Briefings in Bioinformatics, 2026, 27, bbag144
https://doi.org/10.1093/bib/bbag144
Published: 6 April 2026
Problem Solving Protocol

OXFORD

BEASTsim—a benchmarking and analysis platform for spatial transcriptomics simulations

Tomás Bordoy García-Carpintero [1,*,‡], Lucas A. D. T. Dyssel [1,‡], Kristóf Péter [1], Nikolaj F. H. Hansen [1], Lena J. Straßer [2,3], Chit Tong Lio [2], Merle Stahl [2], Markus List [2,3], Richard Röttger [1]

[1]Department of Mathematics and Computer Science, University of Southern Denmark, Campusvej 55, DK-5230 Odense, Denmark
[2]Data Science in Systems Biology, TUM School of Life Sciences, Technical University of Munich, Maximus-von-Imhof-Forum 3, D-85354 Freising, Germany
[3]Munich Data Science Institute (MDSI), Technical University of Munich, Walther-von-Dyck-Straße 10, D-85748 Garching, Germany

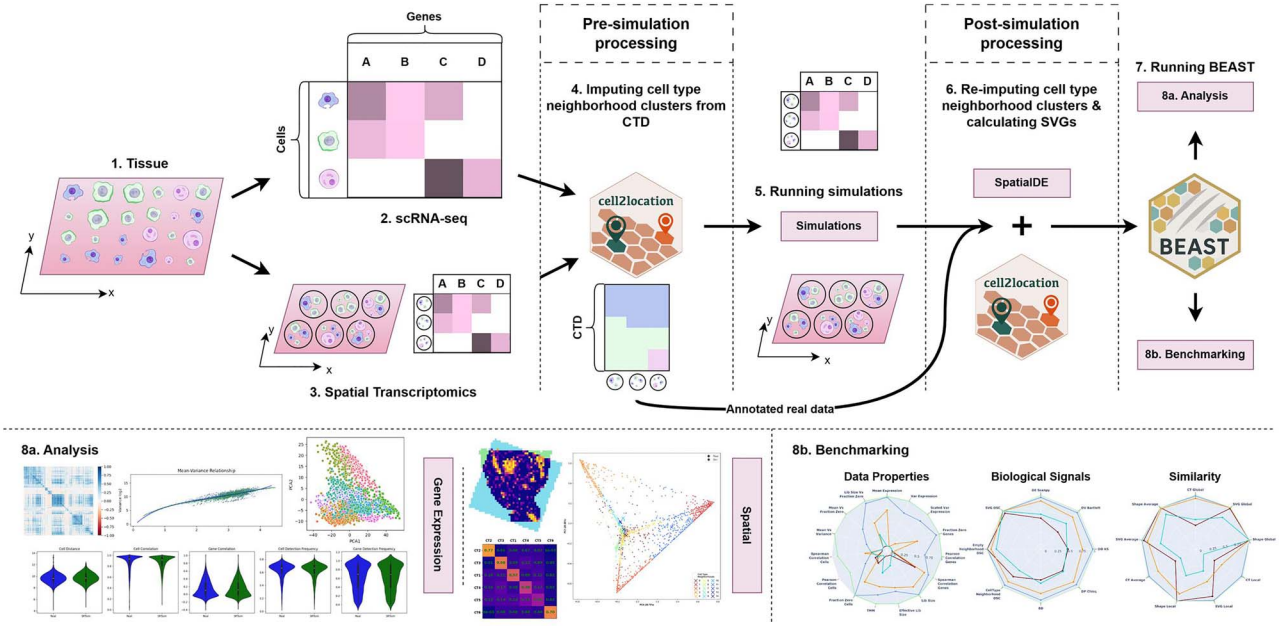
*Corresponding author. E-mail: tobor@imada.sdu.dk
‡Tomás Bordoy García-Carpintero and Lucas A. D. T. Dyssel are shared first authors

Abstract

Advancements in spatial transcriptomics and single-cell RNA sequencing have enhanced our understanding of gene expression within tissues. Spatial transcriptomics retains spatial context at the expense of resolution, often resulting in cell mixtures, whereas single-cell RNA sequencing offers single-cell resolution with the loss of spatial information. Some computational methods aim to integrate data from these two technologies; however, a ground truth for their evaluation is typically lacking. Thus, simulation techniques may be used to generate artificial gold or silver standards, offering the possibility for standardized analysis. This not only requires an accurate replication of real tissue types, but also sufficient sample diversity, calling for a unified evaluation of these properties between present techniques. Existing benchmark metrics and platforms often favor simulations that closely replicate the input data rather than promoting novel tissue layouts. This paper introduces a comprehensive benchmarking platform that evaluates spatial transcriptomics simulation methods across data property distributions, biological signal preservation, and similarity-based metrics. Our framework ensures that simulations go beyond simple data replication, instead introducing biologically meaningful variation. BEASTsim can be easily integrated into analysis pipelines and provides a practical tool for evaluating and developing computational methods, thereby advancing the integration of spatial transcriptomics and single-cell RNA sequencing data to yield more accurate biological insights. As a result, we have utilized BEASTsim to create a decision tree that helps users select the most suitable simulation model based on their data and goals. This work provides a practical tool for evaluating and developing computational methods, thereby advancing the integration of spatial transcriptomics and single-cell RNA sequencing data to yield more accurate biological insights.

Received: October 9, 2025. **Revised:** January 26, 2026. **Accepted:** March 5, 2026

Graphical Abstract



Keywords benchmarking, simulation, spatial transcriptomics, spatial variable genes, cellular neighborhoods

Introduction

Over the past few decades, developments in molecular biology have enhanced our ability to investigate complex biological systems. Among the advancements is the field of transcriptomics, the study of RNA expression, revealing the molecular properties of tissues and cells. Initially, bulk RNA sequencing was used to quantify the average gene expression at the tissue level, offering valuable insights into the average signal across a heterogeneous cell population. The limitation of only studying the average cell was later addressed with the emergence of single-cell RNA sequencing (scRNA-seq), enabling the profiling of gene expression at the cell level rather than at the tissue level, allowing for the discovery of novel cell types (CTs). Recently, a new field of technology emerged known as spatial transcriptomics (ST). ST extends the capabilities of scRNA-seq by preserving the spatial context of the gene expression within a tissue, allowing for the analysis of cellular neighborhoods. Despite its ability to maintain the spatial context, ST often suffers from reduced gene coverage and reduced cellular resolution, making the higher quality scRNA-seq data essential for downstream analysis [1]. In particular, cell mapping is a computational approach designed to integrate ST and scRNA-seq data, thereby combining the spatial context of ST with the high-resolution transcriptomic profiles of scRNA-seq. For biological data, however, no ground truth exists for such a mapping due to the destructive nature of each technology, making it difficult to assess performance rigorously. Simulations can be used to address this challenge by providing an artificial dataset with a known, user-defined ground truth, free from the limitations of current technologies. Despite their potential, the relative youth of the ST field means that no standardized simulation benchmarking framework currently exists, resulting in inconsistent method comparisons and consequently complicating method development. The development of such a benchmarking framework is therefore essential to accelerate the progress of ST research.

