

# Mixed-Drove Spatio-Temporal Co-occurrence Pattern Mining

Mete Celik, *Student Member, IEEE*, Shashi Shekhar, *Fellow, IEEE*, James P. Rogers, and James A. Shine

**Abstract**—Mixed-drove spatio-temporal co-occurrence patterns (MDCOPs) represent subsets of two or more different object-types whose instances are often located in spatial and temporal proximity. Discovering MDCOPs is an important problem with many applications such as identifying tactics in battlefields and games and tracking predator-prey interactions. However, mining MDCOPs is computationally very expensive because the interest measures are computationally complex, datasets are larger due to the archival history, and the set of candidate patterns is exponential in the number of object-types. We propose a monotonic composite interest measure for discovering MDCOPs and novel MDCOP mining algorithms. Analytical results show that the proposed algorithms are correct and complete. Experimental results also show that the proposed methods are computationally more efficient than naïve alternatives.

**Index Terms**—Spatio-temporal Data Mining, Spatio-temporal Co-occurrence Pattern Mining, Composite Interest Measure, Mixed-drove Spatio-temporal Co-occurrence Pattern.

## 1 INTRODUCTION

As the volume of spatio-temporal data continues to increase significantly due to both the growth of database archives and the increasing number and resolution of spatio-temporal sensors, automated and semi-automated pattern analysis becomes more essential. As a result, spatio-temporal co-occurrence pattern mining has been the subject of recent research. Given a moving object database, our aim is to discover mixed-drove spatio-temporal co-occurrence patterns (MDCOPs) representing subsets of different object-types whose instances are located close together in geographic space for a significant fraction of time. Unlike the objectives of some other spatio-temporal co-occurrence pattern identification approaches where the pattern is the primary interest, in MDCOPs both the pattern and the nature of the different *object-types* are of interest.

A simple example of an MDCOP is in ecological predator-prey relationships. Patterns of movements of rabbits and foxes, for example, will tend to be co-located in many time-frames which may or may not be consecutive. Rabbits may attempt to move away from foxes, and the foxes may attempt to stay with the rabbits. Other factors such as available food and water may also affect the patterns.

More example MDCOPs may be illustrated in American football where two teams try to outscore each other by moving a football to the opponent's end of the field.

- M. Celik and S. Shekhar are with the Department of Computer Science, University of Minnesota, Minneapolis, MN, USA, 55455  
E-mail: mcelik, shekhar@cs.umn.edu
- J.P. Rogers and J.A. Shine are with the U.S. Army ERDC, Topographic Engineering Center, VA, USA.  
E-mail: james.p.rogers.11, james.a.shine@usace.army.mil

Manuscript received March 26, 2007; revised December 28, 2007; revised March 15, 2008; accepted April 30, 2008.

Various complex interactions occur within one team and across teams to achieve this goal. These interactions involve intentional and accidental MDCOPs, the identification of which may help teams to study their opponent's tactics. In American football, object-types may be defined by the roles of the offensive and defensive players, such as quarterback, running back, wide receiver, kicker, holder, linebacker, and cornerback. An MDCOP is a subset of these object-types (such as {kicker, holder} or {wide\_receiver, cornerback}) that occur frequently. One example MDCOP involves offensive wide receivers, defensive linebackers, and defensive cornerbacks, and is called a Hail Mary play. In this play, the objective of the offensive wide receivers is to outrun any linebackers and defensive backs and get behind them, catching an undefended pass while running untouched for a touchdown. This interaction creates an MDCOP between wide receivers and cornerbacks. An example Hail Mary play is given in Fig. 1. It shows the positions of four offensive wide receivers (W.1, W.2, W.3, and W.4), two defensive cornerbacks (C.1 and C.2), two defensive linebackers (L.1 and L.2), and a quarterback (Q.1) in four time slots. The solid lines between the players show the neighboring players. The wide receivers W.1 and W.4 cross over each other and the wide receivers W.2 and W.3 run directly to the end zone of the field. Initially, the wide receivers W.1 and W.4 are co-located with cornerbacks C.1 and C.2 respectively and the wide receivers W.2 and W.3 are co-located with linebackers L.1 and L.2 at time slot  $t=0$  (Fig. 1a). In time slot  $t=1$ , the four wide receivers begin to run, while the linebackers run towards the quarterback and the cornerbacks remain in their original position, possibly due to a fake handoff from the quarterback to the running back (Fig. 1b). In time slot  $t=2$ , the wide receivers W.1 and W.4 cross over each other and try to drift further away from their respective cornerbacks

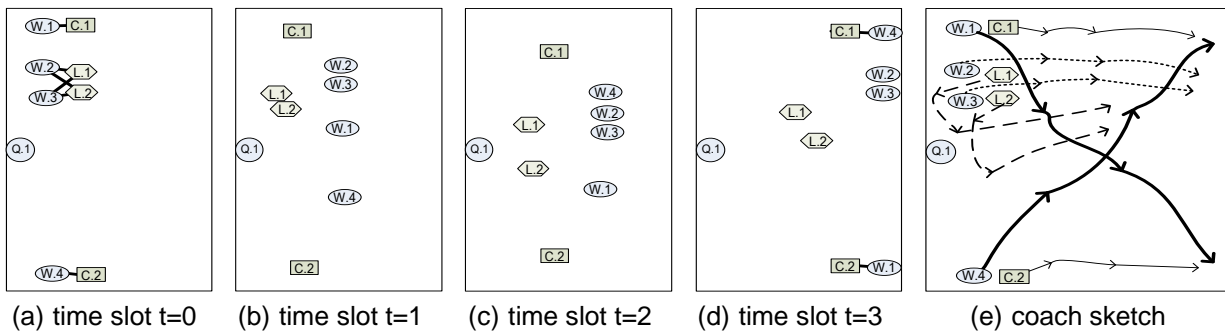


Fig. 1. An example Hail Mary play in American football

(Fig. 1c). When the quarterback shows signs of throwing the football, both cornerbacks and linebackers run to their respective wide receivers (Fig. 1d). The overall sketch of the game tactics can be seen in Fig. 1e. In this example, wide receivers and cornerbacks form an MDCOP since they are persistent over time and they occur 2 out of 4 time slots. However, wide receivers and linebackers do not form an MDCOP due to the lack of temporal persistence.

