

# Towards Association Based Spatio-temporal Reasoning

Hui Yang, Srinivasan Parthasarathy and Sameep Mehta

Department of Computer Science and Engineering

Ohio State University

2015 Neil Avenue, Columbus, Ohio, USA

{yanghu, srini, mehtas}@cse.ohio-state.edu

Keywords: Spatio-temporal reasoning, Spatio-temporal object association, Scientific data mining

## Abstract

In this paper, we present an association based approach towards spatio-temporal reasoning on scientific data. This work is built upon our previous work, where we proposed a general framework to discover multiple types of spatial association patterns in spatial data. We extend the framework to accommodate temporal information by generating spatio-temporal episodes. We then develop algorithms to show that such episodes can be used to reason about critical events and make inferences on time-varying interactions. We also present preliminary results on a simulation dataset drawn from Computational Fluid Dynamics (CFD) to validate the proposed algorithms.

## 1. Introduction

Analyzing spatial data is an important problem in many application domains, including geographical information systems, bioinformatics, scientific and engineering informatics, and computer aided design. The main difference between analyzing such data and data in a transactional form is that an object can influence the properties of objects located in the same spatial neighborhood [15], and therefore must be modeled in the analysis. Analyzing and reasoning about relationships among spatial objects is further complicated if the data is time varying in nature. Data produced from fluid dynamics simulations, is an example that has this additional constraint. Mining spatial relationships in this type of data is an important and interesting problem, since specific relationships may be indicators or predictors of upcoming events (e.g., vortex (hurricane) dissipation, crack propagation and amalgamation in materials).

Unfortunately, mining relationships among spatial objects is an extremely challenging task. First, almost all the related work done to date on this problem models each feature<sup>1</sup>

<sup>1</sup>We define a feature to be an object or entity of interest. A feature can have multiple attributes (including non-spatial ones) associated with it. For example, a vortex is a feature in a fluid flow simulation. Associated with this feature are spatial attributes such as location and non-spatial attributes such as velocity, swirl etc.

as a point in a multi-dimensional space [10; 11; 19]. However, especially in physical sciences, the extent and shape of a feature can play an important role in determining its influence on neighboring objects. Second, there is a strong need to develop techniques to capture and reason about the interactions among the features. These interactions if captured properly can help domain experts to understand the underlying processes in very effective manner. Third, one needs to develop effective techniques to incorporate temporal information in the overall analysis and reason about them. Finally, recent technological advances in computational sciences have resulted in huge amounts of data. Traditional statistical approaches to model such interactions do not scale very well to large datasets.

In our previous work [17], we proposed a general framework to address the first two challenges. We have developed algorithms to efficiently identify different types of Spatial Object Association Patterns (SOAP), where SOAPs are used to characterize the interaction among different types of features (objects) at a given moment. We have identified two SOAP types, namely, *Clique* and *Star* (Figures 2 (a)-(b)), in [17]. Another SOAP type, *Sequence* (Figure 2(c)), is identified in our later work [16]. The framework supports three shape representation schemes to capture the geometric properties of features. It also supports multiple distance metrics to measure distances between spatial objects.

In this work, we propose an association based approach to reason about the time-varying interactions among features. We develop a simple yet effective algorithm to model evolving SOAPS or sets of features in data produced by scientific simulations. We define and identify three types of events to describe SOAPS' evolutionary nature: *formation*, *dissipation*, and *continuation*. Formation and dissipation of a SOAP indicate the start and end of an interaction among involved features. Whereas continuation characterizes the stability of an interaction. The continuation of a SOAP from its formation to its next dissipation is abstracted as a *spatio-temporal episode*. Critical events such as feature amalgamation [14], can then be inferred by analyzing these episodes. Furthermore, by combining episodes associated with multiple SOAP types, we can

also model how the interactions among features evolve over time.

We present preliminary results on scientific simulation data drawn from Computational Fluid Dynamics to demonstrate the efficacy of the proposed approach. Other main challenges for this application include detecting, extracting, and classifying vortices, and then modeling different interactions among vortices. Many techniques have been proposed to detect, extract and classify vortices from such data in the past [7; 8; 18]. We have proposed a framework [17], to model different spatial interactions among objects. In this work we limit our discussion to the problem of spatio-temporal association based reasoning.

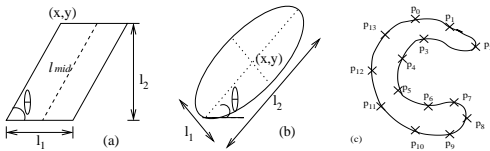
## 2. Background and Basic Concepts

Most of the content in this section can also be found in our previous work [16; 17]. We include it here to make this article self-contained.

### 2.1 Spatial Feature Representation

We propose three different representation schemes: parallelepiped ( or parallelogram in 2D), ellipsoid (or ellipse in 2D), and landmarks based representation, where landmarks are sampled boundary points [12]. These schemes can be used to model features from a variety of scientific domains. For instance, parallelograms are suitable to model non-local structures in protein contact maps. Whereas ellipsoids or ellipses are well-suited for vortices [13]. Finally, landmarks are very effective to model highly irregular-shaped features such as defect structures in materials [9]. The number of landmarks needed to represent a feature is domain dependent. The framework also supports basic shapes such as lines and splines.

As shown in Figures 1(a)-(b), the shape descriptor of a parallelogram or an ellipse can be described as a vector  $A_{basic} = \langle l_1, l_2, \theta \rangle$ . If landmarks are used (Figure 1(c)), the shape descriptor is  $A_{landmark} = \langle (x_i, y_i) : 1 \leq i \leq \nu \rangle$ , where  $(x_i, y_i)$  is the position of the  $i^{th}$  landmark. These shape descriptors can be easily extended to 3-D cases.