To address these challenges, we present BEASTsim, a standardized benchmarking and analysis platform for evaluating ST simulation techniques. BEASTsim is designed to be both accessible for researchers seeking to identify the most suitable simulation method and extensible for developers aiming to integrate new simulation techniques or benchmarking metrics. The framework encompasses a range of benchmarks, from basic data properties such as library size and mean expression to more advanced metrics capturing biological signals, including differentially expressed genes. It also includes a novel similarity-based benchmark aimed explicitly at detecting replicative simulations that merely return a slightly modified version of the ground truth. In addition, BEASTsim provides analysis tools for visualizing benchmarking results through detailed and informative plots and charts.

This study makes the following contributions:

1. A comprehensive benchmarking framework for evaluating ST simulation techniques, with a modular, extensible, and open-source Python implementation.
2. Novel similarity-based benchmarking metrics to detect trivial simulations.
3. A region-based consideration of cell locations, instead of point-wise, enabling the comparison of both reference-free and reference-based simulations.
4. A benchmark study comparing five different ST simulation techniques, with a decision tree provided to guide method selection.

Materials

Datasets

Three publicly available datasets were used from the MERFISH, 10x Visium, and Xenium ST technologies. The Mouse Ileum MERFISH

dataset was generated by [2, 3] and includes ∼5800 cells obtained from cryosections of fresh-frozen mouse ileum, downloadable from bioconductor. The mouse brain (sagittal-anterior) 10x Visium dataset, containing 2695 spatial spots, was produced by 10x Genomics [4] from a fresh-frozen C57BL/6 male mouse brain and is publicly available at 10xgenomics. Lastly, the human colorectal cancer Xenium dataset, containing ∼388 000 segmented cells and a gene panel comprised of 480 assayed genes, was generated by 10x Genomics [5] and is publicly available at 10xgenomics. As a reference for CT annotation for the 10x Visium dataset, we used the cortical cell dataset provided by the Allen Institute [6], from an adult mouse cortex with a total of 14 249 single cells, 23 CTs, and 34 617 genes, downloadable from dropbox. For CT annotation of the Xenium colorectal cancer dataset, we used an scRNA-seq reference derived from the Healthy Reference subset of the Pan-Gastrointestinal (Pan-GI) Cell Atlas [7], restricted to large-intestine-derived tissue and comprising 280 847 single cells annotated into 42 intermediate-resolution CT.

See the Supplementary Section *2* for a more detailed description of each dataset.

Data filtering

The MERFISH and Visium datasets were filtered in two different ways, by selecting spatially variable genes (SVGs) and Spapros genes [8]. This approach focuses our benchmarks solely on these spatial features while also reducing simulation time. This type of filtering was not applied to the Xenium dataset due to the generally low gene counts inherent to the technology, diminishing the applicability of spatial variability-based filtering methods. Instead, the worst quality cells, having extremely small expression counts, have been removed. After this processing, the Xenium dataset contained 273 660 cells. As a result, we acquired five datasets for our benchmarking study: MERFISH SpatialDE, MERFISH Spapros, Visium SpatialDE, Visium Spapros filtered, and Xenium. The specifics of the selection process and the rationale behind the chosen thresholds for the filtering are detailed in the Supplementary Section *4.2*.

Simulation techniques

Several ST simulators exist, differing in their approach to generating synthetic spatially resolved gene expression data. Some simulators generate data exclusively from ST, such as scCube [9], SRTsim [10], and scDesign3 [11]. Others generate from voxel-based ST combined with scRNA-seq data, such as SpatialcoGCN [12], generating voxelized SC data. As a result, these simulations can only be reasonably benchmarked via comparison with a real reference when that reference is voxel-based, such as Visium, and are not applicable for MERFISH or Xenium. Lastly, some methods generate ST profiles purely from scRNA-seq data, such as spider [13], scMultiSim [14], synthspot [15], and spaSim [16]. However, in this study, we focus on simulators that take at least ST data as input and are also capable of returning ST data similar to the input ST data. The primary reason for this choice is the challenge of benchmarking methods that rely solely on scRNA-seq data. Without corresponding spatial ground truth, evaluating the accuracy of spatial reconstructions becomes inherently difficult. By using ST-based simulators, we ensure that spatial structure is preserved and that benchmarking can be performed with a known reference, improving the reliability of method comparisons. The methods in focus are SRTsim [10], scCube [9], scDesign3 [11], and SpatialcoGCN [12]. For a short overview of these methods, see the Supplementary Section *1.3*.

In addition to differences in input data sources, ST simulators can also be classified based on their reliance on spatial reference information. In general, they fall into three groups: reference-based, input-reference-free, and output-reference-free. Reference-based methods explicitly use spatial information from the input data to guide the generation of synthetic tissues. Input-reference-free methods generate spatial patterns without any spatial reference. Output-reference-free methods technically incorporate spatial information but do not use exact cell locations when constructing the new tissue. As a result, their generated tissues may exhibit different shapes and sizes compared with the original data. For clarity and consistency, we will refer to both input- and output-reference-free methods as reference-free throughout the rest of this study. This distinction does not affect our later benchmark results or selection process.

Methods

The methods we used to perform our benchmarking study on the simulated datasets and further analyze the specifics of the different scores are integrated into a modular, open-source Python package called BEASTsim. The Supplementary Section *3* provides a detailed explanation of the framework itself, where in Subsection *3.1* we explain the features of the different modules and their interaction, while in Subsection *3.2* we provide extensive theoretical explanations of the benchmarking and analysis tools.

