Visual Ensemble Analysis of Fluid Flow in Porous Media across Simulation Codes and Experiment

Ruben Bauer[1*], Quynh Quang Ngo[1], Guido Reina[1], Steffen Frey[3], Bernd Flemisch[2], Helwig Hauser[4], Thomas Ertl[1] and Michael Sedlmair[1]

[1*]VISUS, University of Stuttgart, Germany.
[2]IWS, University of Stuttgart, Germany.
[3] University of Groningen, the Netherlands.
[4] University of Bergen, Norway.

*Corresponding author(s). E-mail(s): ruben.bauer@visus.uni-stuttgart.de;
Contributing authors: quynh.ngo@visus.uni-stuttgart.de; guido.reina@visus.uni-stuttgart.de; s.d.frey@rug.nl; bernd.flemisch@iws.uni-stuttgart.de; thomas.ertl@vis.uni-stuttgart.de; michael.sedlmair@visus.uni-stuttgart.de; helwig.hauser@uib.no;

Abstract

We study the question of how visual analysis can support the comparison of spatio-temporal ensemble data of liquid and gas flow in porous media. To this end, we focus on a case study, in which nine different research groups concurrently simulated the process of injecting CO_2 into the subsurface. We explore different data aggregation and interactive visualization approaches to compare and analyze these nine simulations. In terms of data aggregation, one key component is the choice of similarity metrics that define the relationship between different simulations. We test different metrics and find that using the machine-learning model "S4" (tailored to the present study) as metric provides the best visualization results. Based on that, we propose different visualization methods. For overviewing the data, we use dimensionality reduction methods that allow us to plot and compare the different simulations in a scatterplot. To show details about the spatio-temporal data of each individual simulation, we employ a space-time cube volume rendering. All views support linking and brushing interaction to allow users to select and highlight subsets of the data simultaneously across multiple views. We use the resulting interactive, multi-view visual analysis tool to explore the nine simulations and also to compare them to data from experimental setups. Our main findings include new insights into ranking of simulation results with respect to experimental data, and the development of gravity fingers in simulations.

Keywords: Porous media, fluid flow, visual analytics, benchmark study, simulation ensemble

1 Introduction

Injecting CO_2 into subsurface reservoirs might be a key approach in the future to mitigate climate change [1–3]. Toward this approach, however, it is fundamental to gain a better understanding of fluid flow and transport in porous media, an area which has attracted substantial attention in many research fields [4–6]. Concerning geological carbon storage, large experimental efforts have been undertaken at potential storage sites to collect information about, for example, the geology and formation fluids as well as their respective dynamics [7, 8]. These efforts are costly though and often limited to very specific conditions. To overcome these problems, simulation studies such as [9] have become popular, taking advantage of the increasing computational capabilities. To validate the respective simulation models, they should be compared to existing experimental data. In the absence of such ground truth experiment data, however, the validation necessitates the careful exploration and analysis of simulations with different settings to capture all potential phenomenal patterns of fluid flow in porous media.

Interactive visualizations are widely used in the data visualization domain and have shown to be beneficial for exploring and analyzing complex data [10]. The use of such visualizations allows employing techniques like linking and brushing to connect multiple heterogeneous views [11], and concepts like Shneiderman's visual information seeking mantra: overview first – zoom and filter – details on demand [12]. Both are specifically useful for exploring and analyzing complex and or multidimensional data where it is impossible to visualize all important aspects of the data at once or when the data contains multiple (possibly dynamic) facets whose understanding is difficult in static, non-interactive views.

The main goal of our work is to explore how interactive visualization can support exploring, comparing, and analyzing different simulations in the domain of flow in porous media. To this end, we focus on a benchmark study of geological storage of CO_2 in the subsurface [13, 14]. In a larger project consortium, nine different research groups around the globe were tasked with simulating this process. The result of each individual simulation is spatio-temporal data (2D+time) that predicts the behavior of CO_2 flow starting from a joint, pre-defined condition. More precisely, the output of each simulation constitutes 2D images containing saturation and concentration of CO_2 in each cell of a uniform Cartesian grid discretizing the 2D spatial domain, recorded in ten-minute intervals. This spatio-temporal data is complemented by additional measurables such as the pressure at a specified location or the CO_2 mass integrated over a specific region, each providing a separate scalar time series.

After the individual simulations were run, the challenge is now to explore the resulting ensemble of simulations to compare similarities and differences between them, and to set them into context to the underlying experiments that were conducted along with the simulations. This set of exploratory tasks leans itself toward a visual analysis approach in general [15], and ensemble visualization in particular [16, 17]. In an interdisciplinary team of visualization experts and a porous media domain expert, we set out to better understand how ensemble visualization can benefit this area, and how respective visualizations should be designed. We followed a joint design study process [18] and explored different data aggregation and visualization approaches for the data at hand which we combine into one visual analysis tool.

The basis behind most ensemble visualizations is a quantitative similarity metric that allows to relate different ensemble members to each other and to visually compare them in the same space [19]. The choice of metrics largely depends on the domain application though, and so far no universal similarity metrics exist for fluid flow data in porous media. To address this issue, we leverage a variety of metrics to define the similarity between ensemble members, including a machine-learning-based approach, which we adapt for the domain problem at hand. We then leverage interactive data visualization that allows to see and analyze the data from these different angles.

The similarity metrics can then be used in an overview visualization. To this end, we split simulation results into different timeslots (patches) and project them as time curves [20] into a 2D scatterplot. This visualization allows us to find similarities between different simulations at different times. We additionally embed the experimental data into the same space, which allows us to put the nine different simulations into a global context. We extend this overview visualization with different detail visualizations. We use a space-time cube volume rendering [21] to present the full spatio-temporal simulation results of each ensemble member. Another juxtaposed view is used to display the respective scalar time series, and interaction allows to dive deeper into specific questions.

Based on Shneiderman's visual information seeking mantra, we propose a three-step workflow for the analysis with the resulting interactive visualization tool: First, analyze the overview visualization and search for clusters, outliers, and patterns in the projection. Second, explore the spatial maps guided by the findings from the overview. Third, leverage all linked views to achieve an in-depth understanding of the relationship between the projected similarities, spatial dynamics, and measurables across the ensemble. Following this workflow, we discovered several novel findings that suggest further investigations for domain scientists, such as comparisons of the length, shape, and development behavior of gravity fingers and interesting quantifications of similarities between simulation results and experimental data.

In summary, we make the following contributions:

1. We propose a visual comparative analysis approach utilizing a variety of similarity metrics for ensemble data of simulating fluid flow in porous media.
2. Using our approach, we explore data from a benchmark study, revealing new insights about the underlying domain, including various examples showing the benefits of our interactive visual analysis, e.g., ranking simulations with respect to experimental data, and different types of development of gravity fingers.

2 Background and Related Work

In this section, we first provide some background on the simulation benchmark study, including the modelled CO_2 injection process, and a brief summary of its data generation process. We then discuss various examples of how related work has dealt with the visual analysis of similar simulation ensembles.

2.1 Benchmark Study

We focus on analyzing simulation ensemble data of a recent benchmark study by Nordbotten et al. [13]. The benchmark study concerns itself with the injection of CO_2 into subsurface reservoirs. Subsurface reservoirs are geological structures below ground that are suitable for long-term storage of fluids. They usually consist of layers with different porosity and permeability such as a coarse-grained highly permeable region which is suited for storing a large amount of fluid, and a fine-grained low permeable caprock above which prevents the stored fluid from escaping. In these naturally occurring subsurface reservoirs, possibly large amounts of CO_2 can be injected and captured first below the caprock. Over time, more and more of the CO_2 dissolves into the formation water and the CO_2-saturated water sinks downward, increasing long-term storage security [1–3]. This process of convective mixing is driven by the density difference of the original formation water and the CO_2-saturated water and usually manifests itself in the form of so-called “gravity fingers” [13].

For such large-scale real-world scenarios, it is important to assess potential risks and opportunities by trying to model and forecast them [9, 22]. Even with a good understanding of the complex physical processes during and after injection of CO_2 into porous media, the lack of knowledge about the precise conditions in the subsurface introduces many uncertainties. As experiments are costly and field-scale real-time measurements are prohibitive due to the targeted long time spans, this problem is often addressed by uncertainty quantification approaches which require running many forward simulations that cover different conditions [23, 24].

The primary objective of the benchmark study was to "provide a full-physics validation of the state-of-the-art simulation capabilities" [13] for such CO_2 injection processes. With the help of an experimental rig for repeated multiphase 2D flow experiments[1], a laboratory-scale CO_2 injection experiment was conducted which served as reference for the benchmark study [25]. In the beginning of the study, the most important boundary conditions, like geometry, operation process of the injection, and other physical parameters of the experimental rig, were specified and provided to nine expert simulation groups. These groups were then asked to model, simulate, and forecast the actual experiments that were performed with the rig, but without having access to the experimental data. The experiments were run by an independent experiment group and serve as ground truth for the analysis and validation of the simulation data.

Appropriate analysis and comparison methods are required that allow to inspect and compare different simulations with each other and with the experimental data. This is a non-trivial problem, especially for spatio-temporal and multivariate data. We address this problem with our visual approach which is designed to support the analysis of such ensemble data.

2.2 Related Work

From the data types and design methodology perspective, we review in this section related work about visual comparative analysis of ensemble spatio-temporal data. Particularly, we focus on visual analysis methods and similarity metrics.

2.2.1 Visual Analysis for Ensemble Spatiotemporal Data

We refer here to a survey by Obermaier and Joy [26] which categorized existing ensemble visualizations into "feature-based visualization", and "location-based visualization". Our visual approach supports analysis components that fall into both categories. A recent survey of Wang et al. [27] categorizes ensemble visualization from two perspectives: the proposed visualization techniques, and the involved analytic tasks. Technique-wise, we consider multivariate data and mainly address linked juxtaposed views including two views for direct volume rendering. Regarding tasks, we address most of the mentioned tasks directly, except "clustering" and "parameter analysis".

Existing work also provides different design applications for a variety of ensemble spatio-temporal data types and respective domain application tasks. Demir et al. [28] presented a chart-based approach showing statistical properties of 3D volume ensemble members to support comparative analysis. Höllt et al. [29] contributed a visual analysis tool for reservoir simulation

[1] The FluidFlower Concept: Operating Flow Rigs https://fluidflower.w.uib.no/large-scale/

ensembles, utilizing statistical measures. Potter et al. [30] built a visualization framework focusing on statistical measures of simulation ensemble data. Bach et al. [21] provided a descriptive framework for temporal data visualizations based on generalized space-time cubes. Fofonov and Linsen [31] focused their work on multi-run physical simulation data, analyzing the impact of initial conditions and parameter settings on simulation results. Our ensemble consists of only few members with a variety of different parameter settings which makes it unsuitable for a quantitative parameter space analysis. We focus on an interactive visual comparison analysis of the ensemble members using variety of similarity metrics instead of statistical properties of ensemble data.

