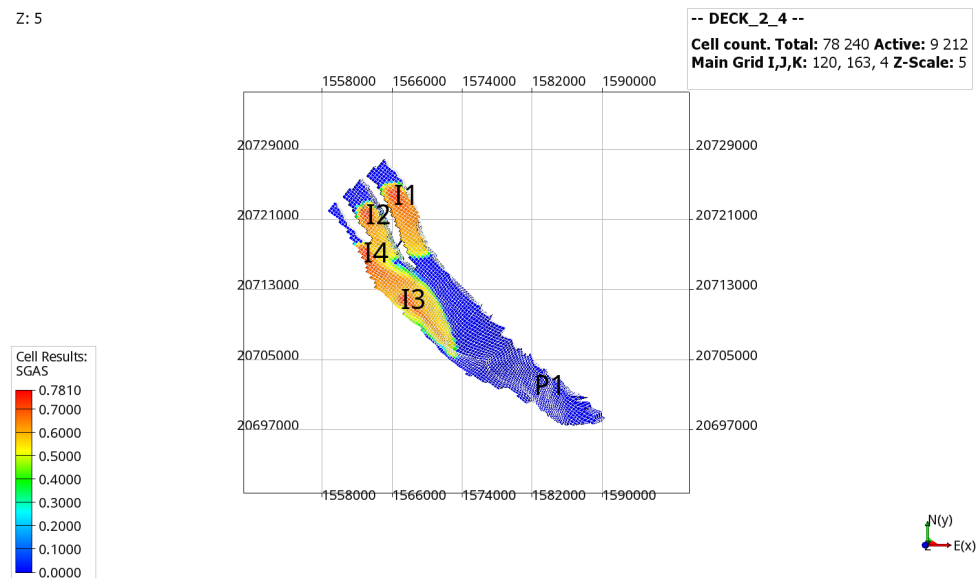


Figure 14. Scenario 2₄—CO₂ saturation distribution at the final time step.

Figure 15 overlays the final fast/slow labels on the grid. Fast cells (red) form a contiguous belt around the injectors and extend along the most dynamically active part of the main plume corridor, whereas slow cells (green) occupy the far-field and low-saturation regions. Boundary cells excluded from the activity analysis are shown in light grey to outline the domain.

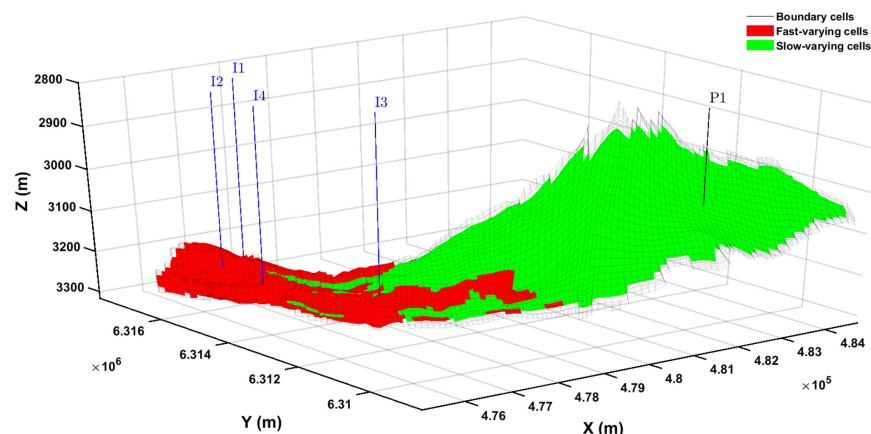


Figure 15. Multi-scenario activity-based partition of the grid into fast (red) and slow (green) interior cells.

The spatial agreement between the fast set and the plume/near-well region confirms that the activity-based classifier has a direct physical interpretation. The fast-varying cells are concentrated around the injection wells and extend laterally along the main CO₂ plume-influenced corridor, where injection-rate changes, pressure interference between wells, and plume expansion generate the strongest local transients. In contrast, the slow-varying cells occupy the broader far-field part of the aquifer and the weakly plume-affected regions, including areas toward the structurally higher part of the model where the pressure

response is smoother and saturation changes are absent, near-zero, or gradual. Thus, the slow-varying region should not be interpreted as a purely mathematical low-activity subset, but as the portion of the reservoir where pressure propagation is damped and plume-related saturation evolution is limited enough for derivative-based proxy updates to be more reliable. Importantly, this does not mean that all CO₂-saturated or plume-affected cells are excluded from the slow set.

Scenario dependence is summarized in Figure 16, which visualizes the fast/slow outcome for the full set of 8243 interior-equivalent grid blocks across the five retained scenarios (columns). Each entry is binary (red = fast, gray = slow) and indicates whether a given block exceeds at least one of the scenario-specific 90th-percentile activity thresholds. The blocks are not displayed in their original grid ordering; instead, rows are sorted in descending order by the number of scenarios in which a block is classified as fast, so the vertical axis represents an ordered list (rank), not a physical cell index.

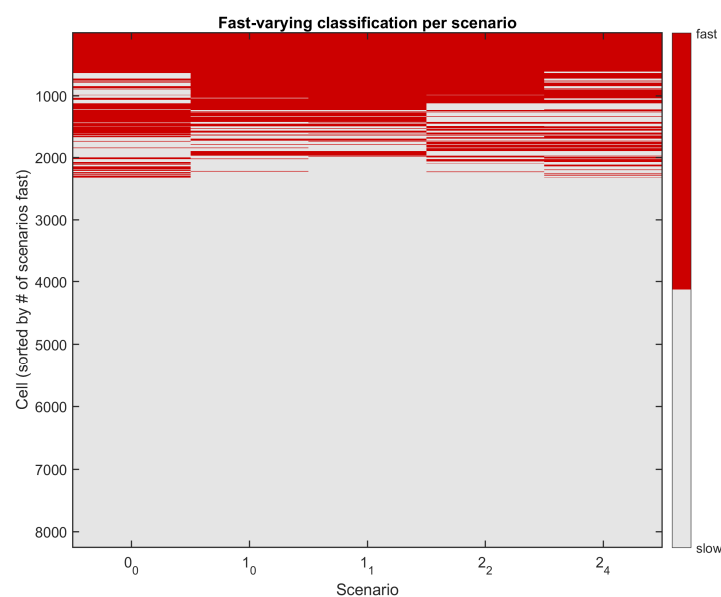


Figure 16. Scenario-wise fast/slow classification of cells across the retained training schedules.

With this ordering, the interpretation is immediate. The upper band, which remains red across multiple columns, corresponds to a compact core subset of blocks that are repeatedly classified as fast under different schedules, consistent with regions that remain intrinsically dynamic due to persistent near-well transients and plume adjacency. Below this core band, intermittent red stripes indicate blocks whose fast behavior is scenario-dependent, reflecting control-induced shifts of the most transient region along established flow paths. The dominant lower portion of the matrix is uniformly gray, showing that most interior-equivalent blocks remain slow in every scenario. This confirms that high-transient behavior is confined to a relatively small subset of the grid, while the proxy-eligible domain is extensive and stable across the tested operating strategies.

Figure 17 breaks down, for each scenario, why a block enters the fast set. A block is counted as *P*-only when it exceeds the pressure-activity threshold $\tau_p^{(s)}$ but not the saturation threshold $\tau_s^{(s)}$; it is *S*-only when it exceeds $\tau_s^{(s)}$ but not $\tau_p^{(s)}$; and it is counted as both when it simultaneously exceeds both thresholds. Because the overall partition used in the workflow is defined by the union across scenarios, the total union fast set contains 2329 blocks, while Figure 17 shows how those fast labels are distributed across the two physical “drivers” within each scenario.

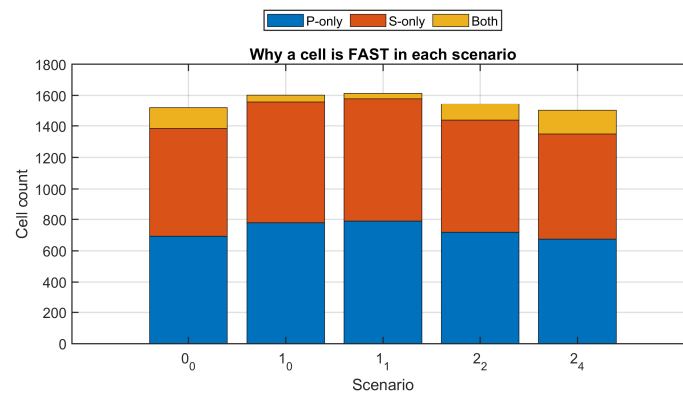


Figure 17. Pressure- and saturation-driven components of the fast set for each training scenario.