There are many applications for which discovering co-occurring patterns of specific combinations of object-types is important. Some of these include military (battlefield planning and strategy), ecology (tracking species and pollutant movements), homeland defense (looking for significant “events”), and transportation (road and network planning) [11], [15].

However, discovering MDCOPs poses several non-trivial challenges. First, current interest measures (i.e. the spatial prevalence measure) are not sufficient to quantify such patterns, so new composite interest measures must be created and formalized [13], [22]. Second, the set of candidate patterns grows exponentially with the number of object-types. Finally, since spatio-temporal datasets are huge, computationally efficient algorithms must be developed [23].

## 1.1 Contributions

This paper is an extended version of our paper published in the proceedings of the 6th IEEE International Conference on Data Mining (ICDM) [5], where we introduced an MDCOP mining problem, proposed a new monotonic composite interest measure, developed two MDCOP algorithms, and evaluated these using real datasets. This extended paper makes the following new contributions:

- It proposes a new and computationally efficient MDCOP mining algorithm (FastMDCOP-Miner)
- It compares the proposed algorithm with those in the ICDM paper [5].
- It presents additional experimental results with synthetic datasets for all MDCOP algorithms.
- It includes an expanded literature survey and a discussion of statistical spatio-temporal interest measures.

- It includes revised comparisons of approaches and experimental design.

## 1.2 Scope and Outline

This paper focuses on MDCOPs (typed collections of moving objects) by extending interest measures for spatial co-location patterns given a user-defined participation index threshold [13], [22]. The following issues are beyond the scope of this paper: (i) determining thresholds for MDCOP interest measures; (ii) similarity measures for tracking moving objects due to the focus on object-types rather than objects; (iii) indexing and query processing issues related to mining objects; (iv) discovering multisets (e.g.  $\{A, A, B\}$ ).

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 presents basic concepts to provide a formal model of MDCOPs and the problem statement of mining MDCOPs. In section 4, we present our proposed MDCOP mining algorithm. Analysis of the algorithms is given in Section 5. Section 6 presents the experimental evaluation and Section 7 presents conclusions and future work.

## 2 RELATED WORK

Data analysis can be broadly categorized into statistical approaches and data mining approaches. In statistical approaches, there are bodies of work in both spatial and temporal analysis. Spatial point patterns are often described by metrics such as the intensity function and Ripley’s  $K$  [20], [21]. Other measures such as complete spatial randomness (CSR) and spatial covariance functions are used to describe the spatial relationships of adjacent areas and continuous variables as random fields [7]. Temporal patterns have been extensively studied in models such as moving averages, first and second order autoregression, integration, and periodic patterns such as seasonality [25]. Granger has looked at co-occurring temporal patterns under an assumption of cointegration [8]. There has also been some recent research in combining spatial and temporal analysis, such as Brix and Diggle’s extended intensity function and the extended  $K(r, t)$  function [1], [18]. Most attempts to combine the fields suffer from limitations such as

the inability to model space-time interactions, treating time as merely another dimension of space and assuming separability and independence between space and time [21]. Statistical research specifically focused on spatio-temporal co-occurrence patterns and their possible interactions has been limited.

Previous data mining studies for mining spatio-temporal co-occurrence patterns can be classified into two categories: mining of uniform groups of moving objects, and mining of mixed groups of moving objects.

To mine uniform groups of moving objects, the problems of discovering flock patterns [17], [10], [9] and moving clusters [14] are defined. A flock pattern is a moving group of the same kind of objects, such as a sheep flock or a bird flock. Gudmundsson et al. proposed algorithms for detection of the flock pattern in spatio-temporal datasets [10], [9]. Kalnis et al. defined the problem of discovering moving clusters and proposed clustering-based methods to mine such patterns [14]. In this approach, if there is a large enough number of common objects between clusters in consecutive time slots, such clusters are called moving clusters. These methods do not take object-types into account, and thus are not effective for mining MDCOPs [5]. To mine mixed groups of moving objects, the problems of discovering collocation episodes [4] and topological patterns [24] are important. Both generalize co-location patterns [13] (subsets of object-types that are frequently located together in space) to the spatio-temporal domain. A collocation episode is a sequence of co-location patterns with some common object-types across consecutive time slots. However, if there is no common object-type in consecutive time slots, the proposed approach will not identify any pattern. For example, the collocation episodes algorithm will not be able to find any pattern from the dataset given in Fig. 1, if the window length (which is used to find co-location patterns) is 2. For this case, the algorithm tries to find co-location patterns that are persistent in 2 consecutive time slots, but there is no such pattern in the dataset because wide receivers and cornerbacks are forming co-locations in time slots  $t=0$  and  $t=3$  and wide receivers and linebackers are forming co-locations in time slots  $t=0$ . Thus, there may not be any co-location patterns and collocation episodes identified in the dataset.

A topological pattern [24] is a subset of object-types whose instances are close in space and time. An interest measure for a topological pattern  $\{A,B\}$  (e.g. participation index or support) is a spatio-temporal join of instances of A and instances of B [13]. This statistic may be high even if many instances of A and many instances of B are not spatially together for a moment in time. The semantics of topological patterns are not well-defined for moving objects. For example, this approach can not find an answer to the question of what fraction of time the pattern occurs. The answer of this approach may also be "empty" to the question of when (which time slots) the pattern occurs since there is no time slot notion. In the dataset given in Fig. 1, this approach will discover the

two patterns of  $\{W,C\}$  and  $\{W,L\}$ . Both patterns have the same support, but pattern  $\{W,C\}$  occurs in 2 time slots out of 4 (a persistent pattern) and pattern  $\{W,L\}$  occurs in 1 time slot out of 4 (a transient pattern) since tracks of objects are represented as spatio-temporal instances. The persistent pattern  $\{W,C\}$  occurs in time slots  $t=0$  and  $t=3$  and its instances  $\{W1,C1\}$  and  $\{W4,C2\}$  occur in time slot  $t=0$  and  $\{W1,C2\}$  and  $\{W4,C1\}$  in time slot  $t=1$ . The transient pattern  $\{W,L\}$  occurs in time slot  $t=0$  and its instances  $\{W2,L1\}$ ,  $\{W3,L1\}$ ,  $\{W2,L2\}$ , and  $\{W3,L2\}$  occur in time slot  $t=0$ .