**Figure 1: Shape Representations: (a)Parallelogram (b)Ellipse (c)Irregular**

### 2.2 Dataset Representation

The dataset  $\mathbb{D}$  consists of  $n$  features extracted from  $r > 1$  maps, denoted as  $M = \{m_1, m_2, \dots, m_r\}$ . In time-varying data, the  $r$  maps correspond to  $r$  time frames or snapshots, which are taken at time  $t_1, t_2, \dots, t_r$  ( $t_1 < t_2, \dots, t_r$ ). The  $n$  features in  $\mathbb{D}$  are categorized into  $l$  types, which are governed by the underlying domain. The  $l$  feature types are identified by  $l$  unique labels, denoted as  $\Sigma = \{c_1, c_2, \dots, c_l\}$ . A

feature's geometric properties such as shape and size are captured based on one of the three representation schemes described in Section 2.1. Thus a feature  $f$  appearing at time  $t_i$  can be described as a vector  $f = \langle t_i, location, A_{geo}, type \rangle$ , where  $type \in \Sigma$ ,  $location$  identifies  $f$ 's position at time  $t_i$ , and  $A_{geo} \in \{A_{basic}, A_{landmark}\}$  models the geometric properties of  $f$  at  $t_i$ .

Note that in the rest of the paper, we refer to a feature corresponding to the above vector as a **spatial object**. We also assume the existence of an ordering among the  $l$  feature types:  $c_1 < c_2 < \dots < c_l$ . Furthermore, we refer to a snapshot's associated time  $t_i$  as its ID.

### 2.3 Object-based Distance Metrics

The framework uses the following metrics to measure the distance between two objects  $o_i$  and  $o_j$  existing in the same snapshot.

- **Point-Point distance:** This is simply the Euclidian distance between object centroids.
- **Line-Line distance:** If  $o_i$  and  $o_j$  are parallelepipeds (or parallelograms), we first identify the line segment between the midpoints of the upper and lower surfaces (or sides) in each object, then compute the shortest distance between these two line segments as the line-line distance between  $o_i$  and  $o_j$ . If  $o_i$  and  $o_j$  are ellipsoids (or ellipses), the line-line distance is the shortest distance between the two major axes.
- **Boundary-Boundary distance:** This is the shortest pairwise distance between the sampled boundary points (landmarks) of  $o_i$  and  $o_j$ .

Notice that the last two metrics take objects' geometric properties into account. The framework also supports Hausdorff distance [1]. Since this distance is not applicable to the applications described in this article, we do not discuss it here.

Two objects  $o_i$  and  $o_j$  have a **closeTo** relationship if the distance between them is  $\leq \epsilon$ , where  $\epsilon$  is a user-specified parameter. Two objects are **neighbors** if they have a *closeTo* relationship. We also define the **isAbove** relationship between  $o_i$  and  $o_j$ . In a coordinate system,  $o_i$  is said to have a *isAbove* relationship with  $o_j$ , if the upper-left corner of  $o_i$ 's Minimum Bounding Box (MBB), denoted as  $(x_i, y_i, z_i)$ , and the upper left corner of  $o_j$ 's MBB, denoted as  $(x_j, y_j, z_j)$ , meets the following condition:  $(z_i > z_j) \vee [(z_i = z_j) \wedge [(y_i > y_j) \vee ((y_i = y_j) \wedge (x_i < x_j))]]$  in 3D, or  $(y_i > y_j) \vee [(y_i = y_j) \wedge (x_i < x_j)]$  in 2D.

### 2.4 Spatial Object Association Pattern (SOAP)

A **Spatial Object Association Pattern (SOAP)** of size  $k$ , denoted as  $k$ -SOAP, characterizes the *closeTo* or *isAbove* relationships among  $k$  object types. The framework supports the discovery of three SOAP types: **Star**, **Clique**, and **Sequence** (Figure 2). They can be abstracted as undirected graphs, where a node corresponds to an object-type  $c_i \in \Sigma$ , and an

edge  $(c_i, c_j)$  indicates a *closeTo* or *isAbove* relationship between  $c_i$  and  $c_j$ . These SOAP types can also be represented as lists with different constraints for different SOAP types. We describe each SOAP type and its corresponding list representation as follows.

- **Star SOAPs** (Figure 2(a)) have a *center* object-type, which is required to have a *closeTo* relationship with all the other object-types in the same SOAP. Let  $c_{center}$  be the center of a Star  $k$ -SOAP  $p$ , and  $\{c_{[i]}: i \in [1, k-1]\}$  be the other  $k-1$  object-types in  $p$ , where  $c_{[1]} \leq \dots \leq c_{[k-1]}$ . The SOAP  $p$  can then be represented as the list  $p=(c_{center}, c_{[1]}, \dots, c_{[k-1]})$ , where  $closeTo(c_{center}, c_{[i]})=true$  ( $i \in [1, k-1]$ ).
- **Clique SOAPs** (Figure 2(b)) require a *closeTo* relationship hold between every pair of involved object-types in the same SOAP. Let  $\{c_{[i]}: i \in [1, k]\}$  ( $c_{[1]} \leq \dots \leq c_{[k]}$ ) be the  $k$  object-types in a *Clique* SOAP, it can then be described by the following list  $(c_{[i]}: i \in [1, k])$ :  $\forall_{i, j \in [1, k]} closeTo(c_{[i]}, c_{[j]})$ .
- **Sequence SOAPs** (Figure 2(c)) of size  $k$ ,  $p = (c_{[i]}: i \in [1, k])$ , satisfy two constraints: (1)  $closeTo(c_{[i]}, c_{[i+1]})=true$  and (2)  $isAbove(c_{[i]}, c_{[i+1]})=true$ , where  $1 \leq i \leq k-1$ .