The resulting decomposition is reservoir consistent. Pressure-driven fast blocks primarily reflect rapid pressure adjustments caused by changes in injection rates and well interference, which are strongest in the near-well region and along high-transmissibility connections between wells. Saturation-driven fast blocks reflect locations where CO₂ displacement produces sufficiently strong temporal changes in gas saturation, for example during rapid plume-front movement, local brine remobilization, or mobility changes as gas saturation evolves. These effects are commonly associated with the active plume body and its advancing edge, but CO₂ presence alone does not define a cell as fast. A plume-affected cell may still be classified as slow if its saturation history changes smoothly or has nearly stabilized over the classification interval. The both category is smaller because the most extreme pressure transients and the strongest saturation changes do not necessarily peak in exactly the same locations or at the same times. Pressure response is often strongest immediately around wells, whereas saturation activity can peak where the moving front produces the largest temporal saturation variations downstream of injection.

The fast/slow partition should therefore not be interpreted as a physical no-flow boundary, and it does not assume that pressure gradients vanish at the transition between fast- and slow-varying cells. Rather, it is an activity-based proxy-eligibility mask constructed from simulator-derived pressure and saturation activity. Cells where rate switching, well interference, high-transmissibility pressure communication, or plume-front effects generate strong local transients are classified as fast and excluded from proxy treatment. Therefore, a cell is retained in the slow set only if its activity remains below the multi-scenario fast threshold. Such cells may still experience pressure propagation, but their pressure response is smoother than that of the excluded fast cells.

Across the range from scenario 0₀ to scenario 2₄, the relative contributions of *P*-only, *S*-only, and both remain broadly similar. This indicates that refining the control schedule changes the magnitude and spatial extent of transients, but it does not fundamentally change the type of dynamics that define the fast set. In other words, the classifier does not drift toward labeling “mostly pressure-fast” or “mostly saturation-fast” as controls become more heterogeneous. Instead, it continues to flag fast blocks through a balanced combination of interference-dominated pressure transients and plume-related saturation transients.

The underlying activity distributions are examined in Figures 18 and 19 for scenarios 0₀ and 2₄. In both cases, the activity measures are strongly right-skewed: the majority of interior-equivalent blocks exhibit very low activity, whereas a comparatively small upper tail concentrates the localized high-transient behavior. The dashed vertical lines mark the scenario-specific 90th-percentile thresholds $\tau_P^{(s)}$ and $\tau_S^{(s)}$, which define the cutoffs for pressure- and saturation-driven fast behavior, respectively. As expected, the refined control in scenario 2₄ shifts the activity distributions toward larger magnitudes and increases the absolute threshold values, reflecting a more dynamic near-well pressure response caused

by rate changes and interference, together with stronger saturation transients. Importantly, the classification still follows a consistent “upper-tail” definition of fast behavior within each scenario.

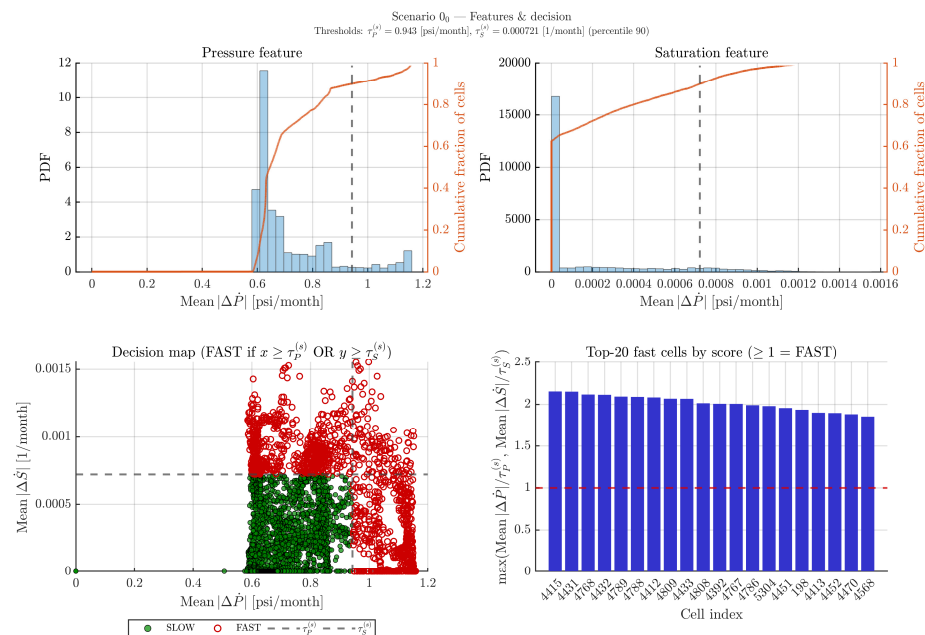


Figure 18. Scenario 0_0 —Activity-feature distributions and fast/slow decision map.

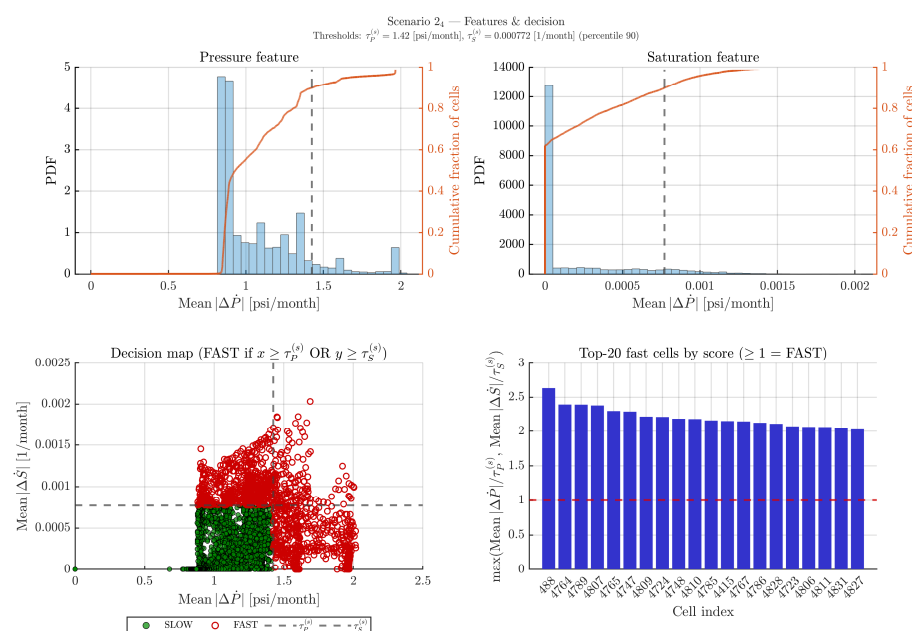


Figure 19. Scenario 2_4 —Activity-feature distributions and fast/slow decision map.

The decision maps (bottom-left panels) place each block in the $\left(\left|\overline{\Delta P}\right|,\left|\overline{\Delta S}\right|\right)$ plane and visualize the classification rule directly. A dense cluster of slow blocks occupies the lower-left region where both activity measures remain below their thresholds. Fast blocks populate the sparser upper tail outside this region, because they exceed at least one threshold. Finally, the lower-right panels display the top-20 ranked fast cells by normalized score, $\max\left\{\left|\overline{\Delta P}\right|/\tau_P^{(s)},\left|\overline{\Delta S}\right|/\tau_S^{(s)}\right\}$, with blue bars and a red dashed line at unity indicating the fast/slow cut-off. These high-score cells correspond to locations adjacent to injectors or

to prominent flow-path features, confirming that the classifier systematically targets the most dynamically sensitive parts of the grid.

This partition provides the operational bridge between physics and learning: it identifies regions where dynamics are most abrupt (near-well transients and plume-front behavior) and therefore conservatively excludes them from proxy prediction, while defining a large smoothly evolving slow domain on which surrogate accuracy is assessed in Section 5.2.

