Eurographics Conference on Visualization (EuroVis) 2015
H. Carr, K.-L. Ma, and G. Santucci
(Guest Editors)

Volume 34 (2015), Number 3

Feature-Driven Visual Analytics of Chaotic Parameter-Dependent Movement

M. Luboschik[1], M. Röhlig[1], A. T. Bittig[1], N. Andrienko[2], H. Schumann[1], C. Tominski[1]

[1]Institute for Computer Science, University of Rostock, Germany
[2]Fraunhofer IAIS Bonn, Germany and City University London, UK

Abstract

Analyzing movements in their spatial and temporal context is a complex task. We are additionally interested in understanding the movements' dependency on parameters that govern the processes behind the movement. We propose a visual analytics approach combining analytic, visual, and interactive means to deal with the added complexity. The key idea is to perform an analytical extraction of features that capture distinct movement characteristics. Different parameter configurations and extracted features are then visualized in a compact fashion to facilitate an overview of the data. Interaction enables the user to access details about features, to compare features, and to relate features back to the original movement. We instantiate our approach with a repository of more than twenty, accepted and novel, features to help analysts in gaining insight into simulations of chaotic behavior of thousands of entities over thousands of data points. Domain experts applied our solution successfully to study dynamic groups in such movements in relation to thousands of parameter configurations.

Categories and Subject Descriptors (according to ACM CCS): Human-centered computing – Visualization – Visualization application domains – Visual analytics

1. Introduction

Visual analytics has become an indispensable means to help us understand the characteristics of movements in space and time [AAB*13a]. Here, we address movements that were synthesized in an effort to simulate processes that are difficult to observe otherwise. Such simulations are typically controlled by parameters whose influence on the simulation outcome is not clear upfront. So, in addition to investigating movement in space and time, there is also the need to understand the movement's dependency on the parameter configuration.

The analysis of parameter dependencies is a rather challenging issue [OJ14], particularly for simulations of complex movements. A reason is that we have to integrate the visual representation of parameter configurations with the corresponding movement in a comprehensible way. On top of that, there might be thousands of different configurations, each resulting in thousands of unconstrained or even chaotic movements. In such cases, severe clutter and over-plotting will make it hard to discern even basic movement characteristics from the data, not to mention gaining insight into the influence of parameters.

The related work reviewed in Section 2 indicates that extracting and visualizing high-level features can be more appropriate than showing the raw data. For example, time-evolving features have been used successfully to explore and compare single group movements [vLBSF14]. However, approaches are still lacking to analyze *all* movements belonging to a specific parameter configuration and to explore these in regard to *all* alternative configurations. To close this gap, we tightly integrate analytic, visual, and interactive means in a novel visual analytics approach for studying chaotic movement data with regard to the underlying parameterizations. The abstract outline of our approach is as follows:

Analytic: We extract high-level features to capture the characteristics of *all* movements belonging to a specific parameter configuration. We consider basic features, group features, and region features. Advanced features further increase the level of abstraction.

Visual: We visualize the features via a novel visual design that integrates (i) an overview of *all* movements conjoining feature and parameter distributions and (ii) detail views reflecting certain aspects of the high-level features back onto the low-level raw data.

Interactive: We provide interaction techniques facilitating the analysis of parameter dependencies of movement features, the exploration of feature evolution with regard to individual parameter configurations, and the comparison of features corresponding to different configurations.

The analytic feature extraction and the interactive feature visualization are detailed in Sections 3 and 4, respectively. In Section 5, we demonstrate our solution by applying it to analyze chaotic movements simulated for thousands of different parameter configurations over thousands of time steps.

2. Motivation and Related Work

Next, we outline the motivation for our research and discuss the related work in visual analytics of movements and parameter dependencies.

2.1. Motivation

Our work is motivated and driven by recent advances in systems biology. In particular, research on spatial simulation has gained momentum as it expands our ability to understand biological phenomena [Kho06, TTNtW10, HBRU13]. The key idea is to abstract from nature's details and create generic models of biological processes. Some of the details abstracted away during the modeling are captured in parameters to be experimented with when simulating the models. Therefore, multiple simulation runs with different parameter configurations are necessary. The simulation generates large data sets containing parameter-dependent spatial and temporal information about the entities and their movements.

As a concrete example, we consider the investigation of dynamic interactions between *receptor proteins* and *lipid rafts* on the surface of human cells [NBPH06]. These interactions play an important role in cellular signaling, for instance, in the cancer-related *Wnt* pathway [KYS09].

Studying such dynamic interactions is a task that is typically difficult to carry out. There are several reasons for that. First, the spatial simulation is based on stochastic Brownian motion. The resulting movement trajectories are in a sense *chaotic*, because they are entangled to a large degree. Second, the entities may pick up, take along, and drop other entities during the simulation. This way, they form *dynamic groups*, which are of high interest, but difficult to grasp. Third, the simulated interactions *depend on the parameter configuration*, where the impact of individual parameters or combinations of specific parameter values is largely unknown.

Our objective is to develop a solution that helps analysts to unveil the influence of parameters on the movement dynamics so that they can evaluate the simulation approach in itself and confirm or reject hypotheses about the underlying biological model. Although we address chaotic movements from systems biology, our approach is generic enough to be applicable to other problems as well.

2.2. Related Work

Our research is related to visual analytics of movement and visual analytics of parameter dependencies.

Visual Analytics of Movement concentrates on (1) visualizing spatial and temporal aspects of individual trajectories and sets of trajectories, (2) visualizing movement attributes along trajectories, (3) detecting stops, interactions between trajectories, and other kinds of events, (4) aggregating movement data in space and time and visualizing the resulting aggregates, and (5) revealing relationships between movement and the environment. A profound overview and systematization can be found in [AAB*13a]. We review existing work with regard to chaotic movements and movements of dynamic groups, which are key aspects of our research.

A general problem when visualizing chaotic movements is the severe over-plotting. Typical approaches to tackle over-plotting include clustering [RPN*08], aggregation with density kernels [WvdWvW09], or flow maps [WSD11], which provide summaries of the underlying data. However, with the chaotic movements that we address in our work, these approaches are likely to fail.