2.2.2 Similarity Metrics for Ensemble Spatiotemporal Data

Visual parameter space/ensemble analysis [16] normally concerns how parameter configurations influence the outcome of a simulation system by comparing ensemble members. One key challenge in spatio-temporal ensemble analysis is finding a suitable similarity/distance metric for an assisted or automated ensemble member comparison. While a manually performed visual comparison of spatio-temporal data can be employed for pattern recognition and comparison of complex time series in the details, it is not suitable for large data sets such as simulation ensembles. A similarity metric could support providing an abstraction overview of the ensemble in form of a scatterplot. The overview is the context that enables us to efficiently and automatically compare, filter, rank, and cluster ensemble members before performing a time-intensive manual analysis and comparison of-and-between individual members. In our visual approach, we utilize various similarity metrics, including an unsupervised machine-learning-based approach.

Tkachev et al. [32] presented S4—a machine learning (ML)-driven similarity metric, which is based on the assumption that spatial proximity implies similar behavior. We utilize for our use-case a tailored version of their approach to learn a similarity metric on the provided simulation data. We configured this version to use a different patch size and network size that is suitable for our data. In a recent work, Huesmann and Linsen proposed SimilarityNet [33], which is a ML model trained in an autoencoder-like fashion on a generated phantom dataset to learn to encode arbitrary spatio-temporal data into a 1D space representation that preserves spatio-temporal behavior. Due to the limitation of only producing 1D embeddings, we exclude this approach from our design, as we use 2D overviews for other metrics.

3 Problem Characterization

We characterize our problem by providing the description of the data, our research questions, and the respective analysis tasks.

3.1 Data

We consider an ensemble of nine different simulations of CO_2 injection processes into porous structures from nine research groups that participated in the benchmark study, labeled austin, csiro, delft-DARSim, delft-DARTS, lanl, heriot-watt, melbourne, stanford, and stuttgart. From each of the reported simulation data, we consider a series of 2D spatial maps, which represent the recorded CO_2 saturation and concentration values for the first 24 hours in ten-minute intervals. A saturation value is the ratio of the volume occupied by the gaseous phase to the available void space for each considered reporting cell, while a concentration value indicates the mass of dissolved CO_2 per volume of liquid phase in that cell.

At each time step, additional measurables such as local pressure values from virtual sensors and integrated quantities for three different regions of interest are reported, which we consider as scalar time series information. The three regions of interest correspond to predefined rectangular regions in the benchmark study: Box A, Box B, and Box C, see Figure 1. They are defined to capture and express specific features and events during the simulations and experiments. In particular, the research groups were requested to provide the total mass of CO_2 inside the domain, pressure at two locations, phase composition in Boxes A and B, as well as convection in Box C.

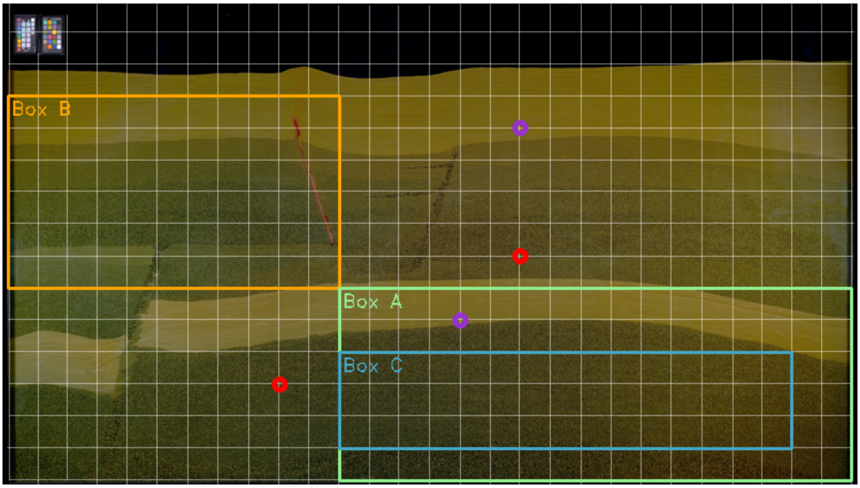


Fig. 1 Photograph image of the benchmark geometry with overlaid laser grid [13, Figure 8]. The red circles indicate the injection points, while the purple circles depict the pressure sensors. Boxes A-C correspond to regions for the evaluation of different system response quantities.

Besides the simulation groups, there was also one experiment group which performed the actual reference experiments with the setup mentioned in Section 2.1. We have access to the segmentation data of four experiment runs which we integrated in our visual approach. In contrast to the simulation data, which provides saturation and concentration values at each grid cell, the experimental data only provides ternary information about whether there is (i) only pure water, (ii) water with dissolved CO_2 or (iii) also gaseous CO_2 in a cell, due to the difficult process of post-processing the experimental data. In the post-processing, saturation and concentration values have to be derived from photographs of the experiment by analyzing the CO_2-induced coloring of the water. The time series data was not available to us for the experimental runs.

3.2 Research Questions and Tasks

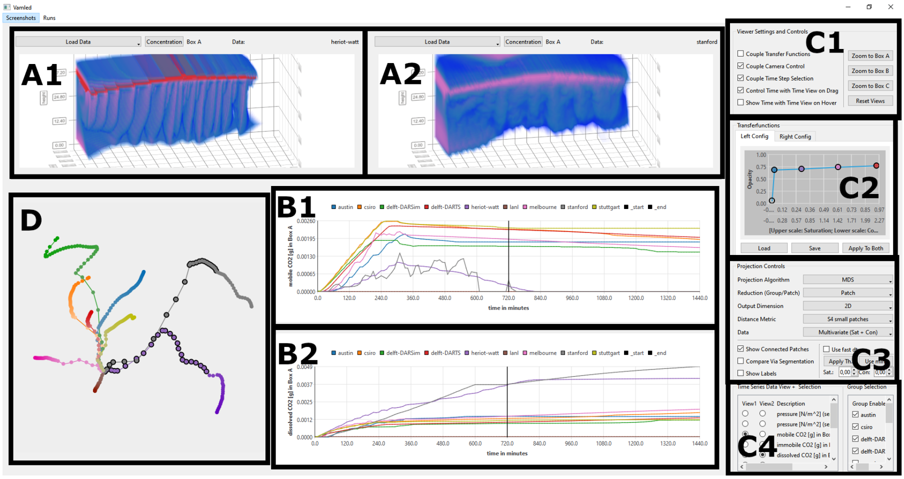
We work alongside a domain expert who has been working with flow and transport processes in porous media for 15 years. We had a series of regular bi-weekly meetings in which we jointly derived a set of research questions that should be possible to analyze with the target visualization tool.

- $\mathbf{Q_1}$: How similar are simulation outcomes across research groups and what are the differences?
- $\mathbf{Q_2}$: How well do the additional measurables capture the dynamics of the spatial maps in the regions of interest?
- $\mathbf{Q_3}$: Concerning features of particular interest: When do the “gravity fingers” (described in subsection 2.1) reach a certain length? When is the spill point reached? (Here, the spill point is the location where CO_2 “spills” over the edge of the modelled reservoir (Figure 1, Box A) after reaching its maximum capacity of mobile free CO_2 gas.)
- $\mathbf{Q_4}$: Which groups’ simulation outcomes match most closely the available experimental data?

We derive a list of tasks for the analysis of fluid flow on an ensemble of porous media simulations.
Comparative analysis of ensemble members: ($\mathbf{T_1}$)-compare simulations of different groups on a certain level of abstraction, to answer **Q1**. ($\mathbf{T_2}$)-find correlations between the different measurables and the dynamics of the spatial maps in the regions of interest, to answer **Q2**. **Spatial detail tasks:** ($\mathbf{T_3}$)-find areas having an interesting CO_2 concentration/saturation. **Temporal detail tasks:** ($\mathbf{T_4}$)-find out when certain events happen. Both ($\mathbf{T_3}$) and ($\mathbf{T_4}$) are to address **Q3**. ($\mathbf{T_5}$)-rank simulations with regard to the experimental results, to answer **Q4**.

4 Visual Analysis Approach

In our design process, we considered the requirements and derived tasks from Section 3.2 as well as the provided data as described in Section 3.1. As such, our visual analysis approach will follow the visual information seeking mantra [12]. Below, we first introduce the data processing employed for our visual approach. Second, we discuss the similarity metrics that we use to compare simulation outcomes. Third, we describe the proposed `data`



Data: heriot-watt

Data: stanford

Viewer Settings and Controls

☐ Couple Transfer Functions
☑ Couple Camera Control
☑ Couple Time Step Selection
☑ Control Time with Time View on Drag
☐ Show Time with Time View on Hover

Projection Controls

Projection Algorithm: MDS
Reduction (Group/Patch): Patch
Output Dimension: 2D
Distance Metric: S4 small patches
Data: Multivariate (Sat + Con)

☑ Show Connected Patches
☐ Compare Via Segmentation
☐ Show Labels
☐ Use fast d...
Sat.: 0,00 Con.: 0,00

View1	View2	Description
○	○	pressure [N/m^2] (se
○	○	pressure [N/m^2] (se
●	○	mobile CO2 [g] in Bo:
○	○	immobile CO2 [g] in I
○	●	dissolved CO2 [g] in E

Group Enable
☑ austin
☑ csiro
☑ delft-DAR
☑ delft-DAR

Fig. 2 Overview of our user interface. (**A1** & **A2**): Space-time cube renderings with dedicated selected group and visualized variable. (**B1** & **B2**): Time series plots of additional measurables. (**C1**): Viewer settings and interaction controls. (**C2**): Transfer function to map saturation and concentration to color and opacity. (**C3**): Controls for selecting projection algorithm, metric, and data. (**C4**): Controls for selecting the groups and features displayed in the line charts. (**D**): Projection of the ensemble.

`representation`, `interaction`, and `controls` components of our visual analysis tool. Fourth, we provide an example workflow for a visual analysis using our visual approach.

4.1 Data Processing

The data processing happens at two stages: the pre-processing required for our visual approach, and the extraction of so-called patches.

4.1.1 Pre-Processing

To make it easier to observe spatio-temporal patterns, we propose to use a static view of the data from each group, namely space-time cube visualization [34]. The spatial maps then have to be densely packed and transformed to a volume (a 3D texture) in which the uniform grid of the spatial maps as well as their time components map to indices of the resulting volume.

For a proper visual comparison, the resulting volume should be of equal size and cover the same geometric and temporal extents for all ensemble members. However, the amount of time steps and the geometric extents of the spatial maps between different groups in the first 24 h varies slightly. To align the data in the volumes, we first fill in missing time steps by repeating the data of the preceding time step, if available. If there is no previous time step, we fill the time step with zeroes. We set the target geometric extents of the volumes to the extents as described by the benchmark description. The width and height of the resulting volumes represent the benchmark geometry from $x = 0.005\,\mathrm{m}$

to $x = 2.855\,\mathrm{m}$, and from $y = 0.005\,\mathrm{m}$ to $y = 1.225\,\mathrm{m}$ with a step size of 0.01m in x- and y direction. The depth of the volume represents the time from $t = 0\,\mathrm{s}$ to $t = 8640\,\mathrm{s}$ (=24 h) with a step size of 600 s (=10 min).