Benchmarking pipeline

We utilized BEASTsim to run an extensive benchmarking of the selected tools. The entire benchmarking process is published on GitLab for complete reproducibility. The following is a brief overview of the steps we followed, also illustrated in the graphical abstract.

1. Filtering the reference ST data for spatially relevant genes to obtain the four datasets we listed in Section Data filtering, and removing noisy and low-quality genes from the Xenium data.
2. Preprocessing the five filtered real datasets obtained in the previous step, according to the default preprocessing routine provided by BEASTsim.
3. Running the simulation tools on the preprocessed data with various settings and hyperparameters.
4. Processing the generated artificial tissues together with the respective real references using BEASTsim's default routine.
5. Run the benchmark metrics on the processed results to generate an unbiased, spatially aware, fair comparison of the methods.
6. Visualization of the benchmarking results.

The Supplement Section *4.1* contains a similar list detailing the pipeline, with additional references to the related Supplement sections.

In particular, the data processing performed at steps 2 and 4 relies on the SpatialDE [17] method to calculate adjusted *P*-values signaling spatial variability for each gene, and the Cell2Location [18] CT deconvolution technique for inferring CT abundances for voxel-based data (e.g. 10x Visium). Cell2location was chosen as it is a state-of-the-art tool for obtaining cell abundances for each voxel, as claimed by several benchmarking studies [19–22]. For single-cell resolution spatial data

(e.g. MERFISH), a CT distribution is calculated by averaging the CTs of spatial clusters of cells, obtained using K-means clustering.

Benchmarking metrics

BEASTsim's benchmarking module is organized into three main categories of evaluation metrics: **Data Properties**, **Biological Signals**, and **Similarity**.

The **Data Property** and **count-based Biological Signal** benchmarks, originally developed for nonspatial single-cell datasets by Yue Cao *et al.* [23], have been adapted and extended in BEASTsim to incorporate spatial information. Specifically, categorical labels are assigned for each voxel by clustering their respective CT distribution vectors. These categories are used to partition the count matrix, enabling differential gene expression analyzes across categories and comparisons of data properties between real and simulated tissues. A detailed description of the data processing steps used to generate these voxel labels is provided in the Supplementary Section *3.1.4*. Hereafter, we refer to these categorical voxel labels as assigned CT neighborhoods.

While the previous benchmarks extend existing methods, the **Spatial Biological Signal** and **Similarity** benchmarks represent novel contributions. These methods rely on BEASTsim's grid transformation procedure, detailed in the Supplementary Section *3.2.1, Grid Transformation*. In short, tissues are divided into a square grid structure, after which, for each region, all voxels located inside it are expressed in a region vector. This is crucial for locally comparing real and simulated tissues with different shapes and numbers of voxels, as each voxel lacks an ID, making it impossible to pair a simulated voxel with its real counterpart. By comparing the average properties of voxels within corresponding grid cells, BEASTsim enables localized comparisons, with the grid resolution controlling the degree of spatial granularity.

Data property benchmarks

In the original version of the data property benchmarking metrics [23], the count matrix is split according to the CTs of the single cells, giving data property values separately per CT. Instead of using CTs, which contain no spatial information, we utilize the assigned CT neighborhoods for this splitting to integrate spatial context into these benchmarking metrics without altering their fundamental behavior.

The final benchmark scores are then calculated based on *P*-values derived from the two-sample KDE comparison test [24] between a sample from the data properties of the real and the simulated ST data. The different data property metrics compared in this way are listed in Supplement Section *1.1.1*, while Supplement Section *3.2.2* explains the precise score formula.

Biological signal benchmarks

We defined two categories of biological signal benchmarks: count-based and spatial.

The count-based benchmarks comprise various types of differentially significant statistical gene detection methods between groups of expression vectors, grouped according to assigned CT neighborhoods. For a more thorough explanation of comparison scores defined based on these tests, see Supplement Section *3.2.2, Count-based Biological Signals*.

The spatial benchmarks are inherently based on the spatial variance of genes and connectivity between CTs. These new scores are the following:

- SVGs: spatially variable gene overlap, assess the spatial variable gene retention of the simulation.
- CT neighborhoods: measures at a tunable level of locality, the probability of CTs being spatially next to each other, then compares the distribution of this CT with CT connectivity between tissues.
- Tissue border neighborhoods: measures at a tunable level of locality, the probability of CTs being spatially next to the tissue border, then compares the distribution of these CTs with border connectivity between tissues.

See the Supplementary Section *3.2.2, Spatial Biological Signals* for comprehensive theoretical explanations of these benchmarks. Figure 1, created with BEASTsim's analysis module, visualizes the raw values of the CT-to-CT and CT-to-border connectivity conditional distribution matrices of an example real and simulated tissue as heatmaps, which are compared and aggregated with the latter two benchmark scores.

Similarity benchmarks

Lastly, similarity benchmarks assess whether patterns in the simulated tissue are structurally preserved rather than merely copied.

- Global patterns: overall spatial overlap of CTs and SVG expressions
- Local patterns: spatially specific overlap of CTs and SVG expressions
- Average patterns: average overlap of CTs and SVG expressions

See the Supplementary Section *3.2.2, Similarity* for comprehensive explanations of these scores. Figure 2, created with BEASTsim's analysis module, visualizes the raw values as varying grid resolution heatmaps, which are aggregated into the mentioned similarity scores.

Results

Overview

Every simulation was performed on the UCloud server equipped with Intel Xeon Gold 6230 CPUs, 46 GB of memory, and an NVIDIA V100 GPU. For an overview of the runtime and memory usage of each method across each dataset, see the Supplementary Section *5.3.1*. The Supplementary also include a usability and sustainability analysis of all methods, focusing on general user experience and maintenance.

For each dataset, simulation performance was evaluated across three benchmark categories: data properties, biological signals, and local similarity. Within each category, raw benchmark metrics were aggregated to enable a comparative assessment of the evaluated simulation techniques. Detailed results and visual summaries of these aggregated scores are presented and discussed in the following section, with additional metric-level analyses provided in the Supplementary Material, Section *5*.

