MarkovHC: Markov hierarchical clustering for the topological structure of high-dimensional single-cell omics data

Zhenyi Wang, [#,1] Yanjie Zhong,[3,5] Zhaofeng Ye,[2] Lang Zeng,[6] Yang Chen,[1] Minglei Shi,[2] Minping Qian*,[3] and Michael Q. Zhang*,[1,2,4]

1 MOE Key Laboratory of Bioinformatics; Bioinformatics Division and Center for Synthetic & Systems Biology, BNRist; Department of Automation, Tsinghua University, Beijing 100084, China

2 MOE Key Laboratory of Bioinformatics; Bioinformatics Division and Center for Synthetic & Systems Biology, BNRist; School of Medicine, Tsinghua University, Beijing 100084, China

3, School of Mathematical Sciences, Peking University, Beijing 100871, China

4, Department of Biological Sciences, Center for Systems Biology, The University of Texas, Richardson, TX 75080-3021, USA

5, Department of Mathematics and Statistics, Washington University in St. Louis, St. Louis, MO 63130, USA

6, Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA

Emails:
Zhenyi Wang[#]: wangzy17@mails.tsinghua.edu.cn
Yanjie Zhong: yanjie.zhong@wustl.edu
Zhaofeng Ye: yezf16@mails.tsinghua.edu.cn
Lang Zeng: langzeng@umich.edu
Yang Chen: yc@mail.tsinghua.edu.cn
Minglei Shi: shiml79@mail.tsinghua.edu.cn
Minping Qian*: qianmp@math.pku.edu.cn
Michael Q. Zhang*: michaelzhang@tsinghua.edu.cn

Highlights:

1. MarkovHC explores the topology hierarchy in high-dimensional data.
2. MarkovHC can find clusters (basins) and cores (attractors) of clusters in different scales.
3. The trajectory of state transition (transition paths) and critical points in the process of state transition (critical points) among clusters can be tracked.
4. MarkovHC can be applied on diverse types of single-cell omics data.

Abstract

Distinguishing cell types and cell states is one of the fundamental questions in single-cell studies. Meanwhile, exploring the lineage relations among cells and finding the path and critical points in the cell fate transition are also of great importance.

Existing unsupervised clustering methods and lineage trajectory reconstruction methods often face several challenges such as clustering data of arbitrary shapes, tracking precise trajectories and identifying critical points. Certain adaptive landscape approach[1-3], which constructs a pseudo-energy landscape of the dynamical system, may be used to explore such problems. Thus, we propose Markov hierarchical clustering algorithm (MarkovHC), which reconstructs multi-scale pseudo-energy landscape by exploiting underlying metastability structure in an exponentially perturbed Markov chain[4]. A Markov process describes the random walk of a hypothetically traveling cell in the corresponding pseudo-energy landscape over possible gene expression states. Technically, MarkovHC integrates the tasks of cell classification, trajectory reconstruction, and critical point identification in a single theoretical framework consistent with topological data analysis (TDA)[5].

In addition to the algorithm development and simulation tests, we also applied MarkovHC to diverse types of real biological data: single-cell RNA-Seq data, cytometry data, and single-cell ATAC-Seq data. Remarkably, when applying to single-cell RNA-Seq data of human ESC derived progenitor cells[6], MarkovHC not only could successfully identify known cell types, but also discover new cell types and stages. In addition, when using MarkovHC to analyze single-cell RNA-Seq data of human preimplantation embryos in early development[7], the hierarchical structure of the lineage trajectories was faithfully reconstituted. Furthermore, the critical points representing important stage transitions had also been identified by MarkovHC from early gastric cancer data[8].

In summary, these results demonstrate that MarkovHC is a powerful tool based on rigorous metastability theory to explore hierarchical structures of biological data, to identify a cell sub-population (basin) and a critical point (stage transition), and to track a lineage trajectory (differentiation path).

Introduction

Cellular activities can be modeled as stochastic dynamical states in the high-

dimensional space of expression levels of genes, proteins, or other molecular quantities[9]. Modeling the dynamics of these processes can greatly improve our understanding of the cell fate decision, carcinogenesis, and early diagnoses of diseases. S. Wright[10] first proposed the adaptive landscape concept in 1932 to describe population dynamics. Waddington's epigenetic landscape[11,12] was proposed later and has shown increasing application in biology[13,14].

Nowadays, high throughput single-cell technologies provide a tremendous amount of data for modeling cellular dynamics, however, the observed data suffer from a high level of noises due to both intrinsic biological stochasticity and technical errors in spite of their potential to overcome heterogeneity as shown, e.g. in exploring the lineage relations among differentiated cells[15-19]and predicting cellular communications[20]. Computationally, clustering of cells and delineating density ridges/energy landscapes for trajectory reconstruction, are the two most important tasks, especially in topological data analysis (TDA)[5,21]. Existing methods remain facing a number of difficult challenges such as clustering non-convex and mixture-density data, detecting rare clusters, tracking precise trajectories and identifying critical points.

Density ridge, a basic idea in topological data analysis is the essence of clustering and trajectory problems but apparently ignored by a lot of existing methods. In contract, reconstructing the underlying landscape maintaining the topological structure can shed light upon these tasks and overcome most of the challenges mentioned above. Meanwhile, cells are known to be regulated by redundant genes, therefore the gene expression data observed in a high dimensional space can be embedded in lower-dimensional manifolds[22]. Different cell types express distinctive combination gene sets, which can be viewed as supports in mathematics. They shape different manifolds[23] and form disconnected blocks if ignoring boundary points. In addition, the definition of cell types greatly depends on classification criteria and is often ambiguous. The hierarchical structure, however, uses a series of levels corresponding to cells clustering in different fineness degrees to overcome the hard threshold problem. Moreover, the hierarchical structure can integrate information concisely[24,25]and reveal the inherent structure in many biological systems[26]. Thus, we developed a novel tool, MarkovHC, to address these issues in single-cell data analysis. MarkovHC is unsupervised (no assumption about the distribution of the data), which makes it intuitive, robust, and widely applicable. It recognizes cell clusters (basins) and cores of clusters (attractors) in different scales, tracks critical points (stage transition points) and transition paths (the trajectory of state transition) between clusters and packs all information in a hierarchical structure, which respects the topology of the data.

In this work, we compared MarkovHC with 19 popular clustering methods[19,25,27-46] (in part C and part D of Supplementary Notes and sheet 1 of Supplementary Table 1) and 4 representative trajectory construction methods (part C of Supplementary Notes) including SPADE[15], monocle series[16,47,48], URD[49] and Waddington-OT[14,26] in general. These existing works may be divided into two classes: data-based and model-based. Since MarkovHC is unsupervised, it belongs to the first class. For a given set of samples, a Markov chain in sample state space is constructed (see Supplementary Notes) and its steady state (an invariant measure of the Markov dynamics) is obtained to

approximating the probability density model with an adjustable coarse-graining scale ("temperature") parameter . Firstly, to validate the effectiveness of MarkovHC and to show its advantages, we applied it to simulated data and several benchmark single-cell datasets. Then, we used MarkovHC to explore the hierarchical structure in mouse ES cell differentiation, the transition path and critical points in human embryo development, and the critical points in gastric cancer. These results demonstrate that MarkovHC is capable of exploring hierarchical structure in high-dimensional data, finding basins and attractors in different scales, tracking critical points, and transition paths among basins systematically. In summary, MarkovHC is a powerful and extendable method based on rigorous mathematical theory for studying the landscape of single-cell data and shows great potential in analyzing extensive types of single-cell omics data.

Results

The MarkovHC algorithm.

In this section, we briefly introduce the proposed algorithm. The general flowchart of MarkovHC is shown in Fig. 1.

A gene expression (or flow cytometry) data matrix ($A=\{a_{gc}$: $g=1,2,..,G$, $c=1,2,\ldots,C\}$ is taken as a generic input (Fig. 1A). In principle, treating each distinct cell as a unique point (vertex) or vector in the G-dimensional "expression" space, an initial (i.e. lowest level) Markov network can be built by defining a transition probability matrix from vertex "density" (i.e. higher local vertex weight means more cells could jump out per unit time) and edge similarity (i.e. more interaction edge strength means more "flow conductance"). The "density" may be defined as the number of cells in the same state (same expression level) or vertex degree, and the similarities among states may be calculated using the cells shared between nearest neighbor (sNN) states (Fig. 1B). Since calculating the global Euclidean distance directly in high-dimensional space tends to introduce noises and biases, sNN is used instead, which is able to explore the natural ("geodesic") distance on the underlying "manifolds" and has been proved to get more robust results[50] in recovering cell subpopulations[51,52] and reconstructing trajectories[23,50,53,54]. In most graphical models, a similarity matrix alone is sufficient to define a Markov transition probability matrix. Since we prefer a pre-processing step (see Methods for details) for robustness, the degree of each node in the sNN network was used to measure the point aggregation density, (see Methods for the density score definition). Hence, the density scores of the nodes were used together with the similarity matrix to define the basic Markov transition probability. (Fig. 1C)

Given the transition probability matrix on each level, MarkovHC can find attractors and basins on each level and construct the hierarchical structure systematically. In this work, we term basins as clusters and attractors as cores of these clusters. Attractors are steady states in multi-stable systems and basins consist of attractors and their subsidiary points. These two notions have been widely used in previous works[55,56] modeling the nonlinear dynamics of organisms and biological

phenomena as dynamics systems.

Based on the metastability theory developed by one of us (Chen et.al. 1996)[4], the hierarchical structure is built in an iterative manner. Firstly, MarkovHC finds the attractors and basins on the first level (details are in part A of Supplementary Notes). Secondly, for each point in attractors, Dijkstra shortest path algorithm[57] is deployed to detect the shortest transition path from it to its nearest neighboring basin using the pseudo-energy matrix (Fig. 1D). The length of the shortest path represents the pseudo-energy cost in the transition from one basin (cluster) to its nearest neighboring basin (cluster). On the second level, state-space consists of basins on the first level. To form the Markov transition matrix on the second level, the transition probabilities among basins are calculated. Then, MarkovHC uses the same approach as that on the first level to find attractors and basins on the second level. Repeating this process and we can obtain attractors and basins on each level as well as the relations among basins on neighboring levels. The hierarchical structure of the levels (Fig. 1E) is built using these attractors, basins, and topological relations.

On each transition energy level, we can identify the transition path and critical points. The transition path is defined as the most probable transiting path between two basins and the critical points are defined as the closest points between two basins on the transition path. In previous works[58], the saddle points in stochastic nonlinear dynamics were regarded as critical points. However, in practice, it's impossible to locate perfect saddle points with discrete data. Hence, we regard boundary points on the transition path as critical points to approximate saddle points.

MarkovHC can characterize clusters, cores, transition paths and critical points among clusters.

In this section, we use simple examples to intuitively show how MarkovHC can find basins, attractors, critical points, transition paths among basins, and constructing the intrinsic hierarchical structure.

Although a large number of clustering methods have been developed, the definition of a cluster is still ambiguous. Inspired by topological data analysis (TDA) [21], MarkovHC algorithm is based on the topological connectivity among points, which gives us a clear and reasonable definition of a cluster. Wasserman[21] proposed a concept called level sets for targeted density function. Here, we generalize this idea to discrete data clustering problems. For any given $\delta > 0$, points with similarities smaller than δ are partitioned into one cluster. The larger δ is used, the larger and fewer clusters are obtained. As δ increases and reaches some critical value, some small clusters will merge into a larger one and form hierarchical structures. The above definition of a cluster is clear but lacked biological application because the interaction of two states in a biological system can often be asymmetric (no detailed balance or time-reversal symmetry). Therefore, we modified the symmetric δ connectivity to a directed graph. For any fixed δ, we can obtain a directed graph, from which we can tell basins, attractors, critical points, and transition paths. We used intuitive plots (Fig. 2A) to illustrate how

MarkovHC builds up a topologically structure-based hierarchy. Note that when we mention the length of a directed line segment below, we refer to some "asymmetric distance" of biological significance, for example, the amount of transition energy consumption. With a given δ, points that can be mutually connected by directed line segments consist of attractors (white border, Fig. 2A), which are the representative points in the highest density regions of basins. Other points that can link to a given attractor are assigned to the corresponding basin. As δ increasing, according to the order of basins' immergence, the hierarchical structure of these points can be constructed.

The Markov chain modeling state transition progress includes the possibility of a state returning to a previous state, by which transition paths can be modeled precisely and reasonably. The asymmetric δ connectivity gives MarkovHC the superiority to find the transition path among the basins and the critical points on the transition paths. On a specific level, with a given start basin, the transition path is the shortest path from its attractors to another basin. Within the transition path of two cell types, only a minority of cells (ridge cells) are critical in the progress[59]. If cells of one type transit to the critical points, these cells have a big chance to transit to another type. Meanwhile, detecting these critical points is of great importance in applications, for example, diagnosing the disease at an early stage[8]. As illustrated in Fig. 2A, the transition path (red box with arrows) and the critical points (black edge points) may be found.

For demonstration, we applied MarkovHC to a set of simulated points, consisting of two spheres with a bridge connecting them in a 3-dimensional Euclidean space (Fig. 2B). MarkovHC successfully identified the two basins (green, blue) and attractors (red) located at the core with high-density respectively. Meanwhile, the transition path (orange) between these two basins were tracked. Note that the critical points (purple) around the gap without topological continuity on the transition path were detected.

To accelerate computation, we used some simple clustering methods such as k-means to cluster original points into hundreds of small clusters at the first step when MarkovHC was used to analyze data sets with more than one thousand single-cells. We regarded these small clusters as new aggregated points for further processing. As long as the primary topology of the data is kept, it should be legitimate and helpful to do pre-clustering[60]. An example with the similar topological structure in Fig. 2B but with more points (2100) was analyzed with MarkovHC similarly and results were shown in Supplementary Fig. 1A-D. The results demonstrate that the base clustering will not destroy the hierarchical structure built by MarkovHC as long as the sizes of small clusters are sufficiently small to avoid clustering heterogeneous points in the same small cluster.

MarkovHC is able to cluster non-convex data, hetero-density data, and differentiation stages in a continuum.

Single-cell data in high-dimensional gene space is complicated. Therefore, clustering algorithms that assume data are convex may not always be suitable for single-cell data. For instance, k-means and complete linkage hierarchical clustering get biased

results in single-cell data clustering[61]. In addition, the heterogeneity and size variety of cell populations also bring obstacles to traditional density clustering. For instance, DBSCAN[62] is not applicable to identify rare cell types. In many cases, such as tissue development and cell differentiation, the single-cell data are in a continuum, which makes it hard to detect reasonable cluster boundaries. One solution would be using the topology of the data, which only depends on δ-connectivity[5]. Several works have proposed clustering methods based on topology[63,64] and achieved improved performance. MarkovHC also takes advantage of topological connectivity, which embeds the topology of the original data into basins.

We tested MarkovHC on simulated data which consists of four groups of points in different shapes in a two-dimensional space (Fig. 2F). Within each group, points at the edge of the basins are colored grey (Fig. 2F) to indicate low aggregation, while central points are colored black to indicate high aggregation. As Fig. 2G shows, density scores (D scores) reflected point aggregation degree. A hierarchical structure from level 23 to 26 is sketched in Fig. 2I. Three basins could be obtained on level 26 (Supplementary Fig. 2C). On level 25, the green basin split into two basins (brown and green, Supplementary Fig. 2B) with the critical points (purple) on the bridge connecting them. On level 24, the brown basin on level 25 was further divided into two basins (brown and dark brown, Supplementary Fig. 2A), while the latter split into two smaller basins (dark and light brown, Fig. 2H).

Next, we performed MarkovHC on a simulated single-cell data with three stages in a continuum generated by splatter[65]. The data include 1000 cells in three stages in a 5000 genes space. These 1000 cells were projected onto the principal component space and colored according to their stages (Fig. 2L, Supplementary Fig. 3B). Meanwhile, these are “low-density valleys” on the density landscape, which correspond to energy barriers separating basins on the pseudo-energy landscape (Supplementary Fig. 3A). On level 17 these basins and corresponding attractors were identified (Fig. 2M). In comparison with other algorithms, adjust rand index (ARI) (Fig. 2N) and normalized mutual information (NMI) (Supplementary Fig. 3C) were calculated, which showed that MarkovHC performed better than other methods on this dataset. Specifically, we checked the transition path (orange) and the critical points (purple) from the brown basin to the blue basin (Fig. 2O). The differentially expressed genes along the transition path (Fig. 2P) and the dramatic changes of gene expression (Fig. 2Q) around the critical points demonstrate faithfully the kinetics of gene expression along the transition path.

In the above simulations, we verified three merits of MarkovHC. Firstly, the sparse basin (grey in Fig. 2H and Supplementary Fig. 2) was successfully detected, demonstrating the ability of MarkovHC to detect small basins (rare types) sensitively. Secondly, non-convex basins with arbitrary shapes (blue in Fig. 2H and Supplementary Fig. 2, helixes in Fig. 2K) could be handled by MarkovHC. Last, MarkovHC could find clusters in the continuum (Fig. 2M). These results demonstrate the potentials for MarkovHC to identify rare cell types, handle non-convex in high-dimensional single-cell data, and track development trajectory.

MarkovHC can consistently achieve accuracy in clustering diverse types of single-cell omics data.

Accurate clustering is the foundation for correct identification of cell populations and new cell types. Moving to the real data, we compared MarkovHC with popular single-cell RNA-Seq clustering methods including Seurat, SIMLR, SC3, k-means, hierarchical clustering with average linkage (ALINK), HDBSCAN, spectral clustering and model-based clustering on five benchmark datasets[34,66] (Fig. 3A-E). The ARIs and NMIs of MarkovHC are comparable to those of these popular clustering methods (Fig. 3F). The comparison among MarkovHC, Seurat with default parameters, and hierarchical clustering with other linkages are in Supplementary Fig. 4. Remarkably, the cell atlas of C. elegans embryogenesis[66] (Fig. 3E) is a golden benchmark to test development trajectory and lineage clustering algorithm, in which 6090 cells were annotated as 27 sub-cell types. MarkovHC constructed the hierarchical structure of this benchmark and reconstructed sub-cell types on level 4 yielding 0.64 and 0.73 on ARI and NMI respectively (Fig. 3F), which outperformed the other methods.

In addition, the classification and hierarchical structures of human peripheral blood mononuclear cells have been well studied in many researches[67-69]. We applied MarkovHC on the 33k PBMCs dataset to validate the applicability of MarkovHC in analyzing datasets with a large number of cells. On level 113, basins correspond to B cells, T cells, NK cells, megakaryocyte, and monocyte (Supplementary Fig. 5) could be identified. To identify subtypes of basins, we checked lower levels on the hierarchical structure. On level 111, the T cell basin was divided into CD4+ T cell basin and CD8+ T cell basin. On level 110, CD4+ T cell basin was further divided into CD4+ naïve, and CD4+ memory T cell basins. Similarly, the subtypes of B cell, CD8+ T cell, and monocyte could also be recovered on lower levels (Supplementary Fig. 5). These results demonstrate the power of the hierarchical structure, with which a specific cell type can be identified using marker genes, and the sub-cell types or cell states can be checked in detail on lower levels.

Moreover, MarkovHC is based on topology among samples without an assumption of data distribution, it can be widely used to build the hierarchical structures of other omics data besides scRNA-Seq data. As long as an appropriate similarity matrix is provided, MarkovHC algorithm can be applied. In single-cell research, another widely used technology in profiling cell populations is mass cytometry. Here, we verified the applicability of MarkovHC on mass cytometry data from Anchang's paper[60], which includes about 150,000 cells and 38 parameters. In this dataset, cells were obtained from a healthy donor's bone marrow and 24 subpopulations were labeled according to the protein expression of 38 surface markers. We randomly sampled 10009 cells from the dataset in proportion and projected these cells into 2-dimensional space by UMAP (Fig. S6). The results showed that MarkovHC got a comparable performance to Seurat (Fig. S7).

Nevertheless, MarkovHC can help to perform integration analyses, for instance, the integration of scRNA-Seq and scATAC data. In Supplementary Fig. 8, there are 464

scRNA-Seq and 96 scATAC-seq samples generated for the retinoic acid-induced mESCs differentiation at day 4 in Zeng *et. al.*'s work[70]. MarkovHC identified three basins in scATAC-Seq (Supplementary Fig. 8A) data and scRNA-Seq (Supplementary Fig. 8B) data respectively. We used tSNE to visualize these data (Supplementary Fig. 8A, B), and identified differentially chromatin accessibility in scATAC-Seq (Supplementary Fig. 8C) and differentially expressed genes in scRNA-Seq (Supplementary Fig. 8D). The peaks locating highly expressed genes are more accessible than those of repressed genes. For instance, PDE10A, RSPH1, and FBXL17 were highly expressed in basin 1. And the peaks, chr17-8725942-8730132, chr17-31394370-31396459, and chr17-63652707-63655013, which correspond to these three highly expressed genes, were open in basin1. In basin 2, highly expressed genes, RERE and SP8, correspond to open peaks, chr4-149929815-149933175 and chr12-120087188-120088390 respectively. And in basin 3, expression of these genes and accessibility of corresponding peaks were low. Therefore, three basins from these two datasets could be matched according to the peak pattern and the gene pattern respectively, which were also consistent with the three subpopulations inferred by Zeng *et. al.*[70]. These results demonstrate the potential of MarkovHC in the integration of multi-omics data.

In order to verify the effectiveness of MarkovHC in discovering new biological knowledge from real data, we applied it to three other single-cell RNA-Seq datasets, i. e., human embryonic stem cell differentiation to definitive endoderm[6], human preimplantation embryos[7], and gastric antral biopsies in the following sections.

MarkovHC's hierarchical structure revealed cell lineages, cell types, and cell stages simultaneously.

In this section, we show that the hierarchical structure obtained by MarkovHC could be used to explore cell lineages, cell types, and cell stages. In human cell atlas[71], cell lineage means the developmental relationship of cells; cell type implies a notion of homeostatic persistence; and cell state refers to more inducible or transient properties. However, the boundaries among these concepts at the gene transcriptional level can be fuzzy, partly may be due to the limit of our knowledge and understanding of cellular dynamics. Fortunately, data-driven approaches, especially hierarchical algorithms can be helpful to refine these concepts.

Single-cell RNA-seq data[6] from experiments profiling snapshots of lineage-specific progenitor cells differentiated from H1 human ES cells in specific induction circumstances was used to illustrate this point. Cells were sorted with specific markers using fluorescence-activated cell sorting (FACS) and captured for single-cell RNA-seq using the Fluidigm C1 system (Scale bar 200μm). In total, 1018 single-cells of six cell types were analyzed in this dataset. All cells were projected into 2-dimensional space using phateR[72], in which six clusters could be found (Fig. 4A). Low-level results could be swamped by a lot of noises, while high level results may average out true biological signals. Therefore, appropriate levels for downstream analyses need to be chosen based

on biological prior knowledge and low dimensional visualization. There were six basins on level 42 (Fig. 4B), around which we did further analyses.

On level 42, the six basins (Fig. 4B) match the six cell types in the ground truth (Fig. 4A). On level 41, we noticed that the neuronal progenitor cells were divided into two basins. Meanwhile, the H1 and H9 ES basin was also divided into two individual basins on level 40. Relations among these basins can be visualized and studied using the hierarchical structure (Fig. 4C). Specifically, on the higher levels, the difference among basins might reflect different cell lineages, while on the lower levels it could correspond to cell types or stages. Meanwhile, the attractors are representative cells of basins, which consist of cells with significantly higher homogeneity than that of other cells in basins (Supplementary Fig. 9).

To closely inspect the difference among these basins, the 50 most significantly up-regulated genes in each basin were obtained by Wilcoxon rank sum test (Fig. 4D left). In checking the biological significance of these basins, we selected 34 lineage-specific markers from Chu's paper[6] to show the expression patterns of basins (Fig. 4D right). In these two figures (Fig. 4D), the pattern of basins obtained by MarkovHC through differentially expressed genes (in sheet 2 of Supplementary Table 1) were consistent with that of the selected lineage-specific markers, which demonstrated that the basins were of biological significance. To reveal the difference between H1 and H9 ES cell basins on level 40 and two small basins on level 41 in neuronal progenitor cells, we enriched gene ontology terms using the differentially expressed genes. On level 40, the differentially expressed genes (Supplementary Fig. 10) between the two basins in ES cells were enriched in terms related to cell division and cell cycle (Fig. 4E), which suggests these ES cells were in different cell cycle stages. In the brown basin, cells were in a more active division stage, while cells in the light brown basin were in the suppressive division stage. On level 41, the up-regulated genes (Supplementary Fig. 11) of cells in the blue basin were enriched in neuron differentiation (Supplementary Fig. 12A), while terms were enriched in synapse organization and forebrain development (Fig. S12B) in the light blue basin, which indicated that two subtypes of cells could be found in neuronal progenitor cells. Obviously, the definitive endoderm cell and endothelial cell are two lineages that would be merged on level 43. From the bottom to top in the hierarchical structure (Fig. 4C), cell stages, cell types and cell lineages are presented, which gives a good example of how hierarchical structure can be used to define cell identities and relationships at different resolutions along the developmental lineage trees.

MarkovHC revealed the transition paths and critical points in human preimplantation embryo development.

MarkovHC can be used to track transition paths among cell types and detect the critical points. This dataset includes scRNA-Seq data[7] of 1529 single-cells collected from 88 human preimplantation embryos at different stages. The overall layout of these 1529 cells in the space of PHATE1-3[72] can reflect the order of samples (Fig. 5A).

Cells collected at the same time point can be heterogeneous due to stochasticity in cell differentiation. We first identified homogeneous clusters (Fig. 5B) of cells with MarkovHC and then tracked the transition paths and critical points among these clusters. Meanwhile, the bar plot (Supplementary Fig. 13) shows that there is only one basin in E3, and then this basin separated into two basins in E4. Basin3 is dominant in E5 early and more new basins emerge after that, which suggests that new cell populations burst at that time. As the work from Petropoulos *et. al.*[7] demonstrated, cells in these data were identified as eight cell groups, including pre-lineages, trophectoderm (TE), inner cell mass (ICM), epiblast (EPI), primitive endoderm (PE), E5mid (these cells collected at seven time points), mural and polar (Fig. 5K). For there were eight basins on level 8, we selected the levels (from level 6 to level 15) around level 8 from the hierarchical structure (Fig. 5C) for further analyses.