5.2. Pressure and Saturation Proxy Accuracy in Slow Cells

Building on the fast/slow partition of Section 5.1, the proxy models are assessed strictly on the slow-varying cell region. For the pressure-derivative proxy, each sample corresponds to a time-window–cell pair (w, c) and the ANN learns the regression map $\hat{P}_c(t) = M_P(\mathbf{x}_{w,c})$, where $\mathbf{x}_{w,c}$ denotes the input vector defined in Section 3.3 and the target is $\dot{P}_c(t \rightarrow t + 1)$ in [psi/month]. Training is performed on the five injection schedules $\{0_0, 1_0, 1_1, 2_2$ and $2_4\}$, which span distinct control patterns and well-rate allocations and therefore expose the network to a representative range of pressure responses on the slow domain without relying on a single operating regime. Scenario 2₁ is reserved as a fully blind case and is excluded from both mask definition and proxy training; it is selected specifically to test schedule-level generalization, since its global and per-well injection trajectory differs from those used for training and thus stresses the learned control–response mapping under out-of-sample conditions. Scaling is identified on the training subset only and then applied unchanged to validation, test, and blind data, while the slow-cell population is split once into TRAIN/VAL/TEST and kept fixed across scenarios to ensure comparability.

Under this setup, the model exhibits essentially identical behavior across TRAIN/VAL/TEST for the five training schedules, with the parity plots collapsing on the 1:1 line and near-unity coefficients of determination (Figure 20). The reported error levels are very small, with MAE ≈ 0.098 psi/month (mean absolute error, i.e., average $|\hat{y} - y|$) and RMSE ≈ 0.16 psi/month (root-mean-square error, i.e., $\sqrt{\text{mean}(\hat{y} - y)^2}$), while $R^2 \approx 0.9998$ in all three splits (Figure 20, top row). The absolute-error distributions are sharply concentrated near zero and the 95th-percentile indicator remains well below 1 psi/month (Figure 20, bottom row), confirming stable generalization within the slow region for the training schedules.

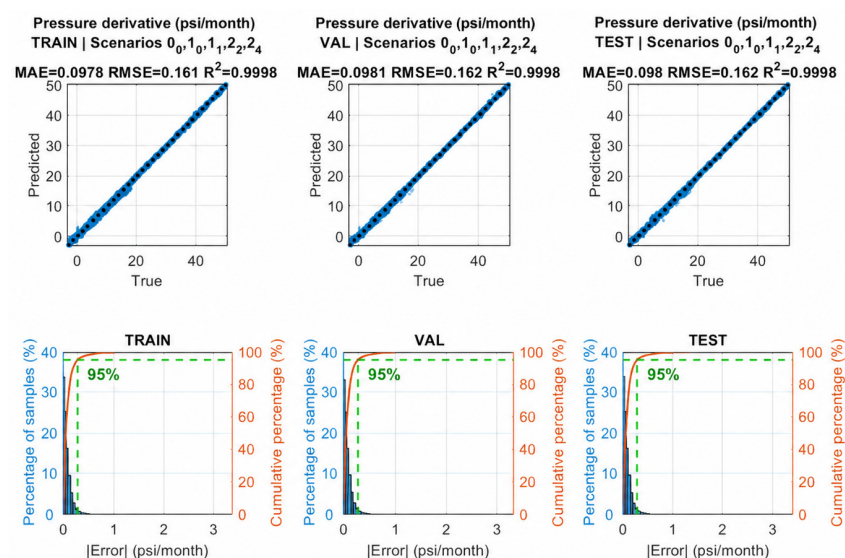


Figure 20. Parity and absolute-error distributions for the pressure-derivative proxy in slow cells (TRAIN, VAL and TEST splits).

A central objective of the proposed framework is not only to restrict surrogate prediction spatially to a conservative slow domain, but also to characterize performance over user-selected time intervals. Because the learning problem is formulated on time windows (which overlap strongly due to the fixed stencil), reporting performance window-by-window would be redundant, while reporting a small number of representative window bands provides a clearer picture of how the surrogate behaves across distinct dynamical regimes. In the results below, blind-scenario performance is therefore summarized over three bands, $w = 5\text{--}20$ (early), $w = 40\text{--}55$ (mid), and $w = 60\text{--}93$ (late), and for each band we report both anchored reconstructions (one-step accuracy) and full-rollout reconstructions (accumulated drift under sustained proxy use). For readability, the absolute-error histograms in the reconstruction plots are displayed over a truncated x-range, with the upper bound set to 20% above the 99.5th percentile of the corresponding absolute-error distribution, while rarer larger-tail errors are nevertheless still captured by the reported MAE, RMSE, R^2 values and the parity plots.

Although the network predicts $\hat{P}$, the quantity of practical interest is pressure. Accordingly, we assess the proxy through pressure reconstruction tests that integrate the predicted derivative over the corresponding step duration. In the anchored one-step reconstruction, the simulator supplies the pressure at the beginning of each update and the proxy supplies the incremental change:

$$\hat{P}_c(t+1) = P_c^{sim}(t) + \hat{P}_c(t)\Delta t \quad (52)$$

where Δt is the corresponding step duration.

Figure 21 reports anchored pressure reconstruction for the blind Scenario 2₁ across the three window bands and shows an essentially perfect collapse on the 1:1 line in all cases ($R^2 = 1.0000$). Absolute errors remain at the sub-psi to few-psi level: MAE/RMSE = 0.98/1.14 psi ($w = 5\text{--}20$), 1.43/2.32 psi ($w = 40\text{--}55$), and 0.37/0.45 psi ($w = 60\text{--}93$). These values are best interpreted in the context of the similarity tolerance used to identify the blind case. In the pressure similarity definition of Equation (46), two pressure values taken at the same (t, c) location in two different scenarios are treated as equivalent when their difference falls within a conservative combined tolerance that is dominated in practice by agreement within a few psi at reservoir pressure levels. Consistent with this, the anchored error distributions remain concentrated near small values, with the mid band ($w = 40\text{--}55$) showing the largest tail, an expected signature of a more transient interval where schedule forcing and local pressure responses are more variable, while the late band ($w = 60\text{--}93$) exhibits the tightest distribution.

To further quantify robustness under sustained use, Figure 22 reports a stricter full-rollout reconstruction across the same three window bands, in which the proxy is applied sequentially so that the predicted $\hat{P}_c(t+1)$ becomes the anchor for the next update (i.e., without re-anchoring to simulator pressure at each step). As expected, small step-wise discrepancies accumulate, producing larger errors than in the anchored case while maintaining very high parity ($R^2 \geq 0.9998$). This behavior reflects integration drift during sequential derivative integration. In the anchored reconstruction, each pressure update is initialized from the corresponding simulator pressure, so the one-step error is reset at every update. In the full-rollout reconstruction, by contrast, the proxy is applied sequentially on the slow-varying cells, and the pressure reconstructed at one step becomes the starting point for the following step. As a result, small derivative-prediction errors are no longer reset and can gradually accumulate as integration drift, because the offline rollout is propagated without re-anchoring to the simulator pressure state and without re-enforcing the full simulator equations at each intermediate step. The rollout MAE/RMSE are 5.34/7.28 psi ($w = 5\text{--}20$), 3.69/5.48 psi ($w = 40\text{--}55$), and 2.80/3.90 psi ($w = 60\text{--}93$). Although the rollout

error tails extend to higher values than the anchored case (reflecting accumulated drift), the error magnitudes remain small relative to the prevailing 7–9 kpsi pressure level and remain within a range that is operationally minor for the intended screening use. Importantly, presenting both anchored and rollout behavior across early/mid/late bands clarifies that the surrogate remains stable not only in late-time regimes but also during earlier and mid-horizon dynamics, even when evaluated on the deliberately challenging least-similar blind schedule.

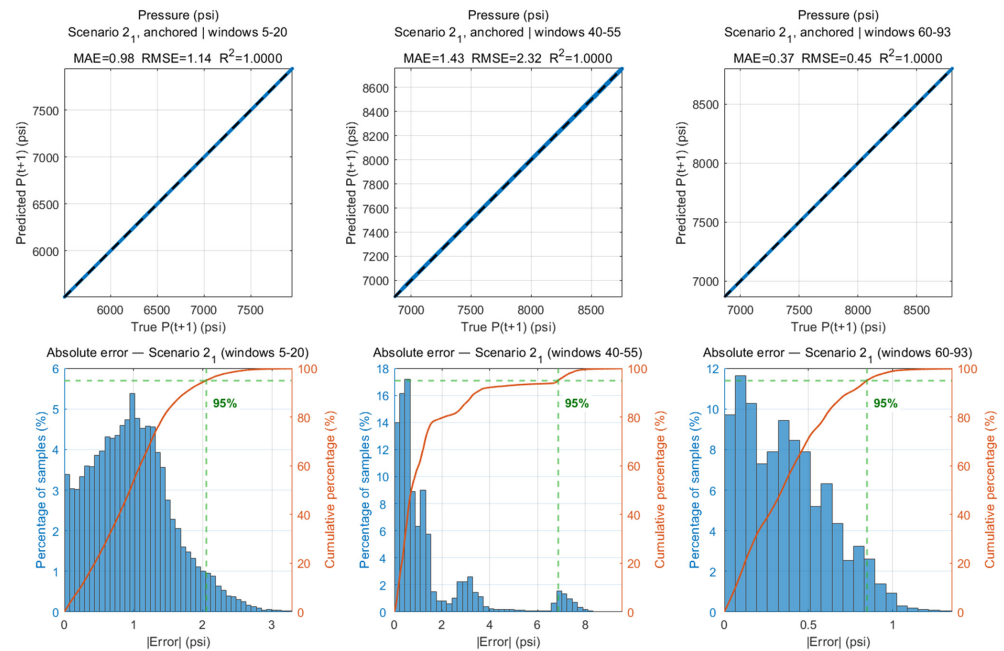


Figure 21. Blind scenario 2₁, anchored pressure reconstruction in slow cells (windows 5–20, 40–55, 60–93): parity and absolute-error distributions.