Another widely accepted approach to deal with complex data is feature visualization [RPS01]. The basic idea is to visualize derived features, rather than the raw data. Feature-based approaches have already been used successfully for analyzing movements of groups [vLBSF14]. Group movements are also studied in [AAB*13b], yet without following a feature-based approach. In both works, the groups are static, that is, group membership is not allowed to change. Dynamic groups are addressed in [RTBW*09], but only for rather few simple movements.

Visual Analytics of Parameter Dependencies deals with visualizing given input parameters of configurable processes along with the corresponding output. A recent survey can be found in [SHB*14].

Related to our work are global-to-local approaches, which provide an overview in the beginning and allow the user to drill down into details. An example is the overview-driven approach for parameter dependencies of large time series data [LRHS14]. However, the specifics of chaotic parameter-dependent movement have not been addressed yet.

To be able to generate overviews it is often necessary to reduce the number of parameter configurations and output size. A typical way to do so is to use surrogate models, which predict or interpolate the corresponding output [PBK10, TWSM*11, BPFG11]. Undirected optimization [MAB*97, BM10] and parameter space partitioning [BSM*13] are approaches that make use of reduced result previews or clustering. A problem with these approaches is that they cannot be applied directly to chaotic movements, which can be difficult to predict, interpolate, reduce, or cluster, if this is possible at all.

With regard to movement analysis and parameter dependencies, we also found related work in the flow visualization literature [VP04,GYHZ13,vPGL*14]. These approaches basically superimpose results of different flows, which are each filtered to a reduced set of streamlines, which in turn are comparable to our raw data trajectories. Given a certain degree of spatial similarity among the resulting trajectories, direct visual comparison of different parameter inputs becomes possible.

Recently, the idea of using features has also been applied to analyze parameter dependencies of complex simulations [MGS*14]. In contrast to the single-valued scalar features used there, we propose using time-evolving features. In doing so, we extend previous feature-based approaches for movement analysis [vLBSF14], as indicated earlier.

In summary, we see several individual solutions, but none that suits our needs directly. Therefore, our goal is to develop a feature-based approach that works with chaotic movements and dynamic groups, and that also supports the analysis of parameter dependencies. To achieve this goal, we (i) introduce tailored features to capture key characteristics of chaotic movements, including dynamic groups, (ii) visualize the features in association with parameter configurations to enable users to analyze their dependencies, and (iii) integrate appropriate interaction to allow users to look into details and compare different aspects of the data.

3. Feature Extraction

We consider data of the following form. A data set $D = \{R_1,\ldots,R_r\}$ consists of r simulation runs. Each run $R_i = (P_i, M_i)$ with $1 \le i \le r$ is a pair of a parameter configuration P_i and the corresponding movement M_i. A movement $M_i = \{T_1,\ldots,T_m\}$ consists of the trajectories of m moving entities. The trajectories are sampled at uniform time intervals so that we obtain for each trajectory a set of n points $T_j = \{\mathbf{t}_1,\ldots,\mathbf{t}_n\}$, where $1 \le j \le m$. The points store information about an entity at a particular time. This includes information that is readily available such as the entity's position or type, but also derived attributes such as speed, acceleration, or the distance to particular other entities.

The analytic part of our approach is to condense the complex data down to information that is manageable and relevant. Our method of choice is feature extraction. The feature extraction is based on two principal steps, which are carried out for each movement M_i. First, the data points are enriched with derived measures. This is to inject into the data meaningful information that can help to characterize the movement. The second step is aggregation. As illustrated in Figure 2 (a), the goal is to reduce M_i with its m trajectories consisting of n points to a single aggregated feature time series of length n. To this end, we consider all points $\mathbf{t}_k$ of all m trajectories and aggregate them to a single feature value f_k. As we do this for all $1 \le k \le n$ time steps, we get n feature values $f_1,\ldots,f_n$ that characterize the movement over time.

The net effect is that we replace the complex movement M_i by a feature $F_i = \{f_1,\ldots,f_n\}$. The difficult problem of visualizing (P_i, M_i) for $1 \le i \le r$ is thus reduced to the problem of visualizing (P_i, F_i), which is much easier to solve as we will see in Section 4.

Apparently, a single feature alone will not suffice to capture the richness of movement data. Therefore, we consider a repository of feature definitions in four categories: (1) basic features, (2) group features, (3) region features, and (4) advanced features. Basic features capture general movement characteristics. Group features address characteristics of dynamic groups, such as fluctuations in memberships and retention periods. Region features perform more complex spatio-temporal aggregations to characterize regions of interest and their evolution over time. Advanced features are a way to extract features over features.

Next, we describe exemplary features from all four categories. For the sake of brevity, we restrict ourselves to brief informal explanations. For a complete list of features, including design rationales developed with domain experts, we refer to the supplemental material.

Basic Features General characteristics of movement trajectories can be captured by aggregating basic properties, such as speed, direction, and distance of movements. In addition to considering such features across all moving entities, our approach can compute them also with respect to entities of different types (e.g., receptor proteins or lipid rafts). This allows us to investigate the entire movement as well as specific subsets of moving entities. Some aspects of the movement behavior can for instance be characterized by averaging the distances of all entities of one type to the closest entity of another type.

Group Features As indicated earlier, previous work on analyzing groups mostly considered *static* groups. We are interested in *dynamic* groups, that is groups that emerge, continue to exist with changing members, and decay.

We build upon previous work on non-spatial groups, for example, tracking changes in group sizes [BvLA*11] or changes of structural properties [TPRH11]. To capture the dynamic behavior of spatial groups and to analyze how the behaviors of group members and non-members differ, we specify several group features:

Group count: basically captures the number of existing groups per time step.

Group affiliation ratio: describes the overall ratio of group members and entities not being contained in any group.

Group load: relates the actual group sizes to the maximum allowed capacity of groups.

Group retention period: captures the time period between entities joining and leaving a group accumulated for all current group members. This measure can further be aggregated for all groups to describe the temporal fluctuation of group memberships.