To reveal the biological processes in each basin, the 50 most significantly up-regulated genes of each basin were identified using Wilcoxon rank sum test (in sheet 3 of Supplementary Table 1). Overall, genes were enriched on gene ontology terms related to the process of embryonic development, while specific groups of gene ontology terms could be found in each basin (Fig. 5D, in sheet 4 of Supplementary Table 1). To infer transition paths among these basins, we calculated the pseudo-energy matrix (Fig. 5E). Because a basin transit to its closest basin (yellow boxes in Fig. 5E) on the energy landscape with the highest probability, we inferred the trajectories (Fig. 5F) and built the development hierarchy (Fig. 5H) which is basically consistent with the knowledge of human embryo development hierarchy (Fig. 5K). To identify each basin, we used the canonical marker genes from Petropoulos *et. al.*[7] and LifeMap Discovery (https://discovery.lifemapsc.com/in-vivo-development/) in the embryonic development (Fig. 5G) and identified four main cell types which were 8-cell embryo (GATA6-, SOX7-, SOX17-), morula cell (GATA6+, SOX7-), ICM (inner cell mass, SOX2+ and PDGFRA+) and TE (trophectoderm, GATA2+ and GATA3+) according to the marker genes expression. To identify sub-groups in TE, we used the differentially expressed genes of mural (WIPF3+, PLCE1+, PDCD4+, SLC15A2+) and polar cells (GCM1+, NRP1+, SLCO4A1+)[7]. To further explore sub-groups in ICM, we checked the sub-basins of ICM on level 4 and found that there were three main basins (Fig. 5I). According to their marker genes expression (Fig. 5J), we identified these three basins as PE (PDGFRA, FGFR2, LAMA4, and HNF1B), EPI (SOX2, TDGF1, DPPA5, GDF3, and PRDM14) and ICM.early. To further check gene-expression “flow” trends along the transition path from 8-cell embryo to ICM, we tracked the cells in the path (Fig. 5L, yellow points indicate cells on the transition path and purple points indicate the critical points from morula cells to ICM). 1631 differentially expressed genes along this path were obtained (Fig. 5M, in sheet 5 of Supplementary Table 1). GATA2, PTGES, PDGFA (Fig. 5N) showed increasing trends, while SLC34A2, MYC, ARGFX (Fig. 5O) displayed decreasing trends. Dramatic changes in gene expression were observed around the critical points (purple points), indicating these points as pivotal stages in the developmental process. These results demonstrate the significance of transition paths and critical points in understanding developmental dynamics and time-series biological data.

MarkovHC detected the critical points from MSCs to gastric cancer cells.

The transition process from normal gastric cells to cancer cells is complex and poorly studied. Many studies have been carried out to better understand the carcinogenesis[73-75]. For example, Zhang *et. al.*[8] conducted a single-cell transcriptomic study on gastric antral biopsies and identified gastric cancer related cell populations. As mentioned above, transition paths and critical points can be helpful to reveal cluster relations. Here, we used MarkovHC to analyze 830 mesenchymal stem cells (MSCs) and 649 MSC-origin early gastric cancer cells (EGCs) (Fig. 6A) from one patient in Zhang *et. al.*'s study[8].

MarkovHC found two basins on level 22 (Fig. 6B), which was basically consistent with the ground truth (Fig. 6A). However, it was obvious that MSCs and EGCs got mixed up in the boundary region (Fig. 6A). Therefore, we checked the lower levels and found that the MSC basin (green) and EGC basin (red) split into three and two basins on level 19 respectively (Fig. 6C). Not surprisingly, basin 3 and 4 (Fig. 6C) were a mixture of EGCs and MSCs. The differentially expressed genes of these two basins to basin 2 (Fig. 6G) suggested that these EGCs might be in two different transition states. Meanwhile, basin 1 and 5 might be two subpopulations in EGCs (Fig. 6G). Based on these observations, two transition paths from basin 2 to basin 1 were plotted (yellow and brown arrows in Fig. 6D). From the pseudo-energy matrix (Fig. 6E), we could see it would take less energy to transit from MSCs (basin 2) to transition states (basin 3 and 4) than directly to EGCs (basin 1 and 5). Namely, MSCs could transit to transition states with a larger probability. Interestingly, we also noticed that it would cost more energy to transit from basin 3 to EGCs (basin 1 and 5) than from basin 4 to EGCs. This might suggest that basin 3 be more stable than basin 4 on the landscape and reflect the heterogeneity of these two transit states, with the latter having a greater potential to process into cancer. In addition, the two EGC basins (basin 1 and 5) could transit to each other suggesting plasticity and instability of EGCs. We presented the inferred hierarchy of these cell populations in Fig. 6F.

To further investigate the gene expression patterns in these basins, Wilcoxon rank sum test was deployed to find the top 50 differentially expressed genes (DEGs) (Fig. 6G, in sheet 6 of Supplementary Table 1), with which the GO terms were enriched in the relevant functional categories (in sheet 7 of Supplementary Table 1). In the heatmap, we observed relative homogeneity in MSCs (basin 2) and obvious heterogeneity in transition states (basin 3) and EGCs (basin 1 and 5). Specifically, highly expressed genes in basin 2 were enriched in response to topologically incorrect protein and unfolded protein related terms. This suggested that although basin 2 cells were MSC-like cells, they might have been stimulated by environment factors and could be resisting to incorrect protein folding. DEGs in basin 3 were enriched in protein targeting and protein localization related terms, while DEGs in basin 4 were enriched in intranuclear biological processes such as DNA strand elongation and DNA conformation change. The differences indicated that these two transition states responded to environmental stimulus differently. Interestingly, the up-regulated genes in basin 1 were enriched in epidermis development and primary alcohol catabolic

process related terms, while those in basin 5 were responses to metal ion and detoxification related terms. These results suggested that the formation of basin 5 might be related to metal ion stimulus, while basin 1 might be associated with alcohol stimulus and epidermis damage. The latter was partially supported by the fact that this patient (p8 in Zhang *et. al.*'s paper[8]) had a chronic alcohol consumption history.

To check the transition trajectories from MSCs to EGCs (path 1 and path 2 in Fig. 6D), we examined the gene expressions of some marker genes along paths (Fig 6.H-J). MSC markers (OLFM4, EPHB2, and SOX9) and EGC markers (KLK10, SLC11A2, SULT2B1, KLK7, ECM1 and LMTK3) were gradually decreasing and increasing respectively (Fig. 6H). DEGs along path 1 and 2 were also identified (Fig. 6I-J), and used for gene function enrichment analysis (in sheet 8 of Supplementary Table 1).

Basically, two clusters of DEGs could be obtained in Fig. 6I and 6J. In path 1, the decreased genes were enriched in response to chemokine, while the increased genes were enriched in cellular response to metal ion suggesting that the potential disease progress might be driven by metal pollution[76]. Specifically, we observed OLFM4[77], a marker for stem cells in the human intestine, deceased from MSCs to EGCs, while CEACAM6[78], which plays important roles in invasion and metastasis in Gastric Cancer, was increased (Fig. 6K). In path 2, the decreased genes were enriched in response to mechanical stimulus, DNA damage response and signal transduction by p53 class mediator resulting in cell cycle arrest, while the increased genes were enriched in neutrophil mediated immunity suggesting this path could be the classic p53 related progression of gastric cancer[79,80]. One key gene, SOX4, deceased along path 2 (Fig. 6L), which was consistent with the results that MiR-596 down-regulated SOX4 expression and was a potential novel biomarker for gastric cancer[81]. Another marker, NEAT1[82], is a long non-coding RNA promoting viability and migration of gastric cancer cells through up-regulation of microRNA-17, which increased along path 2 (Fig. 6L). Furthermore, expression values of these genes dramatically changed around the critical points (purple points in Fig. 6K, L), which suggested these critical points were unstable states locating at the boundaries of the basins. The two paths indicated that there might be two separating trajectories from MSCs to EGCs for this patient, corresponding to two types of potential disease progression routes. These results again demonstrate the great capability and flexibility of MarkovHC in clarifying the relationship of the cell subpopulations and the usefulness of transition paths and critical points in revealing the underlying mechanisms.

Discussion

In this paper, we developed MarkovHC based on the metastability of exponentially perturbed Markov chain to explore the pseudo energy landscape which corresponds to the intrinsic topological structure of the data. And we also developed an R package, "MarkovHC" (https://github.com/ZhenyiWangTHU/MarkovHC). This tool is easy to use and only requires a feature×cell data matrix as the input. We showed that MarkovHC can cluster cells into populations at different resolutions, which could correspond to cell lineages, cell types, and cell states. Furthermore, we showed that the

transition paths and the critical transition points among cell populations tracked by MarkovHC can reveal developmental processes, such as human embryonic development or disease procession.

Besides clustering and identifying differentiation paths, there could be other interesting types of analyses with the help of MarkovHC. First, one could calculate pseudo-time[16,47,48,83] from the hierarchical structure. Though hierarchical ordering itself already has a causal interpretation, one could refine it using the same framework as shown in this paper. Second, one could start out from a given relevant biological level, e.g. based on prior knowledge or other clustering methods, and use MarkovHC to explore the next level up or down in the hierarchy. Thirdly, the hierarchical structure obtained by MarkovHC could be used to integrating other or multi-omics data, e.g. by using UnionCom[84] recently developed by Cao *et. al.*, for the unsupervised topological alignment of single-cell multi-omics integration.

Finally, our method, which is based on a general mathematics theory, is flexible and robust. In the preprocessing, we used sNN to derive the initial transition probability matrix in order to overcome sparsity/dropout problems as well as computational burden. One could certainly use any other advanced manifold-learning methods[34,85] to build the ground similarity matrix. Although such preliminary processing step has little effect[60] on the topological structure at biologically relevant levels, it is possible to further optimize to best balance the tradeoff between efficiency and accuracy for a specific data set.

Methods

Pre-processing of single-cell data.

We employed functions in a popular single-cell toolkit, Seurat (v.3.2.1)[30], to pre-process scRNA-Seq count matrixes. These matrixes were normalized by NormalizeData function with factor 10000, and 3000 highly variable features were selected using FindVariableFeatures function. Then, the normalized matrixes were scaled by ScaleData function. Next, RunPCA function was used to perform linear dimensional reduction and dimensions were selected by an heuristic method generating an “Elbow plot”: principle components were ranked based on the percentage of variance explained by each one, and we always can observe an “elbow”, suggesting that the top PCs before the “elbow” could capture the majority of true signal. And these dimension reduced matrixes were the original inputs to the MarkovHC algorithm.

In particular, the pre-processing steps in DC3[70] were used to normalize the data of 464 scRNA-Seq and 96 scATAC-seq samples (Supplementary Fig. 8) from their work. And the cytometry data (Supplementary Fig. 6) were scaled by log(x + sqrt(x^2 + 1)) and performed linear dimensional reduction by RunPCA function.

Overview of the MarkovHC algorithm.

The input data of MarkovHC algorithm is the matrix of genes by cells or proteins by cells, representing the detected mRNA or proteins expression in each cell. Without loss of generality, let $s_1, s_2, \ldots, s_n$ be a set of d-dimensional data points. Traditionally, the

clustering goal is to divide them into different basins that accords with intuition and are interpretable:

$B_1 = \{s_{11}, s_{12}, ..., s_{1p1}\}, ..., B_k = \{s_{k1}, s_{k2}, ..., s_{1pk}\}$ where $\{B_i, i = 1, ..., \mathrm{k}\}$ usually have no overlap. As we know, sometimes k, the number of clusters, is cannot be decide beforehand and may be varying depending on different goals. Therefore, the Markov algorithm is allowed to explore the intrinsic hierarchical structure in the data points. Supplementary Fig. 14 shows the target hierarchical structure of MarkovHC, where every $\{B_i^j, i = 1, ..., k_j; j = 1, ..., r\}$ is the subset of $\{s_1, ..., s_n\}$ and B_i^p is the subset of some basins on level p+1. Note that we allow B_i^j on the same level have some overlaps defined exactly by the shared points. They are of great significance in the hierarchical structure. If there are points shared by B_i^j and B_k^j , once states in B_i^j reach to these shared points after a long trip, they can easily develop into points in B_k^j in a short time. Of course, sharing some points is not a sufficient condition for combining B_i^j and B_k^j together as. If it is very hard for core points of B_i^j and B_k^j to develop into points, B_i^j and B_k^j will not combine on the next level. Similar to the structure produced by traditional hierarchical clustering algorithms, when we select a level, there is a clustering result. Unlike those traditional hierarchical structures, structure produced by MarkovHC is more concise and meaningful. Every level's clustering result is interpretable. Different levels correspond to different δ mentioned in Fig. 2. If we want to obtain a more detailed clustering result, we can select a lower level.

We can define a measure of the relation between different sample points. Some measures are undirected, generally called similarity or dissimilarity. For example, k-means and standard hierarchical clustering usually use Euclidean distance as the dissimilarity between points. Some measures are directed. For example, DBSCAN[62] gives a directed definition of directly density-reachability. Obviously, a directed relation is more reasonable because in many cases, the status of two sample points is not equal. Let's take a biological example. Assume A is a point representing a hematopoietic stem cell and B is a point representing a red blood cell. A is able to develop into B while the inverse process is impossible. Therefore, we consider using Markov chains to describe the whole dataset. Continue with the example above, we can set the transition probability p(A, B) > 0 while p(B, A) = 0, which means that A can reach B but B cannot return to A. If we have a Markov chain to describe the relations of different points, it is very natural to see points belonging to a same recurrent class as core points of a cluster. Points that are not recurrent states but can reach to more than one recurrent class can be seen as shared points. However, a single Markov chain is not enough to describe the intrinsic structure of data points on the ground that one Markov chain only corresponds to one clustering criterion. As we have mentioned above, for many problems, building

a hierarchical structure can completely show the information contained in data. As a result, we use a family of Markov chains to describe the dataset. We can regard the input dataset as a finite state-space $S=\{1,...,N\}$. $\{P^{\beta}:\beta\in[0,\infty)\}$ are a family of Markov chain transition matrices on S satisfying that there exists:

$$p_{i,j}=p(i\to j)=p(j|i)=\lim_{\beta\to\infty}p^{\beta}(j|i)=p^{\infty}(j|i),\forall i,j\in S$$

For a fixed β, if state i has a stronger directed relation towards state j, $p^{\beta}(j|i)$ would be bigger. The parameter β controls the movement intensity of states. When β becomes bigger, the whole state-space will become more inactive that the expected time taken from core states in one cluster to another cluster will exponentially rise. Finally, when β approaches infinity, the state movement is basically limited in core points of the same cluster. Taking advantage of the difference of movement time scale between points, we can build a hierarchical structure of the state-space. We assume the following limit exists:

$$c_{i,j}=c(i\to j)=\lim_{\beta\to\infty}-\frac{1}{\beta}\log p^{\beta}(j|i)$$

We assume that

$$c_{i,j}>0, if\ p^{\infty}(j|i)=0$$

$c_{i,j}$ can be viewed as pseudo-energy barrier from state i to move to state j.

Under the assumptions given above, Markov chains family, which is composed of Markov chains with transition matrices $\{P^{\beta}:\beta\in[0,\infty)\}$ is an exponential perturbation of the Markov chain with transition matrix P^{∞}. The core of our theory is the pseudo-energy matrix C, which concentrates all asymptotic information provided by the Markov chain family. Actually, the time scale for point ξ to directly move to point η is $e^{\beta C_{\xi\eta}}$. The time scale for points in one cluster to move to another cluster can also be expressed by entries of C. As a result, only with the help of C, we can construct the target hierarchical structure iteratively.

Level 1 clustering result should be the most detailed and basic one. Therefore, we consider the most inactive Markov chain, i.e. the Markov chain with transition matrix P^{β}. Firstly, we find out all recurrent classes and see them as core points of the corresponding level 1 clusters. Actually, if ξ and η belong to a same recurrent class, there exists a path led from ξ to η such that the sum of pseudo-energy along this path is 0, which means that ξ can move to η effortlessly, and vice versa. We denote these level 1 recurrent classes as $\{A_i^1\}$, also called as "attractors". We denote the cluster whose

attractor is $\left\{A_i^1\right\}$ as B_i^1, also called as a basin. Then, we need to assign non-recurrent states to appropriate clusters. Given an attractor A_i^1 and a non-recurrent state τ, we say τ belongs to basin B_i^1 if there exists a directed path from τ to A_i^1 that the sum of pseudo-energy on this path is 0. Though different attractors do no overlap, it is possible that a non-recurrent state τ shared by two or more basins. We give a simple example in Supplementary Fig. 15. Assume data points are 1-dimensional and they are plot at the top of Supplementary Fig. 15. The value of vertical coordinate is the pseudo-energy, which is defined as the negative logarithm of density here. The time scale for points in $A_i^1(A_j^1)$ to reach τ is $e^{T\left(B_i^1\right)\beta}\left(e^{T\left(B_j^1\right)\beta}\right)$ while the time scale for τ to reach $A_i^1(A_j^1)$ is a constant irrespective of β. ($T\left(B_i^1\right)$ can be viewed as the least pseudo-energy cost for points in A_i^1 to reach τ.) Therefore, it takes a long time for the points in attractors to move to the shared points while the shared points can jump into the attractors in a very short time. Of course, the time scales for points in A_i^1 and A_j^1 to move to τ are different. Such a difference is exactly the criterion used to define the relation between level 1 clusters (basins) and to construct higher-level Markov chains.

The construction of level 2 clustering result is based on the degree of difficulty for points in level 1 attractor A_i^1 to escape from their corresponding basin B_i^1, i.e. $T\left(B_i^1\right)$. $T\left(B_i^1\right)$ is the least pseudo-energy cost for points in A_i^1 to reach other basins. Considering the characteristic of critical points, if B_i^1 has critical points shared by other basins, then $T\left(B_i^1\right)$ equals to the least pseudo-energy cost for points in A_i^1 to reach B_i^1's shared points. We construct a new Markov chain to describe the relation between level 1 basins. The new state-space is $S_1 = \left\{B_i^1 \middle| i = 1,...,k_1\right\}$, where k_1 is the number of level 1 basins. We set the transition probability $p\left(B_i^1 \middle| B_j^1\right) > 0$ if and only if $T\left(B_i^1\right) = \min_{1 \le k \le k_1} T\left(B_k^1\right)$ and one of the following two conditions holds:

1. B_i^1 and B_j^1 have common shared points;

2. There exists a path from A_i^1 to B_j^1 that the sum of pseudo-energy equals to

$T\left(B_i^1\right)$. After normalization, we can obtain a transition probability matrix and create a Markov chain on state-space S_1. Similar to level 1 clustering result, level 2 clustering results can be easily obtained on the basis of this new Markov chain. We illustrated it in Supplementary Fig. 16. There are four basins on level 1 (Supplementary Fig. 16A). We wrote down some time scales in the figure. For example, the time scale for points in A_1^1 to move to shared points shared by B_1^1 and B_2^1 is e^{β}. The rest of time scales not shown in the picture are all assumed to be bigger than $e^{4\beta}$. Based on our criteria, $B_1^1 \cup B_2^1 \cup B_4^1$ is a new level 2 basin (Supplementary Fig. 16B), which is denoted as B_2^1. In addition, B_3^1 is another level 2 basin, which is denoted as B_2^2. We should note that the attractor of B_2^1 is $A_1^1 \cup A_2^1$, but not $A_1^1 \cup A_2^1 \cup A_4^1$, for the reason that moving from A_1^1 to B_4^1 is far more difficult than moving from A_4^1 to B_1^1.

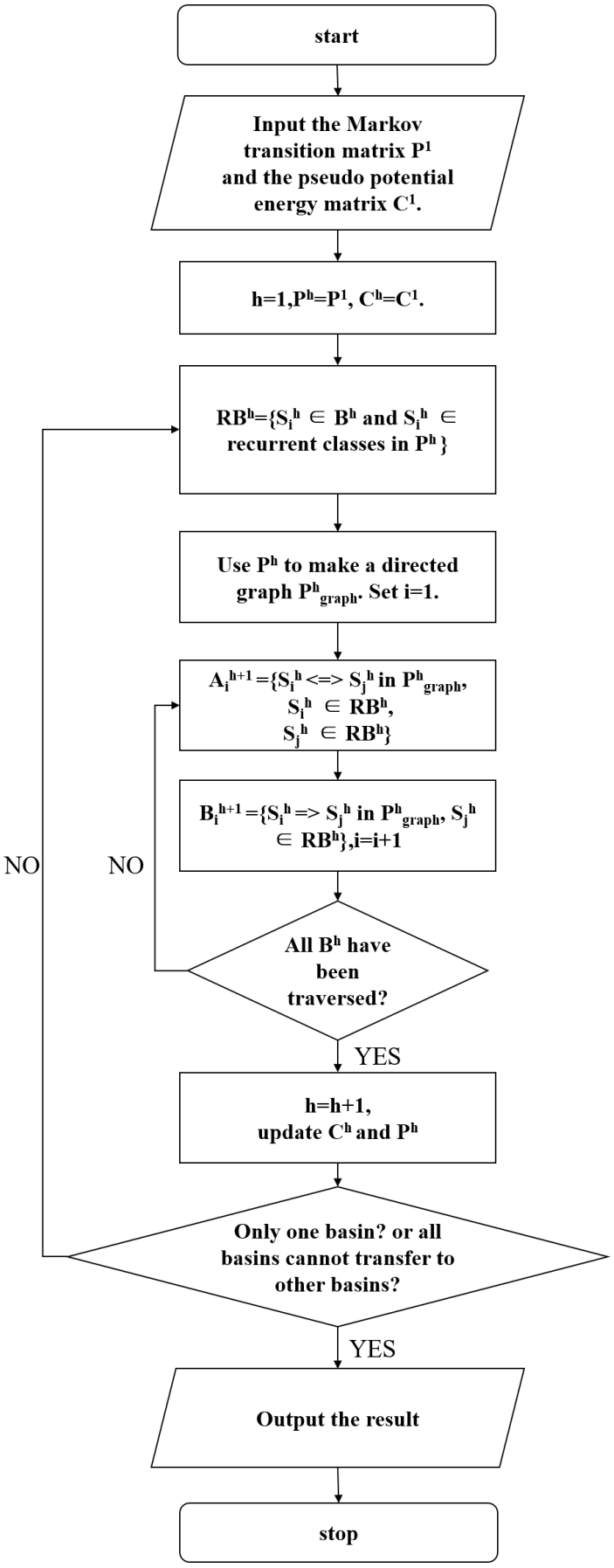
We can construct the higher-level clustering results in a similar way until the complete target hierarchical structure is finished. Actually, we just transform the information contained in matrix C into a hierarchical structure that can better show the intrinsic relations of different data points. A more detailed derivation, which contains some definitions and theorems regarding to metastability of exponentially perturbed Markov chain is in the Supplementary Notes (rigorous mathematical proofs can be found in the original paper[4] of Chen *et. al.*).

The pseudo code.

1 Input the Markov transition matrix P^1 and the pseudo-energy matrix C^1
2 $P^h = P^1$, $C^h = C^1$
3 while (TRUE){
#find attractors and partition basins
4 Find all basins belong to recurrent classes in P^h and we call them RB^h below
5 Use P^h to make a directed graph. In the graph, the vertexes are the basins on the current level and the edges are the transition probability between basins. Set i=1.
6 while (TRUE){
7 Select an untraversed recurrent vertex, say RB_i^h ,as the start vertex, find all reachable vertexes in the directed graph from RB_i^h and merge the nodes in the attractors of these reachable basins to be the attractor of the basin on the level h+1, denoted as A_i^{h+1}.
8 Let RB_i^h to be the target vertex, find all vertexes in the directed graph that can reach RB_i^h and merge the nodes in these basins to be the basin on the level h+1, denoted as B_i^{h+1}. Mark all basins contained in B_i^{h+1} as traversed. i=i+1.
9 if (all B^h have been traversed) {break}
}

```
10 h=h+1
#Update the pseudo-energy matrix on level h+1
   for(each basin pair on level h+1){
   The energy consumed by basin i transiting to basin j is equal to the minimum energy
     consumed by the nodes in basin i's attractor transiting to the nodes in basin j. Update
     the values in pseudo-energy matrix C^h }
#Update the Markov transition matrix P^h on level h+1
   Find the positions corresponding the values smaller than α% quantile of all values in
     C^h. α% is a customized parameter. P^h (positions) = 1, P^h (other positions) = 0, and
     convert P^h to be a Markov probability matrix by normalizing the sum of each row
     of it to 1.
11 if(there is only one basin or all basins cannot transfer to other basins){
        return the result
12    }
13 }
```

The flow chart.



To build the level h+1 in the hierarchical structure, we need to use the Markov transition matrix P^h. At first, we must find all basins belong to recurrent states in the Markov transition matrix P^h. As the mathematical proof in part B of Supplementary Notes, that if a state i is a recurrent state in the Markov transition matrix, then there must be an eigenvalue 1 with a corresponding eigenvector whose i th entry is non-zero. If a state i is a transient state, then for all eigenvalues 1, the i th entry of the corresponding eigenvector must be 0. In this way, we can quickly identify all recurrent states.

The nodes in the attractors of these recurrent state basins can reach each other with positive transition probabilities are merged to be the attractor of the basin on level h+1. For example, suppose there are 10 basins on level h, and $\{B_i^h\}, i=1,2,...,5$ are recurrent states. $\{A_i^h\}, i=1,2$ have a positive transition probability path to each other and $\{A_i^h\}, i=3,4,5$ have a positive transition probability path to each other. Then, the nodes in $\{A_i^h\}, i=1,2$ are merged to be $\{A_1^{h+1}\}$ and the nodes in $\{A_i^h\}, i=3,4,5$ are merged to be $\{A_2^{h+1}\}$. The nodes in other basins, which can reach recurrent state basins with positive transition probability but recurrent state basins cannot reach them, are partitioned to the recurrent state basins. For example, $\{B_i^h\}, i=6,7,8$ have positive transition probability path to $\{B_i^h\}, i=1,2$ but $\{B_i^h\}, i=1,2$ cannot reach them, then all nodes in $\{B_i^h\}, i=6,7,8$ are merged to $\{B_i^h\}, i=1,2$ as $\{B_1^{h+1}\}$.

For implementation convenience, the Markov transition matrix P^h is converted to a directed graph. In the graph, the vertexes are the basins on level h and the edges are the transition probability between basins. Let each recurrent state to be the start vertex to find all reachable vertexes from it. The attractors in these basins are merged to be the attractor of one basin on level h+1. For example, $\{A_1^{h+1}\}$.

Next, these basins are used as target vertexes to find all other basins have a positive transition probability path to them. All nodes in these basins are merged to $\{B_1^{h+1}\}$. After all the cells have been traversed, we can partition all nodes into basins on level h+1.

To build the level h+2, we need to calculate the Markov transition matrix P^{h+1} representing the transition probability among each basin on level h+1 first. And the matrix P^{h+1} is derived from C^{h+1}. On level h+1, the pseudo-energy from $\{B_1^{h+1}\}$ to $\{B_2^{h+1}\}$ is the minimum pseudo-energy from the nodes in $\{A_1^{h+1}\}$ to the nodes in $\{B_2^{h+1}\}$. In details, we find the minimum value in the matrix C^{h+1} at first and set 1 to the

corresponding positions in the Markov transition matrix P^{h+1}. Then, we convert the updated P^{h+1} to a Markov probability matrix by normalizing the sum of each row of it to 1. After we obtain the matrix P^{h+1}. The details of building the level h+2 are same as that building level h+1 based on P^h.

There is an approximation in the Markov transition matrix P^h (h>1) calculation. Strictly according to theory, only the positions, which corresponding positions in the C^{h-1} matrix where the value is strictly equal to the minimum, is set to 1 in P^h. To accelerate basins merging, the entries whose corresponding position in the C^{h-1} matrix having a value smaller than the α% quantile of all values in C^{h-1}, are set to be 1 in P^h. The α% is a customizable parameter.