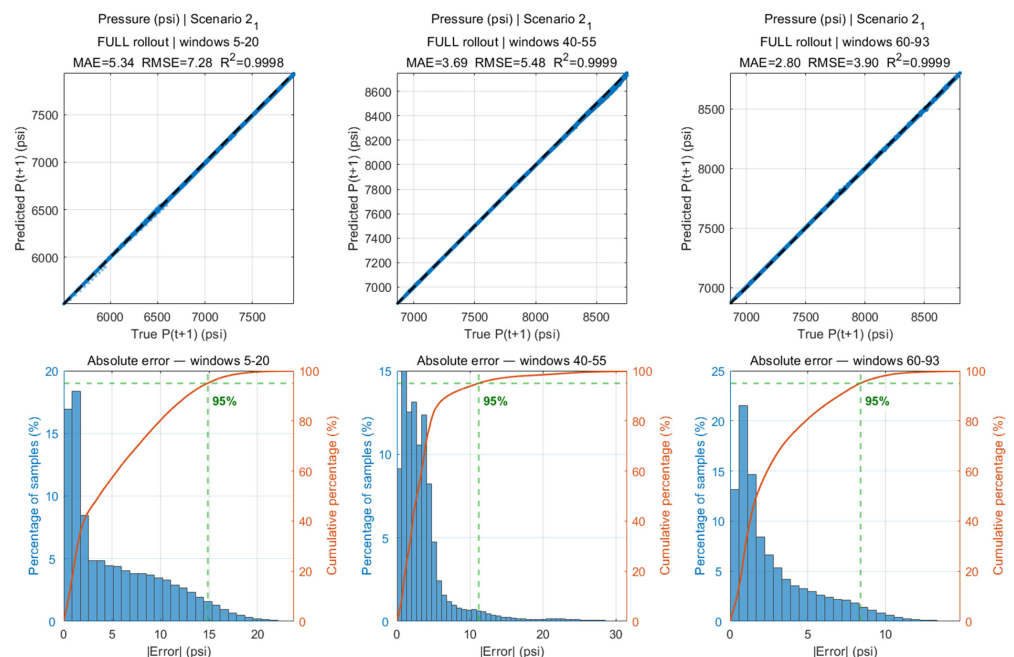


Figure 22. Blind scenario 2₁, full-rollout pressure reconstruction in slow cells (windows 5–20, 40–55, 60–93): parity and absolute-error distributions.

For saturation, the proxy must contend with a markedly different learning geometry than pressure. In the slow-varying region considered in this study, most window-cell pairs

remain effectively brine-dominated over long intervals, so $\dot{S}$ is near zero for most samples, while nonzero saturation change occurs only intermittently, typically at the periphery of plume influence or along weak migration pathways. Accordingly, the saturation proxy is evaluated in the two-stage form described in Section 3.4: a classifier first detects whether a sample is “active” (i.e., belongs to the non-negligible $\dot{S}_c$ regime under the activity definition of Section 3.4), and a regressor is then applied only on the active subset to predict $\hat{S}_c$, with inactive samples set to zero (or a numerically negligible baseline) to preserve the expected quiescent behavior of the slow domain.

Figure 23 reports the confusion matrices of the Stage-A activity classifier for TRAIN/VAL/TEST on the slow cells under the same multi-scenario training setup used for pressure, with scenario 2₁ held out as fully blind. Using the selected operating threshold $\text{thr} = 0.600$ (Section 3.4), the classifier exhibits essentially identical behavior across splits: $\text{Acc} = 0.997$ throughout, with high precision ($P = 0.973/0.972/0.971$ for TRAIN/VAL/TEST) and very high recall ($R = 0.994/0.994/0.993$). In absolute terms, the model correctly identifies almost all truly active samples (e.g., TRAIN: 132,054 true positives with only 800 false negatives, i.e., about 0.6% misses), while keeping false activation in truly inactive samples extremely low (TRAIN: 3608 false positives out of 1,516,966 inactive samples, i.e., about 0.2%). This operating point is reservoir-consistent for the intended hybrid use: missing an active event (false negative) would suppress a real, albeit typically small, saturation increment near the diffuse plume edge, whereas a false activation would introduce spurious “speckle” activity in the far-field. The chosen threshold keeps both failure modes rare, and in particular maintains an active-event miss rate below 1% across splits, which is desirable when the goal is to preserve the correct onset and timing of weak plume influence within the slow domain.

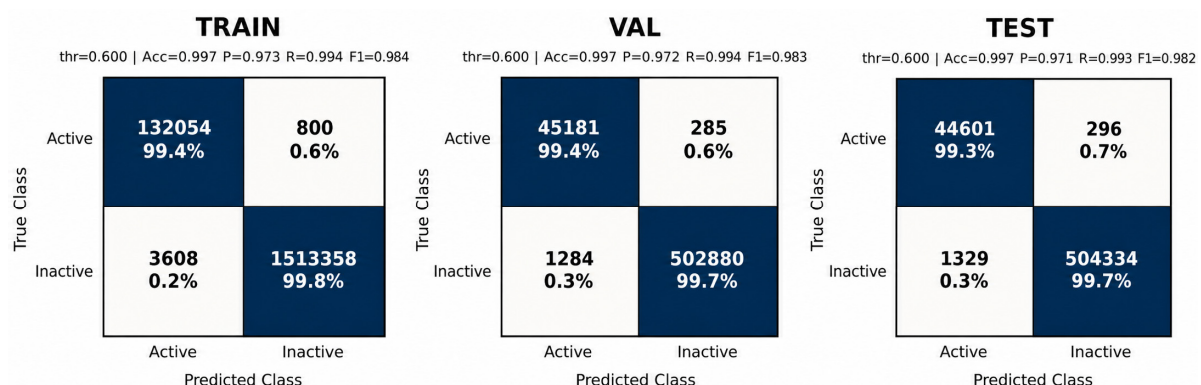


Figure 23. Confusion matrices for the saturation-activity classifier (Stage A) in slow cells for TRAIN/VAL/TEST splits.

Generalization to the fully blind schedule is shown in Figure 24. Despite being excluded from both mask definition and saturation-proxy training, scenario 2₁ yields nearly unchanged classification statistics at the same threshold ($\text{thr} = 0.600$; $\text{Acc} = 0.997$, $P = 0.972$, $R = 0.992$, $F1 = 0.982$). The classifier correctly detects 38,529 of 38,837 true active samples (99.2%), with 308 false negatives (0.8%), while false positives remain limited (1111 cases, about 0.2% of inactive samples). This indicates that the learned activity decision is not schedule-specific: it transfers to an out-of-sample injection trajectory and continues to separate the dominant quiescent slow-cell regime from the rare saturation-change windows that reflect plume-proximal dynamics.

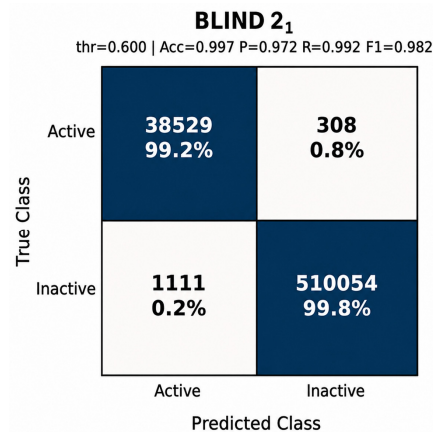


Figure 24. Blind scenario 2₁: confusion matrix for the Stage-A saturation-activity classifier in slow cells, demonstrating schedule-level generalization.