A SOAP is said to be **autocorrelated** if an object-type occurs multiple times. For example,  $(c_1, c_1, c_2)$  is an autocorrelated 3-SOAP, where  $c_1$  occurs twice. An **instance** of a SOAP  $p$  is the set of spatial objects that meet all the requirements specified by  $p$ . For instance,  $(o_i, o_j)$  is an instance of the *Clique* 2-SOAP  $(c_1, c_2)$ , where  $o_i.type=c_1, o_j.type=c_2$ , and  $closeTo(o_i, o_j)=true$ ,

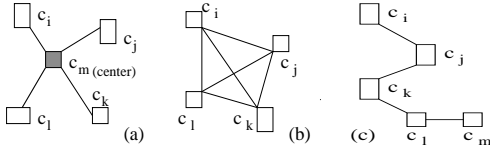


Figure 2: SOAP Types:(a)Star (b)Clique (c)Sequence

We define two measures **support** and **realization** to characterize the importance of a SOAP. The *support* of a SOAP  $p$  is the number of snapshots in the dataset where  $p$  occurs. Assume  $support(p)=s$ , let  $n_i$  be the number of  $p$ 's instances in the  $i^{th}$  snapshot where  $p$  appears,  $realization(p)=\min\{n_i\}$ . A pattern  $p$  is **frequent** if  $support(p) \geq minSupport$ , and **prevalent** if  $realization(p) \geq minRealization$ . Both  $minSupport$  and  $minRealization$  are user-specified parameters.

## 2.5 A Framework Towards Spatio-temporal Reasoning and Inferences

As we mentioned earlier, the work presented here is built upon a general framework proposed in [16; 17]. Figure 3 gives an overview of this framework. In this article, we focus on the two tasks listed in the shaded area, namely, spatio-temporal episodes generation and spatio-temporal reasoning.

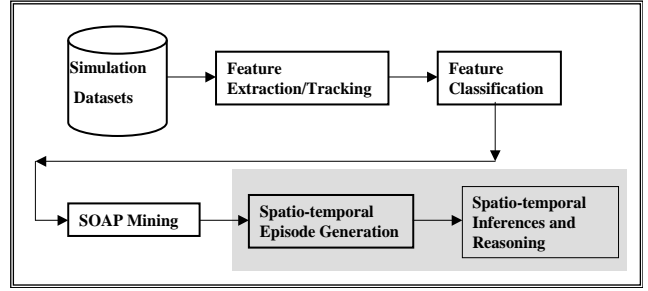


Figure 3: Framework overview

## 3. Spatio-temporal Reasoning and Inference

We have developed algorithms to efficiently extract frequent and prevalent *Star*, *Clique* and *Sequence* SOAPS from spatio-temporal datasets. Please refer to our previous work [16; 17], for implementation details and experimental results. The algorithms provide the following information for each discovered SOAP  $p$ :

- SOAP type, which can be one of the three types, namely, *Star*, *Clique*, and *Sequence*.
- The list of snapshots that contain valid instance(s) of  $p$ .
- Spatial properties such as location, shape and size of all the objects in an instance of  $p$ .

We next discuss how this data can be used to reason about the time-varying interactions among objects. To do this, we first generate spatio-temporal episodes for each discovered SOAP. Based on episodes, we believe that the following issues can be addressed:

- Infer critical events based on individual episodes.
- Model the interaction history for a set of features by combining multiple overlapped episodes.
- Infer future interactions among features based on their interaction history.
- Model the interaction behavior among multiple sets of features.

### 3.1 Generation of Spatio-temporal Episodes

Informally, a spatio-temporal episode of the SOAP  $p$  corresponds to a time interval during which  $p$  stays valid. Since relationships among objects change over time in time-varying data, so do SOAPS change over time as they characterize relationships among objects. We identify the following three evolutionary events for a SOAP  $p$ :

- **Formation** : This event occurs when the number of  $p$ 's instances changes from zero to non-zero.
- **Dissipation**: This event occurs when all the instances of  $p$  become invalid. The dissipation of a SOAP can occur due to many reasons. For example, feature(s) involved in

a SOAP may cease to exist or merge to form a new feature. The objects can also move far apart, which implies there is no interaction or very weak interaction among them.

- **Continuation** from time  $t_s$  to time  $t_e$ : if there exists at least one instance of  $p$  in each snapshot taken between  $t_s$  and  $t_e$ , including  $t_s$  and  $t_e$ .

Formation and dissipation events can occur to a SOAP many times. Therefore, a SOAP can exist in multiple disjoint temporal intervals, where each interval starts at a formation event and ends at a dissipation event. We refer to a SOAP’s continuation in each of the above temporal intervals as a **spatio-temporal episode**. Let  $I_p^t$  be the set of instances of SOAP  $p$  at time  $t$ , an episode of  $p$  in the *discrete* interval  $[t_s, t_e]$  can then be described as:  $E_p[t_s, t_e] = \{ I_p^t : t_s \leq t \leq t_e \}$ .

To generate spatio-temporal episodes for a SOAP, we need to identify its associated formation and dissipation events. Such events can be easily derived by checking the the SOAP’s presence at a given time. Let  $\lambda$  be the number of episodes associated with SOAP  $p$ , then its episodes correspond to  $\lambda$  disjoint time intervals  $\{ [t_s^i, t_e^i] : 1 \leq i \leq \lambda \}$ , where  $t_s^i$  and  $t_e^i$  mark the  $i^{th}$  formation and dissipation of  $p$ .