There's another optional approximation in this algorithm, which can provide considerable acceleration. On each level, we can merge some small basins to its closest qualified basin when some big basins already form. Naturally, we can finish the construction of a hierarchical structure with fewer iterations. This approximation is built on the belief that if a big basin has already existed on some level, it indicates that we've reached a macroscopically meaningful level and hence basins of small sizes can be considered as noise basins. This approximation shows great algorithm efficiency gain and leads to similar results compared with the accurate version when there's no huge density discrepancy in data. However, when there's a huge density discrepancy in data, which means that some clusters have obviously larger density over other clusters, we don't recommend this approximation. This approximation requires three customized parameters to specify the size of noise basins, qualified basins, and big basins respectively. Our empirical recommendation is as follows. We regard basins consisting of less than 10 nodes as noise basins, basins consisting of more than 10 nodes as qualified basins and basins consisting of more than 20 percentage nodes of all as big basins.

Iterating the process above until there is only one basin left or all pseudo-energy consumed by the transition among basins is equal to infinity, which indicates all basins cannot transit to others definitely, the hierarchical structure of the input data is constructed completely.

Technical details of the MarkovHC algorithm.

Step1: Calculate sNN similarity matrix and construct a sNN network.

The sNN (shared Nearest Neighbors) between two points is the number of K nearest neighbor intersection, therefore sNN is a robust similarity measurement. Single-cell data is high dimensional and has serious dropout events, thus we used sNN to measure similarities among cells and computed a cell-by-cell sNN similarity matrix $s_{i,j}$ between node i and node j, which is also equal to the strength of connectivity between the two nodes:

$$s_{i,j} = \#\left\{neighbors \in KNN_i \cap KNN_j\right\}$$

where KNN_i is the K nearest neighbors of node i. Then we use the matrix to construct a network. The nodes of the network are cells and the edges of the network are the similarity among cells.

Step2: Calculate density score of each node.

Accurate density estimation is a challenging task in high dimensional data, especially in single-cell RNA sequencing data. Therefore, we use the degree of each node in sNN network to measure the node aggregation as our density score:

$$D_i = \sum_{j=1}^{n} a(i,j), a(i,j) = \begin{cases} 0, edges(i,j) = \varnothing \\ 1, edges(i,j) \neq \varnothing \end{cases}$$

Step3: Calculate the Markov transition matrix and the pseudo-energy matrix.

The transition probability from node i to node j is defined:

$$p_{i,j} = p(i \to j) = p(j|i) = \lim_{\beta \to \infty} \frac{e^{-\beta \frac{1}{s^{w1}_{i,j}} \left(\frac{D_i}{D_j}\right)^{w2}}}{\sum_{j=1}^{n} e^{-\beta \frac{1}{s^{w1}_{i,j}} \left(\frac{D_i}{D_j}\right)^{w2}}}$$

where w1 and w2 are constant parameters determining the tradeoff between the importance of similarity and density score, and both of them are set to 2 by default. For the convenience of calculation, we can derive the concise formula of p(j|i) :

$$denote\, g_{i,j} = \frac{1}{s^{w1}_{i,j}} \left(\frac{D_i}{D_j}\right)^{w2}, where\, s_{i,j}\, is\, the\, similarity\, of\, i\, and\, j,$$

$$D_i\, is\, the\, centrality\, score\, of\, i.$$

$$\begin{aligned} p(j|i) &= \lim_{\beta \to \infty} \frac{e^{-\beta g_{i,j}}}{\sum_{j=1}^{n} e^{-\beta g_{i,j}}} \\ &= \lim_{\beta \to \infty} \frac{e^{-\beta g_{i,j}} e^{\beta g_{i,0}}}{\sum_{j=1}^{n} e^{-\beta g_{i,j}} e^{\beta g_{i,0}}}, \text{where } g_{i,0} = \min(g_{i,j}), \text{j} = 1, \ldots, \text{n} \\ &= \lim_{\beta \to \infty} \frac{e^{-\beta g_{i,j} + \beta g_{i,0}}}{\sum_{j=1}^{n} e^{-\beta g_{i,j} + \beta g_{i,0}}} \\ &= \lim_{\beta \to \infty} \frac{e^{-\beta (g_{i,j} - g_{i,0})}}{\sum_{j=1}^{n} e^{-\beta (g_{i,j} - g_{i,0})}} \\ &= \begin{cases} 0, & g_{i,j} > g_{i,0} \\ m_i^{-1} \end{cases}, m_i = \{ \#\ of\ j,\ st\ g_{i,j} = g_{i,0} \} \end{aligned}$$

We denote the Markov transition matrix as P1 below.

The consumed pseudo-energy by the direct transition from node i to node j is:

$$c_{i,\text{j}} = \lim_{\beta \to \infty} -\frac{1}{\beta} \log p^{\beta}(j|i)$$

We also can derive the concise formula for calculation convenience:

$$
\begin{aligned}
c_{i,\mathrm{j}} &= \lim_{\beta\to\infty} -\frac{1}{\beta}\log p^{\beta}\left(j|i\right) \\
&= \lim_{\beta\to\infty} -\frac{1}{\beta}\log \frac{e^{-\beta g_{i,j}}}{\sum_{j=1}^{n} e^{-\beta g_{i,j}}} \\
&= g_{i,j} + \lim_{\beta\to\infty} \frac{1}{\beta}\log \sum_{j=1}^{n} e^{-\beta g_{i,j}} \\
&= g_{i,j} + \lim_{\beta\to\infty} \frac{\sum_{j=1}^{n} -g_{i,j} e^{-\beta g_{i,j}}}{\sum_{j=1}^{n} e^{-\beta g_{i,j}}} \\
&= g_{i,j} + \lim_{\beta\to\infty} \frac{\sum_{j=1}^{n} -g_{i,j} e^{-\beta g_{i,j}} e^{-\beta g_{i,0}}}{\sum_{j=1}^{n} e^{-\beta g_{i,j}} e^{-\beta g_{i,0}}}, \; where\, g_{i,0} = \min(g_{i,j}), \mathrm{j} = 1,\ldots,\mathrm{n} \\
&= g_{i,j} + \frac{\sum_{0} -g_{i,0}}{\#\left\{m_i : g_{i,0} = \min(\mathrm{g}_{i,j}), \mathrm{j} = 1,2,\ldots,\mathrm{n}\right\}} \\
&= g_{i,j} - g_{i,0}
\end{aligned}
$$

This pseudo-energy matrix represents the pseudo-energy cost when making a direct transition between nodes. We can calculate geodesics (min-energy path) on the surface of this "energy landscape" by using Dijkstra algorithm[57]. If node i cannot transit to node j, the corresponding entry in the matrix is set to be infinite. We denote this reduced pseudo-energy matrix as C1 below.

Step4: Pre-clustering.

To accelerate the speed of steps below, we can perform pre-clustering at the first stage to cluster all nodes into hundreds of small clusters. These small clusters are homogenous. The number of rough clusters is a customizable parameter in our tools and there are five optional pre-clustering methods, which are k-means, single linkage hierarchical clustering, complete linkage hierarchical clustering, average linkage clustering, and Louvain clustering. The mean density score of the nodes in the small cluster is used as the density score of the small cluster. The number of shared nearest neighbors among nodes is defined as the similarity of them, and the maximum similarity among nodes in two small clusters is used to define the similarity between cluster m and cluster n:

$$
s_{m,n} = \max(s_{i,j}), i \in cluster_m, j \in cluster_n
$$

Users could skip this pre-clustering step, but we recommend this for two reasons. First, such preliminary clustering does no harm and won't largely change the upper-level hierarchical structure, as long as the number of clusters is large enough after this step. Second, pre-clustering can significantly reduce computational time. By default,

we cluster all nodes into 200 small clusters using k-means.

If pre-clustering in this step was performed, we regard small clusters as nodes. For convenience, we still call these 'small clusters' as 'nodes' below.

Step5: Find the attractors and partition the basins of each level to build the hierarchical structure.

Firstly, we explain concepts of attractor, basin, the Markov transition matrix P, and the pseudo-energy matrix C. Attractors and basins are sets of nodes. And basins also can be regarded as clusters and attractors can be regarded as cores of clusters. Matrix P represents the transition probability between basins, and the pseudo-energy matrix represents the energy cost by making the transition between the basins.

On level 1, every node is an attractor and also is a basin according to a mathematical proof (Definition1.1 and Definition1.2 of part A in Supplementary Notes). Namely, there are n attractors and n basins for n nodes data. And there is only one node in each basin. Therefore, level 1 has been built directly according to this conclusion. The attractors are denoted as $\{A_i^1\}, i=1,2,...,\mathrm{n}$, the basins are denoted as $\{B_i^1\}, i=1,2,...,\mathrm{n}$, the $\mathrm{n}\times\mathrm{n}$ Markov transition matrix is denoted as P^1 that represents the transition probability between $\{B_i^1\}, i=1,2,...,\mathrm{n}$, and the $\mathrm{n}\times\mathrm{n}$ pseudo-energy matrix is denoted as C^1 representing the pseudo-energy consumed by transition among $\{B_i^1\}, i=1,2,...,\mathrm{n}$.

On a higher level, such as level h, the attractors are denoted as $\{A_i^h\}, i=1,2,...,\mathrm{n}^h$, the basins are denoted as $\{B_i^h\}, i=1,2,...,\mathrm{n}^h$, the $\mathrm{n}^\mathrm{h}\times\mathrm{n}^\mathrm{h}$ Markov transition matrix is denoted as P^h that represents the transition probability between $\{B_i^h\}, i=1,2,...,\mathrm{n}^h$, and the $\mathrm{n}^\mathrm{h}\times\mathrm{n}^\mathrm{h}$ pseudo-energy matrix is denoted as C^h representing the energy cost by transition among $\{B_i^h\}, i=1,2,...,\mathrm{n}^h$.

Comparison with other methods.

It should be noted that there is a resolution parameter in Seurat which is critical to the results, with increased values leading to more clusters. For unknown data, users can only adjust this parameter based on experience and visualization of data. In the process of our evaluation, we fine-tuned this parameter in Seurat to cluster points into the same number of clusters as that in the ground truth and used the results as the best results of Seurat. For comparison, we also used the default parameters of Seurat to get the default results. As for methods that need to set the number of clusters, we set it equal to that in the ground truth. And the results of MarkovHC were taken from the level with the equal number of basins as that in the ground truth.

Level selection strategy.

Selecting levels appropriately from the cluster hierarchy is important for identifying meaningful cell basins.

For users who have enough prior knowledge about data, we recommend a top-down strategy to do it. Firstly, users can identify cell basins on high levels according to marker genes expression. Secondly, users should track the ambiguous basins from high levels to lower levels and identify the sub-basins by marker genes. Lastly, meaningful basins and the relationships among these basins can be extracted from the hierarchical structure by this strategy.

For users who do not have enough prior knowledge about their data, we suggest them to use the plot of "pseudo-energy vs level plot" to assist level selection. As the level increases, the threshold of pseudo-energy cost for basin transition increases accordingly. We divide the "pseudo-energy vs level plot" into two regimes, a noise regime and a signal regime. Low-level results could be swamped by a lot of noises, so the pseudo-energy change slowly in the noise regime. But the pseudo-energy dramatically changes in the signal regime for basins on the high-level may correspond to big cell clusters with biological significance. So levels around the boundary of the noise regime and the signal regime may have biological meaning. We showed the "pseudo-energy vs level plots" of the simulation data in Fig. 2F (Supplementary Fig. 21A) and four real data sets (Supplementary Fig. 21B. data in Fig. 4; C. data in Fig. 5; D. data in Fig. 6; E. data in Supplementary Fig. 20) used in this paper.

Find the transition paths and critical points.

With the help of Markov chain's directivity, we can determine the transition path, which is the most probable path supported by large deviations theory[86]. And the transition path from basin A to basin B constituted by points on the shortest geodesic path from A's attractors to B on the pseudo energy landscape. We employed Dijkstra algorithm to find the shortest path in implement. After the transition path was obtained, the two closest points between basin A and B on the path were determined as the critical points from A to B.

Differentially expressed genes and gene ontology term enrichment analysis.

Wilcoxon rank sum test in Seurat (v.3.2.1) was employed to detect differentially expressed genes among cell groups. Differentially expressed genes along the transition path were identified by differentialGeneTest function in monocle (v.2.14.0)[16]. And GO terms of gene sets were enriched by enrichGO function in clusterProfiler (v.3.14.3)[87].

Data availability

The links for downloading datasets used in this paper are in the part G of the Supplementary Notes.

Code availability

R package "MarkovHC" and codes for analysis in this paper are freely available at GitHub (https://github.com/ZhenyiWangTHU/MarkovHC).

References

1 Moris, N., Pina, C. & Arias, A. M. Transition states and cell fate decisions in epigenetic landscapes. *Nat Rev Genet* **17**, 693-703 (2016).

2 Chen, Z. *et al.* DensityPath: an algorithm to visualize and reconstruct cell state-transition path on density landscape for single-cell RNA sequencing data. *Bioinformatics* **35**, 2593-2601 (2019).

3 Qian, H. Cycle kinetics, steady state thermodynamics and motors-a paradigm for living matter physics. *J Phys Condens Matter* **17**, 3783-3794 (2005).

4 Chen, D., Feng, J. & Qian, M. Metastability of exponentially perturbed Markov chains. *Science in China Series A-Mathematics* **39**, 7-28 (1996).

5 Wasserman, L. Topological data analysis. *Annual Review of Statistics and Its Application* **5**, 501-532 (2018).

6 Chu, L. F. *et al.* Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome Biol* **17**, 173 (2016).

7 Petropoulos, S. *et al.* Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. *Cell* **165**, 1012-1026 (2016).

8 Zhang, P. *et al.* Dissecting the Single-Cell Transcriptome Network Underlying Gastric Premalignant Lesions and Early Gastric Cancer. *Cell Rep* **27**, 1934-1947 (2019).

9 Huang, S., Eichler, G., Bar-Yam, Y. & Ingber, D. E. Cell fates as high-dimensional attractor states of a complex gene regulatory network. *Phys Rev Lett* **94**, 128701 (2005).

10 Wright, S. The roles of mutation, inbreeding, crossbreeding and selection in evolution, In *Proceedings of the Sixth International Congress of Genetics*, 356–366 (1932).

11 Waddington, C. H. *Principles of Development and Differentiation.* (MacMillan., New York, 1966).

12 Waddington, C. H. *The Strategy of the Genes Routlege Library Editions: 20th Centure Science.* (George Allen & Unwin Ltd., London, 1957).

13 Ao, P. Global view of bionetwork dynamics: adaptive landscape. *Journal of Genetics and Genomics* **36**, 63-73 (2009).

14 Schiebinger, G. *et al.* Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming. *Cell* **176**, 928-943 (2019).

15 Qiu, P. *et al.* Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nat Biotechnol* **29**, 886-891 (2011).

16 Trapnell, C. *et al.* The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol* **32**, 381-386 (2014).

17 La Manno, G. *et al.* RNA velocity of single cells. *Nature* **560**, 494-498 (2018).

18 Bergen, V., Lange, M., Peidli, S., Wolf, F. A. & Theis, F. J. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nat Biotechnol* **38**, 1408-1414, (2020).

19 Kiselev, V. Y. *et al.* SC3: consensus clustering of single-cell RNA-seq data. *Nat Methods* **14**, 483-486 (2017).

20 Efremova, M., Vento-Tormo, M., Teichmann, S. A. & Vento-Tormo, R. CellPhoneDB: inferring cell-cell communication from combined expression of multi-subunit ligand-receptor complexes. *Nat Protoc* **15**, 1484-1506 (2020).

21 Wasserman, L. Topological Data Analysis. *Annual Review of Statistics and Its Application* **5**, 501-532 (2018).

22 van Dijk, D. *et al.* Recovering Gene Interactions from Single-Cell Data Using Data Diffusion. *Cell* **174**, 716-729 (2018).

23 Haghverdi, L., Buettner, F. & Theis, F. J. Diffusion maps for high-dimensional single-cell analysis of differentiation data. *Bioinformatics* **31**, 2989-2998 (2015).

24 Zappia, L. & Oshlack, A. Clustering trees: a visualization for evaluating clusterings at multiple resolutions. *Gigascience* **7**, 274035 (2018).

25 Schwartz, G. W. *et al.* TooManyCells identifies and visualizes relationships of single-cell clades. *Nat Methods* **17**, 405-413 (2020).

26 Alcocer-Cuaron, C., Rivera, A. L. & Castano, V. M. Hierarchical structure of biological systems: a bioengineering approach. *Bioengineered* **5**, 73-79 (2014).

27 Ester, M ., Kriegel, H.-P ., Sander, J. & Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. *In Proc. 2nd Int. Conf. Knowledge, Discovery and Data Mining (KDD-96)* (eds Simoudis, E . et al.) 226–231 (AAAI, 1996)

28 Mihael Ankerst, M. M. B., Hans-Peter Kriegel, Jörg Sander. OPTICS: Ordering Points To Identify the Clustering Structure. (SIGMOD, Philadelphia, 1999).

29 Campello, R. J. G. B., Moulavi, D., Zimek, A. & Sander, J. Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection. *ACM Transactions on Knowledge Discovery from Data* **10**, 1-51 (2015).

30 Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* **36**, 411-420 (2018).

31 Sibson, R. SLINK: an optimally efficient algorithm for the single-link cluster method. *The computer journal* **16**, 30-34 (1973).
32 Sokal, R. R. A statistical method for evaluating systematic relationships. *Univ. Kansas, Sci. Bull.* **38**, 1409-1438 (1958).
33 Defays, D. An efficient algorithm for a complete link method. *The Computer Journal* **20**, 364-366 (1977).
34 Wang, B., Zhu, J., Pierson, E., Ramazzotti, D. & Batzoglou, S. Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat Methods* **14**, 414-416 (2017).
35 Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008**, P10008 (2008).
36 Rodriguez, M. Z. *et al.* Clustering algorithms: A comparative approach. *PLoS One* **14**, e0210236 (2019).
37 Kiselev, V. Y., Andrews, T. S. & Hemberg, M. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat Rev Genet* **20**, 273-282 (2019).
38 Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* **19**, 15 (2018).
39 Lin, P., Troup, M. & Ho, J. W. CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol* **18**, 59 (2017).
40 Zurauskiene, J. & Yau, C. pcaReduce: hierarchical clustering of single cell transcriptional profiles. *BMC Bioinformatics* **17**, 140 (2016).
41 Chen, J., Schlitzer, A., Chakarov, S., Ginhoux, F. & Poidinger, M. Mpath maps multi-branching single-cell trajectories revealing progenitor cell progression during development. *Nat Commun* **7**, 11988 (2016).
42 Guo, M., Wang, H., Potter, S. S., Whitsett, J. A. & Xu, Y. SINCERA: A Pipeline for Single-Cell RNA-Seq Profiling Analysis. *PLoS Comput Biol* **11**, e1004575 (2015).
43 Ertöz, L., Steinbach, M. & Kumar, V. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In *Proceedings of the 3rd SIAM International Conference on Data Mining*, 47-58 (2003).
44 Bhargav, S. & Pawar, M. A review of clustering methods forming non-convex clusters with missing and noisy data. *IJCSE*, 39-44 (2016).
45 Hess, S., Duivesteijn, W., Honysz, P. & Morik, K. The SpectACl of nonconvex clustering: a spectral approach to density-based clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3788-3795 (2019).
46 Van Dongen, S. M. *Graph Clustering by Flow Simulation.* PhD thesis, Univ. Utrecht (2000).
47 Qiu, X. *et al.* Reversed graph embedding resolves complex single-cell trajectories. *Nat Methods* **14**, 979-982 (2017).
48 Cao, J. *et al.* The single-cell transcriptional landscape of mammalian organogenesis. *Nature* **566**, 496-502 (2019).

49 Farrell, J. A. *et al.* Single-cell reconstruction of developmental trajectories during zebrafish embryogenesis. *Science* **360**, 6392 (2018).
50 Bendall, S. C. *et al.* Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. *Cell* **157**, 714-725 (2014).
51 Levine, J. H. *et al.* Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis. *Cell* **162**, 184-197 (2015).
52 Shekhar, K. *et al.* Comprehensive Classification of Retinal Bipolar Neurons by Single-Cell Transcriptomics. *Cell* **166**, 1308-1323 (2016).
53 Setty, M. *et al.* Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nat Biotechnol* **34**, 637-645 (2016).
54 Haghverdi, L., Buttner, M., Wolf, F. A., Buettner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nat Methods* **13**, 845-848 (2016).
55 Huang, S., Li, F., Zhou, J. X. & Qian, H. Processes on the emergent landscapes of biochemical reaction networks and heterogeneous cell population dynamics: differentiation in living matters. *J R Soc Interface* **14**, 20170097 (2017).
56 Ao, P., Galas, D., Hood, L. & Zhu, X. Cancer as robust intrinsic state of endogenous molecular-cellular network shaped by evolution. *Med Hypotheses* **70**, 678-684 (2008).
57 Dijkstra, E. W. A note on two problems in connexion with graphs. *Numerische mathematik* **1**, 269-271 (1959).
58 Jia, C., Qian, M., Kang, Y. & Jiang, D. Modeling stochastic phenotype switching and bet-hedging in bacteria: stochastic nonlinear dynamics and critical state identification. *Quantitative Biology* **2**, 110-125 (2015).
59 Mojtahedi, M. *et al.* Cell Fate Decision as High-Dimensional Critical State Transition. *PLoS Biol* **14**, e2000640 (2016).
60 Anchang, B. *et al.* Visualization and cellular hierarchy inference of single-cell data using SPADE. *Nat Protoc* **11**, 1264-1279 (2016).
61 Andrews, T. S. & Hemberg, M. Identifying cell populations with scRNASeq. *Mol Aspects Med* **59**, 114-122 (2018).
62 Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. A density-based algorithm for discovering clusters. *KDD-96 Proc*. 226-231 (1996).
63 Jeitziner, R. *et al.* Two-Tier Mapper, an unbiased topology-based clustering method for enhanced global gene expression analysis. *Bioinformatics* **35**, 3339-3347 (2019).
64 Wolf, F. A. *et al.* PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biol* **20**, 59 (2019).
65 Zappia, L., Phipson, B. & Oshlack, A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biol* **18**, 174 (2017).
66 Packer, J. S. *et al.* A lineage-resolved molecular atlas of C. elegans embryogenesis at single-cell resolution. *Science* **365**, eaax1971 (2019).

67 Papalexi, E. & Satija, R. Single-cell RNA sequencing to explore immune cell heterogeneity. *Nat Rev Immunol* **18**, 35-45 (2018).
68 Villani, A. C. *et al.* Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science* **356**, eaah4573 (2017).
69 Bendall, S. C. *et al.* Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science* **332**, 687-696 (2011).
70 Zeng, W. *et al.* DC3 is a method for deconvolution and coupled clustering from bulk and single-cell genomics data. *Nat Commun* **10**, 4613 (2019).
71 Regev, A. *et al.* The Human Cell Atlas. *Elife* **6**, e27041 (2017).
72 Moon, K. R. *et al.* Visualizing Structure and Transitions for Biological Data Exploration. Preprint at https://www.biorxiv.org/content/10.1101/120378v3 (2018).
73 Cheng, J. & Fan, X. M. Role of cyclooxygenase-2 in gastric cancer development and progression. *World J Gastroenterol* **19**, 7361-7368 (2013).
74 Boussioutas, A. *et al.* Distinctive patterns of gene expression in premalignant gastric mucosa and gastric cancer. *Cancer Res* **63**, 2569-2577 (2003).
75 Alzahrani, S. *et al.* Effect of Helicobacter pylori on gastric epithelial cells. *World J Gastroenterol* **20**, 12767-12780 (2014).
76 Yuan, W., Yang, N. & Li, X. Advances in Understanding How Heavy Metal Pollution Triggers Gastric Cancer. *Biomed Res Int* **2016**, 7825432 (2016).
77 van der Flier, L. G., Haegebarth, A., Stange, D. E., van de Wetering, M. & Clevers, H. OLFM4 Is a Robust Marker for Stem Cells in Human Intestine and Marks a Subset of Colorectal Cancer Cells. *Gastroenterology* **137**, 15-17 (2009).
78 Wong, C.-M. *et al.* CEACAM6 Promotes Gastric Cancer Invasion and Metastasis by Inducing Epithelial-Mesenchymal Transition via PI3K/AKT Signaling Pathway. *PLoS ONE* **9**, e112908 (2014).
79 Busuttil, R. A. *et al.* Role of p53 in the progression of gastric cancer. *Oncotarget* **5**, 12016-12026 (2014).
80 Jiaqing, L. *et al.* Role of cyclin E and p53 expression in progression of early gastric cancer. *Gastric Cancer* **1**, 160-165 (1998).
81 Chen, Y. *et al.* MiR-596 down regulates SOX4 expression and is a potential novel biomarker for gastric cancer. *Translational Cancer Research* **9**, 1294-1302 (2020).
82 Wang, C. L., Wang, D., Yan, B. Z., Fu, J. W. & Qin, L. Long non-coding RNA NEAT1 promotes viability and migration of gastric cancer cell lines through up-regulation of microRNA-17. *Eur Rev Med Pharmacol Sci* **22**, 4128-4137 (2018).
83 Ji, Z. & Ji, H. Pseudotime Reconstruction Using TSCAN. *Methods Mol Biol* **1935**, 115-124 (2019).
84 Cao, K., Bai, X., Hong, Y. & Wan, L. Unsupervised Topological Alignment for Single-Cell Multi-Omics Integration. Preprint at https://www.biorxiv.org/content/10.1101/2020.02.02.931394v1 (2020).

85 Kang, Z., Peng, C. & Cheng, Q. Kernel-driven similarity learning. *Neurocomputing* **267**, 210-219 (2017).
86 Varadhan, S. R. Asymptotic Probabilities and Differential Equations. *Commun Pur Appl Math* **19**, 261-286 (1966).
87 Yu, G., Wang, L. G., Han, Y. & He, Q. Y. clusterProfiler: an R package for comparing biological themes among gene clusters. *OMICS* **16**, 284-287 (2012).

Acknowledgements

This work was supported by Major Program of National Natural Science Foundation of China (81890991), National Key Research and Development Program of China (2018YFA0801402), to M.Q.Z., Major Program of National Natural Science Foundation of China (81890994) to Y.C.. We thank Dr. Peng Zhang, Yuyao Song and Dr. Kui Hua of Tsinghua University for helpful discussion and suggestion.

Author contributions

Conceptualization: M.P.Q., M.Q.Z., Z.Y.W. and Y.J.Z.;
Methodology: Z.Y.W. and Y.J.Z.;
Software: Z.Y.W.;
Formal Analysis: Z.Y.W.;
Investigation: Z.Y.W., Z.F.Y., L.Z. and M.L.S.;
Resources: M.Q.Z.;
Data Curation: Z.Y.W. and L.Z.;
Writing – Original Draft: Z.Y.W.;
Writing – Review & Editing: M.Q.Z., Z.F.Y., Z.Y.W., Y.J.Z., L.Z., Y.C., M.L.S. and M.P.Q.;
Visualization: Z.F.Y. and Z.Y.W.;
Supervision: M.Q.Z. and M.P.Q.;
Project Administration: M.Q.Z.;
Funding Acquisition: M.Q.Z.

Competing Interests

The authors declare no competing interests.