Figure 25 evaluates the Stage B magnitude regressor on true-active samples only (i.e., $|dS/dt| > \varepsilon_{\text{act}}$) by plotting the absolute magnitude error as histograms with the corresponding ECDFs for TRAIN/VAL/TEST. The distributions are strongly concentrated near zero with thin right tails. The marked ECDF reference (horizontal 95% line and its intersection on the x-axis) shows that the 95th-percentile absolute error remains on the order of $\sim 10^{-3}$ fraction/month ($\approx 1.14 \times 10^{-3}$, 1.18×10^{-3} , 1.23×10^{-3} for TRAIN/VAL/TEST in this run), supporting stable generalization of the magnitude mapping and reservoir-consistent incremental updates in the slow domain.

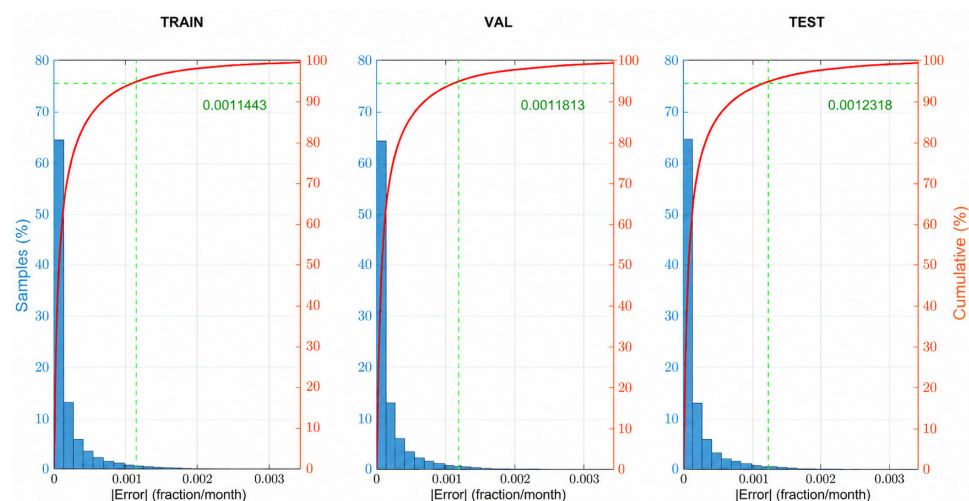


Figure 25. Stage B (Magnitude) performance on true-active samples: Absolute error histograms and ECDFs (TRAIN/VAL/TEST).

Figure 26 reports confusion matrices computed after the full pipeline (Stage A classifier $\rightarrow$ Stage B magnitude regressor $\rightarrow$ probability gating and deadband), where both the ground-truth and predicted derivatives are binarized using the same activity definition as in the training targets: “active” if $|dS/dt| > \varepsilon_{\text{act}}$ and “inactive” otherwise. This diagnostic captures the cumulative impact of (i) activity probability estimation, (ii) magnitude prediction, and (iii) soft gating that suppresses spurious low-amplitude updates while preserving true saturation-change events. Across TRAIN/VAL/TEST, the pipeline achieves near-unit recall ($R \approx 1$; only 41/18/14 false negatives, respectively), confirming that the end-to-end workflow rarely misses true activity in the slow domain—an intentionally conservative design choice because missed-active windows directly degrade plume-front dynamics. Precision is lower ($P \approx 0.69$), indicating that a limited subset of truly inactive

samples are still predicted as active after gating; these correspond to small residual updates that exceed ε_{act} . Overall, the results demonstrate that the pipeline preserves sensitivity to rare saturation-change events while maintaining a bounded false-activation rate prior to saturation-field reconstruction.

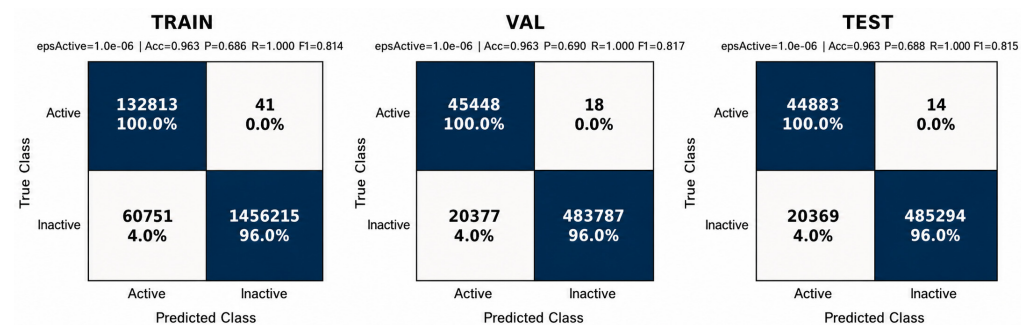


Figure 26. Confusion matrices for the end-to-end two-stage saturation proxy in slow cells for TRAIN/VAL/TEST splits.

Figure 27 extends the same post-pipeline activity assessment to the fully blind injection schedule (Scenario 2₁), using the identical trained normalizers, Stage A and Stage B networks, and the validation-selected gating/deadband parameters. The confusion matrix remains consistent with the split behavior and again shows near-unit recall (19 false negatives out of 38,837 true-active samples in this run), demonstrating robust transfer of the activity decision to an out-of-sample operational trajectory. Precision remains comparable ($P \approx 0.68$), reflecting a controlled level of spurious activations in quiescent regions due to residual low-amplitude updates that exceed ε_{act} . Collectively, these pipeline-level results confirm that the two-stage proxy generalizes its activity behavior to unseen schedules, enabling downstream saturation reconstructions (anchored and rollout) with minimal risk of missed-active dynamics in the slow-varying domain.

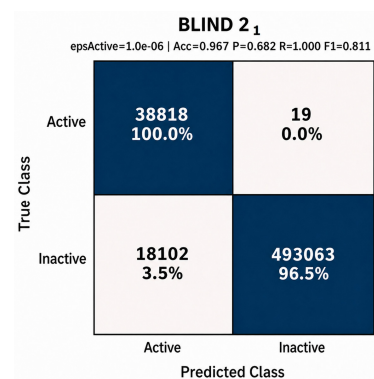


Figure 27. Blind scenario 2₁: confusion matrix for the end-to-end two-stage saturation proxy in slow cells, demonstrating schedule-level generalization.

Because saturation proxy behavior is time-dependent, the saturation results are reported over the same representative window bands used for the pressure reconstruction tests. Consistent with the pressure-reconstruction formulation in Equation (52), saturation performance is reported in terms of reconstructed S_{CO_2} .

Figure 28 reports anchored one-step saturation reconstruction for blind Scenario 2₁ across the three window bands. The parity clouds collapse onto the 1:1 line with $R^2 = 0.9987$ –1.0000, and errors remain very small: $\text{MAE}/\text{RMSE} = 1.323 \times 10^{-5}/2.884 \times 10^{-4}$ ($w = 5$ –20), $7.813 \times 10^{-5}/6.785 \times 10^{-4}$ ($w = 40$ –55), and $1.044 \times 10^{-4}/6.619 \times 10^{-4}$ ($w = 60$ –93). These anchored errors lie predominantly below the entry wise saturation tolerance used to define similarity in Equation (47). In that definition, two saturation

values taken at the same (t, c) location in two different scenarios are treated as equivalent when their difference falls within a conservative combined tolerance that, in practice, corresponds to agreement within about one-thousandth in saturation fraction for the saturation magnitudes encountered here. The absolute-error histograms are therefore consistent with a well-behaved surrogate in the slow domain: it preserves the dominant quiescent background while still resolving the localized nonzero increments associated with weak plume influence.