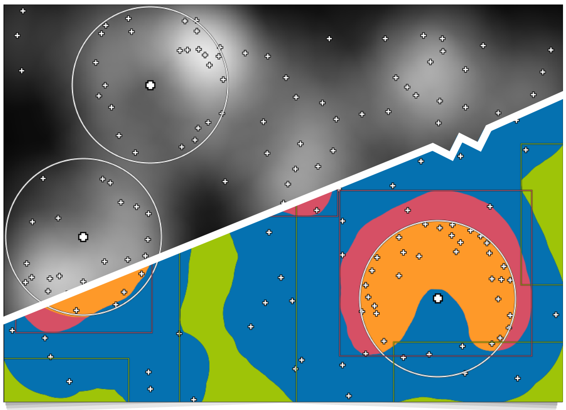


Figure 1: Feature extraction via density maps. Entities (white marks) and groups (circles) superimposed on a density map (top) and extracted regions (bottom) with high (red) and low (green) density and overlaps with groups (orange).

Region Features The feature definitions described so far focus on basic movement characteristics derived directly from the movement trajectories. A limitation of these features is that high-level spatial or temporal characteristics can not be captured well. However, looking at spatial patterns and their temporal evolution is often necessary to fully understand movements and parameter influence.

To better capture spatial aspects, we integrate 2D density maps [DV10] when computing features. Density maps are computed for all time steps, effectively creating a 3D space-time density volume. This allows us to capture generic patterns independently of data size and specific data properties, such as extents of entities or topologies of groups [JYJ11].

The 2D density maps are further analyzed to extract regions of interest with low and high density regarding suitable thresholds. Figure 1 illustrates entities, groups, density map, and extracted regions. Further tracking and aggregating properties of these regions of interest over time enables us to extract spatio-temporal features:

Region count: is the number of disjoint regions of interest (e.g., high or low density regions) per density map.

Region size: corresponds to the aggregated size of all regions of interest per density map.

Region ratio: is generally applied to relate regions of interest with respect to their density (e.g., low and high density regions) and with regard to certain types of entities (e.g., high density regions for one entity and low density regions of another entity).

Advanced Features With the features introduced so far it is possible to study a variety of movement characteristics. To be able to combine features and to generate even higher level abstractions, we introduce the notion of advanced features, i.e., features over features.

Advanced features can be derived by further analytical processing of previously extracted features. For instance, by applying temporal clustering of feature values it is possible to investigate temporal patterns across multiple simulation runs, such as common feature evolution or time periods of specific behavioral variation. We generate features over features via a self-organizing map (SOM), by which we obtain clusters with similar feature characteristics over time. An exemplary use for such features is to verify that stochastic simulations indeed do not exhibit periodic temporal patterns.

In summary, the feature extraction computes analytic abstractions to capture key characteristics of the movement. We consider a wide variety of feature definitions as collected in our feature repository, which is available as supplemental material. Next we describe how the features are visualized in relation to parameter configurations.

4. Feature Visualization and Interaction

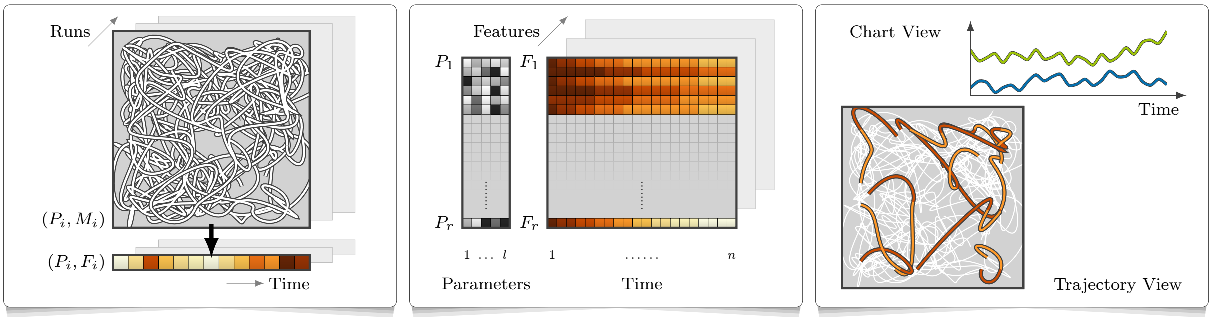
We study parameter configurations and associated movements (P_i, M_i) for multiple simulation runs $1 \leq i \leq r$. In terms of parameters, we define a parameter configuration $P_i = \{p_1, \ldots, p_l\}$ as a set of l parameter values. The number of parameters l is constant for all simulation runs. As illustrated in Figure 2 (a), the analytic feature extraction already reduced the complex movements M_i in space to time series of feature values of the form $F_i = \{f_1, \ldots, f_n\}$. To give a rough measure of the size of our data, the number of parameters l is around ten, simulation runs r can be in the thousands, and time steps n can be in the thousands as well. Section 5 provides more precise numbers for a use case in systems biology.

Our primary objective is to support the exploration and analysis of the aforementioned data. This involves several visualization tasks, which can be differentiated into overview tasks and detail tasks.

Overview: At the overview level, users explore the data with respect to temporal evolution of features in relation to all parameter configurations. The goal is to identify interesting patterns and to analyze them with regard to the underlying movements.

Detail: For a more detailed investigation, the analysis is focused on selected simulation runs with their parameter configurations and corresponding feature values. Focusing on selected runs enables users to compare interesting patterns in detail and to gain a better understanding of the influence of parameters. Since our features sacrifice spatial information for the sake of analytic abstraction, we also have to support linking back the analysis to the spatial domain, at least partially.

With these data definitions and visualization tasks in mind, we developed a dedicated visualization design based on linked overview and detail views. Figure 2 summarizes the overall strategy of our visual analytics solution.



(a) Analysis. (b) Overview. (c) Detail.

Figure 2: Analytically extracted high-level features are visualized via interactive overview and detail representations.

4.1. Overview Visualization

The overview task focuses on exploring movement characteristics and parameter dependencies across all simulation runs. Therefore, we have to visualize all parameter configurations P_i and associated feature time series F_i for all $1 \leq i \leq r$. To this end, the overview presents the data in a compact matrix-like fashion.