### 3.2 Inferences Based on SOAP Events

The afore-mentioned events characterize the stability of interactions among different features. In Table 1, we list some of the potential reasons for these events and information that can be inferred from these events.

Event	Reason	Inference(s)
Formation	Features start interacting	Features moving closer Features may merge
Continuation	Equilibrium is reached	Creation of stable SOAPS
Dissipation	Features stop interacting	Dissipation of one or more features Features merged and formed new feature

**Table 1: Potential reasons and inferences associated with each SOAP event**

Most of the inferences listed are easy to understand. Here we explain only the continuation event and associated inference in greater details. For expository purpose, let us consider a SOAP  $p$  composed of two objects  $A$  and  $B$ . If SOAP  $p$  continues for a long period of time, it can point to some equilibrium. The exact nature of equilibrium is domain specific. For vortices, it is well-known that two vortices rotating in different directions ( one in clockwise and the other in anti-clockwise) tend to destroy each other. Thus a SOAP with long continuation time can imply that  $A$  and  $B$  are rotating in the same direction (clockwise or anti-clockwise). If however,  $p$  dissipates very soon after it is formed, it can imply that the two vortices are rotating in opposite directions.

### 3.3 Inferences Based on Individual Episodes

By analyzing episodes individually, we can make inferences on critical events such as feature amalgamation and the in-

**Algorithm: Infer critical events**  
**Input:**  $\mathbb{E}$ : spatio-temporal episodes of different SOAP types

1. foreach episode  $E_p[t_s^i, t_e^i] \in \mathbb{E}$
2.   Foreach instance  $I_p^t: t \in [t_s^i, t_e^i]$
3.      $A_p^t \leftarrow$  size of the MBB of  $I_p^t$
4.     Fit a simple linear regression model  $A_p = \alpha + \beta t$   
       over  $(A_p^t, t) : t \in [t_s^i, t_e^i]$
5.     Inferences:  
       if  $(\beta > 0) \implies$  (Decreasing interaction)  
       if  $(\beta < 0) \implies$  (Increasing interaction  $\implies$  Object amalgamation)

**Figure 4: Infer critical events based on individual episodes**

involved features’ future interacting behavior. The main analysis steps are described in Figure 4. Let  $E_p[t_s^i, t_e^i]$  be the  $i^{th}$  episode associated with SOAP  $p$ . For each instance of  $p$  appearing during  $[t_s^i, t_e^i]$ , we compute the size of the instance’s Minimum Bounding Box (MBB) (line 3), where the MBB of a SOAP instance encompasses every object in the instance. We then apply a simple linear regression model to quantize the trend that an instance’s MBB varies over time:  $A_p = \alpha + \beta t$  (line 4), where  $A_p$  is the area of a SOAP instance’s MBB, and  $t$  is a time point in the interval  $[t_s^i, t_e^i]$ . Inferences can then be drawn based on this trend (line 5). For instance, if the MBB of an instance decreases from  $t_s$  to  $t_e$ , i.e.  $(\beta < 0)$ , we can infer that the involved objects merge at its corresponding SOAP’s dissipation. Otherwise, we can infer that the interaction among the involved objects becomes weaker over time.

### 3.4 Modeling Features’ Interaction History and Generating Transition Rules

Post-Transition / Pre-Transition	Star	Clique	Sequence
Star	X	Increasing Interaction → Object amalgamation	Decreasing interaction between the center object and some objects → SOAP dissipation or transitions
Clique	Decreasing Interaction → SOAP dissipation	X	Decreasing interaction among some objects → SOAP dissipation
Sequence	Formation of a center object → Clique → Object amalgamation	Increasing Interaction → Object amalgamation	X

**Figure 5: Transitions among different SOAP types and their implications w.r.t. inter-object interactions**

As mentioned earlier, different SOAP types characterize different types of interactions among objects. Therefore, for a set of features, the evolutionary nature of their spatial relationships can be captured by a series of SOAP episodes associated with different SOAP types. We would also like to point out that the three SOAP types are not mutually exclusive in a given interval. For instance, a *Clique* SOAP corresponds to multiple *Star* SOAPS centered at each of the involved object. To address this issue, we impose the following order among the three SOAP types:  $Clique \prec Sequence \prec Star$ , where *Clique* being the most spatially constrained and *Star* the least constrained. In the case where two or more overlapped episodes are identified for the same set of features,

we choose the episode that is associated with the most constrained SOAP type. Figure 5 summarizes the transitions between the three SOAP types. Also given in the figure are the implications and inferences one may make for each transition.

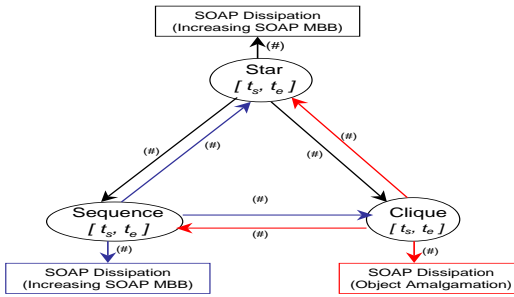
**Algorithm: Model interacting history and Identify Transition rules**  
**Input:**  $\mathbb{E}$ : spatio-temporal episodes of different SOAP types  
 $F = \{c_1, \dots, c_i\}$ : the set of features of interest

1.  $E_F \leftarrow$  all episodes in  $\mathbb{E}$  that are associated with  $F$
2. Sort  $E_F$  in increasing order of  $e.t_s : e \in E_F$
3. if (episodes have overlapped intervals)
4. Keep the episode that is associated with the most constrained SOAP type
5. Construct  $F$ 's relationship transition diagram
6. Make inferences using the Transition Table in Figure 5
7. Identify relationship transition rules for  $F$

**Figure 6: Model interaction history for a set of features and identify transition rules**

The interaction history of a set of features of interest, denoted as  $F = \{c_1, \dots, c_i\}$ , is modeled by combining its overlapped episodes. The resulting model is referred to as the Relationship Transition Diagram of  $F$  as shown in Figure 7. The transition diagram of  $F$  consists of the complete path from  $F$ 's first SOAP formation event till its last SOAP dissipation event. Note that this path is continuous in space and time.