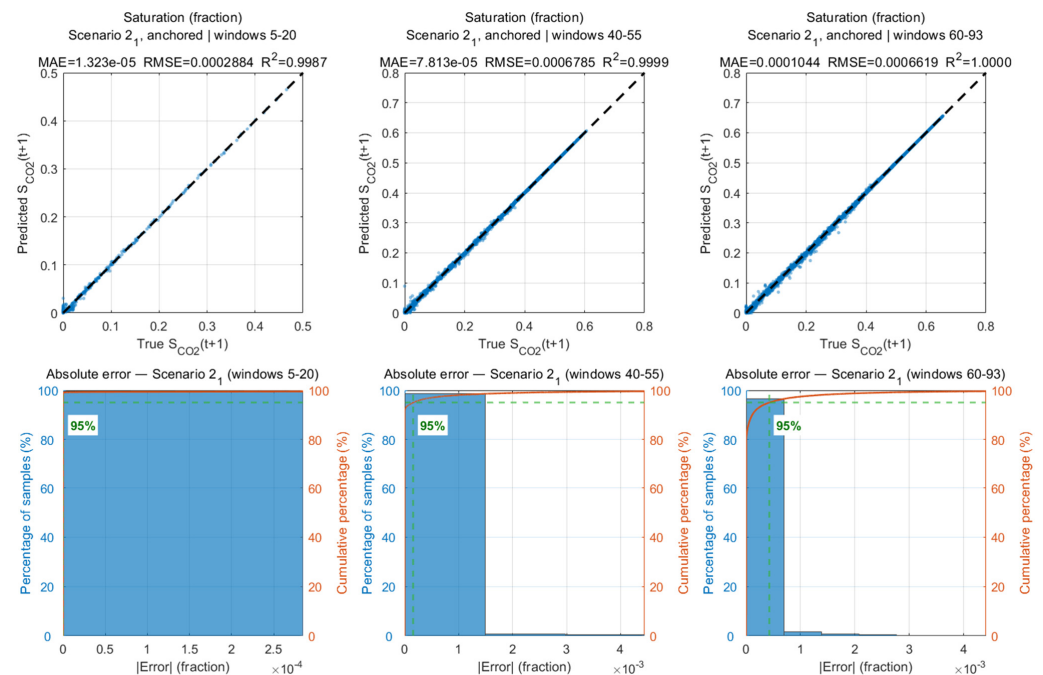


Figure 28. Blind scenario 2₁, anchored saturation reconstruction in slow cells (windows 5–20, 40–55, 60–93): parity and absolute-error distributions.

Figure 29 evaluates the more stringent full-rollout saturation reconstruction across the same blind scenario and window bands, where the predicted $\hat{S}_c(t \rightarrow t+1)$ is propagated forward as the anchor for the next update (i.e., without re-anchoring to the simulator). As expected when integrating a learned derivative over many consecutive steps, small stepwise discrepancies accumulate and the parity cloud widens relative to the anchored case, but agreement remains strong: MAE = 4.833×10^{-5} and RMSE = 0.0008611 ($R^2=0.9887$) for windows 5–20, MAE = 0.0002939 and RMSE = 0.001993 ($R^2=0.9994$) for windows 40–55, and MAE = 0.0007828 and RMSE = 0.003259 ($R^2=0.9995$) for windows 60–93. The absolute-error distributions become broader with more pronounced right tails, reflecting gradual drift under sustained proxy activation. This degradation remains acceptable for the intended role of the framework for three reasons. First, the error magnitude is small on a saturation scale: even the largest reported rollout RMSE (3.259×10^{-3}) corresponds to sub-percent absolute saturation error in fraction units and remains minor relative to the plume-level saturations represented in the evaluated bands (order 0.1 to 1 in the parity plots). Second, the drift is physically benign in practice: saturation is constrained to $[0,1]$, and in the slow domain most cells experience long stretches of weak or zero activity, which limits the opportunity for runaway accumulation; the observed tails therefore represent a conservative upper bound rather than typical behavior. Third, full rollout is intentionally a worst-case stress test because it propagates sequentially without re-initialization; in practical use, drift can be bounded by restricting proxy activation to validated window bands and, when reference states are available, periodically re-initializing the reconstruction, as reflected by the anchored evaluation.

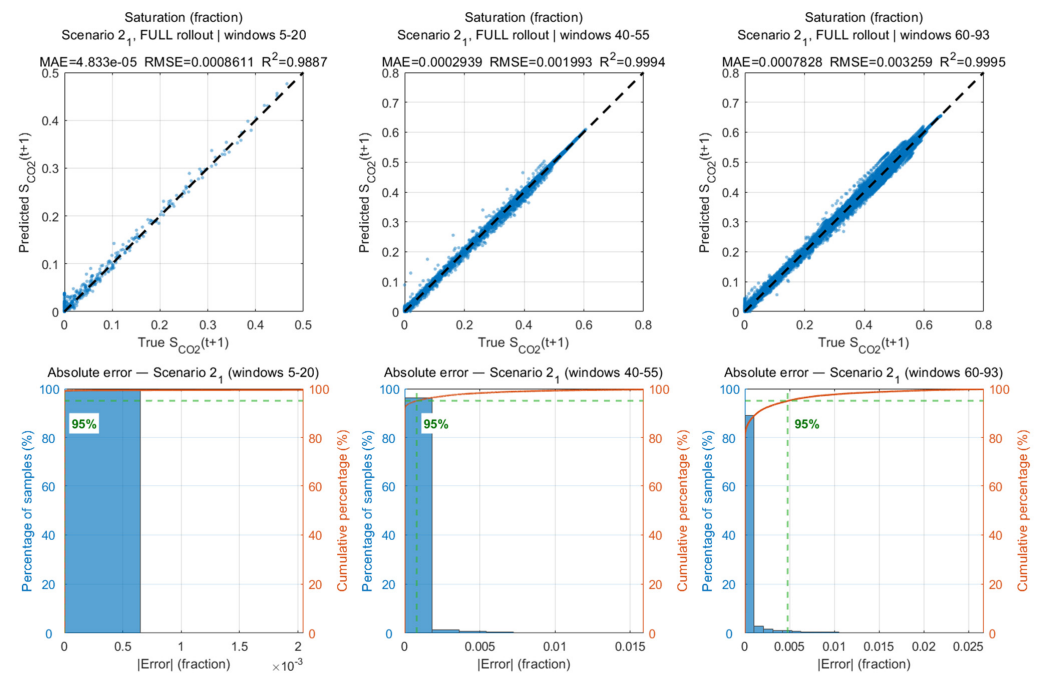


Figure 29. Blind scenario 2₁, full-rollout saturation reconstruction in slow cells (windows 5–20, 40–55, 60–93): parity and absolute-error distributions.

6. Discussion

The proposed framework provides a selective and physically interpretable route for accelerating multi-scenario CO₂ storage simulations. Its main contribution is not the construction of a global surrogate for the entire reservoir, but the development of a proxy-eligibility workflow. The methodology first identifies, at the grid-block level, where pressure and saturation dynamics remain sufficiently smooth for ML updates to be defensible. It then trains and validates derivative-based pressure and saturation proxies only on this activity-screened slow-varying region. This distinction is central to the method: the fast/slow classifier is not a preprocessing step, but a core part of the framework because it determines where the surrogate is allowed to operate and where full-physics treatment must be retained.

The pressure results demonstrate that this selective strategy is effective within the slow-varying domain. The pressure-derivative proxy generalizes consistently across the retained training schedules and transfers to the fully blind schedule, even though the blind injection trajectory was excluded from both mask construction and model training. This schedule-level transfer is essential for screening applications, where the proxy must remain reliable under operating conditions that differ from those used during training. The anchored reconstructions demonstrate near-simulator fidelity when each update is initialized from the simulator state, whereas the full-rollout reconstructions provide a stricter test of accumulated integration drift under sustained proxy use. As expected, rollout errors are larger because local derivative-prediction errors are propagated sequentially without re-anchoring. However, the resulting pressure errors remain small relative to the prevailing 7–9 kpsi reservoir pressure level, indicating that the proxy remains stable within the validated slow-varying region.

The saturation problem is more challenging because CO₂ saturation is transport-controlled and spatially localized, unlike pressure, which propagates more diffusively through the aquifer. In the slow-varying region, many cell-time samples remain outside the active plume movement or experience no meaningful saturation change over a given time window. This creates a strongly zero-inflated target, where most saturation derivatives are

zero or near-zero and only a small fraction of samples contain meaningful nonzero changes. The two-stage saturation proxy is therefore physically motivated: the first stage detects whether a non-negligible saturation-change event exists, and the second stage predicts its magnitude when such an event is present. This structure prevents sparse nonzero saturation-derivative events from being diluted by the dominant zero-change background, while the gating and deadband suppress spurious near-zero activity in quiescent regions.

The methodology is also conservative with respect to rare but important dynamic events. The multi-scenario union rule excludes any grid block that exhibits strong pressure or saturation activity under any retained schedule. Therefore, the most dynamically sensitive regions, including near-well pressure transients, strong pressure-interference pathways, rapid rate-switching responses, and strongly active plume-front behavior, are assigned to the fast region and are not treated by the surrogate. Within the retained slow-varying region, pressure and saturation derivatives are modeled only for cells that have already passed this activity-based screening. Thus, the framework first removes strongly transient dynamics from proxy use and then resolves the weaker derivative variations that remain within the validated slow-domain dataset.