Visual Encoding In the matrix, the i-th row represents the i-th simulation run (P_i, F_i). The first part of a row visualizes the parameter configuration $\{p_1, \dots, p_l\}$ and, separated by a small gap, the remainder of the row visualizes the feature values $\{f_1, \dots, f_n\}$. This arrangement is illustrated in Figure 2 (b). Note that the matrix shows only one feature definition. Yet, switching between different feature definitions from the feature repository is possible at any time.

The cells of the matrix are color-coded using distinct color maps for parameter values and feature values. For the quantitative feature values we apply color maps from ColorBrewer [HB03]. Parameter values are color-coded with hue-less shades of gray. This clearly separates feature values (colors w/ hue) from parameters (colors w/o hue). Darker shades of gray represent lower parameter values and brighter shades stand for higher values. If required, the default color-coding can be interactively adjusted.

Sorting When displaying the data of thousands of simulation runs in a row-wise fashion the order of rows is vital for discovering patterns in the data. Because fully manual sorting is impractical, we provide two ways for automatic sorting: (1) parameter-based sorting and (2) feature-based sorting. Sorting based on parameter values facilitates the interpretation of parameter influence on the data, e.g., for hypothesis testing regarding the parameters. On the other hand, sorting based on feature values helps to identify simulation runs with similar behavior, e.g., to build hypotheses when inspecting the related parameters.

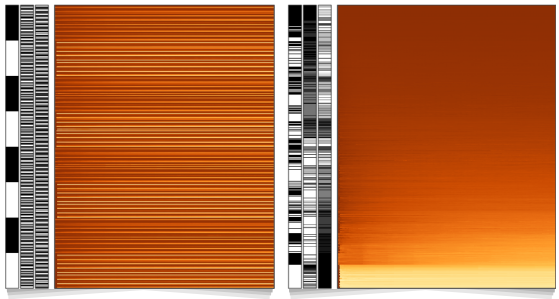


Figure 3: Overviews sorted row-wise by parameter configurations (left) and according to feature behavior (right).

While sorting individual values is trivial, sorting sets of values (our $\{p_1, \dots, p_l\}$ and $\{f_1, \dots, f_n\}$) is rather challenging. We provide various metrics and algorithms for automatic sorting, including Euclidean, Hausdorff, Fréchet, and Levenshtein distance, and average squared error, self-organizing maps (SOM), and dynamic time warping [GH97, SC07]. In general, we achieved good results when using SOM for feature-based sorting and averaged Euclidean distance for parameter-based sorting. Figure 3 depicts the visual effect of sorting on the emergence of patterns.

As sorting according to all parameters or all time steps might not lead to the desired insight, the user can choose to restrict the sorting to subsets of parameters or time steps. To support further data exploration, it is possible to experiment with the different sorting methods and apply them to different parameter subsets or time intervals. If the automatic sorting is still not satisfactory, there is always the option to reorder individual rows or groups of rows manually.

Interactive Exploration Given the size of the data, showing all observed time steps and all parameter values for all simulation runs can easily exceed the available screen space. Therefore, the matrix resides in a zoomable space allowing independent scaling along the axis of simulation runs (rows) and along the time axis (columns). By incorporating interactive zooming and panning, users are enabled to steer the visual analysis process according to their task-specific needs.

To help users in maintaining orientation during exploration, we integrate additional visual cues. Miniature scroll bars indicate where the current view is located with respect to the entire data representation. Further, we use two over-plotting indicators. They tell the user whether simulation runs (rows) and/or time steps (columns) are affected by over-plotting. Red indicators warn the user that perceived patterns could be artifacts due to over-plotting. To resolve such ambiguities quickly, the over-plotting indicators can be clicked to smoothly animate the view to a zoom level where no over-plotting occurs.

Equipped with these interaction facilities, our compact matrix-like visualization provides an overview and supports the identification of basic value distributions and temporal patterns. For example, constant feature values at specific time intervals or uniform feature evolution over time are reflected by rows with ranges of constant or gradually changing colors, respectively. Furthermore, dependencies between parameters and features (and hence the underlying movements) can be discerned by looking vertically for patterns under different orderings. Absence of vertical patterns may also indicate weak or no parameter influence. Although the overview can already lead to specific insights its real value is to initiate more targeted follow-up investigations.

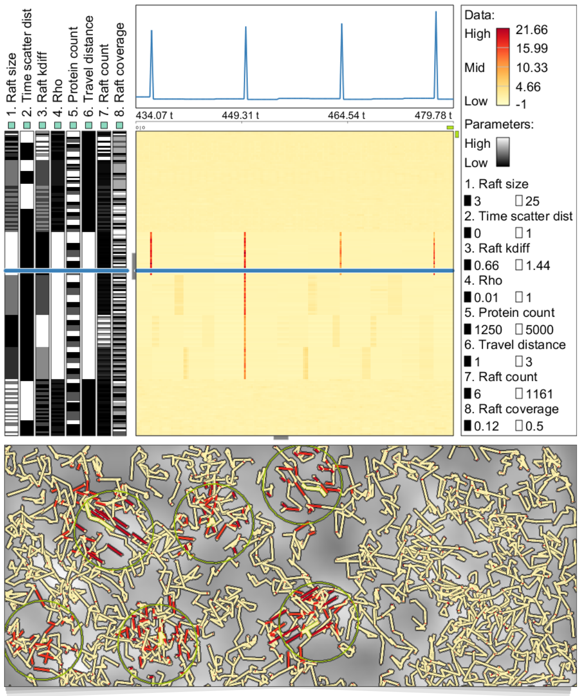


Figure 4: The overview of parameters and features (center) in conjunction with the chart view (top), the trajectory view (bottom), and a legend (right) facilitate spatio-temporal data exploration.

4.2. Detail Visualization

To go beyond overview visualization and basic exploration, we integrate techniques for detailed comparison and re-establishing the spatial context, as illustrated in Figure 2 (c). These techniques are key to exploring movement data in changing analysis scenarios.