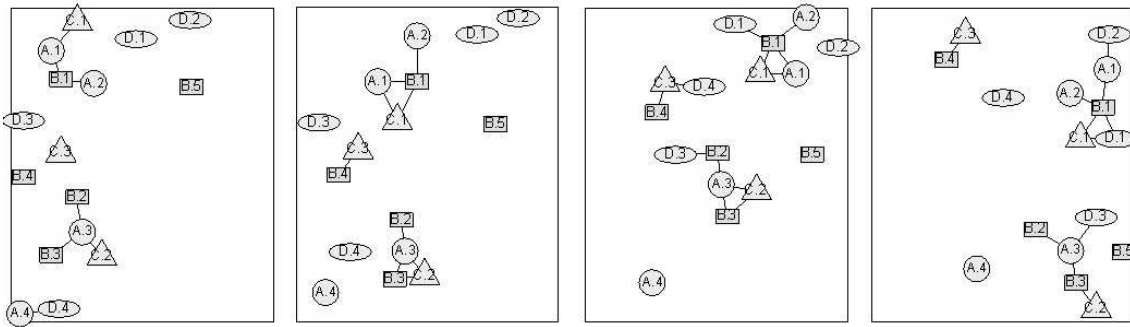
In contrast, our proposed interest measure and algorithms will efficiently mine mixed groups of objects (e.g. MDCOPs) which are close in space and persistent (but not necessarily close) in time. Unlike a number of the techniques just described, our approach will discover persistent patterns that co-occur in most but not all spatio-temporal intervals, so consecutive co-occurrences are not mandatory. For example, our approach will find the MDCOP  $\{\text{wide\_receiver, cornerback}\}$  pattern in Fig. 1, if the fraction of time slots where the pattern occurs over the total number of time slots is no less than a defined threshold, e.g., 0.5. It may reject the pattern  $\{W,L\}$  in Fig. 1 given the lack of time persistence of the  $\{\text{wide\_receiver, linebacker}\}$  pattern. In fact, instances of MDCOP  $\{\text{wide\_receiver, cornerback}\}$  are co-located in 2 time slots out of 4 and instances of  $\{\text{wide\_receiver, linebacker}\}$  are co-located in 1 time slot out of 4. The instances of MDCOP  $\{\text{wide\_receiver, cornerback}\}$  are  $\{W.1, C.1\}$  and  $\{W.4, C.2\}$  in time slot  $t=0$ , and  $\{W.4, C.1\}$  and  $\{W.1, C.2\}$  in time slot  $t=3$ .

### 3 BASIC CONCEPTS & PROBLEM DEFINITION

#### 3.1 Spatial Prevalence Measure

The focus of this study is to discover MDCOPs over a spatio-temporal framework and a neighborhood relation  $R$ . First we explain the modeling of mixed groups of object-types in space, e.g., spatial co-locations [22]. In the next sections, we explain how we model MDCOPs by extending spatial co-location mining to include time information and then propose algorithms to mine these MDCOPs.

Spatial co-location mining algorithms are used to discover sets of mixed object-types that are frequently located together in a spatial framework for a given set of spatial object-types, their instances, and a spatial neighbor relationship  $R$  [13], [22]. For example, in Fig. 2, in time slot  $t=0$ ,  $\{A.1, C.1\}$  is an instance of a co-location if the distance between the objects is no more than a given neighborhood distance threshold. In Fig. 2, the solid lines show the distance between the objects that satisfies the neighborhood distance threshold. The *participation index* is used to determine the strength of the co-location pattern, that is, whether the index is greater than or equal to a threshold [13], [22]. Such a co-location is called *spatial prevalent*. The participation index is defined as the minimum of the *participation*



(a) An input spatio-temporal dataset

Mixed-Drove Spatio-Temporal Co-occurrence Patterns	Spatial prevalence index values				Time prevalence index values
	time slot 0	time slot 1	time slot 2	time slot 3	
A B	3/5	3/5	3/5	3/5	4/4
A C	2/4	2/4	2/4	0	3/4
B C	0	3/5	3/5	3/5	3/4
A B C	0	2/5	2/5	0	2/4

(b) A set of output mixed-drove spatio-temporal co-occurrence patterns

Fig. 2. An input spatio-temporal dataset and a set of output MDCOPs

*ratios* (the fraction of the number of instances of object-types forming co-location instances to the total number of instances). For example, in Fig. 2,  $\{A, B\}$  is a co-location in time slot  $t=0$ , and its instances are  $\{A.1, B.1\}$ ,  $\{A.2, B.1\}$ ,  $\{A.3, B.2\}$ , and  $\{A.3, B.3\}$ . In the dataset, object-type A has 4 instances and three of them (A.1, A.2, and A.3) are contributing to the co-location  $\{A, B\}$ , so the participation ratio of A is  $3/4$ . The participation ratio of B is  $3/5$  since 3 out of 5 instances are contributing to the co-location  $\{A, B\}$ . The participation index of the co-location  $\{A, B\}$  is  $3/5$ , which is the minimum of the participation ratios of object-types A and B. It has been shown that the participation index is anti-monotone in the size of co-locations [13], [22]. In other words,  $participation\_index(P_j) \leq participation\_index(P_i)$  if  $P_i$  is a subset of  $P_j$ . In addition, [13], [22] show that the participation index has a spatial statistical interpretation as an upper bound on the *cross - K* function [7].

### 3.2 Modeling MDCOPs

Given a set of spatio-temporal mixed object-types and a set of their instances with a neighborhood relation  $R$ , an MDCOP is a subset of spatio-temporal mixed object-types whose instances are neighbors in space and time.