The 24-year active-injection period used in the present case study should be interpreted as the validation horizon of the current ML models, not as a restriction of the methodology. This period also represents the most dynamically challenging stage of the CCS workflow considered here, because injection introduces the strongest rate-driven pressure transients and the most active plume migration. Therefore, the direct quantitative validity of the trained pressure- and saturation-derivative proxies refers to the pressure and saturation regimes represented in this active-injection dataset.

The same activity-based framework can be applied to any selected CCS interval, including a longer injection schedule, a full injection-plus-monitoring schedule, or a separate post-injection monitoring period initialized from simulator restart files. In reservoir-simulation terms, the restart files allow the final pressure and saturation solution from the active-injection period to be passed as the initial condition for the monitoring period. After shut-in, no new rate-driven injection transients are introduced. Pressure transients relax, and plume motion becomes governed mainly by gravity, capillarity, dissolution, and gradual redistribution along available flow paths. Consequently, pressure and saturation changes are expected to become smoother than during active injection, fewer cells are expected to remain fast-varying, and the proxy-eligible region will expand. Under these conditions, the already trained injection-period ML proxies remain applicable to post-injection cells whose states and derivative magnitudes remain within, or smoother than, the regimes represented in the training data. If the monitoring interval introduces states outside this range, post-injection simulation data can be added to update the derivative models for the target interval.

The present results should be interpreted as an offline validation of the slow-region component required for a future hybrid simulator. The current reconstructions quantify local state fidelity within the slow domain, but field-scale decision quantities require the coupled evolution of both the simulator-treated fast region and the ML-updated slow region. Similarly, geological uncertainty should be considered in future applications. Because the saturation proxy uses static heterogeneity descriptors such as permeability, porosity, depth-related quantities, and local contrast measures, deployment across multiple geological realizations would require the activity partition and derivative proxies to be trained or revalidated over the relevant ensemble.

Although measured wall-clock acceleration is outside the scope of the present offline study, the partition provides a useful semi-quantitative estimate of the potential computational leverage. The full simulation model contains 9212 active cells, while the

proxy analysis is performed on 8243 interior and interior-equivalent cells for which the required uniform local stencil can be constructed. Of these, 5914 cells are classified as slow and proxy-eligible, while 2329 are retained in the fast/non-proxy subset. The remaining 969 active cells lie outside the present proxy-analysis domain because they do not satisfy the uniform-stencil requirement and would also remain under full-physics treatment in a future coupled implementation. Therefore, the full-physics active-cell count would decrease from 9212 to 3298 cells, corresponding to a theoretical full-model cell-count reduction factor of approximately $9212/3298 \approx 2.8$. Within the evaluated proxy-analysis domain alone, the corresponding reduction is $8243/2329 \approx 3.5$. Standalone neural-network inference is very fast, on the order of 1–2 ms per timestep on a high-performance laptop. This is because, once trained, the ML proxy performs only a direct forward evaluation of the neural-network mappings for the selected slow-varying cells. In contrast, a fully implicit reservoir simulator advances each timestep by solving a coupled nonlinear system of discretized flow equations, typically using Newton iterations. At each Newton iteration, the nonlinear residuals are evaluated, the equations are linearized through the Jacobian matrix, and the resulting linear system is solved to update pressure and saturation until convergence is reached. Difficult timesteps may also require timestep cuts and repeated nonlinear solves. The ML proxy does not perform these operations: it does not assemble a Jacobian, solve a linearized flow system, or iterate to nonlinear convergence. Nevertheless, the reported inference time should not be interpreted as a measured simulator speedup. Realized acceleration will depend on simulator coupling, nonlinear solver behavior, coupling overhead, safeguard logic, and re-anchoring frequency.

The intended simulator integration is a reduced-domain subdomain strategy. In such a future implementation, the full-physics simulator would solve only the fast-varying subdomain, where the strongest pressure and saturation transients occur. The slow-varying cells would be advanced by the ML derivative proxy and would provide pressure and saturation boundary states to the fast subdomain, so that communication between the simulator-treated and proxy-treated regions is preserved. Practical deployment will also require runtime safeguards, including periodic re-anchoring, step-size limits, gating and deadband controls, consistency checks, mass-balance monitoring, and automatic fallback to full physics when the proxy leaves its validated regime. The present study establishes the necessary pre-coupling evidence: a conservative proxy-eligibility map, stable pressure and saturation derivative updates in the slow domain, and schedule-level transfer to a blind injection case. Full simulator coupling and direct runtime assessment are the next implementation steps.

7. Conclusions

This paper establishes an offline, multi-scenario validation framework for selective surrogate treatment of CO₂ storage simulations on a conservatively defined slow-varying domain. A fast/slow partition is obtained by aggregating activity measures across schedules, yielding a proxy-eligibility map that excludes any grid block that exhibits highly transient behavior under any tested operating strategy. For the present case, 5914 of 8243 interior-equivalent blocks (71.7%) are classified as slow and therefore proxy-eligible. On this slow domain, pressure and saturation time-derivative proxies are trained on five schedules and evaluated for schedule-level generalization on a fully blind schedule, with reliability characterized over time-window bands and stability assessed through both anchored and full-rollout reconstructions.

For pressure, the proxy generalizes stably across schedules and maintains near-simulator behavior in anchored reconstruction, while full-rollout reconstruction provides a conservative bound on drift under sustained sequential integration. For saturation, a

two-stage zero-inflated proxy design preserves slow-domain quiescence through explicit activity detection and conditional magnitude prediction, with stabilization mechanisms used to suppress near-zero artifacts while retaining localized increments; activity behavior remains consistent on the fully blind schedule, supporting robust transfer of the active–inactive separation.

Although the present case study considers a 24-year active-injection period, this interval represents the most dynamically challenging stage of the CCS workflow considered here, because injection introduces the strongest rate-driven pressure transients and the most active plume migration. The current ML models are therefore quantitatively validated for the pressure and saturation regimes represented in this active-injection dataset.

The methodology itself is not restricted to this period. For longer CCS time scales, the same activity-based framework can be applied to a longer injection schedule, a full injection-plus-monitoring schedule, or a separate post-injection monitoring interval initialized from simulator restart files. As discussed above, post-injection dynamics are expected to be smoother because no new rate-driven injection transients are introduced. Under these conditions, the trained injection-period ML proxies remain applicable to post-injection cells whose states and derivative magnitudes remain within, or smoother than, the regimes represented in the training data.

Overall, the results provide a rigorous basis for when and where slow-domain surrogate updates are reliable across operating schedules, including a deliberately challenging blind case. End-to-end simulator coupling and measured wall-clock speedups remain outside the scope of the present offline validation study. The theoretical cell-count reductions reported here should therefore be interpreted as semi-quantitative indicators of potential computational leverage, with direct timing comparisons reserved for future coupled-simulator work.

Author Contributions: Conceptualization, E.M.K. and V.G.; methodology, E.M.K.; software, E.M.K. and S.P.F.; validation, E.M.K. and V.G.; formal analysis, E.M.K.; investigation, E.M.K.; data curation, E.M.K.; writing—original draft preparation, E.M.K.; writing—review and editing, V.G.; visualization, E.M.K.; supervision, V.G.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors gratefully acknowledge Giorgos Theocharous for his assistance in the preparation of the figures.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ketzer, J.M.; Iglesias, R.S.; Einloft, S. Reducing Greenhouse Gas Emissions with CO₂ Capture and Geological Storage. In *Handbook of Climate Change Mitigation*; Springer: New York, NY, USA, 2012; pp. 1405–1440. [[CrossRef](#)]
2. Luo, A.; Li, Y.; Chen, X.; Zhu, Z.; Peng, Y. Review of CO₂ Sequestration Mechanism in Saline Aquifers. *Nat. Gas Ind. B* **2022**, *9*, 383–393. [[CrossRef](#)]
3. Sharmina, M.; Edelenbosch, O.Y.; Wilson, C.; Freeman, R.; Gernaat, D.E.H.J.; Gilbert, P.; Larkin, A.; Littleton, E.W.; Traut, M.; van Vuuren, D.P.; et al. Decarbonising the Critical Sectors of Aviation, Shipping, Road Freight and Industry to Limit Warming to 1.5–2 °C. *Clim. Policy* **2021**, *21*, 455–474. [[CrossRef](#)]
4. Ekardt, F.; Wieding, J.; Zorn, A. Paris Agreement, Precautionary Principle and Human Rights: Zero Emissions in Two Decades? *Sustainability* **2018**, *10*, 2812. [[CrossRef](#)]
5. Michaelowa, A.; Hermwille, L.; Obergassel, W.; Butzengeiger, S. Additionality Revisited: Guarding the Integrity of Market Mechanisms under the Paris Agreement. *Clim. Policy* **2019**, *19*, 1211–1224. [[CrossRef](#)]