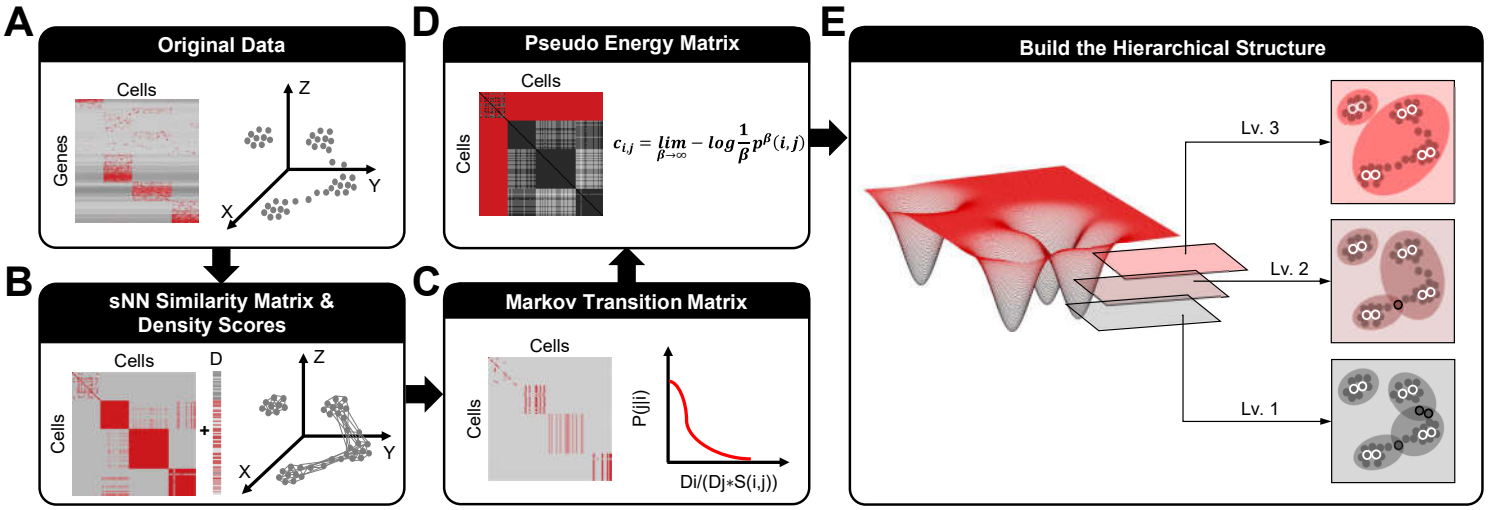


Figure 1. Overview of MarkovHC.

(A) The original input data is the matrix of genes by cells.

(B) We calculate sNN (shared Nearest Neighbors) among cells to get the cell by cell similarity matrix. Then we construct a cellular network using the similarity matrix and calculate each cell's degree (D score) in the network.

(C) The Markov transition matrix is calculated using the similarity matrix and D scores.

(D) The pseudo-energy matrix is calculated based on the Markov transition matrix. (see Methods)

(E) The hierarchical structure is constructed based on attractors, basins, and critical points on each level.

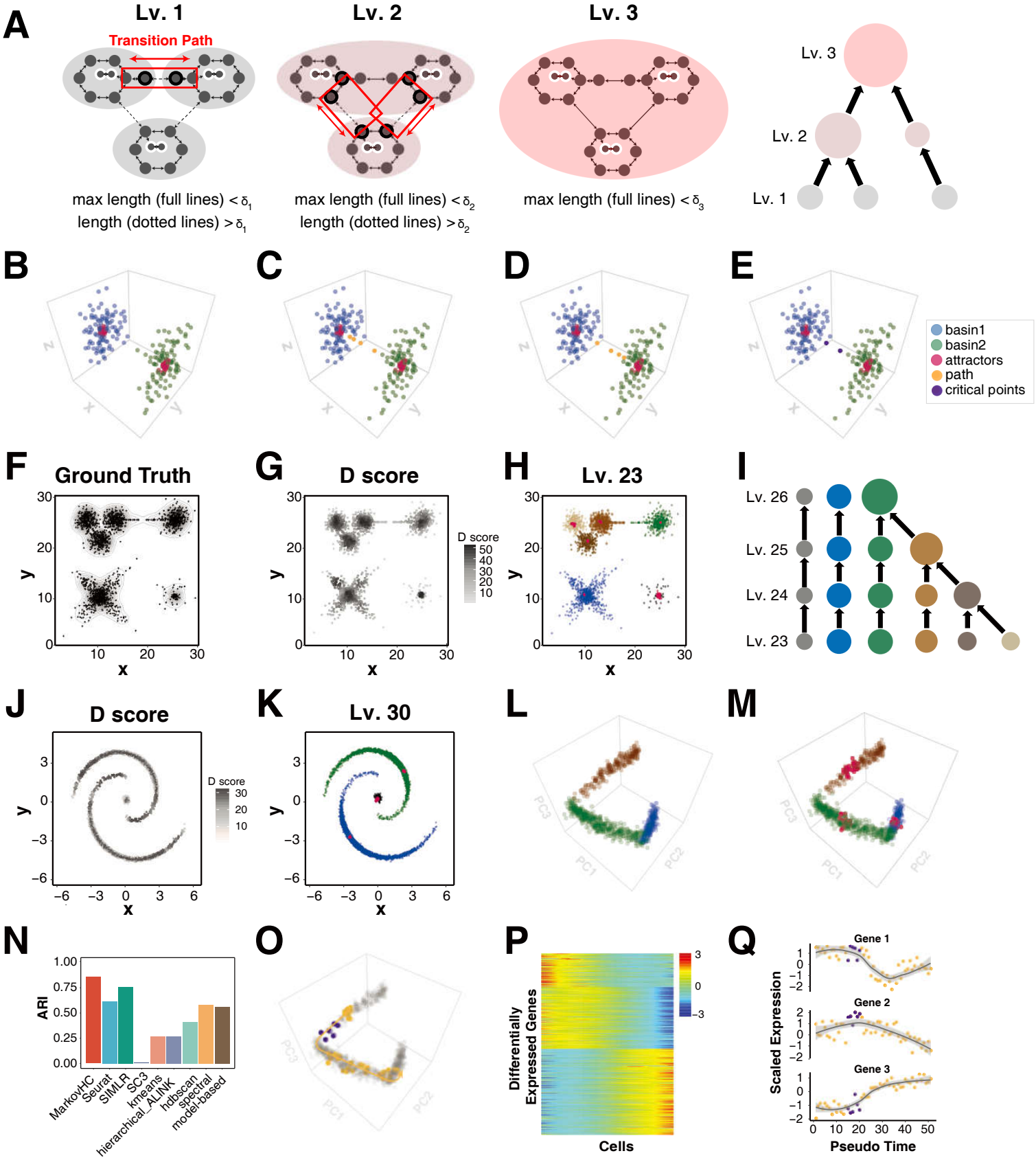

A
Lv. 1
Transition Path
Lv. 2
Lv. 3
max length (full lines) < δ1
length (dotted lines) > δ1
max length (full lines) < δ2
length (dotted lines) > δ2
max length (full lines) < δ3
Lv. 3
Lv. 2
Lv. 1
B
C
D
E
basin1
basin2
attractors
path
critical points
F
Ground Truth
G
D score
D score
H
Lv. 23
I
Lv. 26
Lv. 25
Lv. 24
Lv. 23
J
D score
D score
K
Lv. 30
L
M
N
ARI
MarkovHC
Seurat
SIMLR
SC3
kmeans
hierarchical_ALINK
hdbscan
spectral
model-based
O
P
Differentially Expressed Genes
Cells
Q
Gene 1
Gene 2
Gene 3
Scaled Expression
Pseudo Time

Figure 2. MarkovHC is able to cluster non-convex data, hetero-density data, and differentiation stages in a continuum.

(A)The points that can be connected using lines (pseudo-energy) no longer than δn are partitioned into a basin on level n. In a basin, points in the highest density regions constitute attractors (points with white border). Points connecting basins' attractors constitute the transition path (red box), and the closest points (black border) between these two basins on the transition path are the critical points.

(B-E)These two group points are clustered into two basins (blue and green) by MarkovHC (B). To insight more details of these two basins, we highlighted the attractors (red), the transition paths (yellow) and critical points (purple) respectively. The points linking basin A's attractors and basin B's border constitute the transition path from A to B and vice versa (C, D). The transition path between A and B are the points linking their attractors, and the critical points which indicate the points jump at the boundary of these two basins are the two closest points belong to them on the transition path (E).

(F) The 2-dimensional simulated data with density contours.

(G) D score of each point.

(H) The basins and attractors on level 23 were plotted to show the 2-dimensional layouts. And we observed that MarkovHC can simultaneously detect non-convex basins and hetero-density basins.

(I) The hierarchical structure of these data. On this hierarchical structure, the sizes of basins represent the number of samples and the colors represent different basins.

(J) D score of each point.

(K) Blue, green and black points indicate three basins. And red points show the attractors of each basin.

(L) Simulated single-cell data in three stages (generated by splatter[68]). These data included 1000 cells belonging to three stages in the 5000 genes space and these cells were projected into principle component space.

(M) MarkovHC detected 3 basins on level 17 (brown, green, and blue) and the red points in each basin indicated the attractors.

(N) ARI (adjust rand index) was used to evaluate the clustering accuracy of MarkovHC, Seurat, SIMLR, sc3, k-means, hierarchical clustering with average linkage, HDBSCAN, spectral clustering, and model-based, which showed that MarkovHC performed best on these data.

(O) The orange points indicated the points on the transition path (orange line) from the brown basin to the blue basin. The purple points indicated the critical points.

(P) The heatmap of differentially expressed genes along the transition path in (O).

(Q) Three representative genes with different "gene-flow" trends in (P). The purple points indicated the critical points.

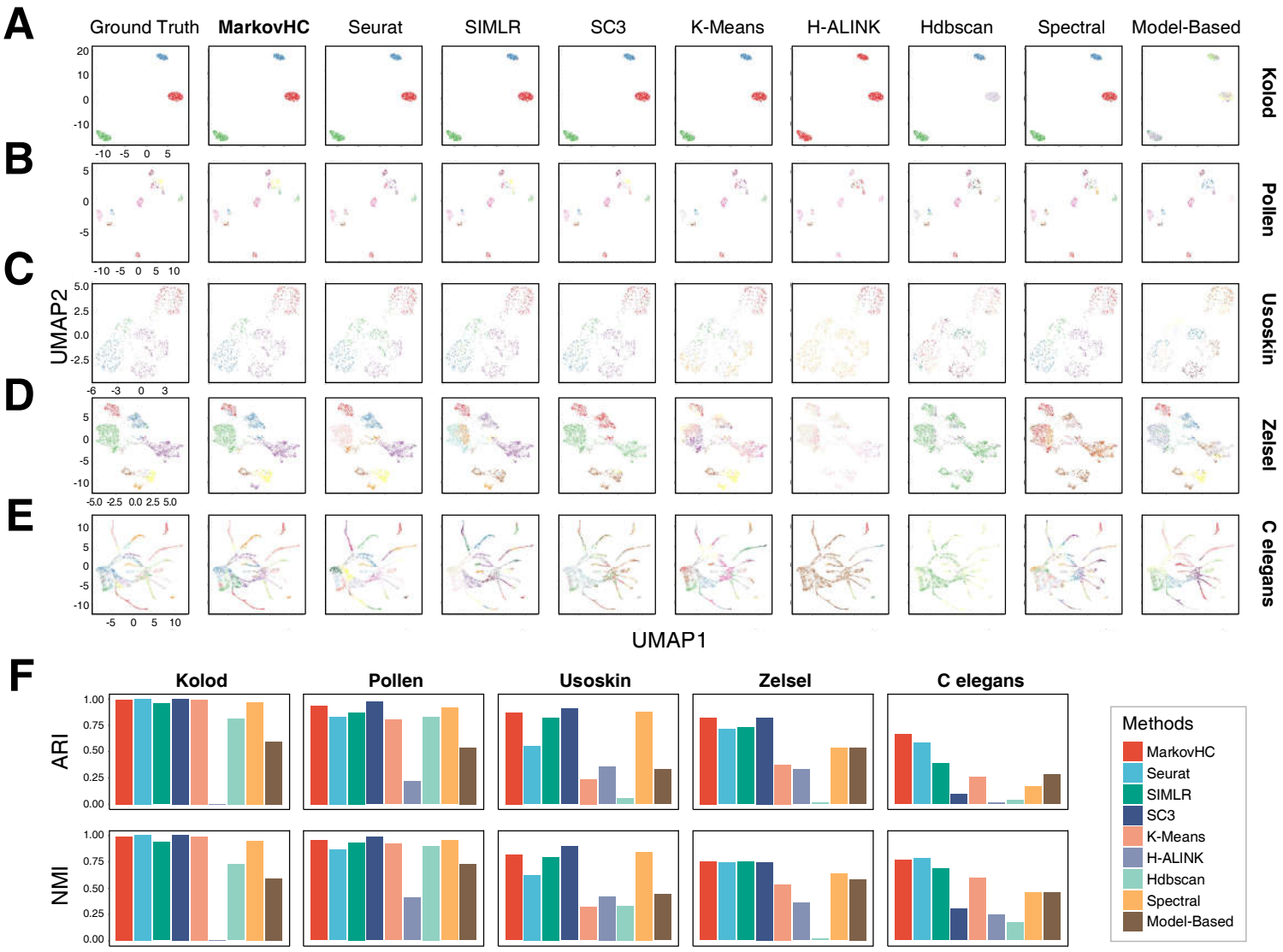


Figure3. MarkovHC can consistently achieve accuracy in clustering diverse types of single-cell omics data.

(A-E) Ground truths of five benchmark single-cell RNA-Seq data sets[34,66] and clustering results of MarkovHC, Seurat, SIMLR, SC3, k-means, hierarchical clustering with average linkage, HDBSCAN, spectral clustering, and EM.

(F) ARI (Rand Adjust Index) and NMI (Normalized Mutual Information) showed MarkovHC can consistently achieve top accuracy in clustering real data.

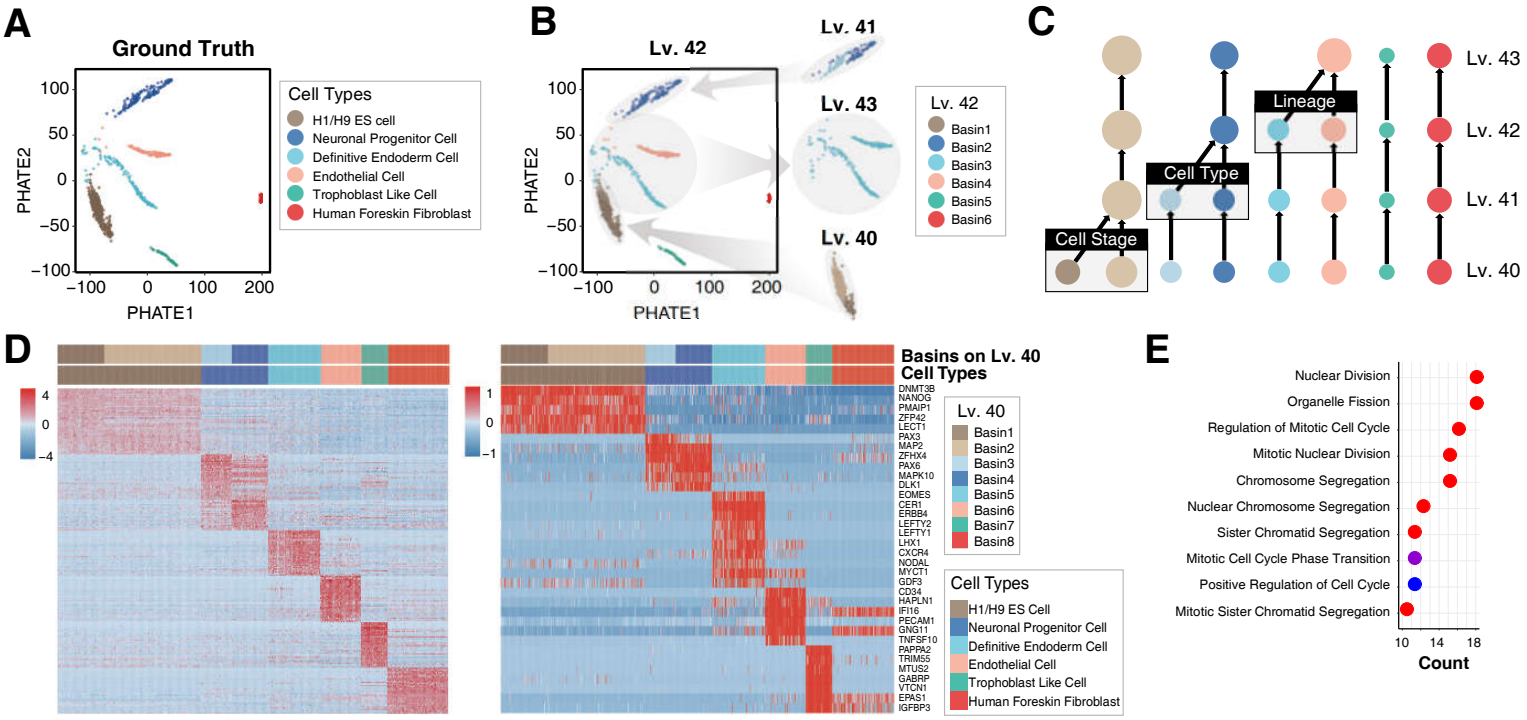


Figure 4. MarkovHC's hierarchical structure revealed cell lineages, cell types, and cell stages simultaneously.

(A) Single-cell RNA-Seq data6 of human ES cell-derived lineage-specific progenitors were projected into 2-dimensional space by phateR. And seven cell types are identified using marker genes.

(B) On level 40, two different cell stages were found in H1 and H9 ES cells. On level 41, two cell types in the neuronal progenitor cells were found. On level 42, cells of different lineages were separated.

(C) On the hierarchical structure, the difference among basins on level 42 might correspond to different cell lineages, while that among basins on level 41 and 40 might correspond to cell types and cell stages

(D) The expression pattern of the 50 most significantly up-regulated genes in each basin (left) and 34 selected marker genes (right).

(E) The differentially expressed genes between basin1 and basin2 on level 40 in H1 and H9 ES cells were enriched in cell division related GO terms, which suggested that these two cell stages relate to the cell cycle.

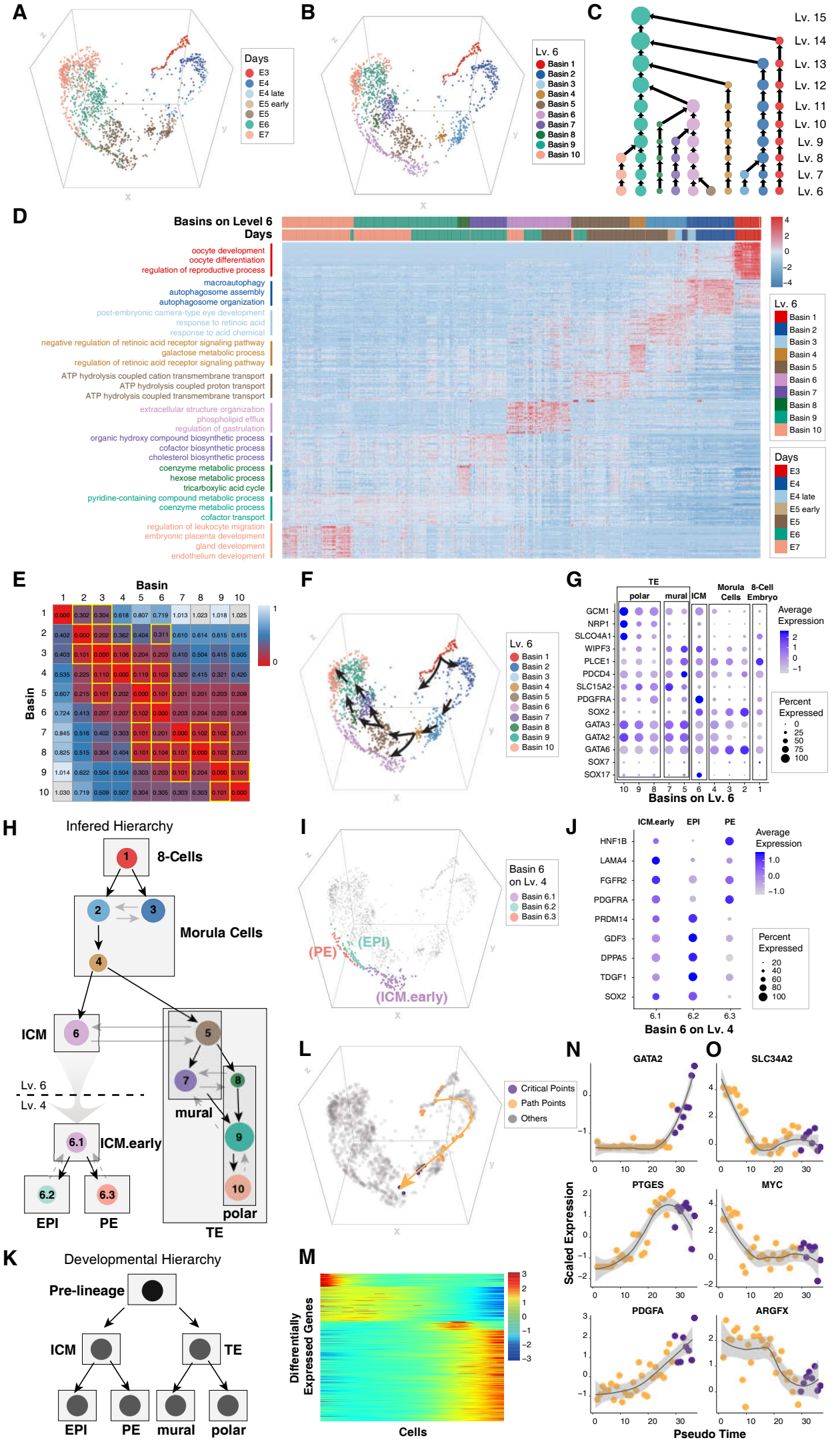

A
Days
E3
E4
E4 late
E5 early
E5
E6
E7
B
Lv. 6
Basin 1
Basin 2
Basin 3
Basin 4
Basin 5
Basin 6
Basin 7
Basin 8
Basin 9
Basin 10
C
Lv. 15
Lv. 14
Lv. 13
Lv. 12
Lv. 11
Lv. 10
Lv. 9
Lv. 8
Lv. 7
Lv. 6
D
Basins on Level 6
Days
oocyte development
oocyte differentiation
regulation of reproductive process
macroautophagy
autophagosome assembly
autophagosome organization
post-embryonic camera-type eye development
response to retinoic acid
response to acid chemical
negative regulation of retinoic acid receptor signaling pathway
galactose metabolic process
regulation of retinoic acid receptor signaling pathway
ATP hydrolysis coupled cation transmembrane transport
ATP hydrolysis coupled proton transport
ATP hydrolysis coupled transmembrane transport
extracellular structure organization
phospholipid efflux
regulation of gastrulation
organic hydroxy compound biosynthetic process
cofactor biosynthetic process
cholesterol biosynthetic process
coenzyme metabolic process
hexose metabolic process
tricarboxylic acid cycle
pyridine-containing compound metabolic process
coenzyme metabolic process
cofactor transport
regulation of leukocyte migration
embryonic placenta development
gland development
endothelium development
E
Basin
F
G
TE
polar
mural
ICM
Morula Cells
8-Cell Embryo
GCM1
NRP1
SLCO4A1
WIPF3
PLCE1
PDCD4
SLC15A2
PDGFRA
SOX2
GATA3
GATA2
GATA6
SOX7
SOX17
Average Expression
Percent Expressed
Basins on Lv. 6
H
Infered Hierarchy
8-Cells
Morula Cells
ICM
Lv. 6
Lv. 4
ICM.early
mural
polar
EPI
PE
TE
I
Basin 6 on Lv. 4
Basin 6.1
Basin 6.2
Basin 6.3
(PE)
(EPI)
(ICM.early)
J
ICM.early
EPI
PE
HNF1B
LAMA4
FGFR2
PDGFRA
PRDM14
GDF3
DPPA5
TDGF1
SOX2
Basin 6 on Lv. 4
L
Critical Points
Path Points
Others
N
GATA2
O
SLC34A2
PTGES
MYC
PDGFA
ARGFX
Scaled Expression
Pseudo Time
K
Developmental Hierarchy
Pre-lineage
ICM
TE
EPI
PE
mural
polar
M
Differentially Expressed Genes
Cells

Figure 5. MarkovHC revealed the transition paths and critical points in human preimplantation embryo development.

(A) The scRNA-Seq data[7] of 1529 human preimplantation embryos cells from E3 stage to E7 stage were projected into 2-dimensional space by phateR. In the data, E3–E7 indicates the embryonic day. E4.late and E5.early indicate cells picked 4–6 hours later and earlier than that in E4 stage and E5 stage, respectively.

(B) All cells were separated into ten basins on level 6 corresponding to ten asynchronous development stages in human preimplantation embryos cells.

(C) The hierarchical structure of these cells.

(D) Enriched GO terms of basin-specific gene sets suggest the main biological processes in each development stage.

(E) Pseudo-energy matrix (yellow boxes indicate low energy cost) could be used to infer transition among basins step by step.

(F) Three trajectories (black arrows) were inferred by the pseudo-energy matrix in (E).

(G) Four main cell types which are 8-cell embryo, morula cell, ICM (inner cell mass), and TE (trophectoderm) were identified according to marker genes expression.

(H) The development hierarchy was built according to the inferred trajectories in (F). The black arrows indicate development directions, the grey arrows indicate transition relations, and the grey dashed arrows indicate reversed links.

(I) ICM (inner cell mass) was split into three basins corresponding to PE (primitive endoderm), EPI (epiblast), and ICM.early (the early stage of inner cell mass) on level 4.

(J) Marker genes were used to identify the three basins in (I).

(K) The ground truth of the development hierarchy.

(L) The transition path (yellow line with the arrow) from 8 cell embryo to ICM were tracked. The yellow points indicate cells along the path and purple points indicate the critical points from morula cells to ICM.

(M) Differentially expressed genes along the transition path in (L).

(N) Representative genes show increasing “gene-flow” trends along the path. And gene expression varies dramatically around critical points (purple points).

(O) Representative genes show decreasing “gene-flow” trends along the path. And gene expression varies dramatically around critical points (purple points).