**Definition 3.1:** Given a spatio-temporal pattern and a set  $TF$  of time slots, such that  $TF = [T_0, \dots, T_{n-1}]$ , the *time prevalence or persistence measure* of the pattern is the fraction of time slots where the pattern occurs over the total number of time slots.

For example, in Fig. 2, the total number of time slots is 4 and pattern  $\{A, B\}$  occurs in all 4 time slots, so its time prevalence index is  $4/4$ . Pattern  $\{A, C\}$  occurs in 3 time slots, namely, time slots  $t=0$ ,  $t=1$ , and  $t=2$ , and its time prevalence index is  $3/4$  (Fig. 2b).

**Definition 3.2:** Given a spatio-temporal dataset of mixed object-types  $ST$ , and a spatial prevalence threshold  $\theta_p$ , the *mixed-drove prevalence measure* of pattern  $P_i$

is a composition of the spatial prevalence and the time prevalence measures as shown below.

$$Prob_{t_m \in TF}(s\_prev(P_i, time\_slot\ t_m) \geq \theta_p), \quad (1)$$

where  $Prob$  stands for probability of overall prevalence time slots and  $s\_prev$  stands for spatial prevalence, e.g., the participation index, described in Section 3.1.

**Definition 3.3:** Given a spatio-temporal dataset of mixed object-types  $ST$  and a threshold pair  $(\theta_p, \theta_{time})$ , MDCOP  $P_i$  is a *mixed-drove prevalent pattern* if its mixed-drove prevalence measure satisfies the following.

$$Prob_{t_m \in TF}[s\_prev(P_i, time\_slot\ t_m) \geq \theta_p] \geq \theta_{time}, \quad (2)$$

where  $Prob$  stands for probability of overall prevalence time slots,  $s\_prev$  stands for spatial prevalence,  $\theta_p$  is the spatial prevalence threshold, and  $\theta_{time}$  is the time prevalence threshold.

For example, in Fig. 2,  $\{A, B\}$  is an MDCOP because it is spatial prevalent in time slots  $t=0$ ,  $t=1$ ,  $t=2$ , and  $t=3$  since its participation indices are no less than the given threshold 0.4 in these time slots, and is time prevalent since its time prevalence index of 1 is above the threshold 0.5. In contrast,  $\{B, D\}$  is not an MDCOP. Although it is spatial prevalent in time slot  $t=2$ , it is not time prevalent since its time prevalence index is no more than the given time prevalence threshold 0.5.

### 3.3 Problem statement

Given:

- A set  $P$  of Boolean spatio-temporal object-types over a common spatio-temporal framework  $STF$ .
- A neighbor relation  $R$  over locations.
- A spatial prevalence threshold,  $\theta_p$ .
- A time prevalence threshold,  $\theta_{time}$ .

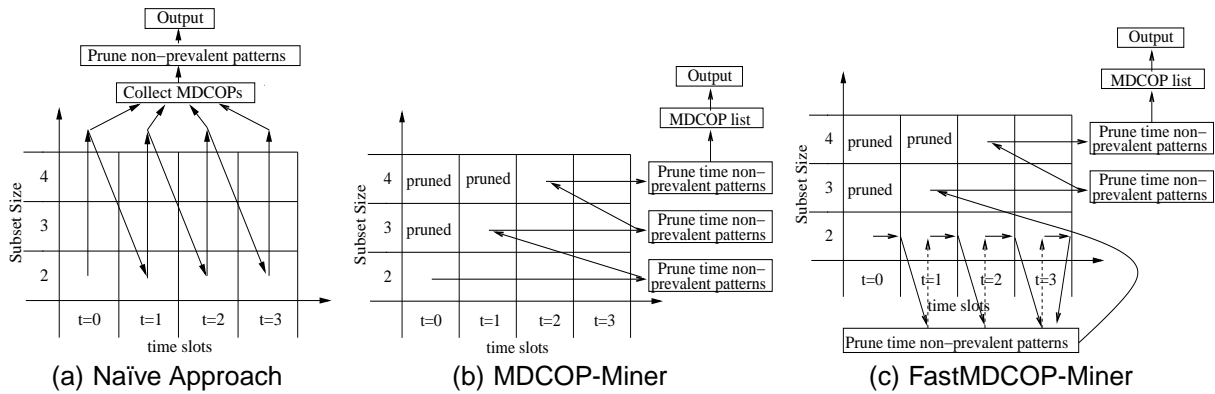


Fig. 3. Comparison of Naïve Approach, MDCOP-Miner, and FastMDCOP-Miner

*Find:*  $\{P_i | P_i \text{ is a subset of } P \text{ and } P_i \text{ is a prevalent MDCOP as in Definition 3.3}\}$ .

*Objective:* Minimize computation cost.

*Constraints:* To find a correct and complete set of MDCOPs.

**Example:** In American football, each play (e.g., Fig. 1) may represent a spatio-temporal dataset and Boolean object-types may be identified by the role of the players (e.g., wide receiver, cornerback, and linebackers). Each object-type is considered as Boolean because we are interested in its presence or absence at any location and time. Fig. 1a-d shows the position of the object-types for four time units. The straight lines between the players show the neighboring objects. The neighbor relation  $R$  may be defined by a distance less than one meter or an average arm's length. For example, in Fig. 1a, wide receiver W.1 is a neighbor of cornerback C.1. However, these players are not neighbors in Fig. 1b since they are separated by more than a meter. In this example,  $\{\text{wide\_receiver, cornerback}\}$  forms a candidate MDCOP, given  $\theta_p=0.5$ , and  $\theta_{time}=0.5$ .

Threshold values selected for MDCOP interest measures (e.g. spatial prevalence measure and time prevalence measure) have important implications on the mining processes and results. Selection of a small interest measure threshold (close to 0) increases the computational complexity of the algorithms and the number of generated prevalent patterns. This may cause generation of insignificant patterns. Selection of a large interest measure threshold (close to 1) decreases the computational complexity of the algorithms and the number of prevalent patterns. This may cause pruning of some of the significant patterns. Nevertheless the selection of interest measure threshold values is dependent on the application and/or purpose of the analysis.