Scalability

To assess the scalability of the simulation techniques, we evaluated the performance on the 273 660-cell Xenium dataset, which is substantially larger than the Visium and MERFISH Datasets. This experiment served as a stress test of computational feasibility under realistic high-throughput conditions. SpatialcoGCN was excluded from this scalability experiment, as it is designed for spot-based ST data and aims to simulate SC resolution from mixed spot-level measurements such as Visium, as explained previously in Section 2.2. As a result,

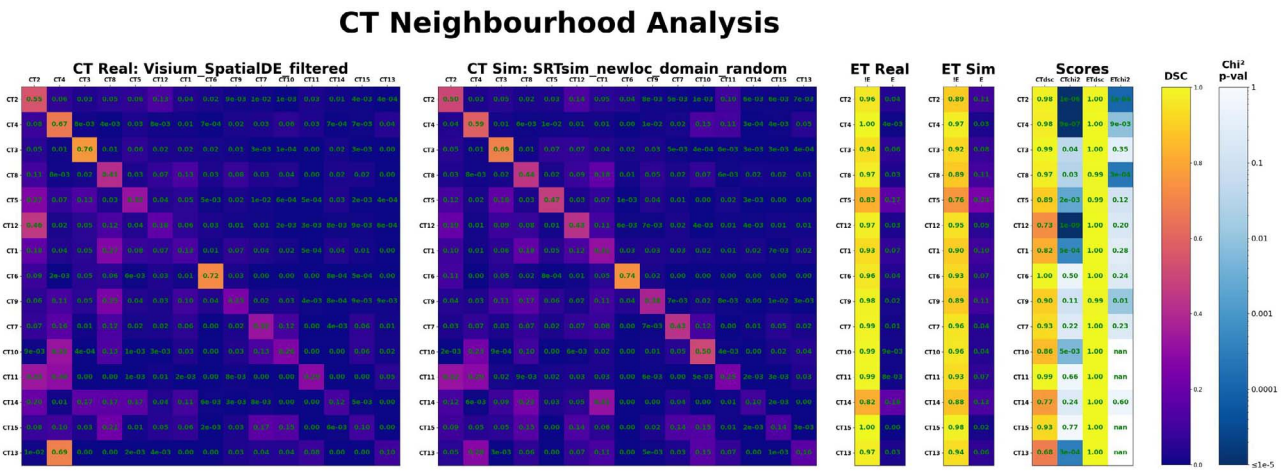


Figure 1. CT neighborhood analysis plot of real and simulated reference-free SRTsim [10] SpatialDE [17] filtered Visium data; details on matrix construction and comparison scores can be found in the Supplementary Section *3.2.2, Neighborhood benchmark* and *3.2.4, Neighborhood analysis*.

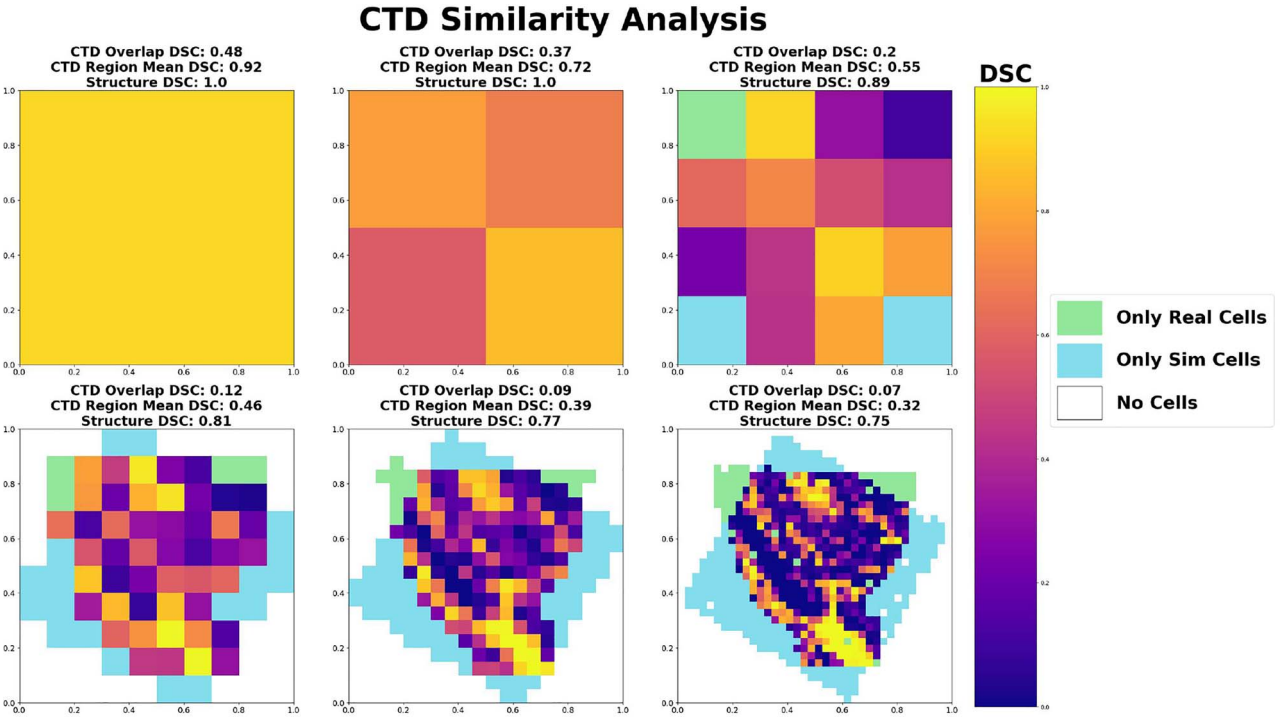


Figure 2. CTD similarity analysis of scCube [9] simulated SpatialDE [17] filtered Visium data, with heatmap values and aggregated benchmark scores detailed in the Supplementary Section *3.2.2, Similarity benchmark* and *3.2.4, Similarity analysis*.

the scalability assessment focused on the remaining simulation techniques.

None of the evaluated techniques was able to complete on the full Xenium dataset within the computational constraints previously mentioned, indicating that the dataset size exceeds the practical capacity of all tested methods. We repeated the experiment by slicing the dataset in half according to the median x-coordinate, reducing it to 136 830 cells. Despite this substantial size reduction, all methods still failed due to memory exhaustion. The experiment was repeated a final time by slicing the dataset again according to the median y-coordinate, reducing it to 68 415 cells, selecting the *top-right quadrant* of the full tissue. At this scale, scCube [9], scDesign3 [11], and the reference-based version of SRTsim [10] ran to completion, whereas the reference-free version of SRTsim [10] failed due to exceeding the allocated memory.