Figure 6 describes the main steps to derive the relationship transition diagram for  $F$ . We consider all of its associated SOAP episodes as input. We first find all the episodes that are associated with  $F$  (line 1). These episodes can be associated with different SOAP types. We then order all these episodes by their formation time (line 2). In the case that multiple overlapped episodes of different SOAP types are identified for  $F$ , we only keep the most constrained SOAP type (lines 3-4). We next construct the transition diagram of  $F$  (line 5). Based on this diagram, we can make important inferences about the spatial relationship among objects in  $F$ . Note that it is possible that  $F$  is not associated with any episode. In such a case, no transition diagram will be constructed.



**Figure 7: SOAP type transition diagram. '(#)' beside each transition indicates the number of this transition.**

Based on relationship transition diagrams, we can further characterize the evolutionary nature of spatial relationships through constructing *relationship transitional rules*. A relationship transitional rule for a set of features  $F$  is in the

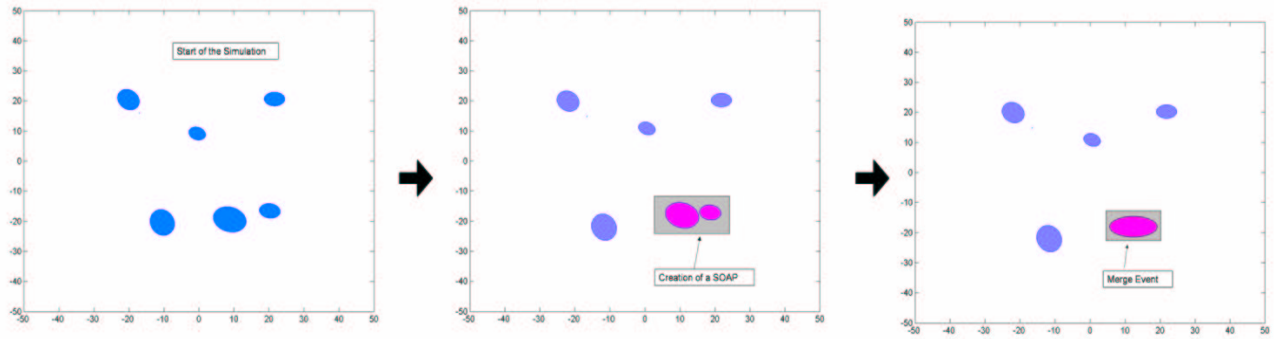
form  $F : T_{pre} \Rightarrow T_{post}(n/N)$ , where  $T_{pre}$  and  $T_{post}$  are the relationship types before and after the transition,  $n$  is the number of this transition that has occurred to  $F$ , and  $N$  is the total number of different transitions that have occurred to  $F$  when the  $T_{pre}$  relationship holds for  $F$ . These rules can facilitate predictions on features' future spatio-temporal behaviors. For instance, according to the rule  $\{A, B, C, D\} : Star \Rightarrow Clique(8/10)$ , if we observe that  $A, B, C$ , and  $D$  form a *Star* SOAP in a certain period, we can predict that they are very likely to form a *Clique* after some time,

### 3.5 Inferences Based on Multiple Sets of Features

So far, we only consider inferences that involve one set of features. We are currently investigating potential approaches that involve multiple sets of features. For example, after the dissipation of SOAP ( $A B C$ ) at time  $t_i$ , we observe that at time  $(t_i + 1)$ , objects  $A$  and  $E$  appear in the same neighborhood where objects  $A, B$  and  $C$  are located at  $t_i$ , we can then infer the following: objects  $B$  and  $C$  merged into  $E$ . We plan to extend this idea by constructing spatio-temporal traces, where a trace is obtained by splicing a series of temporally and spatially overlapped episodes. Since a splicing point of two intervals in a trace usually indicates a significant interaction change, traces can be used to reason about the causal relationships between different evolutionary SOAP events. However, we note that for a given situation, one can often have multiple inferences. For instance, we may also have the following inferences for the above example: (1)  $B$  and  $C$  dissipated at the SOAP's dissipation; and (2)  $B$  evolved into  $E$  and  $C$  dissipated. We are currently investigating other approaches to address this multi-inference issue. One potential solution is to take into account the interaction among different SOAPS. To do this, we need to consider multiple SOAPS in the same vicinity.