## 4 MINING MDCOPs

In this section, we discuss a naïve approach and then propose two novel MDCOP mining algorithms – MDCOP-Miner and FastMDCOP-Miner – to mine MDCOPs. We also give execution traces of these algorithms.

### 4.1 Naïve approach

A naïve approach can use a spatial co-location mining algorithm for each time slot to find spatial prevalent co-locations and then apply a post-processing step to discover MDCOPs by checking their time prevalence. To mine co-locations, Huang, Shekhar and Xiong proposed a join-based approach, Yoo, Shekhar and Celik proposed a partial join-based approach and a join-less approach, and Zhang et al. proposed a multi-way spatial join-based approach [3], [13], [22], [27], [28], [29], [30]. This study will be based on the join-based co-location algorithm proposed by Huang et al., but it is also possible to use other approaches. The naïve approach will generate size  $k + 1$  candidate co-locations for each time slot using spatial prevalent size  $k$  subclasses until there are no more candidates. After finding all size spatial prevalent co-locations in each time slot, a post-processing step can be used to discover MDCOPs by pruning out time non-prevalent co-locations. Even though this approach will prune out spatial non-prevalent co-locations early, it will not prune out time non-prevalent MDCOPs before the post-processing step (Fig. 3a). This leads to unnecessary computational cost.

### 4.2 MDCOP-Miner

To eliminate the drawbacks of the Naïve approach, we propose an MDCOP mining algorithm (MDCOP-Miner) to discover MDCOPs by incorporating a time-prevalence based filtering step in each iteration of the algorithm. The algorithm, first, will discover all size  $k$  spatial prevalent MDCOPs and then will apply a time-prevalence based filtering to discover MDCOPs. Finally, the algorithm will generate size  $k + 1$  candidate MDCOPs using size  $k$  MDCOPs (Fig. 3b). The participation index is used as a spatial prevalence interest measure to check if the pattern is spatial prevalent at a time slot [13]. The time prevalence (i.e., persistence measure in definition 3.1) is used as a time prevalence interest measure. First we give the pseudo code of the algorithm, and then we provide an execution trace of it using the dataset from Fig. 2. Algorithm 1 gives the pseudo code of the MDCOP-Miner algorithm. This pseudo code is used to explain

two algorithms: MDCOP-Miner and FastMDCOP-Miner. FastMDCOP-Miner will be discussed in the next section. The choice of the algorithm is provided by the user. The inputs are algorithm choice  $alg\_choice$  with value *MDCOP-Miner*, a set of distinct spatial object-types  $E$ , a spatio-temporal dataset  $ST$ , a spatial neighborhood relationship  $R$ , and thresholds of interest measures, i.e. spatial prevalence and time prevalence; the output is a set of MDCOPs. In the algorithm, steps 1 include initialization of the parameters, steps 2 through 14 give an iterative process to mine MDCOPs, and step 15 gives a union of the results. Steps 2 through 14 continue until there are no candidate MDCOPs to be generated. The functions of the algorithm are explained below.

**Generating candidate co-occurrence patterns (step 3):**

This function uses an apriori-based approach to generate size  $k + 1$  candidate co-locations  $C_{k+1}$  for each time slot, using all size  $k$  MDCOPs  $MDP_k$  [2].

**Generating spatial co-occurrence instances (step 5):**

The instances of candidate  $C_{k+1}$  are generated by joining neighbor instances of size  $k$  MDCOPs for each time slot. This is similar to the instance generation step of the co-location miner algorithm [13].

**Finding spatial prevalent co-occurrence patterns (step 6):**

All spatial prevalent size  $k + 1$  patterns  $SP_{k+1}$  are found by pruning the ones whose spatial prevalence indices, i.e., participation indices, are less than a given threshold for each time slot. Computation of the participation indices follows the same algorithmic ideas as those in the co-location mining algorithm [13].

In the *for* loop, the algorithm finds size  $k + 1$  spatial prevalent co-location for each time slot. MDCOP-Miner skips steps 8, 9, and 10 which are activated using the FastMDCOP-Miner.

**Forming a time prevalence table (steps 8 and 12):** In steps 8 and 12, the time prevalence indices of the mined spatial prevalent patterns are calculated in FastMDCOP-Miner and MDCOP-Miner algorithms respectively. The time prevalence index of a spatial prevalent co-location is the fraction of the number of time slots where the pattern occurs over the total number of the time slots. Step 8 is activated in FastMDCOP-Miner algorithm and it is used to calculate the time prevalence index of the size  $k$  patterns before generating size  $k$  patterns of the next time slot. Step 12 is activated in MDCOP-Miner and it is used to calculate time prevalence indices of patterns after size  $k$  patterns of all time slots are generated .

**Finding mixed-drove co-occurrence patterns (step 9 and step 13):** These steps discover MDCOPs by checking the time prevalence indices of the spatial prevalent co-locations if they are no less than a given time prevalence threshold  $\theta_{time}$ . The patterns whose time prevalence indices do not satisfy the given threshold are pruned at this stage. The remaining patterns will be MDCOPs and will be used to generate candidate supersets of the MDCOPs in step 3. In step 13, MDCOP-miner prunes time non-prevalent patterns after all size  $k$  patterns in all

**Algorithm 1: MDCOP-Miner and FastMDCOP-Miner**

**Inputs:**

$alg\_choice$ : MDCOP-Miner or FastMDCOP-Miner  
 $E$ : a set of distinct spatial object-types  
 $ST$ : a spatio-temporal dataset  
 <object\_type, object\_id, x, y, time slot>  
 $R$ : spatial neighborhood relationship  
 $TF$ : a time slot frame  $\{t_0, \dots, t_{n-1}\}$   
 $\theta_p$ : a spatial prevalence threshold  
 $\theta_{time}$ : a time prevalence threshold

**Output:** MDCOPs whose spatial prevalence indices are no less than  $\theta_p$ , for time prevalence indices are no less than  $\theta_{time}$ .