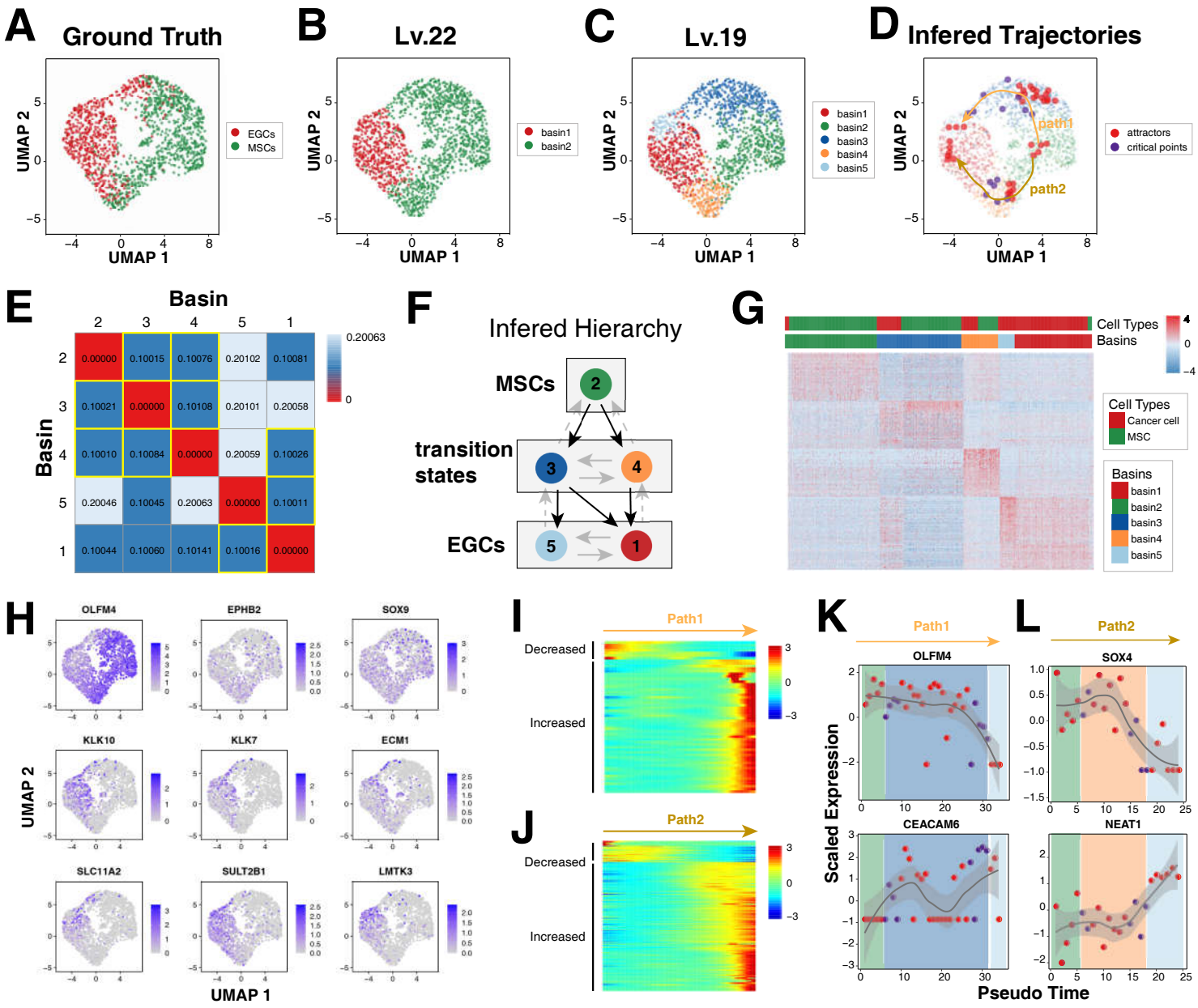

A
Ground Truth
UMAP 2
UMAP 1
EGCs
MSCs
B
Lv.22
basin1
basin2
C
Lv.19
basin1
basin2
basin3
basin4
basin5
D
Infered Trajectories
path1
path2
attractors
critical points
E
Basin
0.20063
0
F
Infered Hierarchy
MSCs
transition states
EGCs
G
Cell Types
Basins
Cancer cell
MSC
H
OLFM4
EPHB2
SOX9
KLK10
KLK7
ECM1
SLC11A2
SULT2B1
LMTK3
I
Path1
Decreased
Increased
J
Path2
K
OLFM4
CEACAM6
Scaled Expression
L
SOX4
NEAT1
Pseudo Time

Figure 6. MarkovHC detected the critical points from MSCs to gastric cancer cells.

(A) 830 mesenchymal stem cells (MSCs) and 649 MSC-origin early gastric cancer cells (EGCs) were projected into 2-dimensinal space using UMAP.

(B) MarkovHC found two basins on level 22.

(C) The basin corresponding to MSCs and EGCs split into three basins and two basins respectively on level 19.

(D) Two potential separating trajectories from MSCs to EGCs were inferred. Red points indicate attractors of each basin and purple points indicate critical points on boundaries of neighboring basins.

(E) Pseudo-energy matrix (yellow boxes indicate low energy cost) could be used to infer transition among basins step by step.

(F) The hierarchy was inferred according to the pseudo-energy matrix in (E). The black arrows indicate differentiation directions, the grey arrows indicate transition relations, and the grey dashed arrows indicate reversed links.

(G) The heatmap of basin-specific gene sets.

(H) MSC markers (OLFM4, EPHB2, and SOX9) and EGC markers (KLK10, SLC11A2, SULT2B1, KLK7, ECM1 and LMTK3) were gradually decreasing and increasing respectively along the trajectories from MSCs to EGCs.

(I) Differentially expressed genes along path 1 in (D).

(J) Differentially expressed genes along path 2 in (D).

(K-L) OLFM4 and CEACAM6 shown different "gene-flow" trends along path 1(K), and SOX4 and NEAT1 shown different "gene-flow" trends along path 2. Red points are attractors, purple points are critical points, and colors of background panels indicate basins on level 19 (C). Meanwhile, expression values of these genes dramatically changed around the critical points.

Supplementary Notes:

MarkovHC: Markov hierarchical clustering for the topological structure of high-dimensional single-cell omics data

Zhenyi Wang, [#,1] Yanjie Zhong,[3,5] Zhaofeng Ye,[2] Lang Zeng,[6] Yang Chen,[1] Minglei Shi,[2] Minping Qian*,[3] and Michael Q. Zhang*,[1,2,4]

1 MOE Key Laboratory of Bioinformatics; Bioinformatics Division and Center for Synthetic & Systems Biology, BNRist; Department of Automation, Tsinghua University, Beijing 100084, China

2 MOE Key Laboratory of Bioinformatics; Bioinformatics Division and Center for Synthetic & Systems Biology, BNRist; School of Medicine, Tsinghua University, Beijing 100084, China

3, School of Mathematical Sciences, Peking University, Beijing 100871, China

4, Department of Biological Sciences, Center for Systems Biology, The University of Texas, Richardson, TX 75080-3021, USA

5, Department of Mathematics and Statistics, Washington University in St. Louis, St. Louis, MO 63130, USA

6, Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA

Contents

Part. A Definitions of metastability of exponentially perturbed Markov chain. 1
Part. B Methods to find recurrent class. 9
Part. C Comparison with popular clustering methods and trajectory construction methods. 13
Part. D Comparison with Markov Cluster Algorithm (MCL). 14
Part. E MarkovHC can reveal non-equilibrium biological processes. 15
Part. F Transition probability estimation. 16
Part. G Data sets. 16
References 17

Part. A Definitions of metastability of exponentially perturbed Markov chain.

Consider a finite state space $S = \{1,2,\dots,N\}$. $\{P^{\beta}: \beta \in [0,\infty)\}$ are a family of Markov chain transition matrices on S satisfying that there exists

$$\lim_{\beta\to\infty} p^{\beta}(i \to j) = \lim_{\beta\to\infty} p^{\beta}(j|i) = p^{\infty}(j|i), \quad \forall i,j \in S.$$

In addition, we assume the following limit also exists,

$$C_{i,j} = \lim_{\beta\to\infty} -\frac{1}{\beta}\log p^{\beta}(j|i) \quad \forall i,j \in S.$$

We assume that

$$C_{i,j} > 0 \quad if \quad P^{\infty}(j|i) = 0.$$

Under the assumptions given above, Markov chains family, which is composed of Markov chains with transition matrices $\{P^{\beta}: \beta \in [0,\infty)\}$ is an exponential perturbation of the Markov chain with transition matrix P^{∞}.

Let K be a subset of S, we define $G(K)$ as the set of maps $g: K \to S$ such that g maps no non-empty subset of K into itself. Let $p \in K$ and $q \in S \setminus K$. We say that $g \in G(K)$ leads from p to q if there are distinct $x_0, \dots, x_k \in K$ satisfying that

$$p = x_0, \quad g(x_k) = q, \quad g(x_j) = x_{j+1}, \quad 1 \le j \le k-1.$$

We define

$$G_{pq}(K) \triangleq \{g | g \in G(K), g\ lead\ from\ p\ to\ q\},$$

$$T(K) = \min_{g\in G(K)} \sum_{\xi\in K} C_{\xi,g(\xi)},$$

$$T_{\xi\eta}(K) = \min_{g\in G_{\xi\eta}(K)} \sum_{\xi\in K} C_{\xi,g(\xi)}.$$

The metastable structure of exponentially perturbed Markov chain is defined iteratively in Chen's paper[1]. We present some fundamental definitions here.

Definition1.1

1. Define $\xi \Rightarrow \eta$ if $C_{\xi\eta} = 0$;
2. Define $\xi \to \eta$ if there is a sequence $\{\xi_i, i = 0,1,\dots,k\}$ such that
   $\xi_0 = \xi, \quad \xi_k = \eta, \quad \xi_i \Rightarrow \xi_{i+1}, \quad i = 0,1,\dots,k-1;$
3. Define $\xi \leftrightarrow \eta$ if $\xi \to \eta$ and $\eta \to \xi$.

Definition1.2

A level 1 attractor A_i^1 is a subset of S such that

1. $\forall \xi,\eta \in A_i^1$, $\xi \leftrightarrow \eta$;
2. If $\xi \in A_i^1$, $\xi \to \eta$, then $\eta \in A_i^1$.

We can see that, A_i^1 is actually a recurrent class of Markov chain with transition matrix P^{∞}.

Definition1.3

The corresponding attractive basin B_i^1 of a level 1 attractor A_i^1 is defined as

$$B_i^1 = \{\xi | \exists \varsigma \in A_i^1,\ s.t.\ \xi \to \varsigma\}.$$

Definition1.4

Define $T_1 = \min_{1\le j<k_1} T(B_j^1)$, where k_1 is the number of level 1 attractors.

Definition1.5

1. Define $A_i^1 \Rightarrow A_j^1$ if there exist $\xi \in B_i^1$, $\eta \in B_j^1 \setminus B_i^1$ such that

$T_{\xi\eta}(B_i^1) = T_1;$

2. Define $A_i^1 \to A_j^1$ if there is a sequence $A_{n_0}^1 = A_i^1, \dots, A_{n_m}^1 = A_j^1$ such that

$$A_{n_l}^1 \Rightarrow A_{n_l+1}^1, \quad l = 0,1, \dots, m-1;$$

3. Define $A_i^1 \leftrightarrow A_j^1$ if $A_i^1 \to A_j^1$ and $A_j^1 \to A_i^1$.

Definition1.6

A level 2 attractor A_i^2 is the union of some level 1 attractors $A_j^1{}'s$, such that

1. $\forall A_j^1, A_k^1 \subseteq A_i^2,\ A_j^1 \leftrightarrow A_k^1$;

2. If $A_j^1 \subseteq A_i^2,\ A_j^1 \to A_k^1$, then $A_k^1 \subseteq A_i^2$.

Definition1.7

The corresponding level 2 attractive basin B_i^2 of a level 2 attractor A_i^2 is defined as

$$B_i^2 = \{\bigcup_j B_j^1 \mid \exists A_k^1 \subseteq A_i^2,\ s.t.\ A_j^1 \to A_k^1\}.$$

Definition1.8

Define

$$V_i^2 = T(B_i^2) - \min_{\varsigma \in B_i^2} T(B_i^2 \setminus \{\varsigma\}),$$

$$T_2 = \min_{1 \le j \le k_2} V_j^2,$$

where k_2 is the number of level 2 attractors.

Suppose that we have defined level k attractors $A_i^k{}'s$, level k attractive basins $B_i^k{}'s$, $V_i^k{}'s$ and T_k. Then we define attractors and attractive basins of higher levels as follows.

Definition1.9

1. Define $A_i^k \Rightarrow A_j^k$ if there exist $\xi \in B_i^k,\ \eta \in B_j^k \setminus B_i^k$ such that

$$T_{\xi\eta}(B_i^k) - \min_{\varsigma \in B_i^k} T(B_i^k \setminus \{\varsigma\}) = V_i^k = T_k;$$

2. Define $A_i^k \to A_j^k$ if there is a sequence $A_{n_0}^k = A_i^k, \dots, A_{n_m}^k = A_j^k$ such that

$$A_{n_l}^k \Rightarrow A_{n_l+1}^k, \quad l = 0,1, \dots, m-1;$$

3. Define $A_i^k \leftrightarrow A_j^k$ if $A_i^k \to A_j^k$ and $A_j^k \to A_i^k$.

Definition1.10

A level $(k+1)$ attractor A_i^{k+1} is the union of some level k attractors $A_j^k{}'s$, such

that

1. $\forall A_m^k, A_j^k \subseteq A_i^{k+1}$, $A_m^k \leftrightarrow A_j^k$;

2. If $A_m^k \subseteq A_i^{k+1}$, $A_m^k \to A_j^k$, then $A_j^k \subseteq A_i^{k+1}$.

Definition1.11

The level $(k+1)$ attractive basin B_i^{k+1} of a level (k+1) attractor A_i^{k+1} is defined as

$$B_i^{k+1} = \{\bigcup_j B_j^k \mid \exists A_m^k \subseteq A_i^{k+1},\ s.t.\ A_j^k \to A_m^k\}$$

Definition1.12

Define

$$V_i^{k+1} = T(B_i^{k+1}) - \min_{\varsigma \in B_i^{k+1}} T(B_i^{k+1} \setminus \{\varsigma\}),$$

$$T_{k+1} = \min_{1 \le j \le s_{k+1}} V_j^{k+1},$$

where s_{k+1} is the number of level (k+1) attractor.

Conventionally, we define $A_i^k \Rightarrow A_i^k$ for any i and k.

Based on definitions presented above, we can construct a hierarchical structure that has been shown in the model part of paper. On a given level k, $A_i^k{'}s$ are sets of core points. A_i^k can show the main features of its corresponding cluster, i.e. B_i^k. $B_i^k \cap B_j^k$ $(i \neq j)$ are sets of critical points. We can consider states in $B_i^k \cap B_j^k$ $(i \neq j)$ as saddle points. Normally, states in A_i^k can hardly develop into states in A_j^k. However, once states in A_i^k reach states in $B_i^k \cap B_j^k$, the probability of reaching states in A_j^k would significantly increase.

As we can see, the essence of the construction of level 1 attractors is to find recurrent classes of Markov chain with transition matrix P^∞. Actually, the essence of higher levels construction is also to find recurrent classes. To achieve this goal, we need to construct a proper higher level Markov chain. On a given level k, our target state space is $S_k = \{A_1^k, A_2^k, \dots, A_{r_k}^k\}$, where r_k is the number of level k attractors and $P_k = (p_k(i|j))$ is the transition matrix on S_k.

We define an $r_k \times r_k$ auxiliary matrix $AU_k = (au_k(i,j))$ as follows,

$$au_k(i,j) = \begin{cases} 1, & if\ A_i^k \Rightarrow A_j^k, \\ 0, & otherwise. \end{cases}$$

Considering that our goal is to find recurrent classes, we can simply set transition matrix $P_k = (p_k(i|j))$ as follows,

$$p_k(i|j) = \frac{au_k(i,j)}{\sum_{m=1}^{r_k} au_k(i,m)}.$$

Apparently, the Markov chain constructed above is only an assistant tool. It doesn't alter the metastable structure of the origin exponentially perturbed Markov chain.

Here, we give further analysis of the metastability of exponentially perturbed Markov chain. Given the basic definition of the metastability of exponentially perturbed Markov chain, we have a basic sketch of our algorithm. The most critical question is how to judge whether $A_i^k \Rightarrow A_j^k$ or not. In order to design an applicable algorithm, we present some additional definitions and lemmas here to simplify the procedure of judging whether $A_i^k \Rightarrow A_j^k$ or not.

Definition1.13

K is a subset of state space S, for $g \in G(K)$, define

$$\prod(g) = \sum_{\xi \in K} C_{\xi, g(\xi)}.$$

Lemma 1.1

For any $i \in \{1,2,\dots,r_1\}$, where r_1 is the number of level 1 attractors, we have

$$T(B_i^1) = \min\{\sum_{k=1}^{l} C_{\xi_{k-1}\xi_k} \,|\, l \geq 1, \xi_0 \in A_i^1, \xi_1, \dots, \xi_{l-1} \in B_i^1, \xi_l \notin B_i^1\}.$$

Lemma 1.2

Assume $i \neq j$, then

$$A_i^k \Rightarrow A_j^k$$

if and only if $T(B_i^1) = T_1$ and at least one of the two following conditions holds:

1. $B_i^1 \cap B_j^1 \neq \emptyset$;
2. There exists $\xi \in A_i^1$, $\eta \in B_j^1 \setminus B_i^1$ and $g \in G_{\xi\eta}(B_i^1)$ such that $\prod(g) = T(B_i^1)$.

proof.

Necessity: If $i \neq j$ and $A_i^1 \Rightarrow A_j^1$, according to definition 1.5, there exists $\mu \in B_i^1$, $\nu \in B_j^1 \setminus B_i^1$ such that $T_{\mu\nu}(B_i^1) = T_1$. Then we have

$$T(B_i^1) \leq T_{\mu\nu}(B_i^1) = T_1 \leq T(B_i^1).$$

Thus, we can obtain that $T(B_i^1) = T_1$. Assume that neither condition 1 nor condition 2 holds. Let h be the map that satisfies$T_{\mu\nu}(B_i^1) = T_1$, i.e. $h \in G_{\mu\nu}(B_i^1)$, $\prod(h) = T_{\mu\nu}(B_i^1) = T_1$. Because condition 2 doesn't hold, $\mu \notin A_i^1$. For any $\xi \in A_i^1$, there is a set of positive integers PI such that for any k in PI, $g^k(\xi) \notin B_i^1$. Denote the

smallest one in PI as $\widetilde{k}$ and define $\Phi \triangleq \{\xi, g(\xi), g^2(\xi), \dots, g^{\widetilde{k}-1}(\xi)\}$. According to lemma 1.1, we have $\prod(h|_\Phi) \geq T(B_i^1)$. Besides, using the former conclusion, we have $\prod(h|_\Phi) \leq \prod(h) = T(B_i^1)$. Therefore, we obtain that

$$\prod(h|_\Phi) = \prod(h) = T(B_i^1).$$

As condition 2 doesn't hold, $g^{\widetilde{k}}(\xi) \notin B_j^1$. As condition 1 doesn't hold, $\Phi \cap B_j^1 = \emptyset$. Thus, there is $\varsigma \in B_i^1 \setminus \Phi$ such that $h(\varsigma) = \nu$. But we know that

$$0 = \prod(h) - \prod(h|_\Phi) \geq C_{\varsigma\nu} \geq 0.$$

Therefore, we have

$$C_{\varsigma\nu} = 0 \Rightarrow \quad \varsigma \in B_j^1 \Rightarrow \quad B_j^1 \cap B_i^1 \neq \emptyset,$$

which contradicts the assumption made at the beginning.

Sufficiency: Assume that $T(B_i^1) = T_1$. To prove $A_i^1 \Rightarrow A_j^1$, we only need to prove that there exists $\mu \in B_i^1$ and $\nu \in B_j^1 \setminus B_i^1$ such that $T_{\mu\nu}(B_i^1) = T_1$. If condition 1 holds, there exists $\mu \in B_i^1 \setminus A_i^1$ and $\nu \in B_j^1 \setminus B_i^1$ such that $C_{\mu\eta} = 0$. In addition, according to lemma 1.1, there exists $\xi_0 \in A_i^1, \xi_1, \dots, \xi_{l-1} \in B_i^1, \xi_l \notin B_i^1$ such that $T(B_i^1) = \sum_{k=1}^{l} C_{\xi_{k-1}\xi_k}$.

If $\mu \in \{\xi_k : k = 1,2,\dots,l-1\}$, we set $\xi_t = \mu$. We define $g \in G(B_i^1)$ as follows. Firstly, define

$$g(\xi_k) \triangleq \xi_{k+1}, k = 0,1,2,\dots,t-1, \quad g(\mu) \triangleq \nu.$$

Then, if $B_i^1 \setminus \{\xi_k : k = 0,1,2,\dots,t\} \neq \emptyset$, randomly pick a $\omega \in B_i^1 \setminus \{\xi_k : k = 0,1,2,\dots,t\}$. There exists $\{\omega_m : m = 0,1,\dots,z, \omega_0 = \omega, \omega_z = \xi_0\} \subseteq B_i^1$ such that $\sum_{m=0}^{z-1} C_{\omega_m\omega_{m+1}} = 0$. Define $\widetilde{m} = \min\{m : \omega_m \in \{\xi_k : k = 0,1,2,\dots,t\}\}$. Define

$$g(\omega_m) \triangleq \omega_{m+1}, m = 0,1,\dots,\widetilde{m}-1.$$

Repeat this process until the definition of g is complete. It is obvious that $g \in G_{\mu\eta}(B_i^1)$ and $\prod(g) \leq T(B_i^1)$. Therefore, we have

$$T(B_i^1) \leq T_{\mu\eta}(B_i^1) \leq \prod(g) \leq T(B_i^1)$$

which results in $T(B_i^1) = T_{\mu\eta}(B_i^1)$.

If $\mu \notin \{\xi_k : k = 1,2,\dots,l-1\}$, we firstly define

$$g(\xi_k) \triangleq \xi_{k+1}, k = 0,1,2,\dots,l-1,; g(\mu) \triangleq \nu.$$

Then we can complete the definition of g in a similar way shown above. Similarly, we can obtain $T(B_i^1) = T_{\mu\eta}(B_i^1)$. If condition 2 holds, i.e. there exists $\xi \in A_i^1$, $\eta \in B_j^1 \setminus B_i^1$ and $g \in G_{\xi\eta}(B_i^1)$ such that $\prod(g) = T(B_i^1)$, we have

$$T(B_i^1) \le T_{\mu\eta}(B_i^1) \le \prod(g) = T(B_i^1),$$

which also results in $T(B_i^1) = T_{\mu\eta}(B_i^1)$.

Lemma1.3

Assume $B_i^2 = \bigcup_{m=1}^{l} B_m^1$, we have

$$T(B_i^2) = C + (l-1)T_1,$$

where $C = \min\{\sum_{m=0}^{n-1} C_{\xi_m\xi_{m+1}} : \xi_0 \in A_i^2; \xi_m \in B_i^2, m = 1,2,\dots,n-1; \xi_n \in (B_i^2)^c\}$.

Proof.

In order to show that $T(B_i^2) = C + (l-1)T_1$, we have to find a map $g \in G(B_i^2)$ satisfying $\prod(g) = C + (l-1)T_1$ at first. Assume that we have found a path

$$S_1 = \{\xi_m : \xi_0 \in A_i^2; \xi_m \in B_i^2, m = 1,2,\dots,n_1-1; \xi_{n_1} \in (B_i^2)^c\},$$

such that $C = C(S_1) \triangleq \sum_{m=0}^{n_1-1} C_{\xi_m\xi_{m+1}}$ (In order to simplify our proof, we use notations like "path" and "$C(S_1)$".). Without loss of generality, we assume $A_1^1 \subseteq A_i^2$ and $\xi_0 \in A_1^1$. Define $g(\xi_m) = \xi_{m+1}$, $m = 1,2,\dots,n_1-1$.

According to definition 1.4, 1.5 and lemma 1.1, there exists a path $S_2 = \{\varsigma_p : \varsigma_0 \in A_2^1; \varsigma_p \in B_2^1, p = 1,2,\dots,n_2-1; \varsigma_{n_2} \in (B_2^1)^c\}$ leading from A_2^1 to $(B_2^1)^c$.

If $S_1 \cap (S_2 \setminus \{\varsigma_{n_2}\}) \cap B_i^2 \ne \emptyset$, assume the first intersection of path S_1 and path S_2 is $\xi_{t_1} = \varsigma_{t_2}$. Define a new path $S_2' \triangleq \{\lambda_p : \lambda_p = \varsigma_p, p = 0,1,\dots,t_2\}$.

If $C(S_2') < T_1$, we create a path concatenating the front part of S_2 and the later part of S_1 : $S_1^* = \{\varphi_p : \varphi_p = \varsigma_p, p = 0,1,\dots,t_2; \varphi_p = \xi_{p-t_2+t_1}, p = t_2+1,\dots,t_2+n_1-t_1\}$. Then we have $C(S_1) - C(S_1^*) = \sum_{m=0}^{t_1} C_{\xi_m\xi_{m+1}} - C(S_2')$. Noticing that $\xi_{t_1} = \varsigma_{t_2} \in B_2^1$, according to the definition of T_1, we have $\sum_{m=0}^{t_1} C_{\xi_m\xi_{m+1}} > T_1$, no matter whether $\xi_{t_1} \in B_1^1$ or not. Therefore, we have $C(S_1) - C(S_1^*) > 0$, which contradicts with the assumption that $C = C(S_1)$.

If $C(S_2') = T_1$, define $g(\lambda_p) = \lambda_{p+1}$, $p = 0,1,\dots,t_2-1$.

If $S_1 \cap (S_2 \setminus \{\varsigma_{n_2}\}) \cap B_i^2 = \emptyset$, define $g(\varsigma_p) = \varsigma_{p+1}$, $p = 0,1,\dots,n_2-1$.

For $B_3^1, B_4^1, \dots, B_l^1$, we continue to define g in the same way as above. At last, for any $\omega \in B_i^2$ that has not been assigned value under map g, similar to the proof of lemma 1.2, we can find a $\varsigma \in B_i^2$ such that $C_{\omega\varsigma} = 0$. Define $g(\omega) = \varsigma$. Then, we finish the definition of g such that $g \in G(B_i^1)$ and $\prod(g) = C + (l-1)T_1$.

Nest, we will show that for any $h \in G(B_i^1)$, we have $\prod(h) \ge C + (l-1)T_1$. For any $\tau \in A_i^2$, under the map h, there exists a path

$$S_\tau = \{\eta_p : \eta_0 = \tau; \eta_p = h^p(\tau) \in B_i^2, p = 1,2,\dots,n_\tau - 1; \eta_{n_\tau} = h^{n_\tau}(\tau) \in (B_i^2)^c\}.$$

Define $S_{\tilde{\tau}} \triangleq \underset{S_\tau}{\operatorname{argmin}} C(S_\tau)$. Without loss of generality, we assume $\tilde{\tau} \in A_1^1$ and $S_{\tilde{\tau}} \cap A_i^2 = \{\tilde{\tau}\}$.