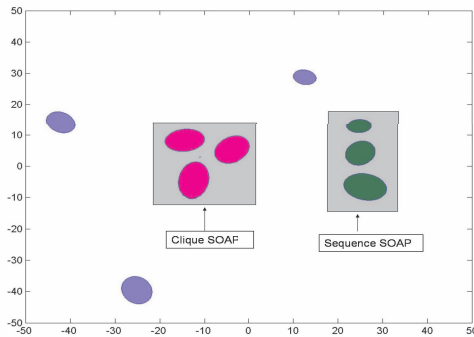
## 4. Evaluation

In this section, we present preliminary results on vortex simulation data to evaluate the proposed approaches. The vortex dataset is generated by implementing a simplistic version of the algorithm proposed by Christian [4]. The dataset consists of 1970 snapshots, with around 200 vortices in each snapshot. The number of vortices in each frame changes over time, as an existing vortex can dissipate, new vortex can be created, or two vortices can merge to form a new vortex. We apply a locally implemented algorithm [8], to detect and extract vortices from the simulation dataset. We then use an entropy-based clustering algorithm to cluster all the detected vortices into groups (or classes) [3], where a vortex is represented as an ellipse. This clustering procedure ensures that similar vortices are put into the same group. Finally, we label each detected vortex by the ID of the group that it is clustered to. We next extract different types of SOAPS by applying our SOAP mining algorithms. The boundary-boundary distance is used to measure the distance between two vortices, and two vortices are considered to be of interaction if they are within the distance of 10 ( $\epsilon = 10$ ). A SOAP is frequent if it appears in at least 10 snapshots, i.e.,  $minSupport=10$ . Finally, the parameter  $minRealization$  is set to 1.



**Figure 8:** (a) Vortices at time 0 (b) SOAP Formation at time 90 (c) Vortex Merging at time 104

Figure 8 shows a fraction of three simulation snapshots at different times. Figure 8(a) shows the initial configuration of the vortices with no SOAPs. Figure 8(b) shows the creation of a SOAP. This SOAP is formed because the two vortices are moving towards each other and eventually their distance becomes less than the distance threshold. Figure 8(c) shows a very interesting result. The two vortices involved in the SOAP came closer and eventually merged into a new vortex. This event is captured by the SOAP's dissipation (resulting from an amalgamation of two features within the SOAP). Note that SOAPs' dissipation does not necessarily imply dissipation of features. We were able to make inference on such events by identifying the changing trend of the corresponding instance's MBB.



**Figure 9:** Different Vortex SOAPs are discovered at time 991.

Figure 9 shows a *Clique* and a *Sequence* SOAP identified in the same snapshot taken at time 991. One can observe that different SOAP types can capture very different spatial relationships (or interactions) among vortices.

#### 4.1 Inferences Based on Individual Episodes

Figures 10-11 demonstrate two opposite cases that lead to a SOAP dissipation event. The SOAP in Figure 10 dissipates due to a vortex amalgamation event. Whereas the SOAP in Figure 11 dissipates because the interaction among the involved vortices becomes weaker and weaker. We are able

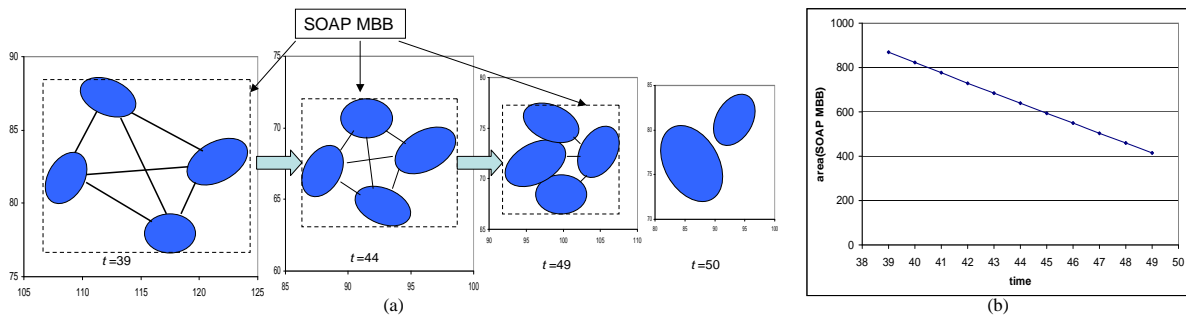
to make inferences on these two types of behaviors by identifying the changing trend of the SOAP's MBB during an episode. Figure 10(a) shows three snapshots of a *Clique* SOAP instance during the episode from time 39 to 49. It also shows a snapshot of the SOAP's neighborhood area immediately after the dissipation. One can observe that the MBB of this instance decreases as the SOAP instance evolves. Such a changing trend is successfully identified by the fitted linear regression model ( Figure 10(b) ). Therefore, we can infer that a vortex amalgamation event may have occurred around the SOAP's dissipation, where three vortices merged into one new vortex. Figure 11(a) illustrates the case where all involved objects are moving away from each other. The fitted linear regression line captures this trend, as shown in Figure 11(b).

#### 4.2 Capturing Features' Interaction History

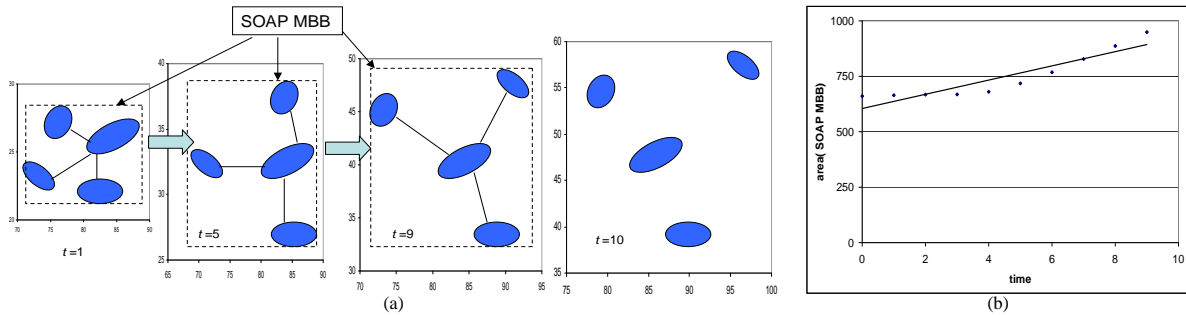
As mentioned earlier, combining episodes associated with different SOAP types can be used to capture the evolving nature of interactions among different features.