**Variables:**

$k$ : co-occurrence size  
 $t$ : time slots  $(0, \dots, n-1)$   
 $T_k$ : set of instances of size  $k$  co-occurrences  
 $C_k$ : set of candidate size  $k$  co-occurrences  
 $SP_k$ : set of spatial prevalent size  $k$  co-occurrences  
 $TP_k$ : set of time prevalent size  $k$  co-occurrences  
 $MDP_k$ : set of mixed-drove size  $k$  co-occurrences

**Algorithm:**

```

1) initialization :  $k = 1, C_k = E, MDP_k(0) = ST$ 
2) while (not empty  $MDP_k$ ) {
3)    $C_{k+1}(0) = gen\_candidate\_co - occ(C_k, MDP_k)$ 
4)   for each time_slot  $t$  in  $(0, \dots, n-1)$  {
5)      $T_{k+1}(t) = gen\_co\_occ\_inst(C_{k+1}(t), T_k(t), R)$ 
6)      $SP_{k+1}(t) = find\_spatial\_prev\_co\_occ(T_{k+1}(t), C_{k+1}(t), \theta_p)$ 
7)     if ( $alg\_choice == "FastMDCOP-Miner"$ ) {
8)        $TP_{k+1}(t) = find\_time\_index(SP_{k+1}(t))$ 
9)        $MDP_{k+1}(t) = find\_time\_prev\_co\_occ(TP_{k+1}(t), \theta_{time})$ 
10)       $C_{k+1}(t) = MDP_{k+1}(t)$  }
11)   if  $alg\_choice == "MDCOP-Miner"$  {
12)      $TP_{k+1} = find\_time\_index(SP_{k+1})$ 
13)      $MDP_{k+1} = find\_time\_prev\_co\_occ(TP_{k+1}, \theta_{time})$  }
14)    $k = k + 1$  }
15) return union ( $MDP_2, \dots, MDP_{k+1}$ )

```

time slots are generated. In step 9, FastMDCOP-Miner prunes time non-prevalent patterns before generating size  $k$  patterns in the next time slot.

The algorithm will run iteratively until there are no more candidate MDCOPs to be generated. The algorithm outputs the union of all size MDCOPs.

*An Execution Trace of MDCOP-Miner:* The execution trace of the MDCOP-Miner is given in Fig. 4 using the dataset given in Fig. 2a. This dataset contains four object-types A, B, C, and D and their instances in four time slots. A has 4 instances, B has 5 instances, C has 3 instances, and D has 4 instances. The instances of each object-type have a unique identifier, such as A.1. Some of the patterns of these object-types form an MDCOP. To discover MDCOPs we propose a monotonic composite interest measure (the mixed-drove prevalence measure) which is a composition of the spatial prevalence and time prevalence measures applied to mixed object-types. The spatial prevalence measure (participation index) shows the strength of the spatial co-location when the index is greater than or equal to a given threshold [13], [22]. The time prevalence measure (time prevalence index) shows the frequency of the pattern over time.

In Fig. 4a, in step 1, candidate pairs of the distinct object-types and their instances are generated for each time slot. The spatial co-locations whose participation

Step 1: Generate pairs and find participation indices

Co-occurrence patterns	time slot t=0						time slot t=1						time slot t=2						time slot t=3										
	AB	AC	AD	BC	BD	CD	AB	AC	AD	BC	BD	CD	AB	AC	AD	BC	BD	CD	AB	AC	AD	BC	BD	CD					
Co-occurrence pattern instances	A.1 B.1	A.1 C.1	A.4 D.4				A.1 B.1	A.1 C.1	B.1 C.1				A.1 B.1	A.1 C.1	B.1 C.1	B.1 D.1	C.3 D.4		A.1 B.1		A.1 D.2	B.1 C.1	B.1 D.1	C.1 D.1					
	A.2 B.1	A.3 C.2					A.2 B.1	A.3 C.2	B.3 C.2				A.2 B.1	A.3 C.2	B.3 C.2	B.2 D.3			A.2 B.1		A.3 D.3	B.3 C.2							
	A.3 B.2						A.3 B.2		B.4 C.3				A.3 B.2		B.4 C.3				A.3 B.2			B.4 C.3							
	A.3 B.3						A.3 B.3						A.3 B.3						A.3 B.3										
P. ratio	3/4	3/5	2/4	2/3	1/4	1/4							3/4	3/5	2/4	2/3			3/5	3/3									
P. index							3/5	2/4		3/5			3/5	2/4		3/5	2/5	1/4	3/4	3/5		2/4	2/3	3/5	3/3	1/5	1/4	1/3	1/4
If PI threshold is 0.4	• {A,D} is pruned												• {C,D} is pruned						• {B,D} and {C,D} are pruned										

(a) Step 1

Step 2: Form time prevalence table

	time slot t=0	time slot t=1	time slot t=2	time slot t=3	time prevalence index
AB	1	1	1	1	4/4
AC	1	1	1	0	3/4
AD	0	0	0	1	1/4
BC	0	1	1	1	3/4
BD	0	0	1	0	1/4
CD	0	0	0	0	0

- If time prevalence index threshold 0.5 (50%) then prune {A,D} and {B,D}
- {A,B}, {A,C}, {B,C} are mixed-drove co-occurrence patterns

Step 3 Generate superset patterns (triplets)

	time slot t=0	time slot t=1	time slot t=2	time slot t=3
ABC		ABC	ABC	ABC
		A.1 B.1 C.1	A.1 B.1 C.1	
		A.3 B.3 C.2	A.3 B.3 C.2	
PR		2/4 2/5 2/3	2/4 2/5 2/3	
PI		2/5	2/5	

Step 4: Find mixed-drove co-occurrence patterns

	time slot t=0	time slot t=1	time slot t=2	time slot t=3	time prevalence index
ABC	-	1	1	-	2/4