We perform the same procedure to fill missing values in the time series as we do for the spatial maps. The resulting time series span the full range from $t = 0\,\mathrm{h}$ to $t = 24\,\mathrm{h}$ with one measurement every ten minutes.

4.1.2 Processing of Patches

We utilize the S4 [32] ML model as a similarity metric for spatio-temporal behavior (see Section 4.2). The S4 model trains on so-called patches of the data. A patch is a subset in the original spatio-temporal data. Considering the spatial maps of the benchmark study as a 3D volume with x, y, and t dimension, a patch in this 3D volume context could be any 3D cuboid in it. We empirically chose the temporal patch size (dimension t) to be always three, which equals 30 minutes in our data. This size allows the model to integrate the temporal component during the computation of latent space features without introducing too much change between two consecutive non-overlapping patches. We utilize two versions of the S4 model with each a different spatial patch size, which we further specify in Section 4.2.

4.2 Similarity Metrics

In the visualization community, similarity metrics are applied to evaluate how similar data items are. The similarity information is often fed to a dimensionality reduction (DR) [35–37] that provides a 2D mapping of the data items. This mapping's outcome is then represented in a scatterplot to serve as an overview of the dataset in a visual analysis pipeline.

In this work, multiple similarity metrics are utilized to capture more information with respect to potential features of the simulation outcomes ($\mathbf{T_1}$). Each metric computes the similarity between two patches with respect to their multivariate facets in general. By using similarity metrics to automatically compute a similarity value between simulations of different groups, we can compare them in an abstract manner without inspecting the spatio-temporal data manually, but to view it in a projection instead. The projection then provides hints for a more detailed, manual analysis of individual patches and groups.

Euclidean Distance and Manhattan Distance

Euclidean distance and Manhattan distance are two of the most popular but simple metrics to compute distances between two points (vectors) P and Q in a multidimensional space $\mathbb{R}^n$. The Euclidean distance is defined as $D(P,Q) = \sqrt{\sum_{i=0}^{n-1} (P_i - Q_i)^2}$. The Manhattan distance is defined as $D(P,Q) = \sum_{i=0}^{n-1} \|P_i - Q_i\|$.

In our case, to compute the distance between two patches, we first flatten both patches (in $\mathbb{R}^{W \times H \times T}$) to each a feature vector with N elements (in $\mathbb{R}^N$ with $N = W * H * T$) by rearranging the dimensions, where W, H are the width and the height of a patch respectively. The Euclidean or Manhattan distance between both patches is then computed by computing the distance between their feature vectors. We consider "feature vectors" as vectors consisting of ordered numerical properties, and which usually serve as input for models to be further processed. While the above metrics are often used, they are not exactly suited to compare spatio-temporal data with patterns that might change in size, (spatio-temporal) position, or orientation, which physical phenomena often exhibit. If two patches capture the exact same pattern, a small change in any of those properties may result in a large distance between these two patches, since the indices of the corresponding elements in the feature vector that capture the patterns might change completely. However, we will show that Euclidean distance still yields reasonable results on the benchmark study ensemble (Section 5.1).

ML Model for Comparing Spatiotemporal Behavior

S4 is a ML model for "Self-Supervised Learning of Spatiotemporal Similarity", which was recently proposed by Tkachev et al. [32]. We expect the S4 model to be a more advanced and better-suited metric for comparing data by its spatio-temporal behavior than conventional metrics like the Euclidean distance or Manhattan distance. The model mainly consists of an encoder part that has to be trained first, before it can be applied to the data. It is trained on patches of the data and learns to encode them into a latent-space feature representation in which two vectors are close by Manhattan distance if their corresponding patches have similar spatio-temporal behavior. The training exploits the assumption that two patches that are close in space and time are also close in terms of their spatio-temporal behavior and vice versa, and thus can be applied to unlabeled data. Specifically, during training, random positive examples (pairs of patches with close spatio-temporal proximity) and random negative examples (pairs of patches from different ensemble members) are drawn from the ensemble and fed to the model, optimizing it to predict whether the provided pair of patches is a positive or negative example.

We trained the model on a spatially downsampled volume by a factor of two. The patch size is chosen to be the remaining full spatial size and a temporal size of three time steps. During inference, we compute the non-overlapping patches of each group and use S4 to compute a 64 feature vector for each patch. The S4 distance between two patches is then the Manhattan distance between their corresponding feature vectors. Our results will show that choosing the full spatial size as patch size yields too few data points and variations for the number of trainable parameters which results in strong overfitting; therefore running short on capturing similarity across the ensemble members. Thus, we trained another model with smaller patch size by spatially subdividing the patches, and reducing the amount of trainable parameters by slightly adjusting the models' hyperparameters to decrease the risk of strong overfitting.

To distinguish between both models in our analysis, we will refer to the first one as just "S4", and to the latter one as "S4 with subdivided patches". For the "S4 with subdivided patches", we chose the spatial patch size to roughly capture the average finger width and length of the simulations in the first 24 h. We believe that this allows the model to better capture local features, like the development of gravity fingers, and show that these changes improve capturing the similarity. During inference with "S4 with subdivided patches", to now compare two patches of full spatial size with each other, we first spatially subdivide these patches in non-overlapping sub-patches and compute the sum of the distances between sub-patches at the same positions instead.

Wasserstein Distance

The Wasserstein distance [38] is a popular measure assessing similarity and has also been used on flow simulation data [39, 40]. It measures the distance between two probability distributions, which in our case, is the distribution of concentration or saturation values in a single patch. To utilize the Wasserstein distance to compare two patches, we represent each patch by a probability distribution. We use the distributions of the variables in a single patch as its representation, losing all spatial information in contrast to the other metrics. Since each patch has two variables, saturation, and concentration, we then compute the differences between the respective distributions of the pair of patches. The distance between two patches would be either the Wasserstein distance between their saturation distributions or their concentration distributions or the average of both, depending on the selected variables of interest.

In our work, the distribution for either variable is computed as follows: First, we scale the variables to be in the range of $[0, 255]$ by using the global minimum and maximum value per variable across all groups, and round each value down to an integer, such that there are at most 256 different values per variable. Next, we compute the histogram per variable in each patch by counting how often a specific value occurs. We divide the histograms by the total amount of elements in the patch, which yields the probability distribution for each variable and each patch. We emphasize that this Wasserstein distance is different from the one employed in [14], where two-dimensional distributions over the spatial domain and the corresponding Euclidean metric are used.

4.3 Data Views

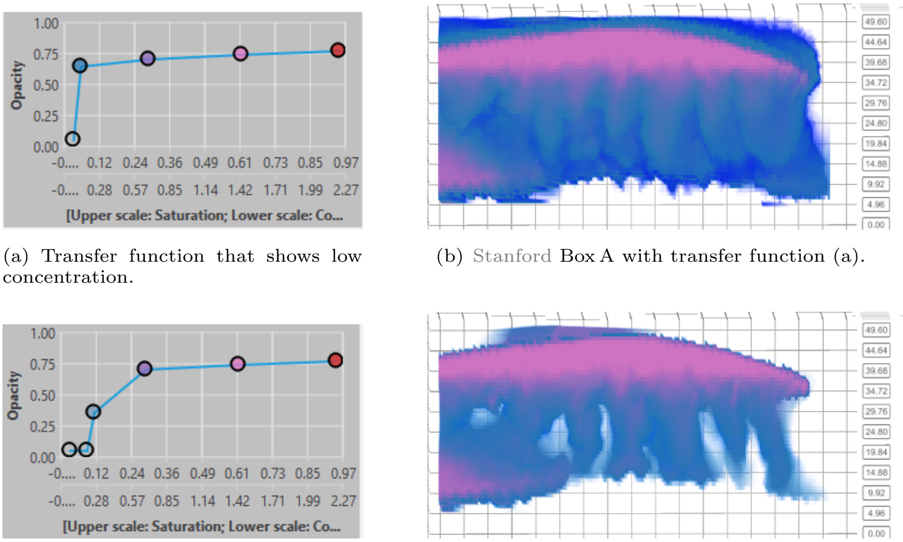
In this subsection, we describe the different views representing our proposed data visual encodings and respective interaction controls in showing how they relate to each other as well as to our defined tasks in Section 3.2. An overview of our implementation instance for all views is provided in Figure 2. Besides the similarity view, which provides an overview of the similarity of the full ensemble, we provide views to link the projected ensemble members or their patches to the actual data from spatial maps or time series information.

4.3.1 Similarity View

In the similarity view, we project the ensemble's similarity in a scatterplot to provide an overview of the full ensemble, using Dimensionality Reduction (DR) algorithms. The similarity metric information among patches or ensemble members is fed to a DR technique to derive the similarity view in form of a 2D scatterplot. Each point in the scatterplot represents either one patch (”patch” mode) or one ensemble member (”group” mode). The distances among the points visually reflect the similarity among the patches or ensemble members. Due to the randomness of DR algorithms's initial configuration, the same input may produce rotated or rearranged scatterplots if projected again. The axes of the scatterplot, therefore, have no meaning, and only the relative distances between the points matter. The similarity view in Figure 2 (**D**) provides hints as to which groups are outliers or if small clusters have formed, allowing the user to explore the ensemble ($\mathbf{T_1}$).

Furthermore, inspired by Machicao et al. [41], we include the experimental data in the projection, which allows us to visually rank the simulation groups by comparing their similarities to the experimental data ($\mathbf{T_5}$). As experimental data is only available in the form of segmentation maps, we have to transform the simulation data into segmentation maps to get them into the same format for comparison. The segmentation maps of the experimental runs were computed by choosing a certain threshold in the image analysis when analysing the original photographs of the experiment. This threshold relates to how much concentration or saturation exists in the image grid cell. Therefore, we also choose a threshold to transform the simulation data into segmentation maps before computing the similarity matrix with our selected similarity metric. We set the default threshold to consider a grid cell to contain CO_2 saturation or CO_2 concentration close to zero (0.001), to avoid the segmentation of actually empty cells due to numerical errors.

Like similarity metrics, DR algorithms may differ in the revealed features and quality of results [35–37]. We chose to integrate three popular DR algorithms that are often used for projecting data into 2D space. We figure multidimensional scaling (MDS) [37] to be one of the most important DR algorithms in our context, as it tries to preserve distances between data points in lower dimensional space. More specifically, provided a set of N points $P \subset \mathbb{R}^n$, MDS tries to find a lower dimensional embedding (or representation) of these points $Q \subset \mathbb{R}^m$ with usually $m < n$ and $q_i \in Q$ is the lower dimensional counterpart of $p_i \in P$ by minimizing a so-called stress function. The stress function S is defined as $S(q_1, ..., q_N) = \sqrt{\sum_{i \neq j=1,..,N} (d_{ij} - \|q_i - q_j\|)^2}$ with d_{ij} being the distance of points p_i and p_j in the original space $\mathbb{R}^n$. Besides MDS, we also integrate uniform manifold approximation and projection (UMAP) [36] and t-distributed stochastic neighbor embedding (t-SNE) [35]. These two DR techniques try to preserve the neighborhoods of data points.