Figure 12 demonstrates how the interaction among four vortices evolves over time and is captured by combining multiple episodes formed at different time. The interaction history of the four vortices is modeled by three SOAP types, corresponding to three episodes. In the first episode, which starts from time 0 and ends at time 18, the four vortices form a *Sequence* SOAP (Figure 12(a) ). As the interaction evolves, the four vortices start to form a *Clique*, which ends at time 34. Following the dissipation of this interaction, the four vortices start to move away from each other. This evolving behavior is captured by the transition of *Clique*  $\Rightarrow$  *Star*. Through identifying the changing trend of the *Star* SOAP's MBB, which increases over time, we can infer that the four vortices will keep moving apart and may cease to interact after some time. This inference is verified by observing the simulation data.

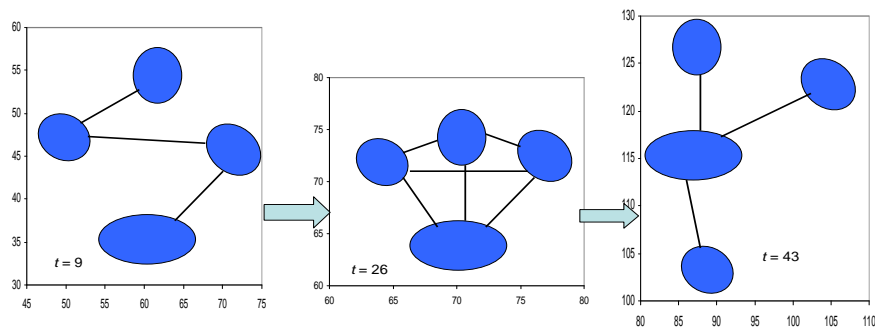
Figure 13 models the other type of evolving relationship among features. The four vortices begin their interaction as a *Sequence* and changes into a *Star* after some time. After staying as a *Star* SOAP for some time, they then evolve into



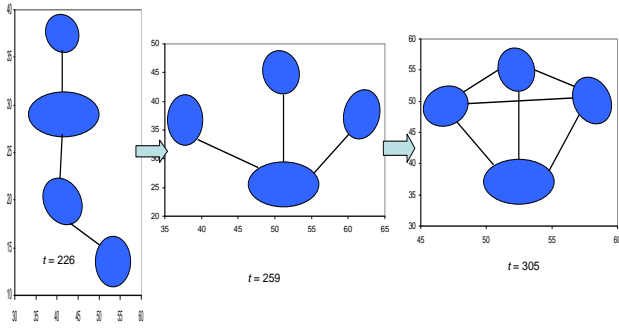
**Figure 10:** The MBB of SOAP (1 15 61 81) decreases in the episode [39, 49], which eventually dissipated at time 50 due to object amalgamation. (a) Snapshots of the SOAP at time 39, 44, 49, and the vortices in the vicinity of the SOAP right after its dissipation (b) The fitted linear regression line: area(MBB) vs. time



**Figure 11:** The MBB of SOAP (19 32 103 109) increases in the episode [0, 9]. (a) Snapshots of the SOAP at time 1, 5, 9, and vortices in the vicinity of the SOAP right after its dissipation (b) The fitted linear regression line: area(MBB) vs. time

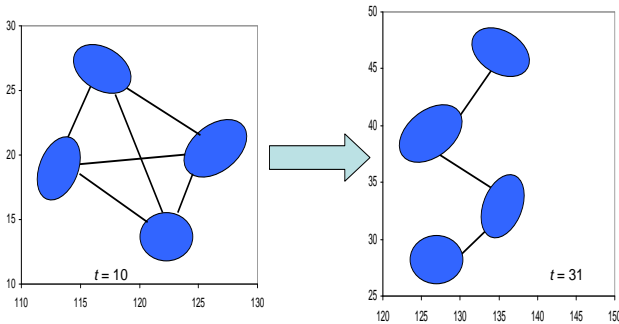


**Figure 12:** Transition: Sequence  $\Rightarrow$  Clique  $\Rightarrow$  Star. The SOAP (23 39 57 58) is a Sequence in the interval [0 18], a Clique in the interval [19 34], and a Star in the interval [35 57].



**Figure 13: Transition: Sequence  $\Rightarrow$  Star  $\Rightarrow$  Clique.** The SOAP (19 27 58 36) is a Sequence in the interval [212 243], a Star in the interval [244 287], and a Clique in [288 313]. It eventually dissipated due to the amalgamation of vortices 19 and 58 at time 314.

a *Clique*. The transition *Star* $\Rightarrow$ *Clique* implies that the interaction among the four vortices increases, which may lead to vortex amalgamation. Such an implication is further strengthened since we see a decreasing trend of the SOAP’s MBB during the time interval [288 313].

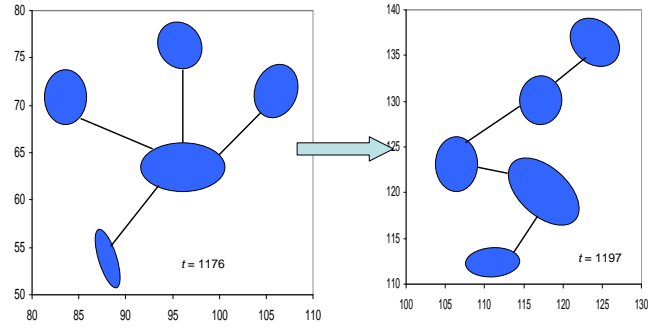


**Figure 14: Transition: Clique  $\Rightarrow$  Sequence.** The SOAP (7 8 42 99) is a Clique in the interval [0 17], then evolves into a Sequence at time 18 and eventually dissipates at time 49

Figure 14 and 15 illustrate the other two transitions that can occur to features: *Clique*  $\Rightarrow$  *Sequence* and *Star*  $\Rightarrow$  *Sequence*. We observe that the transition from *Clique* to *Sequence* often indicates that the involved features will very likely cease to interact after some time. However, such an observation does not hold for the *Star*  $\Rightarrow$  *Sequence* transition.