Main observations

The aggregated benchmarking results are summarized using three bubble plot visualizations, which compare simulation performance across datasets for data properties, biological signals, and local similarity. Figures 3 and 4 report aggregated scores for data properties and biological signals, while Fig. 5 summarizes local similarity.

Consistent with the SpatialSimBench [25] paper, the reference-based version of SRTsim [10] demonstrates the highest performance in nearly every data property and biological signal metric across

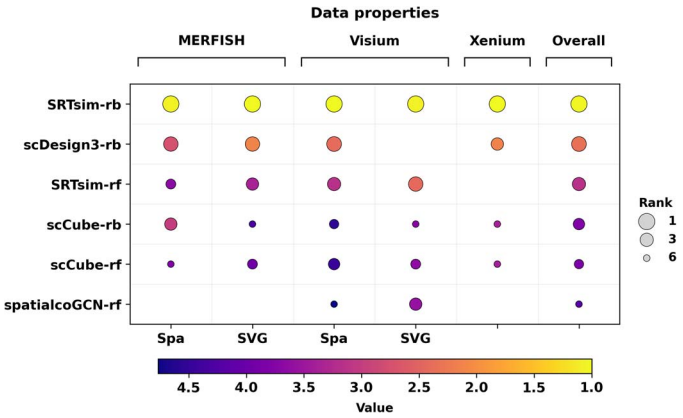


Figure 3. Summary of the data properties across the five evaluated datasets for each simulation method. For each method, the bubble color indicates the average rank computed from the data property metrics for a given dataset. The individual metrics contributing to this average are illustrated for one representative dataset in Fig. 6. The bubble size reflects the corresponding relative ranking of the method compared with other simulation approaches.

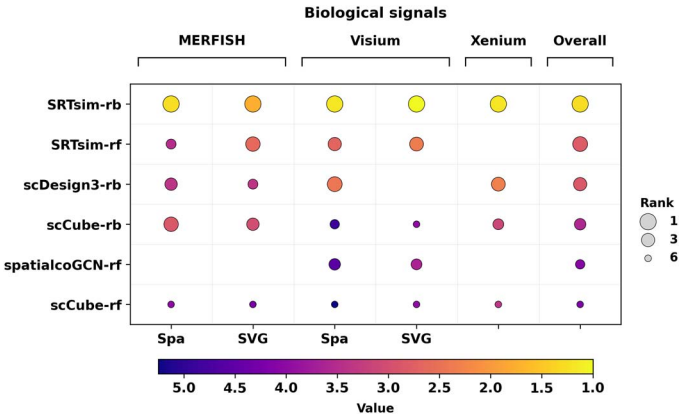


Figure 4. Summary of biological signal preservation across the five evaluated datasets for each simulation method, where bubble color indicates the average rank computed from the biological signal metrics. The bubble size reflects the corresponding relative ranking of the method compared with other simulation approaches.

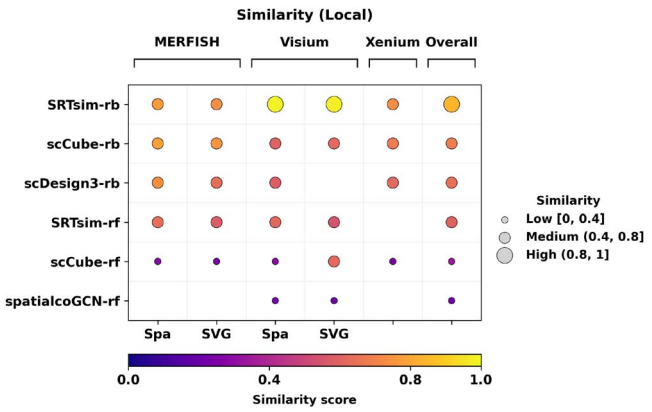


Figure 5. Summary of local similarity across the five evaluated datasets for each simulation method. For each method, the bubble color indicates the average similarity score computed from the local SVG and CT similarity metrics. The bubble size reflects the corresponding categorical similarity classification, defined as low (≤ 0.4), medium (0.4-0.8), or high (> 0.8).

each dataset. Among the remaining methods, only scDesign3 [11] and the reference-free version of SRTsim [10] achieve comparable results. As expected, reference-based methods that rely heavily on the spatial locations of the real data generally outperform reference-free approaches across most metrics. An exception is the reference-based version of scCube [9], which scores among the worst in terms of data properties.

We observed that in these simulations, some genes are being highly over- or underrepresented, in particular, genes that are sparse in the real data are sometimes removed altogether in the simulated scCube [9] datasets, with a global expression of zero. This significantly impacts the metrics that depend on gene presence.

SpatialcoGCN [12], much like scCube [9], underperforms in data property distributions. This is as expected, as it generates SC resolution data and subsequently voxelizes them to produce a spatially structured output. Therefore, the expression values are not taken directly from the ST dataset, but rather the expression of mixtures of SCs from the scRNA-seq data, leading to entirely different scaled datasets. Since SpatialcoGCN [12] was created as a method for benchmarking their cell mapping tool, the specific expression values were not of importance during the design process, and the underperformance is therefore expected.

As illustrated by the raw benchmarking results in Fig. 6, the reference-based version of SRTsim [10] achieves almost perfect local similarity scores, indicating that the method not only preserves global expression patterns but also reproduces fine-grained local structure. Capturing local patterns to this extent suggests that SRTsim [10] closely replicates the reference tissue, providing a plausible explanation for its consistently high performance. This interpretation is further supported by Supplementary Figure *11*, where differences in mean and variance expression are negligible. Importantly, this behavior is not limited to the representative Visium example. As summarized across all datasets in Fig. 5, the reference-based version of SRTsim [10] consistently attains medium to high local similarity scores.

As an example, we included a visualization of all the individual raw metrics for the SpatialDE filtered Visium dataset in Fig. 6. Similarly detailed spider-charts are also available for the other five datasets in the Supplementary Section *5.3*. Note that `GT-real_st`, seen in each spider chart, is a copy of the real dataset with the same annotations, etc.