- {A, B, C} is mixed-drove co-occurrence pattern

(b) Steps 2, 3, and 4

Fig. 4. Execution trace of the MDCOP-Miner algorithm

indices are less than a given threshold are then pruned. A spatial non-prevalent pair {A,D} is pruned in time slot t=0, {C, D} is pruned in time slots t=2 and t=3, and {B,D} is pruned in time slots t=3, because their participation indices are less than the given threshold 0.4. A time prevalence table of pairs of spatial prevalent co-locations is then formed by entering a 1 if the participation index of the corresponding pattern satisfies a given participation index threshold. Time-prevalence indices are then found. For example, in the time prevalence table (step 2 in Fig. 4b), spatial prevalent pattern {A,B} is persistent for all time slots and its time prevalence index is 4/4, and spatial prevalent pattern {A,C} is persistent in time slots t=0, t=1, and t=2 and its time prevalence index is 3/4. The MDCOPs whose time prevalence indices are no less than a given threshold are selected for generating superset candidate MDCOPs.

Spatial prevalent patterns {A,B}, {A,C}, and {B,C} are selected as MDCOPs since they are also time prevalent (their time prevalence indices satisfy the given time prevalence threshold 0.5). In contrast, spatial prevalent patterns {A,D}, {B,D}, and {C,D} are pruned since they are time non-prevalent. Using MDCOPs {A,B}, {A,C}, and {B,C}, the next candidate MDCOP {A,B,C} is generated. The next step is to generate instances of candidate {A,B,C} in time slots where its subsets exist and to check its participation indices in these time slots. Since all subsets of MDCOP {A,B,C} are MDCOPs and exist in time slots t=1 and t=2, there is no need to generate instances of them for time slots t=0 and t=3. In step 3 (Fig. 4b), the instances of candidate MDCOP {A,B,C} are generated and participation indices are found which are 2/5 for time slots t=1 and t=2. In step 4 (Fig. 4b), the time prevalence table is formed for pattern {A,B,C} and its time prevalence index is checked to see if it satisfies the

time prevalence threshold. Candidate MDCOP {A,B,C} is an MDCOP since its time prevalence index 0.5 is equal to the threshold 0.5. Since there are not enough subsets to generate the next superset patterns, the algorithm stops at this stage and outputs the union of all size MDCOPs, i.e., {A,B}, {A,C}, {B,C}, and {A,B,C}.

### 4.3 FastMDCOP-Miner

In this section, we propose a new algorithm, called FastMDCOP-Miner, which improves the computational efficiency of the MDCOP-Miner discussed in Section 4.2. As can be seen in Fig. 3b and in Algorithm 1, MDCOP-Miner waits to prune time non-prevalent patterns until all size  $k$  spatial prevalent patterns are generated for all time slots and then prunes time non-prevalent patterns to discover MDCOPs. However, it is possible to optimize the MDCOP-Miner. We propose to prune time-non prevalent patterns as early as possible by moving "prune non-prevalent patterns" between the time slots shown in Fig. 3c where candidate size 2 pattern generation is illustrated. The pseudo-code of the FastMDCOP-Miner is given in Algorithm 1. When the FastMDCOP-Miner is chosen, the algorithm will activate steps 8, 9, and 10 and deactivate steps 12 and 13. This will allow the algorithm to check the time prevalence of a pattern after every time slot is processed. The functions of the algorithm are as described in Section 4.2. In step 8, FastMDCOP-Miner checks whether the time prevalence indices of size  $k$  patterns (size 2 patterns in Fig. 3c) satisfy the time prevalence threshold before generating size  $k$  patterns for the next time slot. Early discovered time non-prevalent patterns are pruned in Step 9 and time prevalent patterns are used as candidate co-occurrences (Step 10) in the next time slot. For example, assume