We will prove $\prod(h|_{B_2^1 \setminus (B_1^1 \cup S_{\tilde{\tau}})}) \geq T_1$ firstly. Because $A_2^1 \subseteq B_2^1 \setminus (B_1^1 \cup S_{\tilde{\tau}})$, $B_2^1 \setminus (B_1^1 \cup S_{\tilde{\tau}})$ is not empty. We simply denote $B_2^1 \setminus (B_1^1 \cup S_{\tilde{\tau}})$ as M_2. For any $\omega \in A_2^1$, under the influence of map h, there exists a path

$$S_\omega = \{\delta_p : \delta_0 = \omega; \delta_p = h^p(\omega) \in M_2, p = 1,2,\dots,n_\omega - 1; \delta_{n_\omega} = h^{n_\omega}(\omega) \in (M_2)^c\}.$$

We will use reduction to absurdity to prove $C(S_\omega) \geq T_1$. Suppose $C(S_\omega) < T_1$, according to the definition of T_1, we have $h^{n_\omega}(\omega) \in B_2^1$, which means that $h^{n_\omega}(\omega) \in (B_2^1 \cap B_1^1)$ or $h^{n_\omega}(\omega) \in (B_2^1 \cap S_{\tilde{\tau}})$:

If $h^{n_\omega}(\omega) \in (B_2^1 \cap B_1^1)$, we can easily elongate path S_ω into path

$$\tilde{S}_\omega = \{\gamma_p : \gamma_p = \delta_p, p = 1,2\dots,n_\omega; \gamma_p \in B_2^1 \cap B_1^1, p = n_\omega + 1,\dots,n^* - 1; \gamma_{n^*} \in B_1^1 \setminus B_2^1\}.$$

Then we have $C(\tilde{S}_\omega) = C(S_\omega) < T_1$, which contradicts with the definition of T_1.

If $h^{n_\omega}(\omega) \in (B_2^1 \cap S_{\tilde{\tau}})$, splitting at $h^{n_\omega}(\omega)$, we concatenate path S_ω with the later part of path $S_{\tilde{\tau}}$ to get a new path $\hat{S}_\omega$. Because $C(S_\omega) < T_1$, we have $C(\hat{S}_\omega) < C(S_{\tilde{\tau}})$, which contradicts with our assumption.

To conclude, $C(S_\omega) \geq T_1$. Further, we have $\prod(h|_{B_2^1 \setminus (B_1^1 \cup S_{\tilde{\tau}})}) \geq C(S_\omega) \geq T_1$. Similarly, define $M_m \alpha\beta = B_m^1 \setminus ((\bigcup_{t=1}^{m-1} B_t^1) \cup S_{\tilde{\tau}})$, $m = 2,3,\dots,l$. Using the same method as above, we can prove that $\prod(h|_{M_m}) \geq T_1$, $m = 2,3,\dots,l$. Therefore, we have

$$\prod(h) \geq C(S_{\tilde{\tau}}) + \sum_{m=2}^{l} \prod(h|_{M_m}) \geq C + (l-1)T_1.$$

Combining the results of two main parts shown above, we have

$$T(B_i^2) = C + (l-1)T_1.$$

Lemma1.4

Assume $B_i^2 = \bigcup_{m=1}^{l} B_m^1$, we have

$$\min_{\xi \in B_i^2} T(B_i^2 \setminus \{\xi\}) = \min_{\varsigma \in A_i^2} T(B_i^2 \setminus \{\varsigma\}) = (l-1)T_1.$$

Corollary 1.1

Assume $B_i^2 = \bigcup_{m=1}^{l} B_m^1$, we have

$$V_i^2 = C,$$

where

$$C = \min\{\textstyle\sum_{m=0}^{n-1} C_{\xi_m \xi_{m+1}} : \xi_0 \in A_i^2, \xi_m \in B_i^2, m = 1,2,\dots,n-1, \xi_n \in (B_i^2)^c\}.$$

The proof of lemma 1.1 can be found in Chen's paper. Since the proof of lemma

1.4 is similar to lemma 1.3's, we skip it. Lemma 1.3 and 1.4 can directly lead to corollary 1.1. As the principles are similar to lemmas and corollary shown above, we only present some lemmas without proofs as follows:

Lemma1.5

$k \geq 2$. Assume $i \neq j$, then $A_i^k \Rightarrow A_j^k$ if and only if $V_i^k = T_k$ and at least one of the two following conditions holds:

1. $B_i^k \cap B_j^k \neq \emptyset$
2. There exists $\xi \in A_i^k$, $\eta \in B_j^k \setminus B_i^k$ and $S = \{\gamma_p : \gamma_0 = \xi; \gamma_p \in B_i^k, p = 1,2,\dots,n-1; \gamma_n = \eta\}$ such that $C(S) = \min\{\sum_{m=0}^{t-1} C_{\xi_m \xi_{m+1}} : \xi_0 \in A_i^k, \xi_m \in B_i^k, m = 1,2,\dots,t-1, \xi_t \in (B_i^k)^c\}$.

Lemma1.6

$k \geq 2$. Assume $B_i^k = \bigcup_{m=1}^{l} B_m^{k-1}$, we have

$$T(B_i^k) = C + (l-1)T_{k-1} + \sum_{m=1}^{l} \min_{\varsigma \in B_m^{k-1}} T(B_m^{k-1} \setminus \{\varsigma\})$$

where

$C = \min\{\sum_{m=0}^{t-1} C_{\xi_m \xi_{m+1}} : \xi_0 \in A_i^k, \xi_m \in B_i^k, m = 1,2,\dots,t-1, \xi_t \in (B_i^k)^c\}$.

Lemma1.7

$k \geq 2$. Assume $B_i^k = \bigcup_{m=1}^{l} B_m^{k-1}$, we have

$$\min_{\xi \in B_i^k} T(B_i^k \setminus \{\xi\}) = \min_{\omega \in A_i^k} T(B_i^k \setminus \{\omega\}) = (l-1)T_{k-1} + \sum_{m=1}^{l} \min_{\varsigma \in B_m^{k-1}} T(B_m^{k-1} \setminus \{\varsigma\})$$

Corollary1.2

$k \geq 2$. Assume $B_i^k = \bigcup_{m=1}^{l} B_m^{k-1}$, we have

$$V_i^k = C,$$

where

$C = \min\{\sum_{m=0}^{t-1} C_{\xi_m \xi_{m+1}} : \xi_0 \in A_i^k, \xi_m \in B_i^k, m = 1,2,\dots,t-1, \xi_t \in (B_i^k)^c\}$.

Part. B Methods to find recurrent class.

In this part, we show how to decide whether a state is recurrent or not and further find out all attractors. Assume we are considering a discrete Markov chain M_0 on finite state space S_0 with transition matrix P_0. Without loss of generality, assume $S_0 = \{1,2,\dots,n\}$. Before describing our methods, we present some theorems and propositions to lay theoretical foundations.

Theorem2.1

State i is a transient state. π is an invariant distribution. Then $\pi(i) = 0$, where

$\pi(i)$ refers to the ith component of π.

Theorem2.2

Assume M_0 is irreducible and recurrent. π is an invariant distribution. If M_0 is aperiodic, then $\lim_{n\to\infty} p_{ij}^{(n)} = \pi(j)$, where $p_{ij}^{(n)}$ refers to the (i,j) entry of $(P_0)^n$. If M_0 is periodic, $\lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} p_{ij}^{(k)} = \pi(j)$.

Proposition2.1

Assume M_0 is irreducible and recurrent. Then there is a unique π satisfying $\pi P_0 = \pi$ and $\sum_{i\in S_0} \pi(i) = 1$. Furthermore, for any $i \in S_0$, $\pi(i) > 0$.

Proof.

Denote Q to be $P_0 - I_n$, where I_n denotes the $n \times n$ identity matrix. For any $1 \le i \le n$, $\sum_{j=1}^{n} q_{ij} = 0$, where q_{ij} refers to the (i,j) entry of matrix Q. Therefore, $rank(Q) < n$. Then, there exists a nonzero vector η satisfying $\eta Q = 0$, i.e. $\eta P_0 = \eta$. Define $\Pi \triangleq \{\eta: \eta Q = 0, \sum_{i=1}^{n} \eta(i) = 1\}$. For any $\pi \in \Pi$, for any positive integer k, $\pi P_0^k = \pi$. According to theorem 2.2, there exists an invariant distribution μ satisfying that $\lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} p_{ij}^{(k)} = \mu(j)$. Denote $[\mu, \mu, \dots, \mu]_{n\times n}$ as Γ. We have

$$\pi\Gamma^T = \pi \lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} P_0^k = \lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} \pi P_0^k = \pi,$$

which means $\pi = \mu$. Thus, the size of Π is 1. If there exists i such that $\mu(i) = 0$, then select a $j \neq i$ satisfying that $\mu(j) > 0$ and a k satisfying that $p_{ji}^{(k)} > 0$. Therefore, $\mu(i) = \sum_{m=1}^{n} \mu(m) p_{mj}^{(k)} \ge \mu(j) p_{ji}^{(k)} > 0$, which contradicts the assumption.

Proposition2.2

If i is a recurrent state, there exists an invariant distribution π of M_0 s.t. $\pi(i) > 0$.

Proof.

Denote A to be $\{j: j \to i\}$. (We say $j \to k$ if there exists $\{r_0 = j, r_1, \dots, r_t, r_{t+1} = k\}$ satisfying that $p_{r_s r_{s+1}} > 0, s = 0,1,\dots,t$.) Then A is closed and all states in A are recurrent state of M_0. Denote M_1 to be Markov chain on state space A with transition matrix $P_0|_A$. Then M_1 is finite, irreducible and recurrent. According to proposition 2.1, there exists an invariant distribution μ of M_1 subject to that all components of μ are positive. Then we define an n-dimensional vector π as follows:

$$\pi(\mathrm{k}) = \begin{cases} \mu(\mathrm{k}), \mathrm{k} \in \mathrm{A} \\ 0, otherwise \end{cases}$$

π satisfies that π is an invariant distribution of M_0 and $\pi(i) > 0$.

Proposition2.3

i is a transient state. π is a non-zero n-dimensional vector subject to $\pi P_0 = \pi$. Then $\pi(i) = 0$.

Proof.

According to theorem2.2, we know that the limit of $(P_0)^k$ exists and we denote P to be $\lim_{k\to\infty}(P_0)^k$. As i is a transient state, for any $j \in S_0$, $\lim_{k\to\infty} p_{ji}^{(k)} = 0$. Therefore, all components in the ith column of P are 0. Because $\pi P_0 = \pi$, we have

$$\pi P = \pi \lim_{k\to\infty}(P_0)^k = \lim_{k\to\infty}\pi(P_0)^k = \pi.$$

Thus, $\pi(i) = \sum_{j=1}^{n}\pi(j)P(j,i) = 0$.

Proposition2.4

Assume M_0 has k attractors: $A_1, A_2, \dots, A_k$. Then the dimension of solution space of equation $(P_0 - I_n)^T x = 0$ is exactly k.

Proof.

Denote B to be $S_0 \setminus \bigcup_{t=1}^{k} A_t$. (We assume $B \neq \emptyset$ as the $B = \emptyset$ case is just a simpler case.) A non-zero n-dimensional vector x satisfying $(P_0 - I_n)^T x = 0$ means that $xP_0 = x$. According to proposition 2.3, for any $j \in B$, $x(j) = 0$. Denote P_t to be $P_0|_{A_t}$, $t = 1,2,\dots,k$. Denote M_t to be a Markov chain on state space A_t with transition matrix P_t, $t = 1,2,\dots,k$. Then M_t is a closed, irreducible, recurrent Markov chain. Thus, for $t = 1,2,\dots,k$, according to proposition 2.1, there exists a unique vector μ_t subject to $\mu_t P_k = \mu_t$ and $\sum_{j\in A_t}\mu_t(j) = 1$. Combining that $xP_0 = x$ and $x(B) = 0$, we have

$$x(A_t)P_t = x(A_t), \quad t = 1,2,\dots,k.$$

Based on the uniqueness of solution mentioned above, we can easily conclude that the dimension of solution space of equation $(P_0 - I_n)^T x = 0$ is exactly k.

Proposition2.5

Assume the dimension of solution space of equation $(P_0 - I_n)^T x = 0$ is k. Assume the k orthogonal solution vectors are $\pi_1, \pi_2, \dots, \pi_k$. Then for any $i \in S_0$, i is a recurrent state of M_0 if and only if there exists $1 \le t \le k$, satisfying that $\pi_t(i) \neq 0$; i is a transient state of M_0 if and only if for all $1 \le t \le k$, $\pi_t(i) = 0$.

Proof.

If i is a transient state, according to proposition 2.3, we know that for all $1 \le t \le k$, $\pi_t(i) = 0$. If i is a recurrent state, according to proposition 2.2, there exists an invariant distribution μ of M_0, s.t. $\mu(i) > 0$. Therefore, as $\pi_1, \pi_2, \dots, \pi_k$ comprise an orthogonal basis, there exists $1 \le t \le k$, satisfying that $\pi_t(i) \neq 0$.

Based on proposition 2.5, we can distinguish whether a state is recurrent or not by

computing all orthogonal eigenvectors corresponding to eigenvalues 1 of matrix $(P_0)^T$. Then we can find out all attractors and basins in the way we introduce in part C. Actually, when we choose to use MHA to analyze a large number of data points, directly computing eigenvectors is not the wisest choice. We should notice that the state with biggest density must be a recurrent state. We can utilize this information to simplify our algorithm.

Proposition2.6

Assume i is a recurrent state. Denote T_i to be $\{j: j \xrightarrow{M_0} i\}$. (We say $j \xrightarrow{M_0} k$ if there exists $\{r_0 = j, r_1, \dots, r_t, r_{t+1} = k\}$ satisfying that $p_{r_s r_{s+1}} > 0, s = 0,1, \dots, t$.) Denote $P_0(T_i; T_i)$ as $\widetilde{P}_1$. Denote $P_0(S_0 \setminus T_i; S_0 \setminus T_i)$ as P_2. Normalize $\widetilde{P}_1$ by row into P_1. Denote M_1 to be Markov chain on state space T_i with transition matrix P_1. Denote M_2 to be Markov chain on state space $S_0 \setminus T_i$ with transition matrix P_2. For any $j \in T_i$, j is a recurrent state of M_1 if and only if j is a recurrent state of M_0. For any $j \in S_0 \setminus T_i$, j is a recurrent state of M_2 if and only if j is a recurrent state of M_0.

Proof.

We define $j \xrightarrow{M_1} k$ and $j \xrightarrow{M_2} k$ similar to the definition of $j \xrightarrow{M_0} k$ presented above.

Part 1: Assume $j \in T_i$ and j is a recurrent state of M_0. If $j = i$, it is obvious that j is a recurrent state of M_1. If $j \neq i$, because j is a recurrent state of M_0 and $j \xrightarrow{M_0} i$, we have $i \xrightarrow{M_0} j$. Then we have $i \xrightarrow{M_1} j$, which means that j is a recurrent state of M_1.

Part 2: Assume $j \in T_i$ and j is a recurrent state of M_1. If $j = i$, it is trivial that j is a recurrent state of M_0. If $j \neq i$, because $j \xrightarrow{M_0} i$, we know $j \xrightarrow{M_1} i$. Considering that j is a recurrent state of M_1, we have $i \xrightarrow{M_1} j$, which indicates that $i \xrightarrow{M_0} j$. Thus we obtain that j is a recurrent state of M_0.

Part 3: Assume $j \in S_0 \setminus T_i$ and j is a recurrent state of M_0. For any $k \in S_0 \setminus T_i$ subject to $k \neq j$ and $j \xrightarrow{M_2} k$, we have $j \xrightarrow{M_0} k$. As j is a recurrent state of M_0, $k \xrightarrow{M_0} j$ holds. Then we have $k \xrightarrow{M_2} j$. Thus we can conclude that j is a recurrent state of M_2.

Part 4: Assume $j \in S_0 \setminus T_i$ and j is a recurrent state of M_2. According to proposition 2.2, there exists a invariant distribution μ whose size is equal to $|S_0 \setminus T_i|$ satisfying that $\mu(j) > 0$. Then we define an n-dimensional vector π as follows:

$$\pi(k) = \{\mu(k), \quad k \in S_0 \setminus T_i, 0, \quad otherwise.$$

We easily know that $\pi P_0 = \pi$ and $\pi(j) > 0$. According to theorem 2.1, we can conclude that j is a recurrent state of M_0.

Assume the state with biggest density is i. And we use the same notations in proposition 2.6. We divide the origin Markov chain M_0 into two sub-chains M_1 and

M_2. It is easy to know that M_1 only contain a single recurrent class, which can be determined by computing the single eigenvector corresponding to eigenvalue 1 of matrix P_1. According to proposition 2.6, this recurrent class is an attractor of M_0. Next, we apply the same method on M_2 because the state with biggest density among states in $S_0 \setminus T_i$ is also a recurrent state. We do it iteratively until we get all attractors.

Part. C Comparison with popular clustering methods and trajectory construction methods.

We compared MarkovHC with 19 popular clustering methods (table S1) and 4 representative trajectory construction methods in theory.

These clustering methods including partition-based methods such as k-means[2], density-based methods such as DBSCAN[3], OPTICS[4] and HDBSCAN[5], graph-based methods such as Louvain clustering (in Seurat[6]), Markov Cluster Algorithm (MCL)[7] and scanpy[8], linkage-based methods such as SLINK[9] (also called single-linkage clustering), ALINK[10] (also called average-linkage clustering) and CLINK[11] (also called complete-linkage clustering), spectral method[12], model-based method[13], multikernel learning-based methods such as SC3[14], SIMLR[15], and hierarchical clustering methods such as classic hierarchical clustering, CIDR[16], pcaReduce[17], mpath[18], SINCERA[19], TooManyCells[20], and BackSPIN[21]. These methods have their advantages and disadvantages. For example, k-means methods are classic but often suffer from poor classifications when the data is non-convex and low robustness induced by initial seeds selection. DBSCAN, on the other hand, can handle convex and non-convex clusters but relies on two sensitive parameters to define a dense region, i. e., the radius of the neighborhood and the minimum number of points. HDBSCAN can be seen as a conceptual and algorithmic improvement over OPTICS, which is a density-based clustering method and provides a complete clustering hierarchy composed of all possible density-based clusters. This method extracts the hierarchy from MST (minimum spanning tree)[22], which introduces biases from its non-unique topology. Louvain clustering methods are widely used, but cannot generate hierarchical structures of single-cell data directly. Even though different clustering results can be used to build a clustering tree by tuning the resolution parameter, the process heavily relies on experiences and might be tricky to skip cell populations of significantly biological meaning due to overly rough partition. SLINK, ALINK, and CLINK can build hierarchical structure, however, the choice of linkages in measuring the similarity among samples is tricky and different orders of processed samples will lead to totally different results. SC3 and SIMLR perform well when applied to some scRNA-Seq datasets, but cannot provide hierarchical structures automatically either. Among these clustering algorithms, Markov Cluster Algorithm (MCL)[7] invented by Van Dongen et al. is the most similar algorithm to ours. We will compare MarkovHC and MCL detailedly in part D. All problems mentioned above, including non-convex clustering, hetero-density clustering, recognition of small clusters referring to rare cell types, and construction of robust hierarchical structures reflecting data topology, can be solved by MarkovHC.

Meanwhile, a large number of algorithms have been proposed to track cell transition paths. In Schiebinger's paper[23], they proposed Waddington-OT and comprehensively reviewed 62 methods, which were categorized into three classes. In Ji's paper[24], they compared their TSCAN with 11 other methods. We are not going to make a very detailed comparison in this paper, but only choose four representative methods for discussion, which are Spanning-tree progression analysis of Density-normalized events (SPADE)[25], monocle series[26-28], URD[29], and Waddington-OT[23]. SPADE aims to extract a cellular hierarchy from high-dimensional cytometry data. But it has two inherent shortcomings. Firstly, the density-dependent down-sampling may ruin the original structure in data and obscuring the density information of samples. Secondly, MST (minimum spanning tree) is not unique and very sensitive to parameter selection in the example case[30](in Fig. S17). Monocle projects original data into a two-dimensional space and orders cells in pseudo time via MST. Hence, the problems of MST also exist in the original monocle. As we reviewed above, MST is widely used in designing trajectory tracking algorithms which may lead to bias. To break these limitations, URD uncovers developmental trajectories from the viewpoint of extending diffusion maps. Briefly, URD operates discrete random walks on a cell graph to approximate the continuous process of diffusion, which can reflect cell differentiation. Although URD can get an approximate transition path by the Monte Carlo method, it cannot get a precise transition path and cannot find critical transition points in the transition path. Waddington-OT is the latest algorithm-based on optimal-transport theory up to date. It can track the ancestors and descendants for every single-cell, however is unable to find critical points on trajectories either. In contrast, MarkovHC calculates a precise transition path consistent with theory and can detect critical transition points simultaneously.

Part. D Comparison with Markov Cluster Algorithm (MCL).

Markov Cluster Algorithm (MCL)[7], a graph clustering algorithm based on flow simulation, was invented by Van Dongen et al. in 2000. The MCL is one of the most used clustering algorithms in bioinformatics, but we found there were two points of MCL could be improved. The first one is that the clustering results are sensitive to the parameters used in the expansion and inflation steps. Therefore, it is not clear which clustering result is meaningful. The second one is that MCL cannot get a hierarchical structure. Fig. S18 is from Van Dongen's thesis[7]. We can see that although the parameter R can control the scale of clustering, the results with different scales cannot be combined to a hierarchical structure. For example, the subgraph in the red circle is split when R is set to 1.4 (in the left top panel).

We ran MCL on a 2-dimensional simulation data (in Fig. S19), which was also used in our manuscript to test MarkovHC (in Fig.2 F). The KNN graph (Fig. S19A) was inputted into MCL. Here, e is the expansion parameter and i is the inflation parameter. And we tested MCL using 18 combinations of parameters. 45 clusters were get using the default parameters (Fig. S19B), in which e and i were equal to 2. Then we set e and i to 1, 180 clusters were get (Fig. S19C). After several adjustments, four clusters were

detected (Fig. S19D). Although we tried more combinations of parameters, we couldn't get better results. And MCL even collapsed under most parameter combinations.

In theory, MCL is based on operating Markov matrix product iteratively, but MarkovHC is based on the global pseudo-energy matrix derived from the Markov matrix. And MCL may miss meaningful partitions but get too rough or too meticulous results for the difficulty in parameter selection. However, MarkovHC will not miss them benefiting from the hierarchical structure. So, we believe MarkovHC has better practical significance.

In summary, although both MCL and MarkovHC cluster points by calculating the geodesic distance and the basic ideas of them are quite similar, MarkovHC has four advantages over MCL. Firstly, the clustering results of MCL are sensitive to the expansion parameter and the inflation parameter, which are difficult to select. But there are no such sensitive parameters in MarkovHC. Secondly, MCL cannot get a hierarchical structure. But MarkovHC can get a hierarchical structure easily. Thirdly, the transition probability of MarkovHC includes density information which is ignored in the transition probability of MCL. Fourthly, the workflow of MarkovHC not only can be used in a secrete-time Markov chain but also can be used in a continuous-time Markov chain.

Part. E MarkovHC can reveal non-equilibrium biological processes.

In Fig. S20, we used MarkovHC to analysis 280 mouse ESCs in Buettner *et. al.*'s paper[31]. These cells were stained with Hoechst 33342 and Flow cytometry sorted for G1, S, and G2M stages of the cell cycle (Fig. S20A; 96 cells in G1, 96 cells in G2M, and 88 cells in S). We observed three basins match these three stages on level 17 (Fig. S20B). And minimum entries of each row in the pseudo-energy matrix suggest the direction of the cell cycle (yellow boxes in Fig. S20C left).

For the definition of pseudo-energy is

$$c_{i,j} = c(i \to j) = \lim_{\beta \to \infty} -\frac{1}{\beta} \log p^{\beta}\left(j|i\right),$$

the transition probability matrix can be calculated by

$$p^{\beta}\left(j|i\right) \sim e^{-\beta \times c_{i,j}},$$

We can get the pseudo-energy matrix of these basins (Fig. S20C left). Without loss of generality, we let β=5000 in this dataset. Let diagonal entries equal to 0 and perform row normalization to get the transition probability matrix (Fig. S20C right). From the transition probabilities among these three basins, we can get a non-equilibrium 'flow circle' which is 1 (G1) → 3 (S) → 2 (G2M) → 1 (G1).

These results demonstrate that MarkovHC can reveal non-equilibrium biological processes.

Part. F Transition probability estimation.

To quantitatively evaluate how good P_{ij} can fit the 'real transition probability', we compared transition probabilities calculated by quadratic programming and MarkovHC.

The authors of Waddington-OT paper[23] reprogramed secondary (2°) MEFs from E13.5 embryos and collected cells to do single-cell RNA sequencing. We checked these time series scRNA-Seq data and found that sub-clusters in Day2, Day2.5 (the night in Day2), and Day3 can be clustered by Seurat[6] (Fig. S22) and matched according to the differentially expressed genes (Fig. S23). So we used data in these 3 time point and deployed quadratic programming to calculate the transition probability matrix. The quadratic programming (QP) form of this issue is shown below.

$$P_{i,j} = \begin{bmatrix} x_1 & x_4 & x_7 \\ x_2 & x_5 & x_8 \\ x_3 & x_6 & x_9 \end{bmatrix},$$

is the transition probability matrix, and we have the cell proportions at 3 time points which can be denoted as 3 column vectors, d_1, d_2 and d_3. So the QP form is

$$\min \sum_{n=1}^{3} \left\| d_{n+1}^T - d_n^T \right\|_2^2,$$
$$s.t. \begin{cases} x_{i,j} \geq 0, x_{i,j} \in P \\ x_1 + x_4 + x_7 = 1 \\ x_2 + x_5 + x_8 = 1 \\ x_3 + x_6 + x_9 = 1 \end{cases}$$

Although the entries in P_{ij} among clusters (Fig. S22) calculated by QP (the left matrix in Fig. S24) and the P_{ij} among basins (Fig. S25) calculated by MarkovHC (the right matrix in Fig. S24) are different, the transition relationships among these basins indicated by P_{ij} are consistent.

Part. G Data sets.

1. The codes generating simulation data in Fig. 2 are in the 'MarkovHC' repository (https://github.com/ZhenyiWangTHU/MarkovHC).
2. The data and labels in Fig. 3A-D were downloaded from 'SIMLR' repository (https://github.com/BatzoglouLabSU/SIMLR).
3. The data and lineage labels of C. elegans embryogenesis in Fig. 3E were downloaded from GSE126954.
4. The 33k PBMCs dataset were downloaded from 10X genomics support (https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/pbmc33k).

5. The mass cytometry of PBMC were downloaded from the supplementary materials of Anchang's paper[30] (https://www.nature.com/articles/nprot.2016.066).
6. The scRNA-Seq and scATAC-seq data of the retinoic acid-induced mESCs differentiation at day 4 were downloaded from GSE115968 and GSE107651.
7. The scRNA-Seq data were downloaded from GSE75748. In these data[32], the labeled cell types include neuronal progenitor cells (NPCs, ectoderm derivatives, n = 173), DE cells (endoderm derivatives, n = 138), endothelial cells (ECs, mesoderm derivatives, n = 105), trophoblast-like cells (TBs, extraembryonic derivatives, n = 69), undifferentiated H1 (n = 212) and H9 (n = 162) human ES cells, and human foreskin fibroblasts (HFFs, n = 159).
8. The scRNA-Seq data of 1529 single-cells were downloaded from E-MTAB-3929 (https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-3929/).
9. The scRNA-Seq data of 830 mesenchymal stem cells (MSCs) and 649 MSC-origin early gastric cancer cells (EGCs) were downloaded from GSE134520.
10. The scRNA-Seq data of mouse ESCs in Buettner *et. al.*'s paper[31] was downloaded from E-MTAB-2805 (https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-2805/).