Interpretation of results

While reference-free approaches generally perform worse than reference-based methods across most metrics, this does not imply that their performance is inherently poor. It is important to note that the scores presented in Figs 3 and 4 are relative, ranking model performance rather than showing absolute difference. To assess the true performance of an individual model, one must examine the raw metric values. A notable example of a reference-free method performing better than expected is the reference-free version of scCube [9]. As seen in the raw biological signal values (Fig. 6) compared with the aggregated ranks (Fig. 4), scCube [9] appears to perform poorly relative to other methods, but its absolute scores remain reasonably good. The key distinction is that it achieves high global similarity scores while maintaining low local similarity scores, suggesting that the model effectively

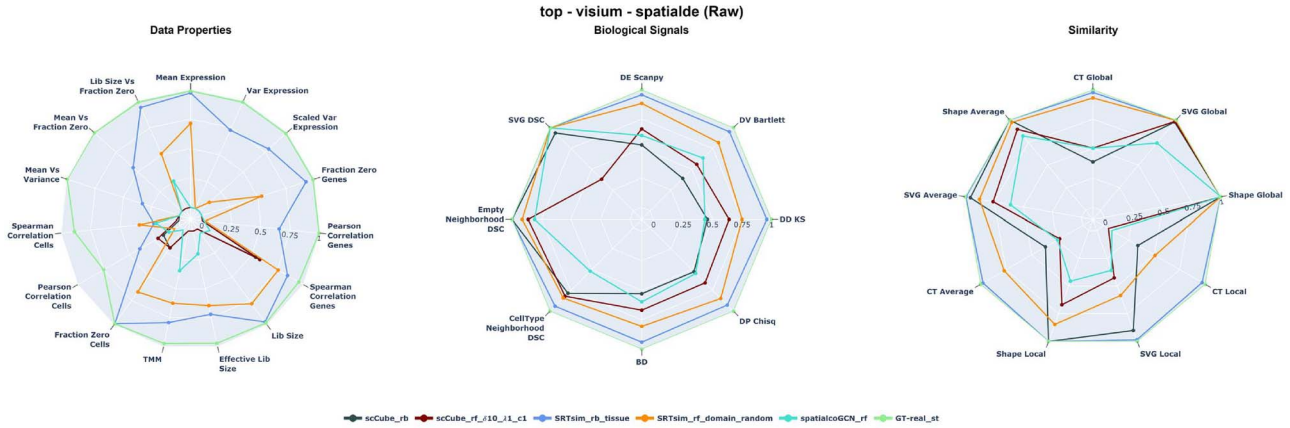


Figure 6. Raw spider chart comparison of SpatialDE [17] filtered Visium data generated by scCube [9], SRTsim [10], and SpatialcoGCN [12]. Due to the random sampling of cells used for the Pearson and Spearman cell correlation benchmarks, even the raw scores of the same tissue are not a perfect 1.

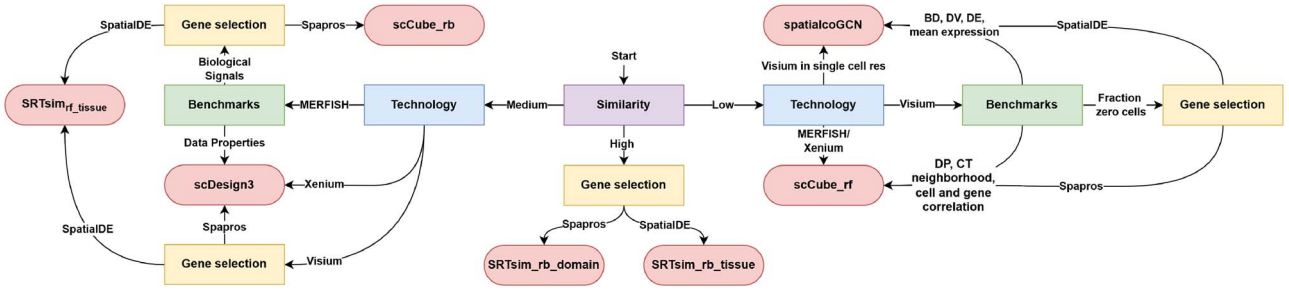


Figure 7. Decision tree for selecting an appropriate ST simulation method. The decision tree guides method selection based on spatial similarity requirements, gene selection strategy, benchmarking priorities, and target ST technology. The root node represents spatial similarity, categorized as low, medium, or high as defined in Fig. 5, reflecting whether the simulated data should closely resemble real tissue or intentionally diverge from it. For high similarity requirements, the reference-based version of SRTsim [10] is recommended across technologies and gene selection strategies. For medium similarity, the reference-free version of SRTsim [10] is recommended for SpatialDE [17]-filtered datasets, while scDesign3 [11] is preferred for Spapros [8]-filtered Visium and Xenium data; for Spapros [8]-filtered MERFISH, scDesign3 [11] is suggested when data property estimation is prioritized, whereas the reference-based version of scCube [9] is preferred when biological signal preservation is emphasized. For low similarity requirements, the reference-free version of scCube [9] is recommended in most settings, assuming the use of a normalized expression matrix. When Visium data are specifically required in single-cell resolution, SpatialcoGCN [12] is suggested

captures the overall cell distribution while also generating a new and diverse tissue layout. This behavior is characteristic of reference-free approaches and is supported by the raw metric results across all datasets. An extended interpretation of these results is provided in Supplementary Section *5.3*.

A visualized overview of the interpretation of the results can be found in Fig. 7, providing a guideline for determining which simulation technique is more suitable for a given ST technology and a set of properties.

Discussion

Interpretation of similarity benchmarks

An important aspect of the similarity benchmarks is that they are not primarily designed to assess the quality of a simulation method, but rather their type or nature. It is up to the user to decide which method best fits their purposes, and these benchmarks help inform that decision. According to this basic idea, a low score in similarity benchmarks is not a necessary indicator of an undesirable technique. For example, reference-free simulations are expected to produce low local similarity scores, as the tissue shape and CTs' spatial occurrences may be drastically different from those in real tissue. At the same time, global similarity can easily be large, as CTs and genes occur in overall similar numbers, albeit at different locations. Therefore, these scores provide a way for us to distinguish between highly similar reference-based methods, which may only introduce a slight noise on top of the real data, and techniques aiming at creating entirely new tissues with possibly similar overall biological properties as the reference.