(a) Transfer function that shows low concentration.

(b) Stanford Box A with transfer function (a).

(c) Transfer function without low concentration.

(d) Stanford Box A with transfer function (c).

Fig. 3 Stanford spatial maps renderings for Box A with two different transfer functions. By selecting a transfer function which omits low concentration values, the shape of the higher concentrated fingers becomes visible.

As mentioned above, we provide two types of similarity views in this work. Choosing the “group” mode aggregates the individual patch distances to project only one point per group. In this case, the distance $D(A, B)$ between two groups A, B where each is partitioned into N patches, is computed via $D(A, B) = \sum_{i=1}^{N} \text{metric}(a_i, b_i)$, with metric being the used similarity metric for the patches a_i and b_i from A and B respectively. Choosing the “patch” mode will compute pairwise distances between all patches from all groups and projects those to provide hints about the temporal similarity between the groups. Figure 4 shows MDS projections of the ensemble using the “group” option, while Figure 6 shows MDS projections of the ensemble using the “patch” option.

4.3.2 Space-Time Cube View

To support focusing on details for spatio-temporal patterns of each ensemble member, we propose to use a static view representing a series of 2D maps as a volume with time as the third dimension. The space-time cube rendering [21] renders the processed volume (Section 4.1.1) of either saturation or concentration information of data. The saturation or concentration values of the space-time cube are mapped to colors with respective luminance and opacity defined by a color transfer function, see Figure 2 (**A1** & **A2**). We argue that this representation of the data provides better insight than juxtaposing 2D spatial data representation images in a sequence as it smoothly maintains the temporal evolution pattern of the data.

To make it more domain-application friendly, we propose to allow the user to interactively change/define the transfer function while analyzing the data using the view. Particularly, the user can choose colors and opacity for any saturation or concentration values in our data by using the transfer function diagram, see Figure 2 (**C2**). Values that do not have an explicit value have the linearly interpolated color and opacity between their left and right point applied to them. By properly configuring the transfer function, the user can highlight or hide ranges of saturation and concentration values of the space-time cube volume. In Figure 3, we show two examples of a space-time cube rendering with different transfer functions that each show different concentration values. Besides the interactive transfer function, the view also provides fundamental interaction such as rotation, zoom-in, and zoom-out to make it easier for the user to analyze and study the data cube.

The space-time cube view alone can support the user to target tasks ($\mathbf{T_3}$) and ($\mathbf{T_4}$). Being static, the view supports comparing the spatial distribution of concentration/saturation of different time steps globally ($\mathbf{T_3}$). From the comparison, the user can look for the area of interest and can determine the time steps at which certain events happen ($\mathbf{T_4}$).

We propose to use two space-time cube views in our design to target ($\mathbf{T_1}$) and ($\mathbf{T_2}$), see Figure 2 (**A1** & **A2**). When the two views are used to represent concentration and saturation information of the same ensemble data, the user can analyze and study the correlation/relationship between the two types of information ($\mathbf{T_2}$). Together with the similarity view, if the two views are used to represent one type of information for two ensemble members, the user can comparatively analyze and study the two ensemble members in detail ($\mathbf{T_1}$).

4.3.3 Line Charts View

We use line charts to visualize the time series data of the given three regions of interest, i.e., regions Box A, Box B, and Box C. Each line chart is used to represent one measurable of one group over time. The line charts with different colors for different groups are superimposed in one view, see Figure 2 (**B1** & **B2**). This allows us to visually compare the development of a measurable among different groups over time.

We propose to contain two juxtaposed views in which each can visualize one of those line charts for one measurable at a time, see Figure 2 (**B1** & **B2**). This provides a convenient side-by-side comparison pattern over simulation groups of two different measurables of different regions. As a decluttering mean, we provide the user the ability to interactively select and deselect one or more groups and to choose the selected measurable of interest that are shown in the views, see Figure 2 (**C1**). To target ($\mathbf{T_2}$), we propose to link the saturation and concentration of the spatial maps that are visualized in the space-time cubes to the corresponding time series measurables mobile CO_2 and dissolved CO_2. The detail of the interaction operation will be presented in Section 4.3.4. By analyzing line charts of different regions of interest, the user can also identify events of interest and respective regions related to the provided measurables, i.e., targeting ($\mathbf{T_3}$) and ($\mathbf{T_4}$).

4.3.4 Interaction

Besides the interaction operations that we designed for each aforementioned view, we use brushing and linking to coordinate among views. The similarity view is linked to the space-time cube view and the time series view to support the top-down analysis approach ($\mathbf{T_1}$). The user can select two patches or two ensemble members of interest to be displayed in the two space-time cube views, e.g., Figure 2 (**A1** & **A2**), to compare them in detail. When the similarity displays the patches, hovering over the patches' representations will slice the space-time cube of the corresponding group to the selected patches. The similarity view is further linked to the time series view and vice versa. Hovering over a patch's representation in the similarity view will select the range of this patch in the line charts view, e.g., Figure 2 (**B1** & **B2**). Selecting a time range in the time series views will highlight the corresponding patches of the groups in the similarity view which are also currently visible in the space-time cube view.

By providing a convenient interaction to inspect the measurables and spatial maps at different time steps simultaneously, the interlinking of the three views effectively targets tasks ($\mathbf{T_2}$), ($\mathbf{T_3}$), and ($\mathbf{T_4}$). Linking between the time-cube view and the line charts view allows the user to analyze correlation between time series input and saturation and concentration information ($\mathbf{T_2}$). Meanwhile, linking the similarity view with the other two views allows the user to observe the overview pattern before focusing on details about some specific group, the specific time step, and the specific spatial region that constitutes the pattern, i.e., targeting ($\mathbf{T_3}$), and ($\mathbf{T_4}$).

4.4 Workflow

Based on the overview first – zoom and filter – details on demand mantra, we propose the following workflow with our visual analysis approach.

Step 1: Analyze the Similarity View

Regarding overview first, the user can take a look at the similarity view showing the "group" mode of the ensemble. The similarity view encodes the different groups by color. The user can identify clusters of some groups being closer to each other than others, or identify outliers.

If the user is interested in the temporal development of the ensemble members or how they diverge throughout the simulation, the "patch" mode can be enabled. Hovering over a patch of one group highlights the corresponding patches of the other group at the same time step (Figure 7(a)). By doing that, the user can see how fast and when two groups diverge.

If the user is interested in how well simulation data compares to the experimental results, the experimental data can be included in the projection results.

The user can further inspect the overview under the changing similarity metrics and DR techniques to see how the results change.

Step 2: Explore the Spatial Maps

From observing the similarity pattern of ensemble members in the similarity view with "group" mode, the user can compare the details pair of the members using space-time cube views. The space-time cube rendering shows the whole outer shape of the simulation at once and makes it easy to roughly estimate if the two simulations behave visually similar over time. Inspecting the space-time cube might verify if one outlier in the projection indeed has completely different behavior in the spatial maps than the others.

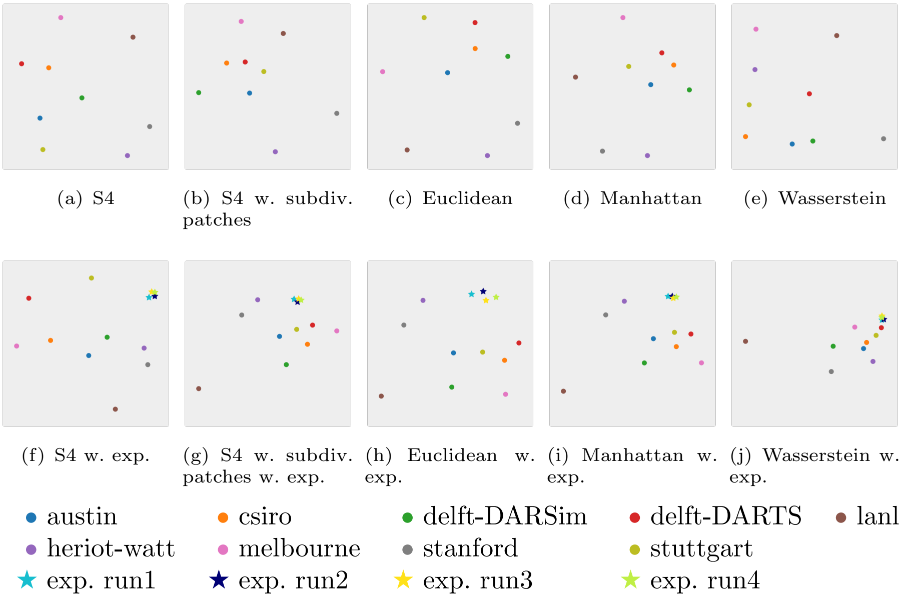
If the similarity view is in "patch" mode, the user can navigate the resulting projection via zooming to inspect early time steps that greatly overlap. By hovering over the patches, the user can compare the corresponding actual spatial maps in the space-time cube views. The time series view reveals the corresponding time steps of the hovered patches, allowing the user to see whether close patches are also similar in the given time series data.

Step 3: Analyze The Ensemble In Detail By Leveraging All Views

Having got a broad overview of the ensemble, the user can analyze the ensemble members in more detail. The user can investigate the difference between concentration and saturation data by comparing the respective two space-time cube views (Figures 9(a), 9(c)). Next, the user can inspect how the CO_2 concentration evolves over time in the simulations by interactively defining the transfer function. When the user is interested in the CO_2 concentration and wants to better inspect the finger development in the different boxes of the simulation geometry, they can zoom to the respective boxes by spatially slicing the volume to contain only the box of interest. After that the user can select the line chart that shows the corresponding dissolved CO_2 in our box of interest. The user can select the range where the line chart first indicates existing dissolved CO_2 in the box of interest up to when the amount of dissolved CO_2 does not seem to change anymore. This slices the space-time cube temporally to the selected range. With that, the user can now inspect the space-time cube and its early stages of finger development in the box of interest.