References

1 Chen, D., Feng, J. & Qian, M. Metastability of exponentially perturbed Markov chains. *Science in China Series A-Mathematics* **39**, 7-28 (1996).

2 MacQueen J. Some methods for classification and analysis of multivariate observations. *LeCam LM, Neyman J (eds) Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, University of California Press*, 281–297, (1967).

3 Ester, M ., Kriegel, H.-P ., Sander, J. & Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. 2nd Int. Conf. Knowledge, Discovery and Data Mining (KDD-96)* (eds Simoudis, E . et al.) 226–231 (AAAI, 1996).

4 Mihael Ankerst, M. M. B., Hans-Peter Kriegel, Jörg Sander. OPTICS: Ordering Points To Identify the Clustering Structure. (SIGMOD, Philadelphia, 1999).

5 Campello, R. J. G. B., Moulavi, D., Zimek, A. & Sander, J. Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection. *ACM Transactions on Knowledge Discovery from Data* **10**, 1-51, (2015).

6 Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and

species. *Nat Biotechnol* **36**, 411-420, (2018).

7 Van Dongen, S. M. *Graph Clustering by Flow Simulation*. PhD thesis, Univ. Utrecht (2000).

8 Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* **19**, 15, (2018).

9 Sibson, R. SLINK: an optimally efficient algorithm for the single-link cluster method. *The computer journal* **16**, 30-34 (1973).

10 Sokal, R. R. A statistical method for evaluating systematic relationships. *Univ. Kansas, Sci. Bull.* **38**, 1409-1438 (1958).

11 Defays, D. An efficient algorithm for a complete link method. *The Computer Journal* **20**, 364-366 (1977).

12 Hess, S., Duivesteijn, W., Honysz, P. & Morik, K. The SpectACl of nonconvex clustering: a spectral approach to density-based clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3788-3795 (2019)

13 Banfield, J. D. & Raftery, A. E. Model-Based Gaussian and Non-Gaussian Clustering. *Biometrics* **49**, 803-821, (1993).

14 Kiselev, V. Y. et al. SC3: consensus clustering of single-cell RNA-seq data. *Nat Methods* **14**, 483-486, (2017).

15 Wang, B., Zhu, J., Pierson, E., Ramazzotti, D. & Batzoglou, S. Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat Methods* **14**, 414-416, (2017).

16 Lin, P., Troup, M. & Ho, J. W. CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol* **18**, 59, (2017).

17 Zurauskiene, J. & Yau, C. pcaReduce: hierarchical clustering of single cell transcriptional profiles. *BMC Bioinformatics* **17**, 140, (2016).

18 Chen, J., Schlitzer, A., Chakarov, S., Ginhoux, F. & Poidinger, M. Mpath maps multi-branching single-cell trajectories revealing progenitor cell progression during development. *Nat Commun* **7**, 11988, (2016).

19 Guo, M., Wang, H., Potter, S. S., Whitsett, J. A. & Xu, Y. SINCERA: A Pipeline for Single-Cell RNA-Seq Profiling Analysis. *PLoS Comput Biol* **11**, e1004575, (2015).

20 Schwartz, G. W. et al. TooManyCells identifies and visualizes relationships of single-cell clades. *Nat Methods* **17**, 405-413, (2020).

21 Zeisel, A. et al. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138-1142, (2015).

22 Prim, R. C. Shortest Connection Networks and Some Generalizations. *At&T Tech J* **36**, 1389-1401, (1957).

23 Schiebinger, G. et al. Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming. *Cell* **176**, 928-943, (2019).

24 Ji, Z. & Ji, H. TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. *Nucleic Acids Res* **44**, e117, (2016).

25 Qiu, P. et al. Extracting a cellular hierarchy from high-dimensional cytometry

data with SPADE. *Nat Biotechnol* **29**, 886-891, (2011).

26 Trapnell, C. et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol* **32**, 381-386, (2014).

27 Qiu, X. et al. Reversed graph embedding resolves complex single-cell trajectories. *Nat Methods* **14**, 979-982, (2017).

28 Cao, J. et al. The single-cell transcriptional landscape of mammalian organogenesis. *Nature* **566**, 496-502, (2019).

29 Farrell, J. A. et al. Single-cell reconstruction of developmental trajectories during zebrafish embryogenesis. *Science* **360**, 6392 (2018).

30 Anchang, B. et al. Visualization and cellular hierarchy inference of single-cell data using SPADE. *Nat Protoc* **11**, 1264-1279, (2016).

31 Buettner, F. et al. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nat Biotechnol* **33**, 155-160, (2015).

32 Chu, L. F. et al. Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome Biol* **17**, 173, (2016).

Supplementary Figures:

MarkovHC: Markov hierarchical clustering for the topological structure of high-dimensional single-cell omics data

Zhenyi Wang, [#,1] Yanjie Zhong,[3,5] Zhaofeng Ye,[2] Lang Zeng,[6] Yang Chen,[1] Minglei Shi,[2] Minping Qian*,[3] and Michael Q. Zhang*,[1,2,4]

1 MOE Key Laboratory of Bioinformatics; Bioinformatics Division and Center for Synthetic & Systems Biology, BNRist; Department of Automation, Tsinghua University, Beijing 100084, China

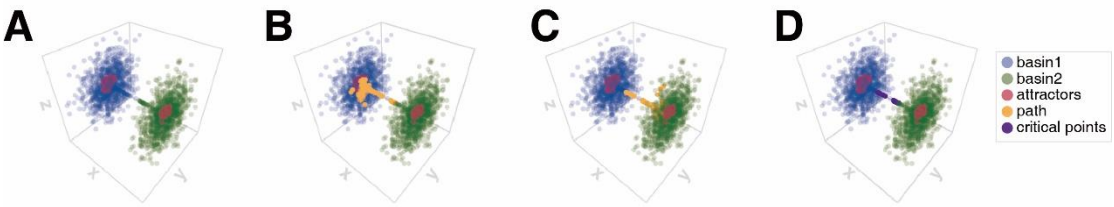
2 MOE Key Laboratory of Bioinformatics; Bioinformatics Division and Center for Synthetic & Systems Biology, BNRist; School of Medicine, Tsinghua University, Beijing 100084, China

3, School of Mathematical Sciences, Peking University, Beijing 100871, China

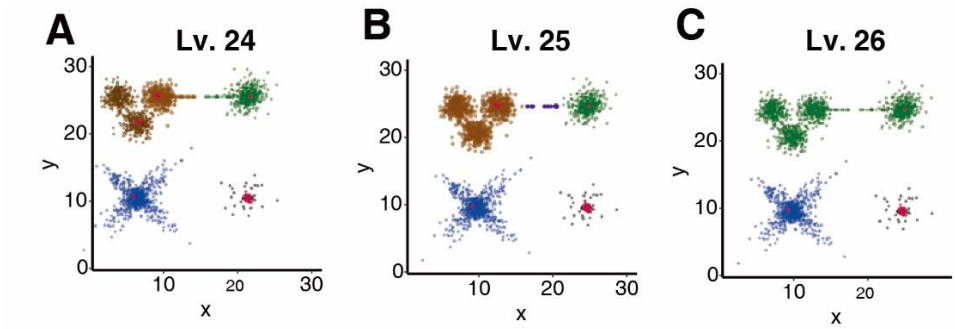
4, Department of Biological Sciences, Center for Systems Biology, The University of Texas, Richardson, TX 75080-3021, USA

5, Department of Mathematics and Statistics, Washington University in St. Louis, St. Louis, MO 63130, USA

6, Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA

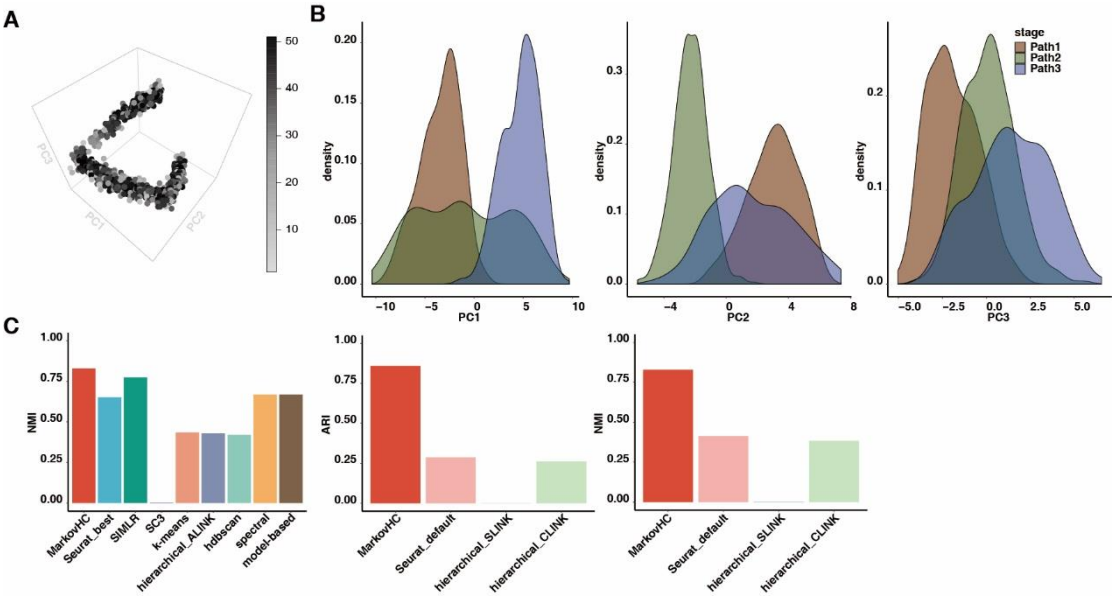


Supplementary figure 1 | Basins, attractors, transition path, and critical points obtained by incorporating a pre-clustering strategy to save computational time.



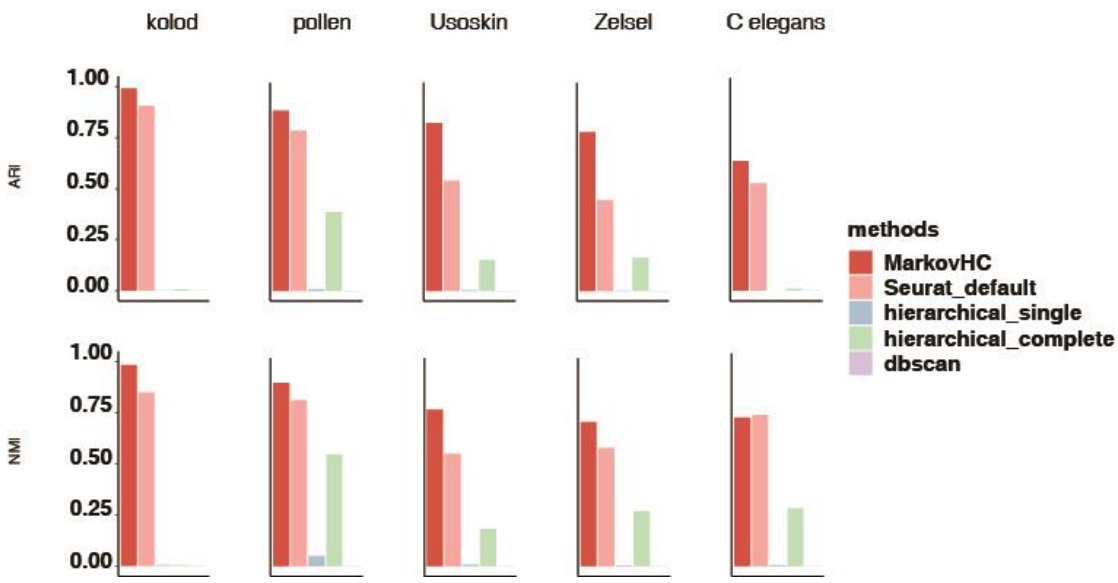
Supplementary figure 2 | The basins, attractors and critical points on level 24, 25 and 26.

A. On level 24, the brown basin on level 25 was further divided into two basins.
B. On level 25, the green basin on level 26 split into two basins (brown and green) with critical points (purple) on the bridge connecting them.
C. Three basins could be found on level 26.

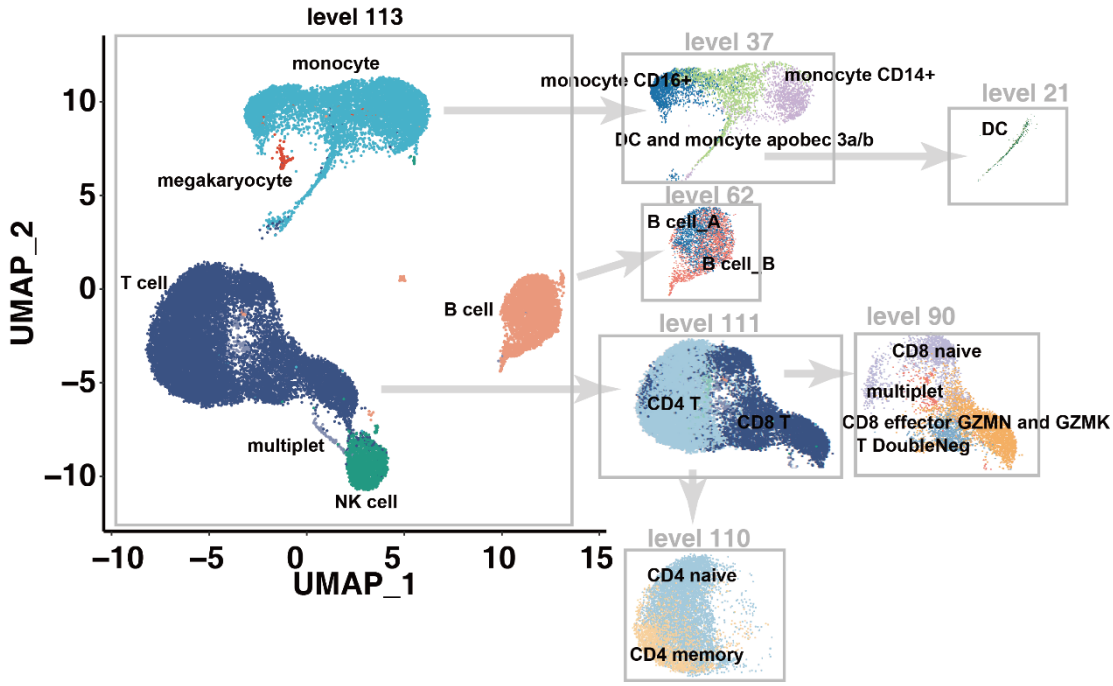


Supplementary figure 3 | More results of the simulated single cell data with three stages in continuum.

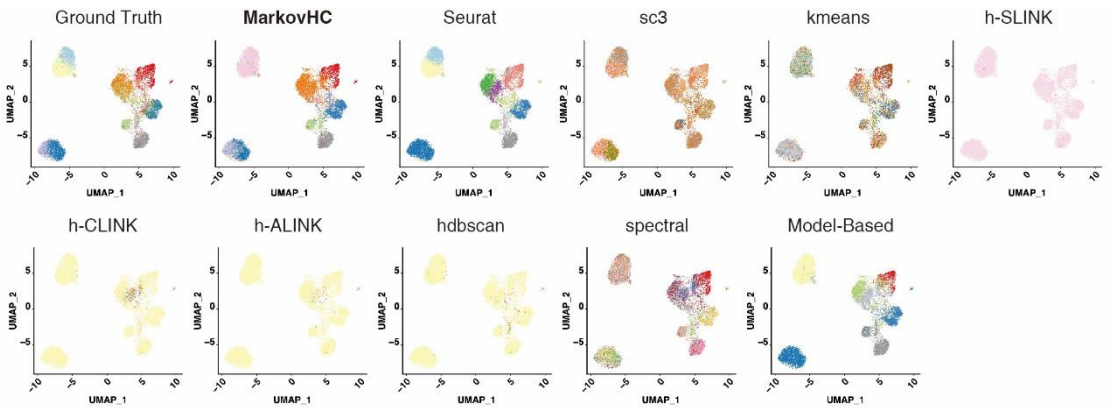
A. D scores of these points.
B. The distribution of each stage on the first, second, and third principle component.
C. Comparisons (NMI and ARI) with Seurat (with parameters fine-tuned), Seurat with default parameters, SIMLR, sc3, k-means, hierarchical clustering with different linkages, HDBSCAN, spectral clustering, and model-based clustering.



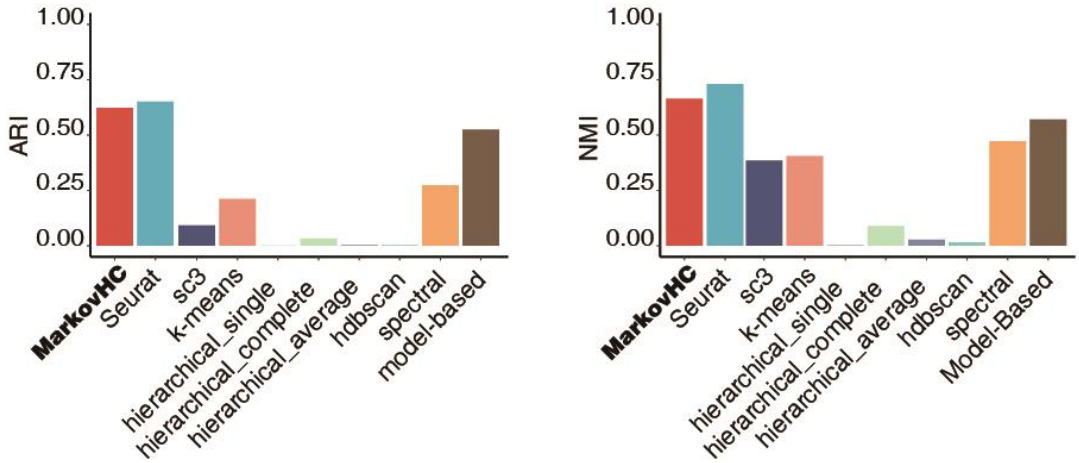
Supplementary figure 4 | The comparison among MarkovHC, Seurat with default parameters and hierarchical clustering with other linkages. ARI and NMI show MarkovHC's clustering accuracy is the best one on these data.



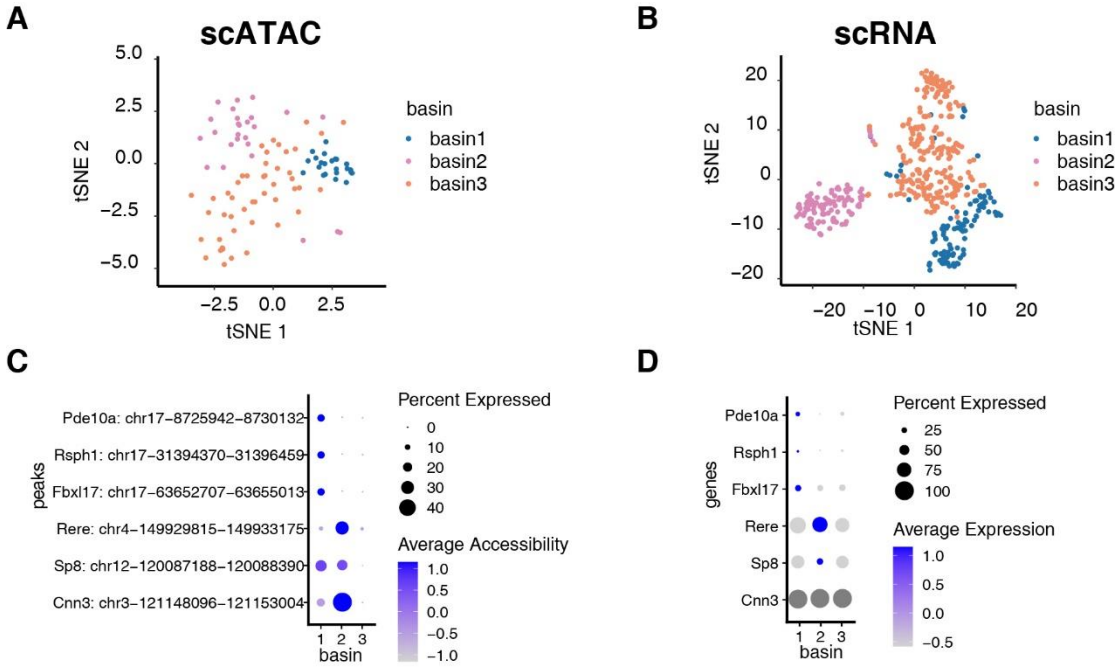
Supplementary figure 5 | Basins in the 33k PBMCs dataset. On level 113, basins correspond to B cells, T cells, NK cells, megakaryocyte and monocyte could be identified. On level 111, the T cell basin was divided into CD4+ T cell basin and CD8+ T cell basin. On level 110, CD4+ T cell basin was further divided into CD4+ naïve and CD4+ memory T cell basins. Similarly, the subtypes of B cell, CD8+ T cell and monocyte could also be recovered on lower levels.



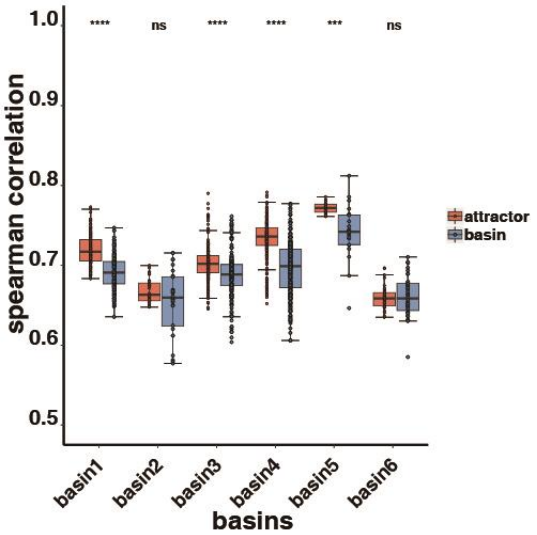
Supplementary figure 6 | Clustering results of MarkovHC and other popular clustering methods. 10009 cells were projected these cells into 2-dimensional space by UMAP and colored according to the clustering results.



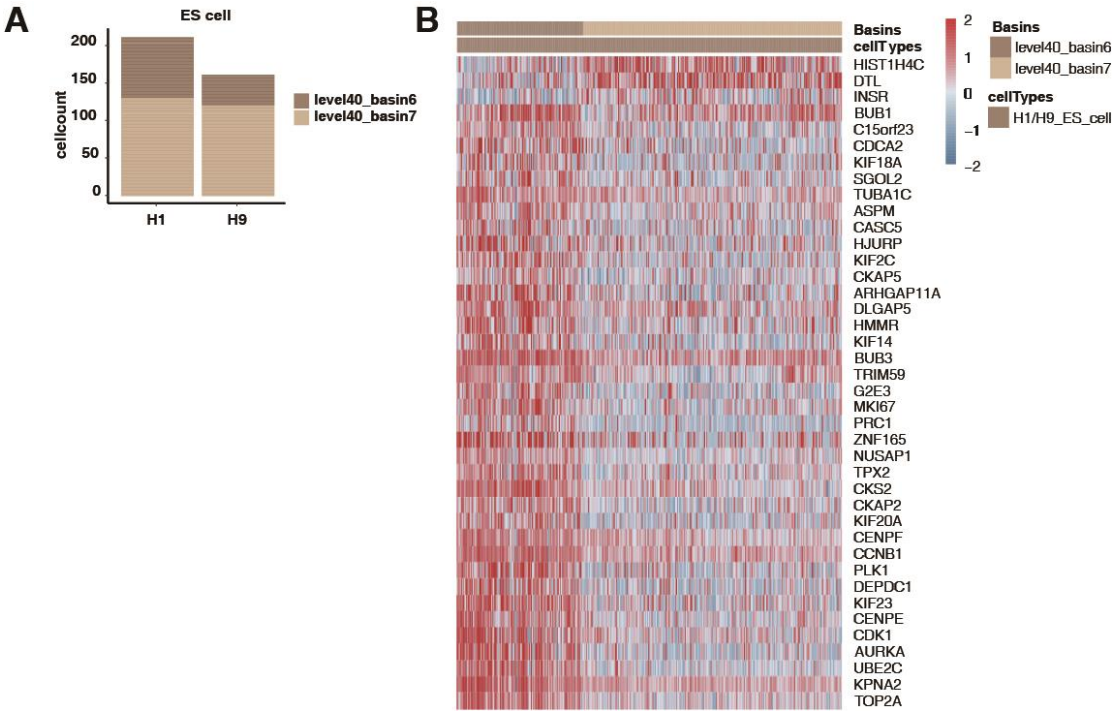
Supplementary figure 7 | Compare MarkovHC's clustering accuracy with those of other popular clustering methods on cytometry data. The results showed that MarkovHC got comparable performance to Seurat on this dataset. We noticed that sc3, k-means, simple hierarchical clustering methods and hdbscan couldn't get good results when applied on this dataset. Spectral clustering and model-based clustering were good enough either. Meanwhile, we also run SIMLR on this dataset but the program was stuck several days for some reason. So we didn't get its performance.



Supplementary figure 8 | MarkovHC identified three basins in scATAC-Seq data and scRNA-Seq data.



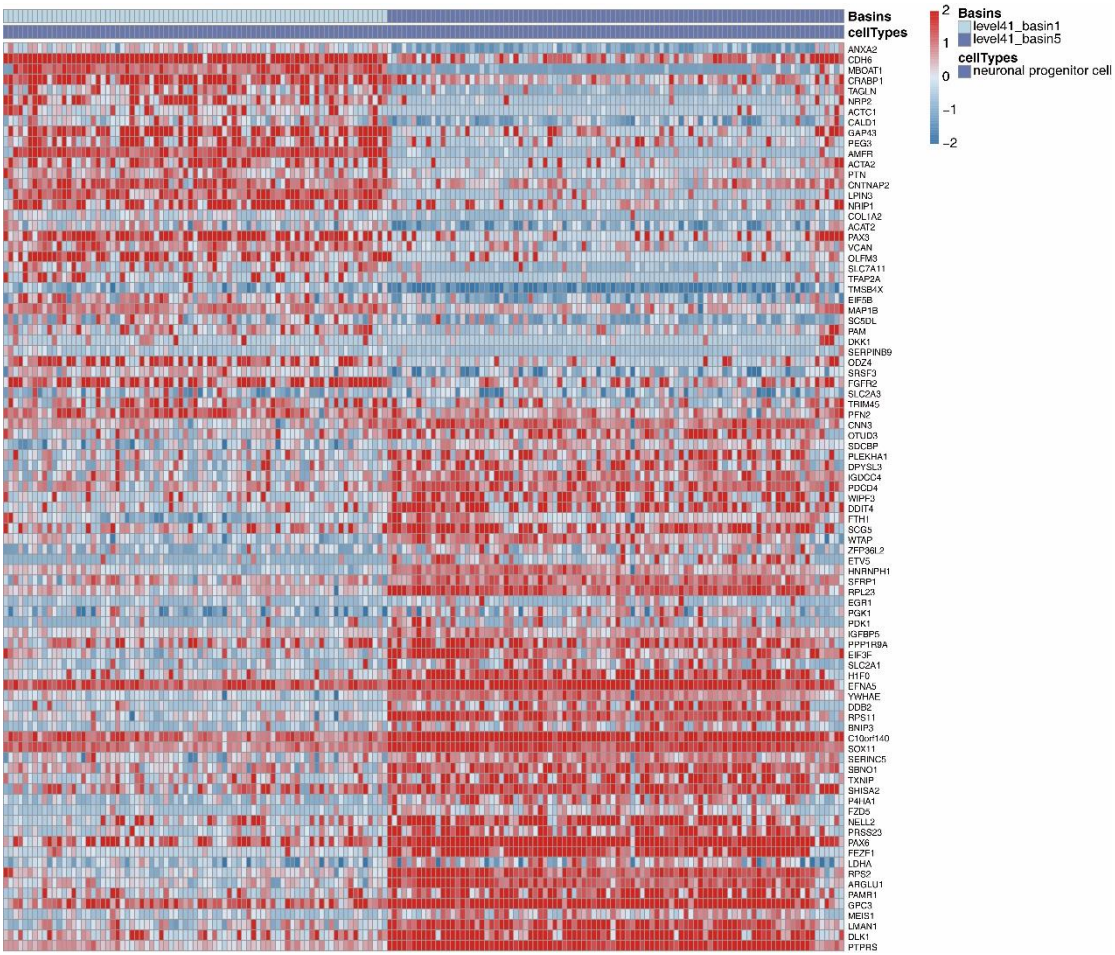
Supplementary figure 9 | The spearman correlations among attractors and others in the basins. The spearman correlations of each cell pair in attractors are significantly higher than those of the same number of random selecting cell pairs in other basin cells which demonstrates that our hypothesis of the feature of attractors is correct in real biology data.