Novelties of BEASTsim

ST is a relatively new technology, and as a result, most computational tools have only been introduced within the past few years, providing a newfound need for robust benchmarking. A recent contribution, *Spatial SimBench* [25], like our BEASTsim framework, adopts benchmarking metrics previously introduced by *SimBench* [23], extending them from SC to ST simulation benchmarking. In contrast to *Spatial SimBench* [25], our framework further extends the nonspatial benchmarks from *SimBench* [23] by incorporating assigned CT neighborhoods. Furthermore, we provide additional benchmarks for SVG detection and neighborhood-based analysis. Since a simulation technique that simply replicates the input data with some slight added noise would most likely outperform most genuine simulations, *Spatial SimBench* [25] may unintentionally favor such approaches.

Our framework, therefore, also contributes similarity-based benchmarks to fix this inherent design flaw by identifying simulation techniques that produce data highly similar to the input. See the Supplementary Section *6.2* for a more detailed demonstration of this effect.

Finally, we provide explicit support for reference-free simulation methods through our grid transformation, enabling the comparison of ST datasets with differing cell coordinates, cell numbers, and tissue shapes.

Limitations

Due to the min-max normalization of spatial coordinates inherent to the grid-transformation method, we recommend removing significant spatial outlier cells and avoiding datasets with substantially varying cell densities, as these may negatively impact the related benchmarks.

It is our experience that the data processing step of BEASTsim, explained in Supplement Section *3.1.4*, can consume a significant amount of computation time, depending on the properties of the input data, due to its reliance on Cell2Location [18] and SpatialDE [17]. It should be noted that while the framework allows for using these methods, it does not inherently require their output as long as the input data already have a CT or CT distribution assigned for each voxel and a probability value for each gene, interpreted as the significance of spatial variability.

The Cell2Location method [18] requires an integer count matrix to fit its model; therefore, it is not directly applicable to methods with normalized output, such as scCube. To bridge this discrepancy, we performed an additional pre-simulation processing step in case it was necessary. We note, however, that this may harm the end scores of scCube [9].

Since we rely solely on Cell2Location [18] for deconvolution, the results and conclusions may differ if alternative deconvolution methods were employed.

Future work

Henceforth, the BEASTsim framework will allow for the comparison and analysis of newly proposed simulation methods, providing the opportunity to continuously broaden our benchmarking study as the field of ST simulation develops. We invite the ST research community to use our framework for trying out new techniques and for helping with their improvement by pointing out their strengths and weaknesses.

As for improvements of BEASTsim, future iterations could expand the existing ST simulation benchmarks by incorporating relevant components from SpatialSimBench [25], thereby enabling broader and more diverse comparisons of simulation techniques. Another useful possibility would be extending the framework with wrapper functions to allow the user to run the simulation techniques through BEASTsim itself.

In addition, BEASTsim could be extended to support benchmarking of other computational ST tools, such as cell mapping techniques, creating a unified evaluation platform for ST technologies. This would include state-of-the-art cell mapping tools, such as the recent refinement [26] of Tangram [27], as well as cell-cell communication analysis methods, which are becoming increasingly important in ST research [28, 29].

Conclusion

Overall, this study provides a practical and informative tool for evaluating ST methods. By highlighting how different approaches perform across three key areas, our framework helps users select the most suitable method for their specific goals.

A critical aspect of simulation evaluation is ensuring that a model is not rewarded for simply reproducing the input tissue too closely. Without a proper similarity metric, simulations that lack meaningful variability could still receive high scores in other benchmarks. Our similarity benchmark addresses this by measuring how closely the simulation resembles the original data, helping to identify when a simulation lacks novelty. This is especially useful in cases where researchers intentionally introduce variation to explore different biological scenarios, as it provides a way to validate that the simulation remains distinct while preserving the key patterns of interest. In addition, our framework supports researchers working toward generating ST ground truths, offering tools to replicate key biological patterns, increase data availability, and design simulations tailored to specific experimental needs. The framework is implemented in Python and is publicly available, making it accessible and easy to integrate into existing analysis pipelines. To further assist users in selecting the most appropriate simulation model for their specific use case, we also provide a decision tree (Fig. 7) that guides model choice based on data characteristics and intended application.

Key Points

- **Comprehensive benchmarking framework**: BEASTsim is a unified, modular Python framework for the evaluation of ST simulation techniques. It is publicly available, designed for easy extension, and supports the integration of tools from other projects or user implementations. The benchmarking framework tests data property distributions, biological signal preservation, and the spatial similarity of the tissues. Additionally, BEASTsim includes an analysis module that offers multiple tools for visualizing and comparing spatial features, gene expression patterns, and tissue structures.
- **Similarity-based metrics**: BEASTsim introduces a similarity-based metric to detect simple data replication, thereby highlighting biologically meaningful variation over replication.
- **Compatibility with reference-free and reference-based simulations**: The grid-based transformation allows fair benchmarking of ST simulations regardless of whether they preserve spatial coordinates or generate them *de novo*.
- **A benchmark study comparing five different ST simulation techniques**: Our benchmark study tests the SRTsim [10], scDesign3 [11], scCube [9], and SpatialcoGCN [12] ST simulation techniques across several of their own hyperparameters, compares their best versions to each other, resulting in a decision tree that guides users based on preferences of reference-based or reference-free simulations, as well as differing benchmark scores.

Acknowledgment

We acknowledge the use of the UCloud computing infrastructure provided by the eScience Center at the University of Southern Denmark (SDU) for supporting the computational aspects of this work.

The graphical abstract includes modified cell illustrations by John Chilton from SciDraw (https://scidraw.io), licensed under CC BY 4.0. Original DOIs: 10.5281/zenodo.3926101, 10.5281/zenodo.3926211, and 10.5281/zenodo.3926537. It also uses the Cell2Location logo (with permission) from their original GitHub page https://github.com/BayraktarLab/cell2location.

Author contributions

Toms Bordoy Garca-Carpintero (Data curation, Methodology, Software, Writing—original draft, Writing—review & editing), Lucas A. D. T. Dyssel (Data curation, Methodology, Software, Writing—original draft, Writing—review & editing), Kristf Pter (Data curation, Methodology, Software, Writing—original draft, Writing—review & editing), Nikolaj F. H. Hansen (Software, Writing—review & editing), Lena J. Straer (Data curation, Writing—original draft, Writing—review & editing), Chit Tong Lio (Writing—review & editing), Merle Stahl (Writing—review & editing), Markus List (Supervision, Writing—review & editing), and Richard Rttger (Supervision, Writing—review & editing)

Supplementary material

Supplementary material is available at *Briefings in Bioinformatics* online.