Detailed Comparison More targeted investigation typically means comparing selected subsets of the data in detail. However, color-coded visual representations are less suited for analyzing and comparing numerical values in detail [LMK07]. Therefore, we integrate a separate chart view.

As illustrated in Figure 4 (top), the chart view shows time series plots representing feature values for simulation runs selected from the overview. Selections and plots are associated with unique highlighting colors to make them easier to distinguish and track across the visualization. In Figure 4, the time series plot in blue corresponds to the selected simulation run indicated by a blue bar across the main matrix. The chart view is positioned above the overview and is aligned horizontally to maintain the temporal context. Additionally, zoom and pan operations in time are linked to preserve temporal alignments between the overview and the line plots.

By showing selected time series as line charts we facilitate a more precise analysis and direct comparison of simulation runs. This allows the user to focus on specific parameter combinations (e.g., similar parameter values) and to compare related feature characteristics. The other way around, the user can also start with interesting patterns of feature values and inspect their relation to the associated parameters.

Relating Back to Space Our approach is based on analytical abstractions of the rather complex and even chaotic movements. These abstractions make it possible to reduce the amount of data to be displayed at a time. Yet, this comes at the cost that the spatial context and the influence of individual movement trajectories is lost to some extent.

To compensate for this, we incorporate an additional trajectory view that relates features back to the raw movement data $M_i = \{T_1, \ldots, T_m\}$, but for selected simulation runs only, as indicated in Figure 4 (bottom). The movements are shown as trajectories $T_j = \{\mathbf{t}_1, \ldots, \mathbf{t}_n\}$. Spatial aspects and feature characteristics are married by combining a spatial layout based on the trajectory points $\mathbf{t}_1, \ldots, \mathbf{t}_n$ with a color-coding based on the feature values $f_1, \ldots, f_n$. Moreover, the trajectory view can be blended with a selected 2D density map generated during the feature extraction. Figure 4 (bottom) shows a gray-scale density map in the background.

As color-coding and density maps establish a connection between feature values and locations where the raw movements took place, the spatial context is partially restored. Linked zooming and panning in time further strengthens the connection to temporal aspects of the raw movement trajectories. Focusing on a selected temporal interval of interest also significantly reduces visual clutter. This helps to investigate relationships between certain patterns of feature values and observed movement behaviors. For example, the user might be able to relate feature values representing low movement speeds to spatial conglomerations of trajectories.

The chart view for detailed comparison and the trajectory view for linking to spatial aspects complete our feature-driven visual analytics approach. In the next section, we apply this approach to a problem from systems biology.

5. Application to Systems Biology

The approach presented so far is a general solution applicable to different types of parameter-dependent movements. Yet our work has been largely motivated and driven by applications in systems biology. In the following, we present a use case where domain experts apply our solution to study dynamic interactions between receptor proteins and lipid rafts on the surface of human cells.

The data were generated using an ML-Space simulator in combination with movement synthesis based on Brownian motion [BHMU11]. Several properties describe the lipid rafts and proteins, including position and size. Lipid rafts and proteins move according to a diffusion parameter k_D. The Brownian motion is simulated by individually calculating displacement vectors with a random direction and a normally distributed average length depending on the smallest entity. During the movement, dynamic groups are formed by proteins docking to lipid rafts. Proteins enclosed in lipid rafts move along with them depending on their fluidity factor *rho*, which also controls the probability of proteins leaving the lipid raft.

Movement updates also include collision detection. Overlaps between entities of the same type (i.e., protein with protein and lipid raft with lipid raft) are prohibited and are resolved stochastically. Collisions of a protein with a lipid raft are handled by pushing the protein a little further so that it is either fully inside or outside the lipid raft depending on if the protein is entering or leaving the lipid raft.

In summary, eight parameters control the simulation, including fluidity, entity size, entity counts, and traveled distances. The domain experts determined 1,968 meaningful parameter configurations for which simulations were run. Each simulation run describes the chaotic movement of up to 1,161 lipid rafts and 5,000 proteins depending on the parameter configuration. The individual simulations covered 4,000 time steps.

The domain experts applied our solution to analyze the simulation outcome. In a pre-process, all features of the repository were computed to allow the experts to quickly switch between different movement characteristics. Among others, the following results could be obtained.

Insights Related to Groups Figure 5 shows the parameter dependency of the *average group size* feature, which captures the average number of proteins inside lipid rafts over time. Although this feature is rather simple, it nicely illustrates parameter dependencies in our data. For this purpose, a hierarchical sorting has been applied based on the parameters *raft size*, *protein count*, and *rho*. At first glance, Figure 5 shows an overall temporal trend of low to high group sizes from left to right (a) and also a trend across simulation runs from bottom to top (b). The temporal trend (a) reflects the fact that lipid rafts start empty and collect proteins incrementally. The trend (b) represents the dependency of group sizes on the parameter *raft size*.

A second observation can be made by looking at the row-wise bands (c), which show different shades aligned with the

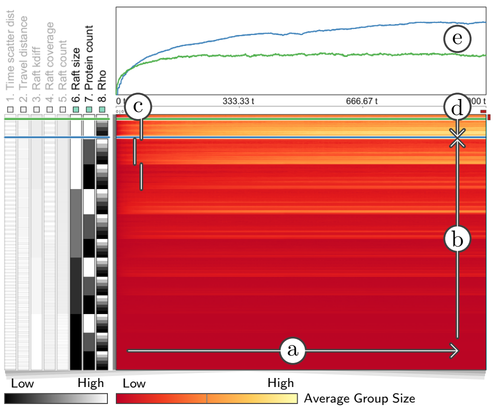


Figure 5: The visualization shows that average group size depends on the size of lipid rafts and the fluidity *rho*.