5 Results

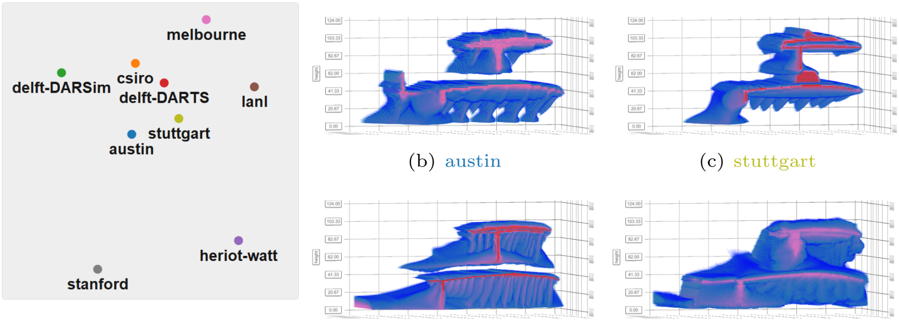
In this section, we provide answers to the aforementioned research questions in Section 3.2 as well as other findings by using our visual analysis approach. We first provide the results related to comparing the groups and finding similarities/differences by using our similarity metrics ($\mathbf{Q_1}$). Then, we present several



(a) S4 (b) S4 w. subdiv. patches (c) Euclidean (d) Manhattan (e) Wasserstein

(f) S4 w. exp. (g) S4 w. subdiv. patches w. exp. (h) Euclidean w. exp. (i) Manhattan w. exp. (j) Wasserstein w. exp.

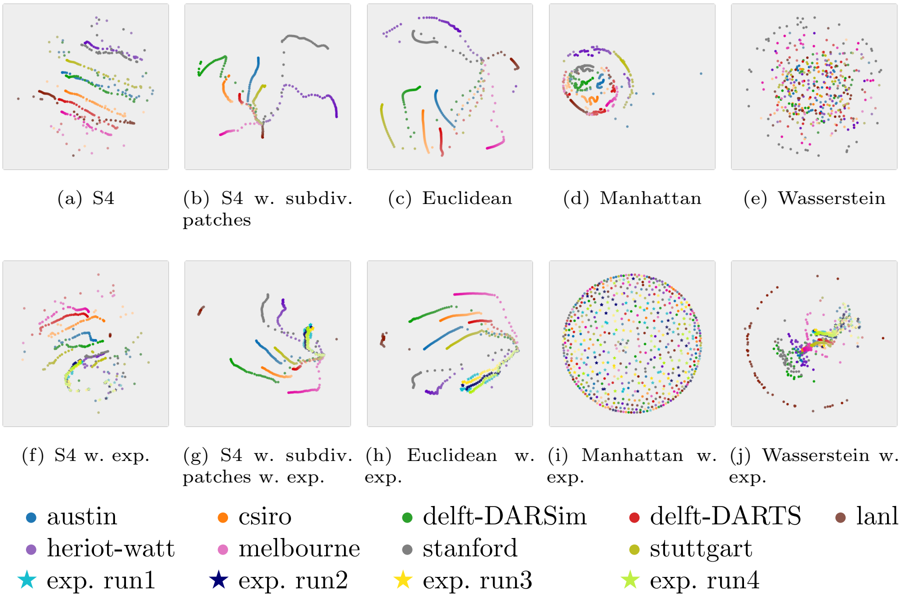
Fig. 4 (a)-(e): 2D plots of the ensemble simulation spatial maps for different similarity metrics with aggregated patches ("group" mode). (f)-(j): same as (a)-(e) but using the segmented spatial maps and including the experimental data in the projection. The experimental data is plotted as star shapes instead of circles. The segmentation threshold is 0.001.



(a) 2D projection via MDS and S4 with subdivided patches. (b) austin (c) stuttgart (d) heriot-watt (e) stanford

Fig. 5 A side-by-side comparison between the similarity view (using saturation and concentration data) (a) and selected spatial maps renderings (of just concentration data) (b)-(e). The projection suggests that (b) and (c) are close and that (d) and (e) are rather distant.

findings related to ($\mathbf{Q_2}$) and ($\mathbf{Q_3}$). Finally, we present the outcomes related to ranking with respect to experimental data ($\mathbf{Q_4}$).


(a) S4

(b) S4 w. subdiv. patches

(c) Euclidean

(d) Manhattan

(e) Wasserstein

(f) S4 w. exp.

(g) S4 w. subdiv. patches w. exp.

(h) Euclidean w. exp.

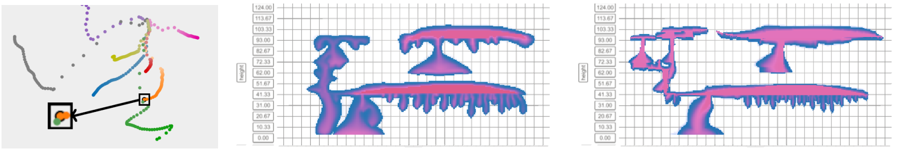
(i) Manhattan w. exp.

(j) Wasserstein w. exp.

Fig. 6 (a)-(e): 2D plots of the ensemble simulation spatial maps for different similarity metrics with all patches ("patch" mode). (f)-(j): same as (a)-(e) but using the segmented spatial maps and including the experimental data in the projection.

5.1 Comparison Across Research Groups via Different Similarity Metrics – Q1

Using different similarity metrics helps us to identify several differences in the simulation outcomes across different research groups. Figure 4 shows an overview of the ensemble with and without experimental data in "group" mode. First, we receive a striking observation that the experimental data representation appear close to each other in the overviews with every similarity metric (Figures 4(f)-4(j)). This outcome validates the correctness of the overviews. Looking closely at the overviews without embedding experimental data, e.g., Figures 4(a)-4(e), we also can see similar pattern deriving from the different metrics, e.g., two sites heriot-watt and stanford are formed in one group, lanl always stands alone, and the other site forms one group with the same spatial arrangement in each view except for Figure 4(e). We find that these patterns align with the manual visual comparison of the space-time cube renderings. In Figure 5, we show the projection using the "S4 with subdivided patches" side-by-side to the space-time-cube renderings of austin, stuttgart, stanford, and heriot-watt. The projection Figure 5(a) suggests that austin and stuttgart match closely and heriot-watt and stanford stand out from the rest. While this is only vaguely resembled by the static space-time-cube renderings Figure 5(b)-(e), these similarities and differences become more clear when comparing the simulations interactively. Nevertheless, this comparison shows that the used metric does not significantly take the feature "reaching the spill



(a) 2D projection of patches via MDS and S4 with subdiv. patches.

(b) Csiro patch rendering at 24 h.

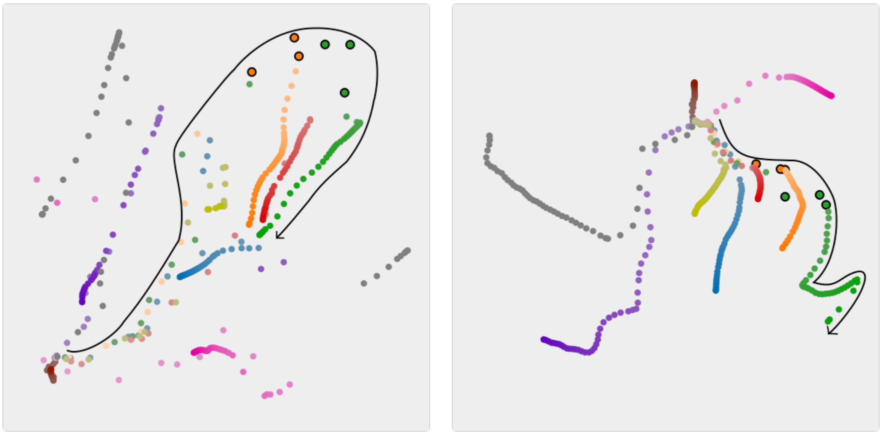
(c) Delft-DARSim patch rendering at 5 h.

Fig. 7 (a) shows the MDS projection of individual patches with enlarged cutout of the last csiro patch at 24 h and the delft-DARSim patch at 5 h. Both patches are close to each other in the projection which suggests that the respective spatial maps should also look similar, which we can verify by comparing (b) and (c). As both patches are from completely different time steps, this suggests that the simulation of delft-DARSim progresses faster than the simulation of csiro.

point" into account in comparison to the overall shape of the simulations, as e.g., austin has reached the spill point whereas stuttgart and the other two did not. Overall, the projections in Figure 4 show few differences among the different similarity metrics. Only the Wasserstein distance shows quite a different distribution of the groups in the projection.

Figure 6 shows the corresponding projections of the ensemble in "patch" mode. For most metrics, the projection of all patches results in prominent time-curves with the time curve arrangements representing a similar pattern to the aggregated counterparts of Figure 4. Though, the time curve representations differ more clearly for different similarity metrics. For example, the projections for S4 with small patches and Euclidean distance in Figure 6(b) and Figure 6(c), show a common starting point for all groups. This is also true when applying segmentation and including the experiment results (Figures 6(g)-6(h)). For the projection using Manhattan distance without the experimental data, we also find time curves, but this time with a less clear common starting point (Figure 6(d)). In contrast to the above, the projection using the S4 metric in Figure 6(a) differentiates the groups quite well, but it does not reveal any common starting point. We also find no clear time-curves or starting point for the Manhattan distance with experimental data (Figure 6(i)), as well as in both projections which use the Wasserstein distance (Figures 6(e) and 6(j)).

The time curve overview in Figures 6(b) and 6(c) reveal that the simulations progresses at different speeds. The time curves of the simulations all start close to each other and move away from the center over time. While some take big steps to extent far from the center like delft-DARSim and stanford, others like csiro and delft-DARTS stay rather close. For example, the projection in Figure 6(b) suggests that the last time step of csiro is one of the closest time steps to delft-DARSim. Looking at the space-time cube rendering of both patches side-by-side, we can visually verify that the last patch of csiro at 24 h is indeed quite similar to delft-DARSim at 5 h (Figure 7). After 5 h, delft-DARSim's simulation progresses and changes further which is also reflected in the simulation.



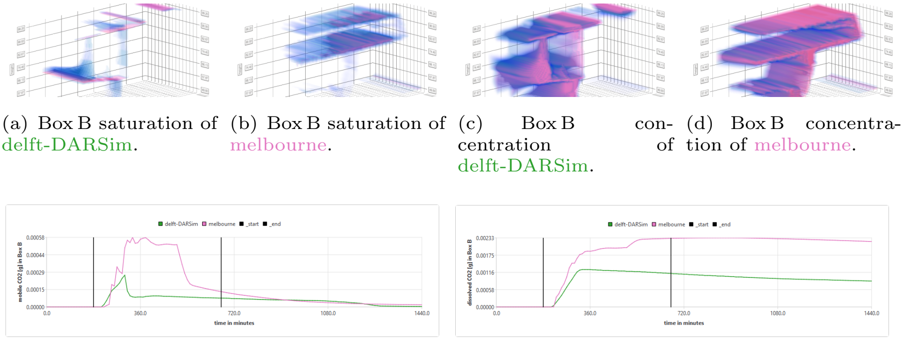
(a) MDS, S4 w. subdiv. patches, saturation. (b) MDS, S4 w. subdiv. patches, concentration.

Fig. 8 The MDS projection of the individual patches of the full simulation ensemble. We use saturation to encode for the time. For instance, the time direction of the group delft-DARSim follows the juxtaposed black arrow. The three patches around the injection stop (after 5 h) are highlighted in (a) and (b) with black halos. The projection in (a) uses only saturation, and in (b) only concentration data to compute distances. After injection stop, the time-curves in (a) make a turn back to the origin (the patches that belong to the first few time steps) because saturation keeps dissolving without replenishment, thus later time steps become more similar to earlier ones. In (b), they still diverge, though slower. Presumably due to the decrease of saturation and hence, a decrease in the rate of new concentration which reduces the change between time steps of the concentration spatial maps.