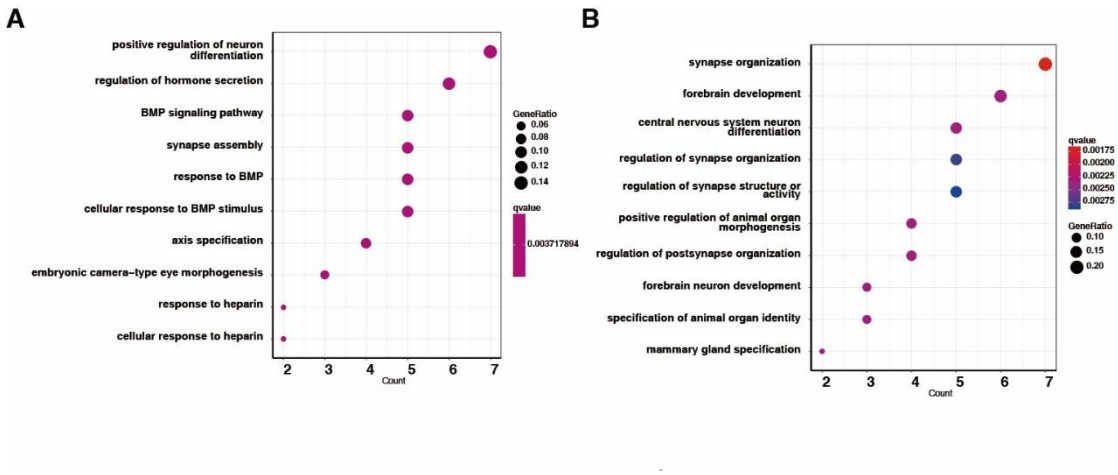
Supplementary figure 10 | Two basins in the H1/H9 ES cells.

A. The bar plot shows the proportion of these two basins in H1 and H9.

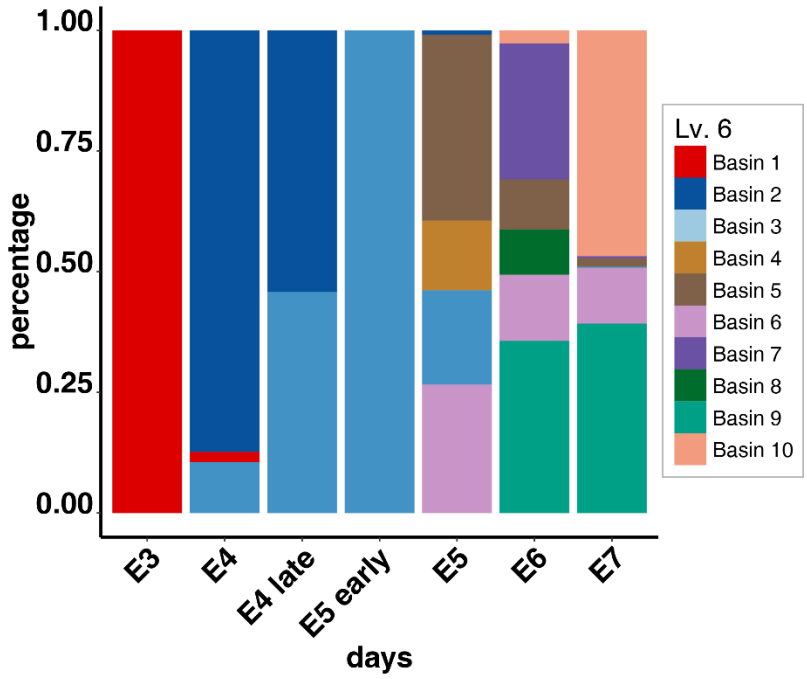
B. The heatmap of differentially expressed genes between these two basins.



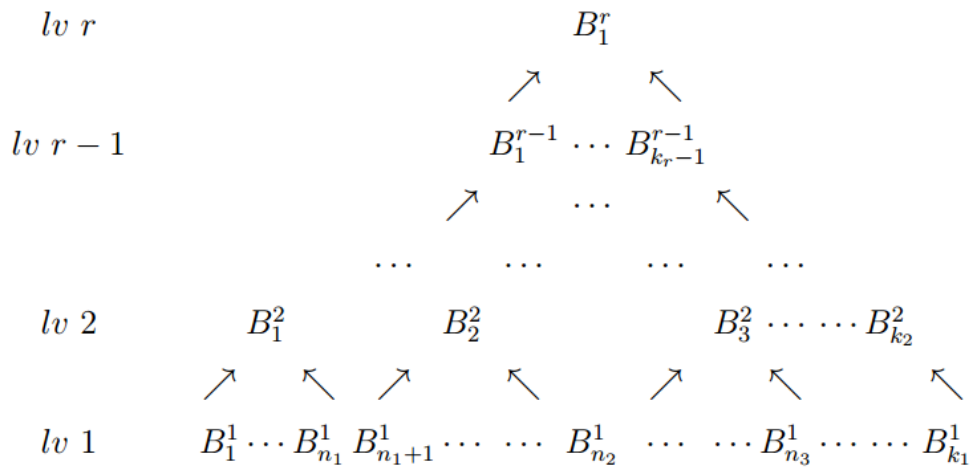
Supplementary figure 11 | The heatmap of differentially expressed genes between two basins in neuronal progenitor cells.



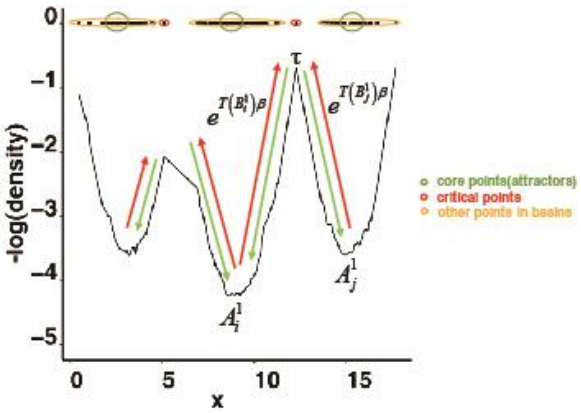
Supplementary figure 12 | GO terms of up-regulated genes in two subtypes in neuronal progenitor cells.



Supplementary figure 13 | The proportion of basins in each time point.

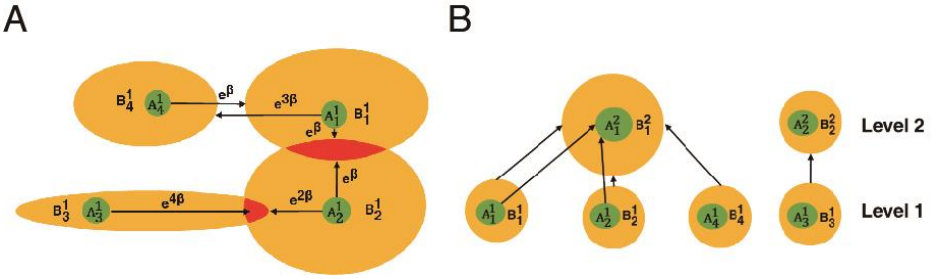


Supplementary figure 14 | The target hierarchical structure of MarkovHC. $\left\{B_i^j, i=1,...,k_j; j=1,...,r\right\}$ is the subset of $\left\{s_1,...,s_n\right\}$ and B_i^p is the subset of some basins on level p+1.



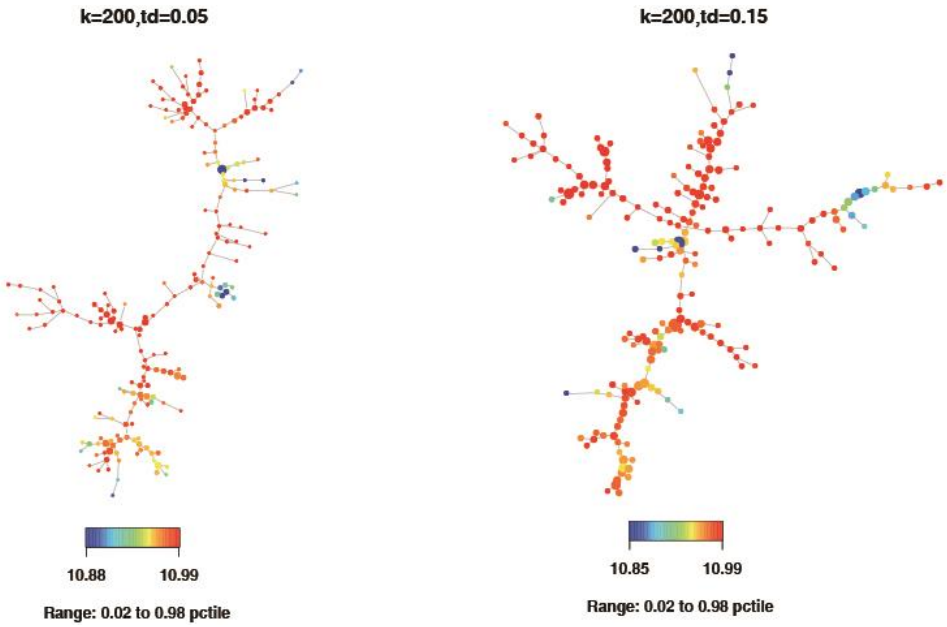
Supplementary figure 15 | 1-dimensional example of landscape. The points are 1-dimensional and they are plotted at the top of the panel. The value of vertical coordinate is the pseudo potential energy, which is defined as the negative logarithm of density here. The time scale for points in $A_i^1(A_j^1)$ to reach τ is $e^{T(B_i^1)\beta}\left(e^{T(B_j^1)\beta}\right)$ while the time scale for τ to reach $A_i^1(A_j^1)$ is a constant irrelative with β. ($T\left(B_i^1\right)$ can be viewed as the least consumption of pseudo potential power for points in A_i^1 to reach τ.) Therefore, it takes a long time for points in attractors to move to shared points while shared points can jump into attractors in a very short time. Of course, the time scales for points in A_i^1 and A_j^1 to move to τ are different. Such a difference is exactly the criterion used to define the relationship between level 1 clusters (basins) and to construct higher-level

Markov chains.

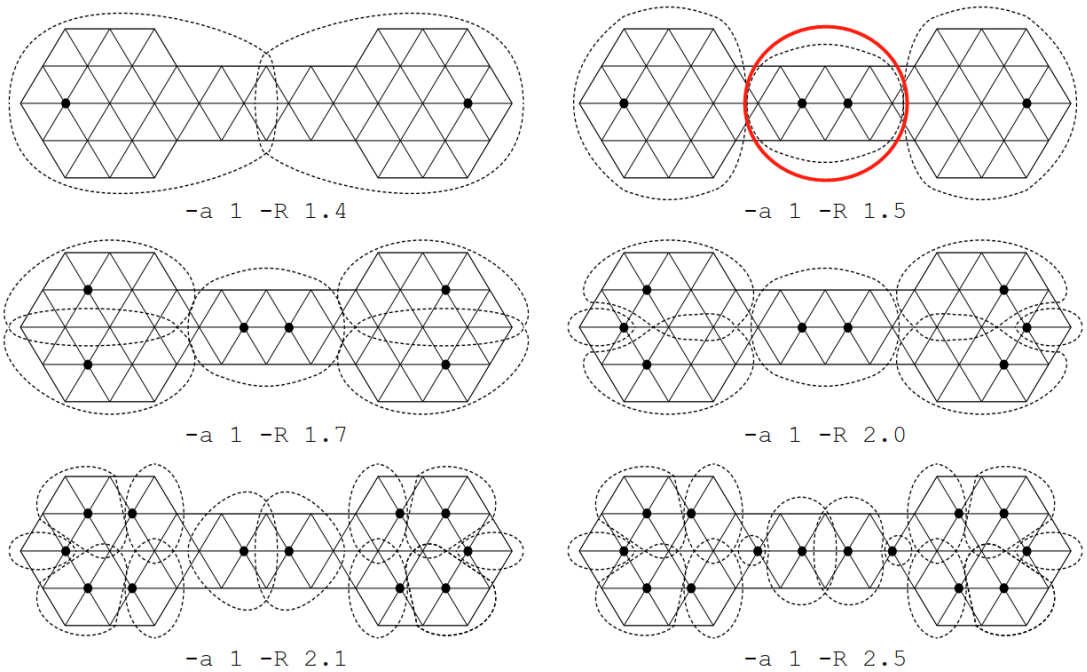


Supplementary figure 16 | The illustration of the criteria for basin merging.
A. Four basins on level 1.

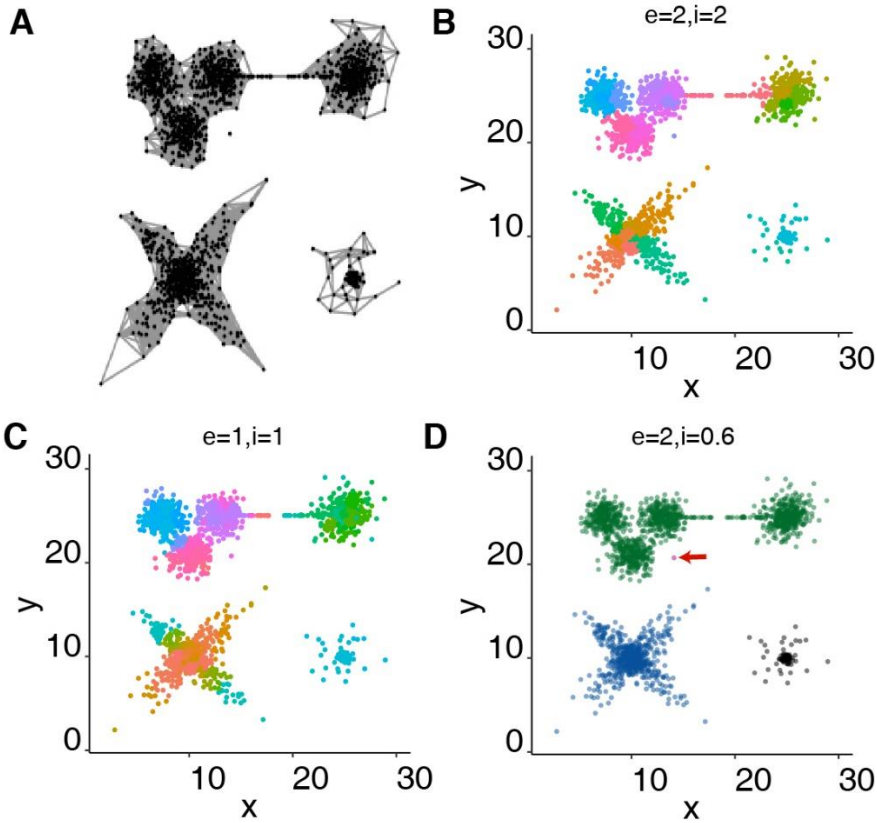
B. Basing on our criteria, $B_1^1 \cup B_2^1 \cup B_4^1$ is a new level 2 basin.



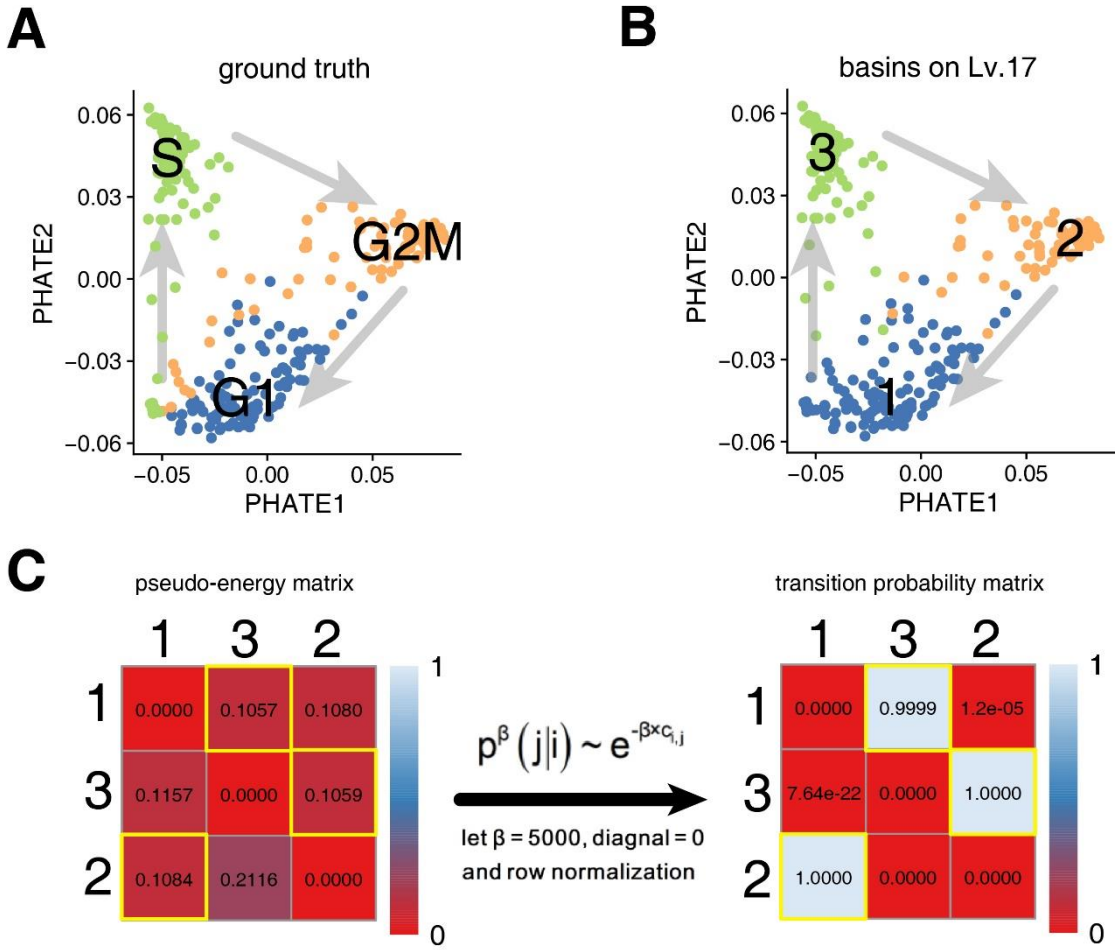
Supplementary figure 17 | Minimum spanning trees constructed by SPADE is not robust. We applied spade on the cytometry data of PBMC twice, and only changed one of insignificant parameters, downsampling_target_pctile, in the algorithm. The structures of MST (minimum spanning tree) were very different. For the stochastic process in the algorithm, even we ran it twice with same parameters, the MSTs were different either.



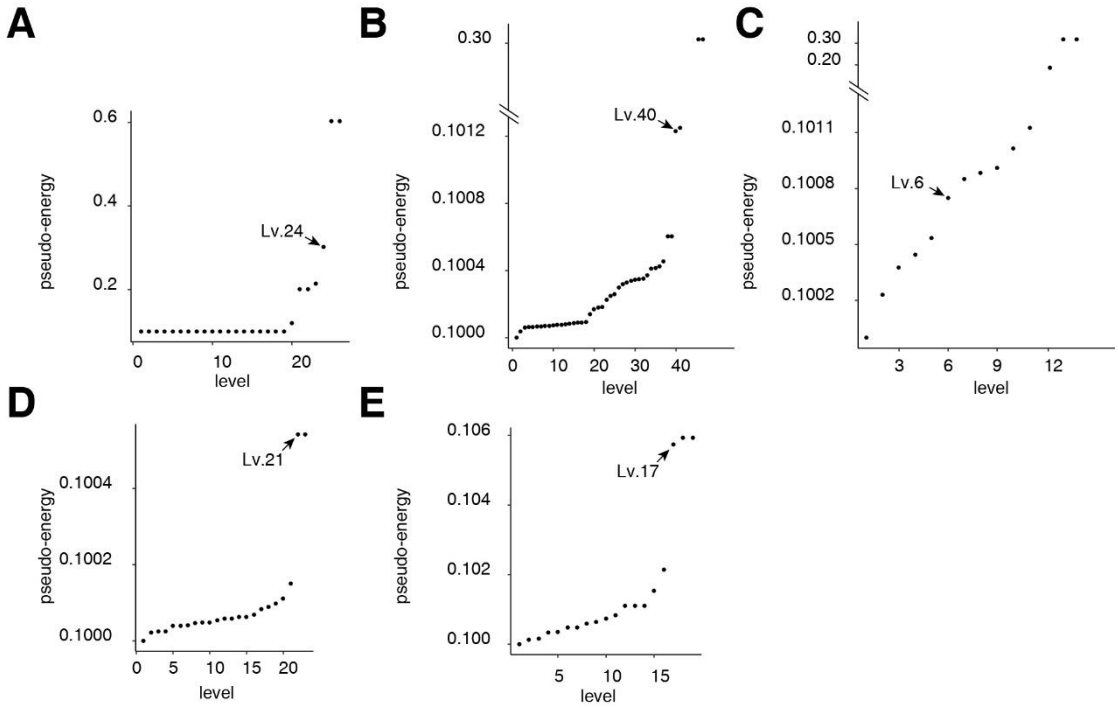
Supplementary figure 18 | Clusterings resulting from the MCL algorithm for the graph in Van Dongen's Ph.D. thesis.



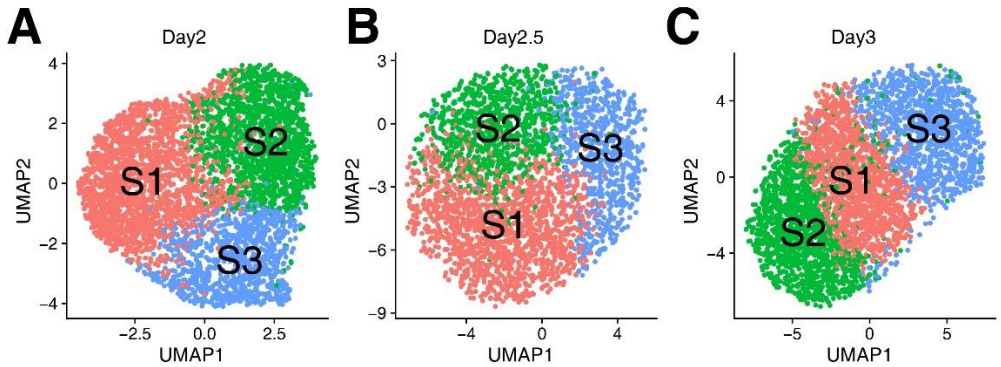
Supplementary figure 19 | Clustering results of MCL on simulation data in Fig.2 F.



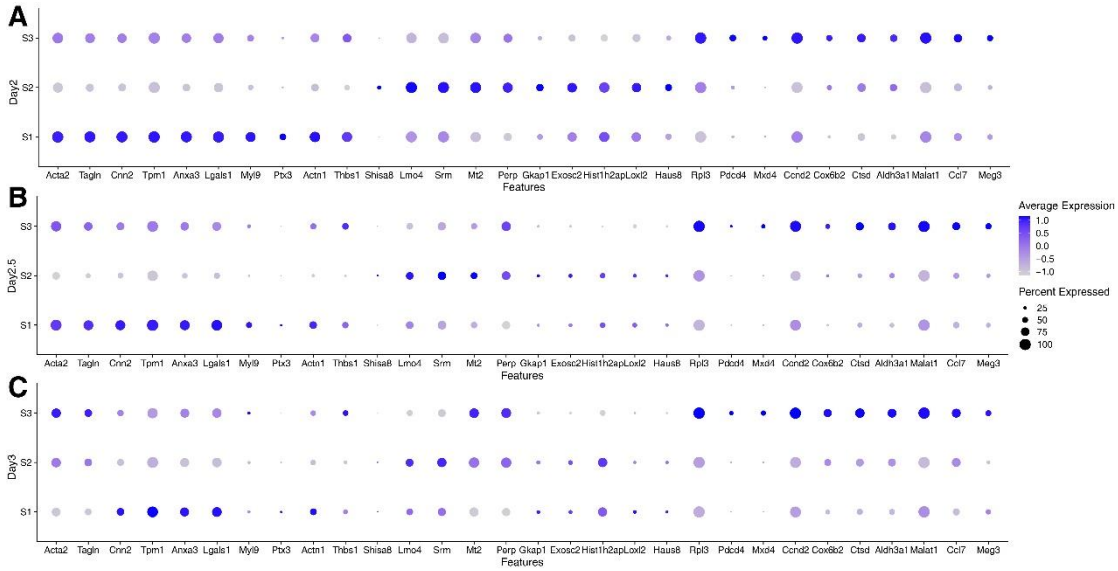
Supplementary figure 20 | MarkovHC can reveal non-equilibrium biological processes.



Supplementary figure 21 | Level selection strategy.



Supplementary figure 22 | The sub-clusters in Day2, Day2.5 (the night in Day2), and Day3.



Supplementary figure 23 | The pattern of differentially expressed genes.

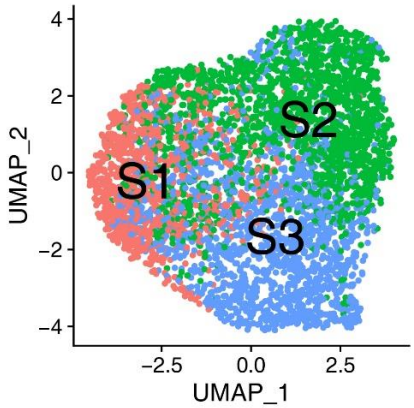
QP

	S1	S2	S3
S1	0	0.451	0.549
S2	1	0	0
S3	0.517	0.483	0

MarkovHC

	S1	S2	S3
S1	0	0.4508943	0.5491057
S2	0.5255155	0	0.4744845
S3	0.6175497	0.3824503	0

Supplementary figure 24 | P_{ij} matrix calculated by QP (left) and MarkovHC (right)



Supplementary figure 25 | Three basins clustered by MarkovHC (right)