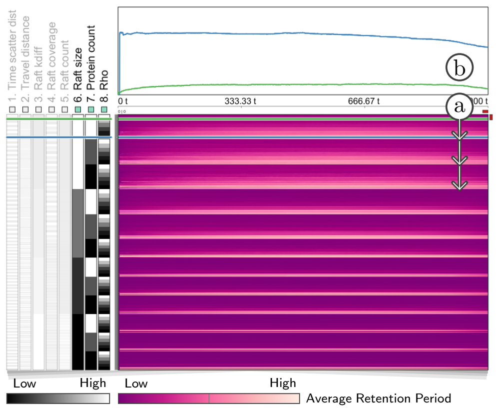


Figure 6: Visualizing average group retention confirms the influence of parameters *rho* and *raft size*.

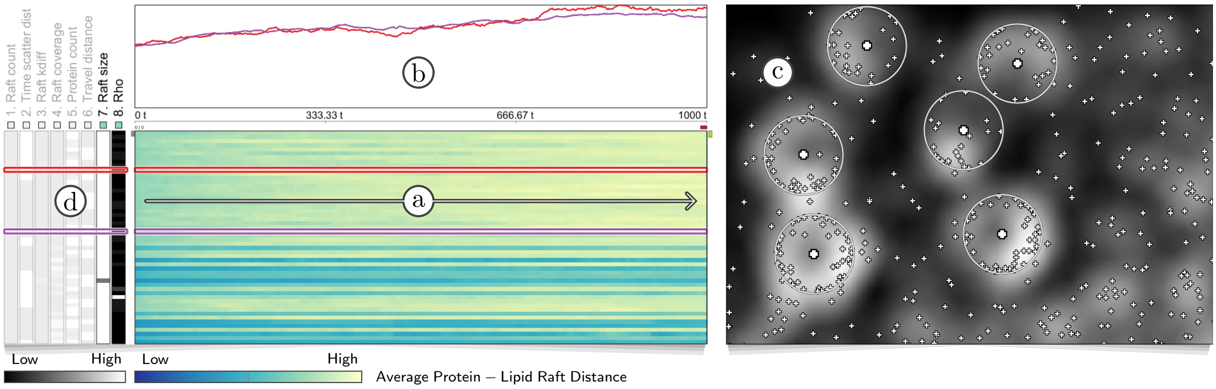


Figure 7: Visualizing the *average distance of non-member proteins to the nearest lipid raft* feature in conjunction with the detail line chart and a selected 2D density map helps in studying the *sweeping effect*.

parameter *protein count*. This dependency is due to the fact that large groups can only emerge if the number of potential group members is sufficiently high.

A third pattern (d) is visible within the bands. It appears to be related to the parameter *rho*. To study this pattern further, two simulation runs with potentially large groups (large raft size and high protein count) were selected and are shown in the chart view (e) in greater detail. The line chart reveals that low values of *rho* lead to constantly increasing group sizes (blue line), whereas high values of *rho* result in stagnation of group sizes below their potential maximum (green line).

Switching to the *average retention period* feature, as shown in Figure 6, while maintaining the selection and order of the simulation runs leads to further findings. Similar intra-band gradients (a) show the logical influence of *rho* on the duration for which proteins remain inside lipid rafts (b). It becomes clear that the stagnating group sizes (green line (e) in Figure 5) are caused by low retention periods (green line (b) in Figure 6). In other words, because of the high fluctuation, the lipid rafts drop the same number of proteins as they pick up, which inhibits further growth of groups.

Confirming the Sweeping Effect One particular pattern our domain experts were anticipating is the so-called *sweeping effect*. The sweeping effect relates to the fact that the space around lipid rafts is only sparsely populated. This phenomenon is due to the lipid rafts' random movement, which causes them to collect nearby proteins, effectively emptying the space around them. Visualizing the raw data trajectories of lipid rafts and proteins helps to identify this effect visually, yet only for a limited number of moving entities [LTB*12].

To investigate this effect, the domain experts set up the hypothesis that the empty space slowly emerges over time and thus the sweeping effect should become apparent by a steady increase of the average distance of non-member proteins to the nearest lipid raft. A corresponding feature was specified and extracted from the data. Figure 7 shows a SOM-sorted visualization of the feature. For several simulation runs, the hypothesized steady increase is particularly visible (a), making the sweeping effect quantitatively graspable for the first time. The result of the effect can be emphasized further by displaying the feature values of selected simulations as line charts (b) or by showing the 2D density maps for selected time points (c). Regarding parameter dependencies of the effect, the experts identified the parameters *raft size* and *rho* to have major influence. For example, (d) shows that only large lipid rafts with a low fluidity *rho* are capable of gathering and holding surrounding proteins in a way that establishes a noticeable effect.

The previous paragraphs illustrated how our approach can be applied to gain insight into chaotic movement simulations from systems biology. Next we briefly outline how we designed our solution together with simulation experts.

User Participation Our solution is the result of a participatory design process starting from prior work [LTB*12] and [LRHS14]. We cooperated with a group of five domain experts. The cooperation was of mutual benefit. We could build upon their domain expertise and devise and specify interesting movement behaviors as features. Collaborative data analysis sessions and informal user feedback helped us to design the visualization and the associated interaction so that they are indeed helpful to the analysts. While some design decisions were driven by the addressed data and tasks (e.g., compact color-coded matrix representation), others were inspired by the domain experts (e.g., chart view for comparison and trajectory view for linking back to space).

In turn, the experts benefited from our solution as it provided them with valuable *new* insight into their simulations, of which we could describe only a few here. The ability to explore and even compare different aspects of the chaotic movements of up to 25,000 entities across multiple parameter configurations was identified to be a major advantage.

Generalization While our use case focuses on data from systems biology, we envision applications in other fields as well. Particularly promising are parameter-dependent simulations of crowd behavior in mass events, which certainly involve dynamic groups and rather chaotic movement trajectories. But also collected real-world data without dependencies on parameters could be interesting to analyze. Examples are dynamic groups in sports (e.g., breakaway groups in cycling) or animal behavior in the wild (e.g., wolves patrolling their territory).

The visualization and interaction part of our solution will be one-to-one applicable to such data. In scenarios where no parameters are involved, the visualization could instead show the conditions or influential factors under which the data have been recorded. Also many of the features we described will be useful. Yet specific analytical questions may require the adjustment of existing features or the development of new ones to better address the particularities of the application.