## 5. Related Work

Our research on spatio-temporal association patterns shares some of the objectives with approaches for spatial reasoning. Bailey-kellogg and Zhao [2] propose a methodology for reasoning about such problems called qualitative spatial reasoning (QSR). Their work is methodology driven and mainly focuses conceptual topics such as data representations and manipulations. They also discuss the use of different spatial primitives to model objects of different shapes and spatial relationships among objects. Our framework is an efficient realization of their conceptual methodology for scientific data.



**Figure 15: Transition: Star  $\Rightarrow$  Sequence.** The SOAP (205 211 218 230) forms a Star at time 1170, evolves into a Sequence at time 1180, and finally dissipates at time 1202.

In addition we also support the discovery of spatio-temporal episode patterns that is not explicitly considered in their work. Fernyhough et al. implemented techniques to detect events by identifying frequently occurring spatial relationships [5; 6]. However, their proposed technique only considers pairwise relationships. Therefore, interactions involving more than two features will be missed.

## 6. Conclusion and Ongoing Work

In this paper, we propose spatio-temporal association based approaches to make inferences on important events and reason about features’ interacting behaviors. Preliminary results on vortex simulation data validate the efficacy of these approaches.

We are currently extending our framework to include the discovery of other types of association patterns, for example, spatio-temporal patterns concerning both topological and neighborhood relationships. We are also in the process of implementing approaches that combines episodes associated with multiple sets of features. Our approaches currently target the discovery of important frequent spatial interactions, but in many cases rare interactions can also be important. We plan to extend our work to address this limitation.

## 7. References

- [1] M. J. Atallah. A linear time algorithm for the hausdorff distance between convex polygons. *Information Processing Letters*, 17:207–209, 1983.
- [2] Chris Bailey-Kellogg and Feng Zhao. Qualitative spatial reasoning: extracting and reasoning with spatial aggregates. *AI Magazine*, 24(4):47–60, 2004.
- [3] C. Cheng, A.W. Fu, and Y. Zhang. Entropy-based subspace clustering for mining numerical data. In *KDD’99*, pages 84–93. ACM Press, 1999.
- [4] J.P. Christian. Numerical simulation of hydrodynamics by the method of point vortices. *Journal of Computational Physics*, 135, pages 189–197, 1971.
- [5] A. G. Cohn and S. M. Hazarika. Qualitative spatial representation and reasoning: an overview. *Fundamental Informatics*, 46(1-2):1–29, 2001.

- [6] J. H. Fernyhough, A. G. Cohn, and D C. Hogg. Event recognition using qualitative reasoning on automatically generated spatio-temporal models from visual input. In *IJCAI Workshop on Spatial and Temporal Reasoning*, 1997.
- [7] C. Henze. Feature detection in linked derived spaces. In *IEEE Conference on Visualization*, 1998.
- [8] M. Jiang, T. Choy, S. Mehta, S. Parthasarathy, R. Machiraju, D. Thompson, J. Wilkins, and B. Gatlin. Feature mining paradigms for scientific data. In *SIAM Conference on Data Mining*, 2003.
- [9] S. Mehta, K. Hazzard, R. Machiraju, S. Parthasarathy, and J. Wilkins. Detection and visualization of anomalous structures in molecular dynamics simulation data. In *IEEE Conf. on Visualization*, 2004.
- [10] Y. Morimoto. Mining frequent neighboring class sets in spatial databases. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 353–358, 2001.
- [11] R. Munro, S. Chawla, and P. Sun. Complex spatial relationships. In *The Third IEEE International Conference on Data Mining*, 2003.
- [12] C. R. Rao and S. Suryawanshi. Statistical analysis of shape of objects based on landmark data. *Proceedings of National Academy of Science, USA.*, 93(22):12132–12136, October 1996.
- [13] A. Sadarjoen, F.H. Post, and D.C. Banks et al. Selective visualization of vortices in hydrodynamic flows. In *Proceedings of the conference on Visualization*, pages 419–422. IEEE Computer Society Press, 1998.
- [14] D. Silver and X. Wang. Volume tracking. In Roni Yagel and Gregory M. Nielson, editors, *IEEE Visualization*, pages 157–164, 1996.
- [15] Waldo R. Tobler. A computer movie simulating urban growth in the detroit region. *Economic Geography* 46:234-230, 1970.
- [16] H. Yang, S. Mehta, and S. Parthasarathy. A generalized framework for mining spatio-temporal patterns in scientific data. In *Technical Report OSU-CISRC-5/05-TR14, Ohio State University*, 2005.
- [17] H. Yang, S. Parthasarathy, and S. Mehta. Mining spatial object associations for scientific data. In *Proceedings of IJCAI (to appear)*, 2005.
- [18] Kenneth Yip. Structural inferences from massive datasets. In *IJCAI*, pages 534–541, 1997.
- [19] X. Zhang, N. Mamoulis, D. W. Cheung, and Y. Shou. Fast mining of spatial collocations. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 384–393. ACM Press, 2004.