Competing interests

M.L. consults for mbiomics GmbH. All other authors declare no competing interests.

Funding

Funding comes from the Novo Nordisk Foundation Data Science Collaborative Research Programme, grant Number NNF22OC0076414.

Data availability

The preprocessed versions of all real and simulated datasets used in our benchmarking study are available at Zenodo.

All of the data visualizations (Figs 1, 2, 3, 4, 5, 6) present in this paper can be regenerated using our dedicated result recreation script, which can be found on our GitLab repository. The code used in this study is written in Python and is publicly available. It can be accessed via our GitLab repository.

References

1. Li X, Wang C-Y. From bulk, single-cell to spatial RNA sequencing. *Int J Oral Sci* 2021; **13**:36.
2. Petukhov V, Xu RJ, Soldatov RA *et al.* Cell segmentation in imaging-based spatial transcriptomics. *Nat Biotechnol* 2022;**40**:345–54.
3. Geistlinger L, Lee T, Crowell H *et al.* MerfishData: collection of public MERFISH datasets. 2025. R package version 1.8.0.
4. 10XGenomics. Mouse brain serial section 1 (sagittal-anterior). 2020. Accessed: September 24, 2024.
5. 10x Genomics. FFPE human colorectal cancer data with human immuno-oncology profiling panel and custom add-on. Website. Accessed: 2025-08-12.
6. Tasic B, Yao Z, Nascimento M *et al.* Adult mouse cortical cell taxonomy by single cell transcriptomics. *Nature* 2018;**563**:72–8.
7. Teichmann S, Elmentaite R, Oliver A. Pan-gi cell atlas. Website. Accessed: 2025-08-12.
8. Kuemmerle LB, Luecken MD, Firsova AB *et al.* Probe set selection for targeted spatial transcriptomics. *Nat Methods* 2024;**21**:2260–70.
9. Qian J, Bao H, Shao X *et al.* Simulating multiple variability in spatially resolved transcriptomics with scCube. *Nat Commun* 2024;**15**:5021.
10. Shang L, Zhou X, Zhu J. SRTsim: spatial pattern preserving simulations for spatially resolved transcriptomics. *Genome Biol* 2023;**24**:39.
11. Song D, Wang Q, Yan G *et al.* scDesign3 generates realistic in silico data for multimodal single-cell and spatial omics. *Nat Biotechnol* 2024;**42**:247–52.
12. Yin W, Wan Y, Zhou Y. SpatialcoGCN: deconvolution and spatial information–aware simulation of spatial transcriptomics data via deep graph co-embedding. *Brief Bioinform* 2024;**25**:bbae130.
13. Yang J, Yang Q, Wei N *et al.* Spider: a flexible and unified framework for simulating spatial transcriptomics data. *Bioinformatics* 2026;**42**:btaf562.
14. Li H, Zhang Z, Squires M *et al.* scMultiSim: simulation of single cell multi-omics and spatial data guided by gene regulatory networks and cell-cell interactions. *Nature Methods* 2025;**22**:982–93.
15. Sang-aram C, Browaeys R, Seurinck R *et al.* Spotless, a reproducible pipeline for benchmarking cell type deconvolution in spatial transcriptomics. *eLife* 2024;**12**:RP88431.
16. Feng Y, Yang T, Zhu J *et al.* Spatial analysis with SPIAT and spaSim to characterize and simulate tissue microenvironments. *Nat Commun* 2023;**14**:2697.
17. Svensson V, Teichmann SA, Stegle O. SpatialDE: identification of spatially variable genes. *Nat Methods* 2018;**15**:343–6.
18. Kleshchevnikov V, Shmatko A, Dann E *et al.* Cell2location maps fine-grained cell types in spatial transcriptomics. *Nat Biotechnol* 2022;**40**:661–71.
19. Li B, Zhang W, Guo C *et al.* Benchmarking spatial and single-cell transcriptomics integration methods for transcript distribution prediction and cell type deconvolution. *Nat Methods* 2022;**19**:662–70.
20. Li H, Zhou J, Li Z *et al.* A comprehensive benchmarking with practical guidelines for cellular deconvolution of spatial transcriptomics. *Nat Commun* 2023;**14**:1548.
21. Slabowska AO, Pyke C, Hvid H *et al.* A systematic evaluation of state-of-the-art deconvolution methods in spatial transcriptomics: insights from cardiovascular disease and chronic kidney disease. *Front Bioinf* 2024;**4**:1352594.
22. Yan L, Sun X. Benchmarking and integration of methods for deconvoluting spatial transcriptomic data. *Bioinformatics* 2023; **39**:btac805.
23. Cao Y, Yang P, Yang JYH. A benchmark study of simulation methods for single-cell RNA sequencing data. *Nat Commun* 2021; **12**:6911.
24. Duong T, Goud B, Schauer K. Closed-form density-based framework for automatic detection of cellular morphology changes. *Proc Natl Acad Sci U S A* 2012;**109**:8382–7.
25. Liang X, Torkel M, Cao Y *et al.* Multi-task benchmarking of spatially resolved gene expression simulation models. *Genome Biol* 2025;**26**:57.
26. Stahl M, Straßer LJ, Lio CT *et al.* Refinement strategies for tangram for reliable single-cell to spatial mapping. *Bioinformatics* 2025;**41**:i552–60.
27. Biancalani T, Scalia G, Buffoni L *et al.* Deep learning and alignment of spatially resolved single-cell transcriptomes with tangram. *Nat Methods* 2021;**18**:1352–62.

28. Lihong P, Wei X, Chendi H *et al*. CellDialog: a computational framework for ligand–receptor–mediated cell–cell communication analysis III. *IEEE J Biomed Health Inform* 2023;**28**: 580-591.

29. Wei W, Zhao W, Longlong L *et al*. Prediction of ligand–receptor interactions based on CatBoost and deep forest and their application in cell–cell communication analysis. *J Chem Inf Model* 2025;**65**:6341–66.

Briefings in Bioinformatics, 2026, **27**, bbag144
https://doi.org/10.1093/bib/bbag144
Problem Solving Protocol