6. Huang, Z.; Huang, Y.; Zhang, S. The Possibility and Improvement Directions of Achieving the Paris Agreement Goals from the Perspective of Climate Policy. *Sustainability* **2024**, *16*, 4212. [\[CrossRef\]](#)
7. Tahir, M.U.; Guo, S. Preliminary Investigation of Fracture Behavior during Carbon Dioxide Fracturing of Natural Hydrogen Reservoir with Hard-Core Imperfections. *Reserv. Sci.* **2026**, *2*, 34–51. [\[CrossRef\]](#)
8. Wu, J.; Ansari, U. From CO₂ Sequestration to Hydrogen Storage: Further Utilization of Depleted Gas Reservoirs. *Reserv. Sci.* **2025**, *1*, 19–35. [\[CrossRef\]](#)
9. Ali, J.; Ansari, U.; Ali, F.; Javed, T.; Hullio, I.A. Application of Machine Learning for Effective Screening of Enhanced Oil Recovery Methods. *Reserv. Sci.* **2026**, *2*, 65–80. [\[CrossRef\]](#)
10. Fotias, S.P.; Ismail, I.; Gaganis, V. Optimization of Well Placement in Carbon Capture and Storage (CCS): Bayesian Optimization Framework under Permutation Invariance. *Appl. Sci.* **2024**, *14*, 3528. [\[CrossRef\]](#)
11. Ismail, I.; Fotias, S.P.; Gaganis, V. A Holistic Framework for Optimizing CO₂ Storage: Reviewing Multidimensional Constraints and Application of Automated Hierarchical Spatiotemporal Discretization Algorithm. *Energies* **2025**, *18*, 5926. [\[CrossRef\]](#)
12. Watson, F.; Klemetsdal, Ø.; Lye, K.O. Multifidelity Monte Carlo Method for Rapid Uncertainty Quantification in CO₂ Storage Applications. *Comput. Geosci.* **2025**, *29*, 32. [\[CrossRef\]](#)
13. Michael, K.; Allinson, G.; Golab, A.; Sharma, S.; Shulakova, V. CO₂ Storage in Saline Aquifers II—Experience from Existing Storage Operations. *Energy Procedia* **2009**, *1*, 1973–1980. [\[CrossRef\]](#)
14. Cameron, D.A.; Durlofsky, L.J. Optimization of Well Placement, CO₂ Injection Rates, and Brine Cycling for Geological Carbon Sequestration. *Int. J. Greenh. Gas Control* **2012**, *10*, 100–112. [\[CrossRef\]](#)
15. Babaei, M.; Pan, I.; Korre, A.; Shi, J.-Q.; Govindan, R.; Durucan, S.; Quinn, M. CO₂ Storage Well Rate Optimisation in the Forties Sandstone of the Forties and Nelson Reservoirs Using Evolutionary Algorithms and Upscaled Geological Models. *Int. J. Greenh. Gas Control* **2016**, *50*, 1–13. [\[CrossRef\]](#)
16. Jin, Z.L.; Durlofsky, L.J. Reduced-Order Modeling of CO₂ Storage Operations. *Int. J. Greenh. Gas Control* **2018**, *68*, 49–67. [\[CrossRef\]](#)
17. Bahrami, P.; Sahari Moghaddam, F.; James, L.A. A Review of Proxy Modeling Highlighting Applications for Reservoir Engineering. *Energies* **2022**, *15*, 5247. [\[CrossRef\]](#)
18. Shahkarami, A.; Mohaghegh, S.; Gholami, V.; Haghighat, A.; Moreno, D. Modeling Pressure and Saturation Distribution in a CO₂ Storage Project Using a Surrogate Reservoir Model (SRM). *Greenh. Gases Sci. Technol.* **2014**, *4*, 289–315. [\[CrossRef\]](#)
19. Amini, S.; Mohaghegh, S.D.; Gaskari, R.; Bromhal, G.S. Pattern Recognition and Data-Driven Analytics for Fast and Accurate Replication of Complex Numerical Reservoir Models at the Grid Block Level. In Proceedings of the SPE Intelligent Energy Conference & Exhibition, Utrecht, The Netherlands, 1–3 April 2014. [\[CrossRef\]](#)
20. Vida, G.; Shahab, M.D.; Mohammad, M. Smart Proxy Modeling of SACROC CO₂-EOR. *Fluids* **2019**, *4*, 85. [\[CrossRef\]](#)
21. Wen, G.; Hay, C.; Benson, S.M. CCSNet: A Deep Learning Modeling Suite for CO₂ Storage. *Adv. Water Resour.* **2021**, *155*, 104009. [\[CrossRef\]](#)
22. Li, Z.; Kovachki, N.; Azizzadenesheli, K.; Liu, B.; Bhattacharya, K.; Stuart, A.; Anandkumar, A. Fourier Neural Operator for Parametric Partial Differential Equations. *arXiv* **2020**, arXiv:2010.08895.
23. Wen, G.; Li, Z.; Azizzadenesheli, K.; Anandkumar, A.; Benson, S.M. U-FNO—An Enhanced Fourier Neural Operator-Based Deep-Learning Model for Multiphase Flow. *Adv. Water Resour.* **2022**, *163*, 104180. [\[CrossRef\]](#)
24. Wen, G.; Li, Z.; Long, Q.; Azizzadenesheli, K.; Anandkumar, A.; Benson, S.M. Real-Time High-Resolution CO₂ Geological Storage Prediction Using Nested Fourier Neural Operators. *Energy Environ. Sci.* **2023**, *16*, 1732–1741. [\[CrossRef\]](#)
25. Tang, H.; Kong, Q.; Morris, J.P. Multi-Fidelity Fourier Neural Operator for Fast Modeling of Large-Scale Geological Carbon Storage. *J. Hydrol.* **2024**, *629*, 130641. [\[CrossRef\]](#)
26. Wu, H.; Lubbers, N.; Viswanathan, H.S.; Pollyea, R.M. A Multi-Dimensional Parametric Study of Variability in Multi-Phase Flow Dynamics during Geologic CO₂ Sequestration Accelerated with Machine Learning. *Appl. Energy* **2021**, *287*, 116580. [\[CrossRef\]](#)
27. Tang, M.; Ju, X.; Durlofsky, L.J. Deep-Learning-Based Coupled Flow-Geomechanics Surrogate Model for CO₂ Sequestration. *Int. J. Greenh. Gas Control* **2022**, *118*, 103692. [\[CrossRef\]](#)
28. Han, Y.; Hamon, F.P.; Jiang, S.; Durlofsky, L.J. Surrogate Model for Geological CO₂ Storage and Its Use in Hierarchical MCMC History Matching. *Adv. Water Resour.* **2024**, *187*, 104678. [\[CrossRef\]](#)
29. Li, B.; Bhark, E.W.; Gross, S.; Billiter, T.C.; Dehghani, K. Best Practices of Assisted History Matching Using Design of Experiments. *SPE J.* **2019**, *24*, 1435–1451. [\[CrossRef\]](#)
30. Diplas, D.; Fotias, S.P.; Ismail, I.; Bellas, S.; Gaganis, V. A Fully Automated Design of Experiments-Based Method for Rapidly Screening Near-Optimal CO₂ Injection Strategies. *Energies* **2026**, *19*, 1361. [\[CrossRef\]](#)
31. Chen, J.; Gildin, E.; Kompantsev, G. Optimization of Pressure Management Strategies for Geological CO₂ Storage Using Surrogate Model-Based Reinforcement Learning. *Int. J. Greenh. Gas Control* **2024**, *138*, 104262. [\[CrossRef\]](#)
32. Kanakaki, E.M.; Ismail, I.; Gaganis, V. Accelerating Numerical Simulations of CO₂ Geological Storage in Deep Saline Aquifers via Machine-Learning-Driven Grid Block Classification. *Processes* **2024**, *12*, 2447. [\[CrossRef\]](#)

33. Kanakaki, E.M.; Fotias, S.P.; Gaganis, V. Application of an Automated Machine Learning-Driven Grid Block Classification Framework to a Realistic Deep Saline Aquifer Model for Accelerating Numerical Simulations of CO₂ Geological Storage. *Processes* **2025**, *13*, 2658. [[CrossRef](#)]
34. Brier, G.W. VERIFICATION OF FORECASTS EXPRESSED IN TERMS OF PROBABILITY. *Mon. Weather. Rev.* **1950**, *78*, 1–3. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.