Finally, it is obvious that our feature-driven approach is applicable to non-chaotic and also constrained movements, with or without the consideration of dynamic groups.

6. Discussion and Conclusion

We presented a visual analytics approach for parameter-dependent movements. The analytic extraction of features of different kinds opens up new possibilities for exploring unconstrained, crowded, and chaotic movements where the moving entities group dynamically. With the help of an overview visualization of parameters and features, users can spot interesting patterns. Selecting individual simulation runs allows the user to conduct in-depth inspections using a chart view and a trajectory view. Coordinated interaction facilitates data exploration. We can conclude that through combining analytic, visual, and interactive means, our approach is a useful aid for analyzing complex movements.

A limitation of our solution as well as any other feature-driven approach is that the expressiveness of the visualization is limited by the expressiveness of the features. This is why we rely on a feature repository to capture many data characteristics. Basic features support basic analytic tasks, whereas more complex features can provide more high-level insights. However, finding meaningful feature definitions that match specific data sets or analysis tasks is challenging in general. An interesting question is how we could support the user in designing feature definitions on the fly and in steering the feature extraction process while it is running. This requires assistance in evaluating how well a feature captures certain characteristics and to which extent individual trajectories influence the outcome of the feature extraction.

The examples we described here demonstrated that our approach is suitable for around ten of parameters. The limited number of parameters allowed us to simply unroll the parameter space to a linear order of parameter configurations and show the features with regard to them. Basic parameter dependencies could be revealed. Yet, complex influences of a larger number of parameters and high-dimensional correlations may be difficult to grasp. It remains to be studied how such scenarios can be handled. An interactive aggregation of multiple parameters or parameter configurations into clusters may be one option to investigate in the future.

We further plan to improve our approach based on the feedback from domain experts as well as on the insights gained from applying it to real world problems. One particular issue is that our visualization shows only one feature at a time. Comparing features by switching between different visualizations is not the best solution, because users have to memorize considerable amounts of information in their short-term memory, which makes the comparison error-prone. More work is needed to be able to show multiple features simultaneously. Cognitive constraints and screen space limitations will be challenging factors to deal with. Hence, it also makes sense to extend our advanced features to be able to capture characteristics of multiple source features. Such higher-level features would accumulate much more information, but would remain straight-forward to visualize.

Acknowledgements

The authors wish to thank the participating domain experts and Gennady Andrienko for conceptual input. This research has been supported by the German Research Foundation (DFG) via the research grant VASSiB (part of SPP 1335).

References

[AAB*13a] Andrienko G., Andrienko N., Bak P., Keim D., Wrobel S.: *Visual Analytics of Movement*. Springer, 2013. doi:10.1007/978-3-642-37583-5. 1, 2

[AAB*13b] Andrienko N., Andrienko G., Barrett L., Dostie M., Henzi P.: Space Transformation for Understanding Group Movement. *IEEE Trans. on Visualization and Computer Graphics 19*, 12 (2013). doi:10.1109/TVCG.2013.193. 2

[BHMU11] Bittig A. T., Haack F., Maus C., Uhrmacher A. M.: Adapting Rule-based Model Descriptions for Simulating in Continuous and Hybrid Space. In *CMSB* (2011), ACM. doi:10.1145/2037509.2037533. 7

[BM10] Bruckner S., Möller T.: Result-Driven Exploration of Simulation Parameter Spaces for Visual Effects Design. *IEEE Trans. on Visualization and Computer Graphics 16*, 6 (2010). doi:10.1109/TVCG.2010.190. 2

[BPFG11] Berger W., Piringer H., Filzmoser P., Gröller E.: Uncertainty-Aware Exploration of Continuous Parameter Spaces Using Multivariate Prediction. *Computer Graphics Forum 30*, 3 (2011). doi:10.1111/j.1467-8659.2011.01940.x. 2

[BSM*13] Bergner S., Sedlmair M., Möller T., Abdolyousefi S. N., Saad A.: ParaGlide: Interactive Parameter Space Partitioning for Computer Simulations. *IEEE Trans. on Visualization and Computer Graphics 19*, 9 (2013). doi:10.1109/TVCG.2013.61. 2

[BvLA*11] Bremm S., von Landesberger T., Andrienko G., Andrienko N., Schreck T.: Interactive Analysis of Object Group Changes over Time. In *EuroVA* (2011). doi:10.2312/PE/EuroVAST/EuroVA11/041-044. 3

[DV10] Demšar U., Virrantaus K.: Space-Time Density of Trajectories: Exploring Spatio-Temporal Patterns in Movement Data. *Int. Journal of Geographical Information Science 24*, 10 (2010). doi:10.1080/13658816.2010.511223. 4

[GH97] Garland M., Heckbert P. S.: Surface simplification using quadric error metrics. In *SIGGRAPH* (1997), ACM Press. doi:10.1145/258734.258849. 5

[GYHZ13] Guo H., Yuan X., Huang J., Zhu X.: Coupled Ensemble Flow Line Advection and Analysis. *IEEE Trans. on Visualization and Computer Graphics 19*, 12 (2013). doi:10.1109/TVCG.2013.144. 3

[HB03] Harrower M. A., Brewer C. A.: ColorBrewer.org: An Online Tool for Selecting Color Schemes for Maps. *The Cartographic Journal 40*, 1 (2003). doi:10.1002/9780470979587.ch34. 5

[HBRU13] Haack F., Burrage K., Redmer R., Uhrmacher A.: Studying the Role of Lipid Rafts on Protein Receptor Bindings with Cellular Automata. *IEEE/ACM Trans. on Computational Biology and Bioinformatics 10*, 3 (2013). doi:10.1109/TCBB.2013.40. 2

[JYJ11] Jeung H., Yiu M. L., Jensen C. S.: Trajectory Pattern Mining. In *Computing with Spatial Trajectories*. Springer, 2011. doi:10.1007/978-1-4614-1629-6_5. 4

[Kho06] Kholodenko B. N.: Cell-Signalling Dynamics in Time and Space. *Nature Reviews Molecular Cell Biology 7*, 3 (2006). doi:10.1038/nrm1838. 2