We empirically find that the projections which use the "S4 with subdivided patches" similarity metric match closest to the perceived differences in the spatial maps. The projections which are derived from Euclidean distance show similar patterns.

5.2 Correlation between the Dynamics of the Spatial Maps and the Time Series Data – Q2

Temporal Behavior of Saturation and Concentration Projections: Using saturation and concentration separately in similarity metric computation, we find significant differences in the temporal behavior of saturation compared to concentration. Figure 8 shows two MDS projections using the "S4 with subdivided patches" as similarity metric. The projection in Figure 8(a) with "patch" mode relates to the saturation data and Figure 8(b) shows the projection of the concentration data. We find that in both projections, the time curves have a common point of origin but behave increasingly different throughout the simulation. For the concentration data, the time curves keeps mostly moving away from the center, such that the latest patches are one of the out-most points in the projection. However, for the saturation data, they first briefly diverge, but then make a turn back in the direction of the origin. By interacting with the similarity view and the line charts views, we



(a) Box B saturation of delft-DARSim.

(b) Box B saturation of melbourne.

(c) Box B concentration of delft-DARSim.

(d) Box B concentration of melbourne.

(e) Time series of mobile CO_2 in Box B for delft-DARSim and melbourne.

(f) Time series of dissolved CO_2 in Box B for delft-DARSim and melbourne.

Fig. 9 The space-time cube visualizations of Box B for delft-DARSim and melbourne (a)-(d) match the trend of the corresponding mobile CO_2 and dissolved CO_2 time series (e)-(f). Subfigures (a) and (b) show the suddenly increasing saturation in Box B after reaching the spill point as well as the decreasing saturation intensity over time as more and more CO_2 dissolves, thus, concentration increases (see (c) and (d)).

find that this turnaround happens right after injection stop at $t = 5\,\mathrm{h}$ when the saturation, i.e., mobile CO_2, has reached its maximum in all of the boxes. Hovering over the saturation maximum data of any of the boxes highlights the most distant patches in the similarity view (Figure 8(a)). The mobile CO_2 only decreases after injection stop, which explains that the patches after this point become more similar again to the previous patches. We validate this by visually inspecting the space-time cube renderings for the saturation data.

Although the patches in the concentration plot keep diverging, they diverge much slower after the injection stop, which can be seen in Figure 8(b) where the patches around the injection stop are highlighted via black halos. This could be explained by the remaining physical processes being mainly dissolution with a decreasing rate due to a decreasing amount of remaining saturation. Thus, the differences between consecutive patches after injection decrease.

Correlation between Spatial Maps and Time Series Data: By zooming to the boxes of interest and interacting with the time series views and the space-time cube renderings, we can inspect the boxes of interest in more detail and detect when certain events happen. For example, the visual inspection of Box B allows to see when the spill point of Box A is reached. When Box A has reached its maximum capacity of CO_2 gas, injecting more CO_2 results the CO_2 gas to "spill" over the spill point and leak into Box B through a coarse-grained riff (Figure 1 to the left of Box A). If this happens, we can see increasing CO_2 saturation and concentration in the space-time cube rendering of Box B which should also correlate to increasing mobile CO_2 and dissolved CO_2 time series data in Box B. We find that this correlation between spatial maps and time series data matches well, which we show for groups delft-DARSim and melbourne in Figure 9. There, we notice that the first saturation in Box B is visible after 220 minutes for delft-DARSim and after 230 minutes for melbourne (not considering outlier lanl). Most other groups start showing CO_2 saturation at around 250 minutes in Box B.

While it would be possible to use the time series data itself for detecting when the spill point is reached, the visualization provides a suitable approach to validate the time series data and that the measured CO_2 in Box B is indeed due to leakage from Box A. Furthermore, this approach also provides means to quickly identify when the spill point is reached for the experiment runs for which no time series data is available to us. We identify the time of reaching the spill point to be after 250 min, 260 min, 270 min, 270 min for the experiment runs 5, 2, 1, and 3, respectively.

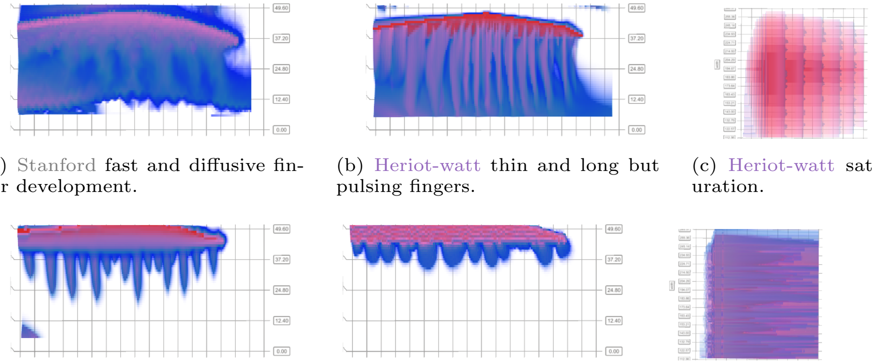
The results demonstrate how interacting with various views helps us understand important aspects of the data. The overview offers hints to the spatial map dynamics, while the space-time cube and time series views can be used to confirm those and link to specific events in time.

5.3 Shape and Development of Fingers Differ Throughout the Ensemble – Q3

Groups	Shape			Length		Development Behavior		
	Thin	Wide	Diffusive	Short	Long	Initial	Recurring	Continuous
Stuttgart	●				●		●	
Stanford			●	●		●		
Melbourne		●		●				●
Heriot-Watt	●				●		●	
Delft-DARSim	●				●		●	
Delft-DARTS	●				●			●
Csiro	●				●		●	
Austin	●				●			●
Experiment Run 1	●				●			●

Table 1 Classification of the finger development across the groups by overall shape, length, and behavior. We classify the experiment runs the same and list experiment run 1 as representative for all experiment runs in the table.

By slicing into the volume for Box A, we find different shapes and types of developments of the fingers that are visible by looking at the concentration data of the different groups. We categorize them by overall shape, length, and development behavior. The individual categories are further subdivided as follows: overall shape in "thin", "wide", and "diffusive"; length in "short" or "long"; and development behavior in "initial pulse", "recurring pulses", and "continuous pulses". We provide our classification in Table 1. Based on our classification, we find that most groups develop thin and long fingers, and consider only stanford and melbourne to have neither long nor thin fingers. Melbourne has rather short and wide fingers, while for stanford they appear short and diffusive. With "diffusive" we describe the fingers to have no clear shape in the early stages, and that the distribution of CO_2 concentration in Box A seems to be fuzzy.



(a) Stanford fast and diffusive finger development.

(b) Heriot-watt thin and long but pulsing fingers.

(c) Heriot-watt saturation.

(d) Stuttgart thin and slow finger development.

(e) Melbourne slow and rather wide/round finger development.

(f) Melbourne saturation.

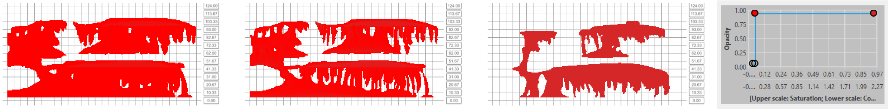
Fig. 10 The volume visualization of Box A for the first 24 h reveals different types of finger development. For example, some groups show a "pulsing" finger development. We can also see this easily by slicing into the volume to Box A and adjusting the transfer function accordingly (c).

During the analysis of the fingers in Box A, we noticed that instead of the CO_2 gradually dissolving into the water and dropping down in the form of fingers, this dissolving process often happens in the form of "pulses". We therefore classify the groups in "initial", "recurring" and "continuous" regarding the development behavior of fingers. With "initial" we refer to groups for which the fingers develop suddenly and only once in one initial "drop". "Recurring" refers to groups for which we noticed recurring pulses, where each pulse introduces more CO_2 concentration which developed in a short time period. "Continuous" refers to the groups where we cannot find recurring behavior or an initial fast drop, but for which the CO_2 instead gradually dissolves into fingers which thus continuously grow.

We find that only stanford has one initial fast drop. The other groups have either recurring pulses or a continuous growth of the fingers.

We provide some examples for different fingers in Figure 10. After further investigating the pulsing behavior, we also find that we see a brief period of low saturation during each pulse for the groups which have pulsing behavior, as we show in Figure 10(c). Compared to groups with continuous finger development, we see a continuous trail of low saturation over time instead, as we show in Figure 10(f).

In this section, we demonstrated the utilization of space-time cube views to conduct detailed investigations and classify spatial dynamics behavior of gravity fingers, particularly focusing on Box B. By slicing into the space-time cube and applying an appropriate transfer function, we also identified simulation artifacts that may offer valuable insights for enhancing the simulation models.



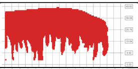
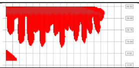
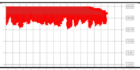
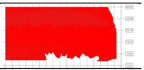
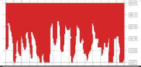
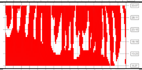
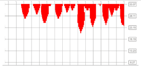
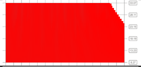
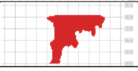
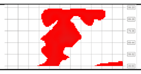
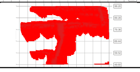
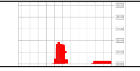
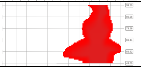
(a) Threshold 0. (b) Threshold 0.042. (c) experimental run 1 (d) Transfer function.

Fig. 11 Side-by-side comparison of the concentration spatial maps of delft-DARSim with a threshold of in (a) 0.0, and in (b) 0.042, to the segmentation map of the concentration data of experimental run 1 (c). All show a rendering of the last patch at 24 h. (d): The used transfer function for selecting the threshold.

Boxes	good match	moderate match	slight match	no match
Box A		🟡 🔵 🟢 🟠	🩷 🔴	⚫ 🟣 🟤
Box C		🔵 🟢 🟡	🟠	🔴 ⚫ 🟣 🟤 🩷
Box B	🟠	🩷 🟢	🔴 🔵	⚫ 🟣 🟤 🟡

🔵 austin 🟠 csiro 🟢 delft-DARSim 🔴 delft-DARTS 🟤 lanl
🟣 heriot-watt 🩷 melbourne ⚫ stanford 🟡 stuttgart

Table 2 Our interpretation of how well the simulation groups match the experiment run 1 regarding the visual similarity based on shape and size of the concentration spatial maps, at time step 24*h* and in boxes A, B, and C respectively. The simulation spatial maps were segmented with a threshold of 0.042. We interpret the matching of each group as "good match", "moderate match" (almost matching, but minor differences exist), "slight match" (only initial signs of the same spatial structure), and "no match" (completely different structure, e.g., nothing or everything segmented).

Boxes	target	good match	moderate match	slight match	no match
Box A	🩵 		🟡 	🔴 	⚫ 
Box C	🩵 		🟢 	🟠 	🟣 
Box B	🩵 	🟠 	🩷 	🔵 	🟤 

🔵 austin 🟠 csiro 🟢 delft-DARSim 🔴 delft-DARTS 🟤 lanl
🟣 heriot-watt 🩷 melbourne ⚫ stanford 🟡 stuttgart 🩵 experiments 1

Table 3 Example images of our interpretation from Table 2. We provide one example for "good match", "moderate match", "slight match", and "no match" with respect to the experimental data for each box at time 24 h (using experiment run 1 as target for the comparison). Neither for Box A nor for Box C do we find a "good match".

5.4 Visual Comparison with Experiment Group – Q4

The central difference when comparing the simulation groups with each other versus the comparison including the experiment group is that the experimental data is only available to us as segmentation maps. In the experimental grid, a chemical ingredient was added which changes its color in contact with CO_2. This allows to visually detect the presence of CO_2 during image processing, by applying appropriate thresholds on the amount of color per grid cell. For our computational comparison, we also have to segment the simulation groups based on a specific threshold. In Figures 4(f)-4(j) and Figures 6(f)-6(j), we use a segmentation threshold of 0.001 for the concentration and saturation values and include the experiment runs in the projection.

We find in "group" mode (Figures 4(f)-4(j)) that the experimental runs form one cluster that is distant from the rest of the groups. When projecting all patches (Figures 6(f)-6(j)), the experimental runs still form a separate cluster and stick together. However, they appear to be rather close to the groups stanford and heriot-watt, though, only for the very first few patches, after which stanford and heriot-watt then quickly diverge from them.

At this point, we cannot say whether any of the groups are actually close to the experiments based on our metrics and projections. Our ML-model was not trained on segmentation data and we cannot expect the results to be reliable for such data. Applying the Wasserstein metric on segmentation maps essentially comes down to counting ones and zeros and comparing the countings among the groups which ignores any potential shapes in the spatial maps. While Manhattan distance and Euclidean distance are better suited for comparing spatial shapes, they are still heavily affected from the chosen threshold for the segmentation.

Therefore, we instead try to verify the projection by visually comparing the experimental runs with the different simulation groups in the space-time cube renderings. By adjusting the transfer function to a function which maps all concentration values to either fully transparent or fully opaque colors depending on a threshold, we can see how the simulation data would look like if it were transformed to segmentation maps by applying this threshold. Thus, we can visually figure out which threshold, if applied to the simulation data, most closely resembles the experimental data. For example, delft-DARSim visually resembles the experiment run 1 at 24 h far better with a threshold of 0.042 on the concentration data instead of a threshold of 0.001 (Figure 11). Though, we do not find any suitable threshold for neither stanford nor heriot-watt, and notice that a visual comparison of the full spatial maps is not appropriate as local similarities and differences exist at the same time. Hence, we answer the question of which simulation group matches best to the experiments with respect to smaller regions of interest, the boxes A, B, and C.

Qualitative Interpretation of Visual Results: For our qualitative comparison of the concentration data, we fix the segmentation threshold of the transfer function to 0.042 for all simulation groups since it has been shown to improve the quality not just for group delft-DARSim but for all groups. We further notice only marginal differences between the visual results among the different experimental runs and thus choose experiment run 1 as their representative for this comparison. To provide enough time for the spatial structures to develop, we choose to only look at the last patches, i.e., the latest time step at $24h$ of the simulations and the experiment. For each group and box, we visually interpret and roughly categorize whether it is a good, moderate, or slight match to the experiment run. We list our interpretation in Table 2 and list some examples of this interpretation in Table 3. We interpret a "good match" between two instances as to having the same spatial structure in terms of shape and size. With "moderate match" we mean that the shape is similar, but either slightly too dominant or modest. For "slight matches", we witness only initial signs of the same shape which is not fully visible. If the shape does not match at all, or one of the boxes is empty, we consider them as having "no match". Based on this analysis, we find that csiro and delft-DARSim match the experimental runs best, followed by austin, stuttgart, melbourne, and delft-DARTS in this order. Heriot-watt, stanford, and lanl fail to match the experimental runs in this comparison.

We note that the main strength of visualization is hypothesis identification [15] and future work will need to quantitatively verify these findings. We did a first stab using the Euclidean distance metric, but found that this does not reveal the same patterns.

6 Discussion

In this section, we discuss some strengths and shortcomings and potential future work.

6.1 Strengths and Shortcomings

Similarity Metric: In the previous section, we provided multiple results, which show the utility of our visual approach. Our visual approach incorporates multiple commonly used similarity metrics like Euclidean, Manhattan, and Wasserstein distances. One strength of our approach is the incorporation of multiple such similarity metrics, but also of a more sophisticated ML approach as similarity metric, which can be used to project the full ensemble into an overview. While the incorporation of the ML model as a similarity metric allows integrating a metric that should consider features in the data such as spatio-temporal behavior, it is not clear what the model actually computes, as it acts as a black box to us. Furthermore, it may not be suited to be used on segmentation data, although it produces promising results.

Dimensionality Reduction: Another strength is the interactive linking between our overview, space-time cube renderings, and line charts. While the similarity view allows to quickly understand similar groups overall and how the simulations of the groups temporally relate to each other, it hides all the details of the actual simulation data. Also, the overview does not show the projection quality such as the remaining stress of the MDS projection. We addressed both issues by linking the views in our visual approach which enables efficient navigation and exploration of the ensemble dataset on different levels of detail by brushing in the overview or line charts. Hence, it is possible to manually verify the correctness of the projections to a certain degree by visually comparing the data details. Though, a proper visualization of the projection quality might be a good fit for future work.

Volume Rendering: One additional benefit in our visual approach was the selection of space-time cubes for the visual representation of the spatial maps.

A space-time cube is well suited for 2D+T data to provide a static overview of multiple 2D spatial maps that shows how saturation and concentration propagates through the geometry. However, a transfer function is necessary to properly leverage the capabilities of a space-time cube representation and configuring a transfer function can be difficult. It introduces many pitfalls in the visual analysis when a transfer function is not properly configured. Nevertheless, we showed the flexibility of the space-time cube rendering with an interactive transfer function and how it can be used to visually compare simulation to experiment runs with different thresholds on the saturation or concentration data.

Generalization: Even though we developed our approach in regard to the data of the benchmark study, it can be generalized to other data as well. Proper pre-processing as described in Section 4.1.1 might be required. For the overview, it is just a matter of choosing a similarity metric that is capable of computing the similarity between patches in the data that relate to a time component. The concept of patches can easily be extended to 3D+T(ime)+V(ariables) data which can also be processed by our chosen ML model. A drawback is that the ML model has to be trained on the data before it can be used with our visual approach and that too little data and poor hyperparameter settings can introduce overfitting. Euclidean, Manhattan, and Wasserstein distances do not have these issues and are still viable options for 3D+T+V data.

One core strength of choosing a space-time cube for visualizing the data is that we can show a static overview of multiple time steps at once, even for 2D+T data. We can still use the same concept for 3D+T data, although at least one dimension has to be collapsed or limited to one slice of it before we can render the remaining data as a 3D volume. Furthermore, our visual approach currently visualizes only one variable of the simulation data per space-time cube rendering. In our implementation, we chose two such renderings as juxtaposed views, which allows a side-by-side comparison of two space-time cube renderings. This limits the capabilities of visualizing more than two ensemble members or variables at once. The interaction with space-time cubes allows to zoom to specific regions of interests that are defined by the benchmark study description. Other data might not have these specific regions of interests and this interaction should be changed to allow to zoom to arbitrary regions of interests.

Scalability: In terms of ensemble size, we figure that bigger ensembles might introduce visual clutter in the overview and line chart views. Besides that, our approach lacks techniques for parameter space analysis [16], which is a common task for big ensembles. We discuss options to address these limitations and other future work below.

6.2 Future Work

The previous discussion and mentioned issues and drawbacks provide multiple directions to extend our visual approach in future work. For example, it is currently necessary to train the ML model we use as similarity metric. This model could be replaced with another model that does not require retraining but still captures spatio-temporal features, such as [33].

Furthermore, our visual approach should be able to naturally process arbitrary 3D+T+V spatio-temporal data. For a better integration of more variables in the space-time cube visualizations, we propose to superimpose multiple space-time cubes of different variables. Combined with a transfer function that can be configured for each space-time cube individually, this superimposition can extend the space-time cube visualization to show multivariate data in a single view.

In addition to our research questions (Section 3.2), the domain experts have expressed great interest to not only discover and analyze the outcome but also understand the reason behind why simulation outcomes vary. However, the differences between two model runs can be manifold and range from possibly different underlying balance equations, discretization approaches and constitutive relations over varying spatial parameters and grid resolutions up to diverse numerical solution approaches and parameters. Therefore, we leave this research question for future work where possibly more selected variations of a computational model are evaluated.

Our current work focused on an in-depth analysis of the ensemble of different simulation runs. Our hope is that this analysis will serve as a starting point for the next steps to agglomerate the ensemble into joint insights and decisions. For that, further extensions of our approach to make it amenable for broad communication will be necessary [15]. Meaningful physical interpretations of the applied similarity metrics will be of particular interest.

7 Conclusion

We presented an approach for the interactive visual analysis of simulation codes and experiment ensembles of porous media fluid flows. An overview with a variety of similarity metrics allows to identify and compare spatio-temporal patterns, such as the differences in the development of gravity fingers, and also to identify correlations among attributes measured from the spatial maps, such as the mobile CO_2 and dissolved CO_2 in different regions of interest. Detail views display CO_2 concentration and saturation in a space-time cube format, and support the navigation through the ensemble data.

We applied our approach to data from a benchmark study with nine different simulation models. Our analysis revealed new insights into ranking of simulation results with respect to experimental data, correlation between CO_2 saturation and concentration, and gravity finger development. As next steps, we plan to expand our collaboration to involve domain experts from all nine research sites and to jointly derive decisions and lessons learned from this large scale simulation endeavor.

Acknowledgments. Funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC

2075 – 390740016 and Project Number 327154368 – SFB 1313. Partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 251654672 – TRR 161, project A08. We acknowledge the support by the Stuttgart Center for Simulation Science (SimTech). Parts of this work have been done in the context of CEDAS, the Center for Data Science at the University of Bergen.

Declarations

Funding

Funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC2075 – 390740016 and Project Number 327154368 – SFB 1313. Partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 251654672 – TRR 161, project A08.

Competing interest

The authors have no relevant financial or non-financial interests to disclose.

Author contributions

All authors contributed to the conceptual design of our work, and made substantial contributions in writing and revising this manuscript. The implementation of the concepts, data preparation, analysis, and writing of the first draft were performed by R. Bauer. All authors read and approved the final manuscript.