[KYS09] Kikuchi A., Yamamoto H., Sato A.: Selective Activation Mechanisms of Wnt Signaling Pathways. *Trends in Cell Biology 19*, 3 (2009). doi:10.1016/j.tcb.2009.01.003. 2

[LMK07] Lam H., Munzner T., Kincaid R.: Overview Use in Multiple Visual Information Resolution Interfaces. *IEEE Trans. on Visualization and Computer Graphics 13*, 6 (2007). doi:10.1109/TVCG.2007.70583. 6

[LRHS14] Luboschik M., Rybacki S., Haack F., Schulz H.-J.: Supporting the Integrated Visual Analysis of Input Parameters and Simulation Trajectories. *Computers & Graphics 39* (2014). doi:10.1016/j.cag.2013.09.004. 2, 8

[LTB*12] Luboschik M., Tominski C., Bittig A. T., Uhrmacher A. M., Schumann H.: Towards Interactive Visual Analysis of Microscopic-Level Simulation Data. In *SIGRAD* (2012), Linköping University Electronic Press. 8

[MAB*97] Marks J., Andalman B., Beardsley P. A., Freeman W., Gibson S., Hodgins J., Kang T., Mirtich B., Pfister H., Ruml W., Ryall K., Seims J., Shieber S.: Design Galleries: A General Approach to Setting Parameters for Computer Graphics and Animation. In *SIGGRAPH* (1997), ACM Press. doi:10.1145/258734.258887. 2

[MGS*14] Matković K., Gračanin D., Splechtna R., Jelović M., Stehno B., Hauser H., Purgathofer W.: Visual Analytics for Complex Engineering Systems: Hybrid Visual Steering of Simulation Ensembles. *IEEE Trans. on Visualization and Computer Graphics 20*, 12 (2014). doi:10.1109/TVCG.2014.2346744. 3

[NBPH06] Nicolau D. V., Burrage K., Parton R. G., Hancock J. F.: Identifying Optimal Lipid Raft Characteristics Required To Promote Nanoscale Protein-Protein Interactions on the Plasma Membrane. *Molecular and Cellular Biology 26*, 1 (2006). doi:10.1128/MCB.26.1.313-323.2006. 2

[OJ14] Obermaier H., Joy K.: Future Challenges for Ensemble Visualization. *IEEE Computer Graphics and Applications 34*, 3 (2014). doi:10.1109/MCG.2014.52. 1

[PBK10] Piringer H., Berger W., Krasser J.: HyperMoVal: Interactive Visual Validation of Regression Models for Real-Time Simulation. *Computer Graphics Forum 29*, 3 (2010). doi:10.1111/j.1467-8659.2009.01684.x. 2

[RPN*08] Rinzivillo S., Pedreschi D., Nanni M., Giannotti F., Andrienko N., Andrienko G.: Visually Driven Analysis of Movement Data by Progressive Clustering. *Information Visualization 7*, 3–4 (2008). doi:10.1057/palgrave.ivs.9500183. 2

[RPS01] Reinders F., Post F. H., Spoelder H. J.: Visualization of Time-Dependent Data with Feature Tracking and Event Detection. *The Visual Computer 17*, 1 (2001). doi:10.1007/PL00013399. 2

[RTBW*09] Reda K., Tantipathananandh C., Berger-Wolf T., Leigh J., Johnson A.: SocioScape – a Tool for Interactive Exploration of Spatio-Temporal Group Dynamics in Social Networks. InfoVis Poster, 2009. 2

[SC07] Salvador S., Chan P.: Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intelligent Data Analysis 11*, 5 (2007). 5

[SHB*14] Sedlmair M., Heinzl C., Bruckner S., Piringer H., Möller T.: Visual parameter space analysis: A conceptual framework. *IEEE Trans. on Visualization and Computer Graphics 20*, 12 (2014). doi:10.1109/TVCG.2014.2346321. 2

[TPRH11] Turkay C., Parulek J., Reuter N., Hauser H.: Interactive visual analysis of temporal cluster structures. *Computer Graphics Forum 30*, 3 (2011). doi:10.1111/j.1467-8659.2011.01920.x. 3

[TTNtW10] Takahashi K., Tănase-Nicola S., ten Wolde P. R.: Spatio-temporal Correlations Can Drastically Change the Response of a MAPK Pathway. *Proc. of the National Academy of Sciences 107*, 6 (2010). doi:10.1073/pnas.0906885107. 2

[TWSM*11] Torsney-Weir T., Saad A., Möller T., Hege H.-C., Weber B., Verbavatz J.-M.: Tuner: Principled Parameter Finding for Image Segmentation Algorithms Using Visual Response Surface Exploration. *IEEE Trans. on Visualization and Computer Graphics 17*, 10 (2011). doi:10.1109/TVCG.2011.248. 2

[vLBSF14] von Landesberger T., Bremm S., Schreck T., Fellner D. W.: Feature-Based Automatic Identification of Interesting Data Segments in Group Movement Data. *Information Visualization Journal 13*, 3 (2014). doi:10.1177/1473871613477851. 1, 2, 3

[VP04] Verma V., Pang A.: Comparative Flow Visualization. *IEEE Trans. on Visualization and Computer Graphics 10*, 6 (2004). doi:10.1109/TVCG.2004.39. 3

[vPGL*14] van Pelt R., Gasteiger R., Lawonn K., Meuschke M., Preim B.: Comparative blood flow visualization for cerebral aneurysm treatment assessment. *Computer Graphics Forum 33*, 3 (2014). doi:10.1111/cgf.12369. 3

[WSD11] Wood J., Slingsby A., Dykes J.: Visualizing the Dynamics of London's Bicycle-Hire Scheme. *Cartographica 46*, 4 (2011). doi:10.3138/carto.46.4.239. 2

[WvdWvW09] Willems N., van de Wetering H., van Wijk J. J.: Visualization of Vessel Movements. *Computer Graphics Forum 28*, 3 (2009). doi:10.1111/j.1467-8659.2009.01440.x. 2