Corresponding author

Correspondence to Ruben Bauer.

Data availability

The datasets which are used in this manuscript, mainly the benchmark study results, are available in the Fluidflower repositories, https://github.com/fluidflower. The code repository is available at https://github.com/rbnbr/VisualSpatioTempEnsembleAnalysis.

References

[1] Bachu, S., Bonijoly, D., Bradshaw, J., Burruss, R., Holloway, S., Christensen, N.P., Mathiassen, O.M.: CO_2 storage capacity estimation: Methodology and gaps. International Journal of Greenhouse Gas Control **1**(4), 430–443 (2007). https://doi.org/10.1016/S1750-5836(07)00086-2

[2] Pacala, S., Socolow, R.: Stabilization wedges: Solving the climate problem for the next 50 years with current technologies. Science **305**(5686), 968–972 (2004). https://doi.org/10.1126/science.1100103

[3] Metz, B., Davidson, O., De Coninck, H., Loos, M., Meyer, L.: IPCC Special Report on Carbon Dioxide Capture and Storage. Cambridge University Press, Cambridge (2005)

[4] Bear, J.: Modeling Phenomena of Flow and Transport in Porous Media vol. 1. Springer, Swiss (2018)

[5] Sahimi, M.: Flow and Transport in Porous Media and Fractured Rock: from Classical Methods to Modern Approaches. John Wiley & Sons, USA (2011)

[6] Kamrava, S., Sahimi, M., Tahmasebi, P.: Simulating fluid flow in complex porous materials by integrating the governing equations with deep-layered machines. npj Computational Materials **7**(1), 127 (2021). https://doi.org/10.1038/s41524-021-00598-2

[7] Lindeberg, E., Vuillaume, J.-F., Ghaderi, A.: Determination of the CO_2 storage capacity of the Utsira formation. Energy Procedia **1**(1), 2777–2784 (2009). https://doi.org/10.1016/j.egypro.2009.02.049. Greenhouse Gas Control Technologies 9

[8] Niemi, A., Bensabat, J., Shtivelman, V., Edlmann, K., Gouze, P., Luquot, L., Hingerl, F., Benson, S.M., Pezard, P.A., Rasmusson, K., *et al.*: Heletz experimental site overview, characterization and data analysis for CO_2 injection and geological storage. International Journal of Greenhouse Gas Control **48**, 3–23 (2016)

[9] Class, H., Ebigbo, A., Helmig, R., Dahle, H.K., Nordbotten, J.M., Celia, M.A., Audigane, P., Darcis, M., Ennis-King, J., Fan, Y., Flemisch, B., Gasda, S.E., Jin, M., Krug, S., Labregere, D., Naderi Beni, A., Pawar, R.J., Sbai, A., Thomas, S.G., Trenty, L., Wei, L.: A benchmark study on problems related to CO_2 storage in geologic formations. Computational Geosciences **13**(4), 409–434 (2009). https://doi.org/10.1007/s10596-009-9146-x

[10] Ward, M.O., Grinstein, G., Keim, D.: Interactive Data Visualization: Foundations, Techniques, and Applications. CRC press, Boca Raton, Florida, USA (2010)

[11] Keim, D.A.: Information visualization and visual data mining. IEEE Transactions on Visualization and Computer Graphics **8**(1), 1–8 (2002). https://doi.org/10.1109/2945.981847

[12] Shneiderman, B.: The eyes have it: a task by data type taxonomy for information visualizations. In: Proceedings 1996 IEEE Symposium on Visual Languages, pp. 336–343 (1996). https://doi.org/10.1109/VL.1996.545307

[13] Nordbotten, J.M., Fernø, M., Flemisch, B., Juanes, R., Jørgensen, M.: Final benchmark description: Fluidflower international benchmark study (2022). https://doi.org/10.5281/zenodo.6807102

[14] Flemisch, B., Nordbotten, J.M., Fernø, M., Juanes, R., Class, H., Delshad, M., Doster, F., Ennis-King, J., Franc, J., Geiger, S., Gläser, D., Green, C., Gunning, J., Hajibeygi, H., Jackson, S.J., Jammoul, M., Karra, S., Li, J., Matthäi, S.K., Miller, T., Shao, Q., Spurin, C., Stauffer, P., Tchelepi, H., Tian, X., Viswanathan, H., Voskov, D., Wang, Y., Wapperom, M., Wheeler, M.F., Wilkins, A., Youssef, A.A., Zhang, Z.: The FluidFlower international benchmark study: Process, modeling results, and comparison to experimental data. Transport in Porous Media, this S.I. (2023)

[15] Munzner, T.: Visualization Analysis and Design. CRC press, Boca Raton, Florida, USA (2014)

[16] Sedlmair, M., Heinzl, C., Bruckner, S., Piringer, H., Möller, T.: Visual parameter space analysis: A conceptual framework. IEEE Transactions on Visualization and Computer Graphics **20**(12), 2161–2170 (2014). https://doi.org/10.1109/TVCG.2014.2346321

[17] Fofonov, A., Linsen, L.: Projected field similarity for comparative visualization of multi-run multi-field time-varying spatial data. Comput. Graph. Forum **38**(1), 286–299 (2019). https://doi.org/10.1111/cgf.13531

[18] Sedlmair, M., Meyer, M., Munzner, T.: Design study methodology: Reflections from the trenches and the stacks. IEEE Transactions on Visualization and Computer Graphics **18**(12), 2431–2440 (2012)

[19] Wang, J., Hazarika, S., Li, C., Shen, H.-W.: Visualization and visual analysis of ensemble data: A survey. IEEE Transactions on Visualization and Computer Graphics **25**(9), 2853–2872 (2019). https://doi.org/10.1109/TVCG.2018.2853721

[20] Bach, B., Shi, C., Heulot, N., Madhyastha, T., Grabowski, T., Dragicevic, P.: Time curves: Folding time to visualize patterns of temporal evolution in data. IEEE Transactions on Visualization and Computer Graphics **22**(1), 559–568 (2015)

[21] Bach, B., Dragicevic, P., Archambault, D., Hurter, C., Carpendale, S.: A descriptive framework for temporal data visualizations based on generalized space-time cubes. Comput. Graph. Forum **36**(6), 36–61 (2017).

https://doi.org/10.1111/cgf.12804

[22] Pruess, K., Garciia, J., Kovscek, T., Oldenburg, C., Rutqvist, J., Steefel, C., Xu, T.: Code intercomparison builds confidence in numerical simulation models for geologic disposal of CO_2. Energy **29**(9–10), 1431–1444 (2004). https://doi.org/10.1016/j.energy.2004.03.077

[23] Walter, L., Binning, P.J., Oladyshkin, S., Flemisch, B., Class, H.: Brine migration resulting from CO_2 injection into saline aquifers – an approach to risk estimation including various levels of uncertainty. International Journal of Greenhouse Gas Control **9**, 495–506 (2012). https://doi.org/10.1016/j.ijggc.2012.05.004

[24] Sun, W., Durlofsky, L.J.: Data-space approaches for uncertainty quantification of CO_2 plume location in geological carbon storage. Advances in Water Resources **123**, 234–255 (2019). https://doi.org/10.1016/j.advwatres.2018.10.028

[25] Ferno, M.A., Haugen, M., Eikehaug, K., Folkvord, O., Benali, B., Both, J.W., Storvik, E., Nixon, C.W., Gawthrope, R.L., Nordbotten, J.M.: Room-scale CO2 injections in a physical reservoir model with faults (2023)

[26] Obermaier, H., Joy, K.I.: Future challenges for ensemble visualization. IEEE Computer Graphics and Applications **34**(3), 8–11 (2014). https://doi.org/10.1109/MCG.2014.52

[27] Wang, J., Hazarika, S., Li, C., Shen, H.-W.: Visualization and visual analysis of ensemble data: A survey. IEEE Transactions on Visualization and Computer Graphics **25**(9), 2853–2872 (2019). https://doi.org/10.1109/TVCG.2018.2853721

[28] Demir, I., Dick, C., Westermann, R.: Multi-charts for comparative 3d ensemble visualization. IEEE Transactions on Visualization and Computer Graphics **20**(12), 2694–2703 (2014). https://doi.org/10.1109/TVCG.2014.2346448

[29] Höllt, T., Ravanelli, F.M.d.M., Hadwiger, M., Hoteit, I.: Visual analysis of reservoir simulation ensembles. In: Rink, K., Middel, A., Zeckzer, D. (eds.) Workshop on Visualisation in Environmental Sciences (EnvirVis). The Eurographics Association, Groningen, the Netherlands (2016). https://doi.org/10.2312/envirvis.20161099

[30] Potter, K., Wilson, A., Bremer, P.-T., Williams, D., Doutriaux, C., Pascucci, V., Johnson, C.R.: Ensemble-vis: A framework for the statistical visualization of ensemble data. In: 2009 IEEE International Conference on Data Mining Workshops, pp. 233–240 (2009). https://doi.org/10.1109/

ICDMW.2009.55

[31] Fofonov, A., Linsen, L.: Multivisa: Visual analysis of multi-run physical simulation data using interactive aggregated plots. In: VISIGRAPP (2018)

[32] Tkachev, G., Frey, S., Ertl, T.: S4: Self-supervised learning of spatiotemporal similarity. IEEE Transactions on Visualization and Computer Graphics **28**(12), 4713–4727 (2022). https://doi.org/10.1109/TVCG.2021.3101418

[33] Huesmann, K., Linsen, L.: Similaritynet: A deep neural network for similarity analysis within spatio-temporal ensembles. Computer Graphics Forum **41**(3), 379–389 (2022). https://doi.org/10.1111/cgf.14548

[34] Hägerstrand, T.: What about people in Regional Science? Papers of the Regional Science Association **24**(1), 6–21 (1970). https://doi.org/10.1007/BF01936872

[35] van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(86), 2579–2605 (2008)

[36] McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv (2018). https://doi.org/10.48550/ARXIV.1802.03426

[37] Kruskal, J.B.: Nonmetric multidimensional scaling: a numerical method. Psychometrika **29**(2), 115–129 (1964)

[38] Kantorovich, L.V.: Mathematical methods of organizing and planning production. Management science **6**(4), 366–422 (1960)

[39] Frey, S., Ertl, T.: Progressive Direct Volume-to-Volume Transformation. IEEE Transactions on Visualization and Computer Graphics **23**(1), 921–930 (2017). https://doi.org/10.1109/TVCG.2016.2599042. Conference Name: IEEE Transactions on Visualization and Computer Graphics

[40] Frey, S., Ertl, T.: Flow-based temporal selection for interactive volume visualization **36**(8), 153–165. https://doi.org/10.1111/cgf.13070. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13070. Accessed 2021-01-27

[41] Machicao, J., Ngo, Q.Q., Molchanov, V., Linsen, L., Bruno, O.: A visual analysis method of randomness for classifying and ranking pseudo-random number generators. Information Sciences **558**, 1–20 (2021). https://doi.org/10.1016/j.ins.2020.10.041