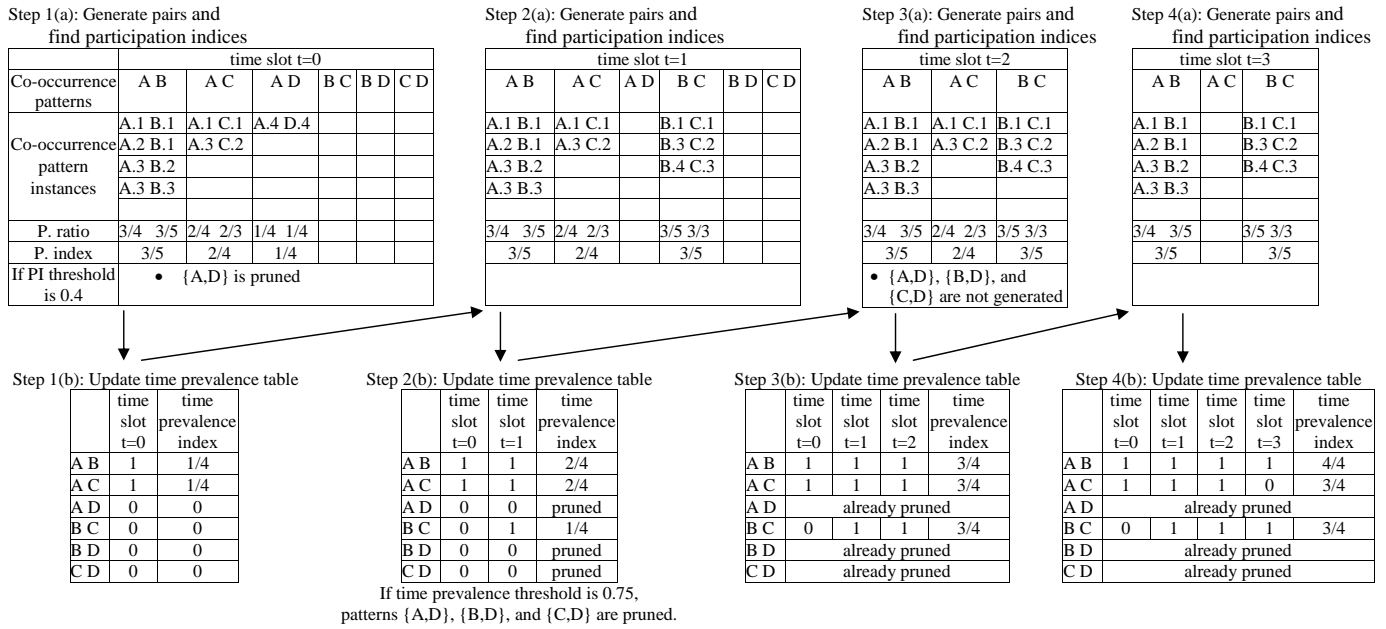


Fig. 5. Execution trace of the FastMDCOP-Miner algorithm

that there are 10 time slots and the time prevalence threshold is 0.5. In this case, a size  $k$  pattern should be present for at least 5 time slots to satisfy the threshold. If the time prevalence index of a pattern is 0 for the first (or any) 6 time slots, there is no need to generate it and check its prevalence for the rest of the time slots. Even if it is time persistent for the remaining 4 time slots, it will not be able to satisfy the given threshold.

*An Execution Trace of FastMDCOP-Miner:* The execution trace of the FastMDCOP-Miner is given in Fig. 5 using the dataset given in Fig. 2a, which has 4 time slots. Assume that the spatial prevalence threshold is 0.4 and the time prevalence threshold is 0.75. If a pattern is not consistent in more than 1 out of 4 time slots, it can be pruned whenever it is discovered. In Step 1(a) pairs and their instances are generated. Pattern {A, D} is pruned at this step since it is spatial non-prevalent. Based on the outcomes of Step 1(a), the prevalence table is updated by entering a 1 for spatial prevalent patterns (Step 1(b)). The time prevalence table initially contains all possible pairs of subsets of object-types. The algorithm checks if time non-prevalent patterns can be discovered at this step. Since the results of one time slot are not enough to make a decision, instances for patterns given in the time prevalence table are generated for time slot  $t=2$  (Step 2(a)) and the time prevalence table is updated (Step 2(b)). Patterns {A, D}, {B, D}, and {C, D} are pruned in Step 2(b), since they are not time consistent in time slot  $t=0$  and  $t=1$ , (they will not be time prevalent even if they are spatial prevalent in the remaining time slots  $t=2$  and  $t=3$ ). These patterns are not present in 2 or more of the 4 time slots. In Step 3(a) instances are generated for patterns {A, B}, {A, C}, and {B, C} which are candidate MDCOPs and their time prevalence indices are updated

(Step 3(b)). At this step no pattern is pruned since there is no possible time non-prevalent pattern. Similarly, in Step 4(a) instances are generated for patterns {A, B}, {A, C}, and {B, C} which are candidate MDCOPs and their time prevalence indices are updated (Step 4(b)). At this step no pattern is pruned since there is no possible time non-prevalent pattern and algorithm outputs MDCOPs {A, B}, {A, C}, and {B, C}. Next, the algorithm continues to discover possible MDCOPs by generating candidate triple patterns.

## 5 ANALYSIS OF THE MDCOP-MINER

This section gives the analysis of the mixed-drove prevalence index measure, and correctness and completeness derivations for the MDCOP mining algorithms.

### 5.1 The Mixed-Drove Prevalence Index Measure is Monotonic

**Lemma 5.1:** The spatial prevalence measure participation index and participation ratio are monotonically non-increasing in the size of the MDCOPs at each time slot [13], [22].

*Proof:* The participation ratio  $pr$  is monotonically non-increasing because an instance of a spatial object-type that is contributing to a co-location  $c_i$  is also contributing to a co-location  $c_j$  where  $c_j \subseteq c_i$ . The participation index  $pi$  is also monotonic because 1) the participation ratio is monotonic and 2)

$$\begin{aligned}
 pi(c \cup f_{k+1}) &= \min_{i=1}^{k+1} \{pr(c \cup f_{k+1}, f_i)\} \\
 &\leq \min_{i=1}^k \{pr(c \cup f_{k+1}, f_i)\} \\
 &\leq \min_{i=1}^k \{pr(c \cup f_i)\} = pi(c)
 \end{aligned}$$

□

**Lemma 5.2:** A mixed-drove prevalence index measure is monotonically non-increasing with the size of MDCOP over space and time. In other words, it is monotonically non-increasing, if MDCOP  $P_i$  is a subset of MDCOP  $P_j$  and

$$\begin{aligned} & Prob_{t_m \in TF}(s\_prev(P_i, t_m) \geq \theta_p), \text{ and} \\ & Prob_{t_m \in TF}(s\_prev(P_j, t_m) \geq \theta_p), \end{aligned}$$

where  $Prob$  stands for the probability of overall prevalence time units,  $s\_prev$  stands for spatial prevalence,  $\theta_p$  is the spatial prevalence threshold, and  $t_m$  is the time slot.

*Proof:* The basic proof sketch follows. Let

$$TS(P_j, \theta_p) = \{t_m | pi(P_j, t_m) \geq \theta_p\}.$$

Lemma 5.1 implies that participation index  $pi(P_j, t_m) \geq \theta_p$  for all  $t_m \in TS(P_j, \theta_p)$ , since  $P_i$  is a subset of  $P_j$ . Thus,

$$Prob_{t_m \in TF}[s\_prev(P_i, t_m) \geq \theta_p] \in \theta_{time},$$

where  $\theta_{time}$  is the time prevalence threshold.  $\square$

## 5.2 Correctness and Completeness

**Theorem 5.1:** The FastMDCOP-Miner, MDCOP-Miner, and naive approach are complete.

*Proof:* The FastMDCOP-Miner, MDCOP-Miner, and naive approach are complete if they find all MDCOPs that satisfy a given participation index threshold and time prevalence threshold. We can show this by proving that none of the functions of the algorithm miss any patterns, i.e., filter out a prevalent MDCOP.

The  $gen\_candidate\_co\_occ$  function does not miss any patterns given the anti-monotone nature of the MDCOP interest measure. The input of this function is size  $k$  MDCOPs and the output is candidate size  $k + 1$  MDCOPs. If  $c_1 = \{f_1, \dots, f_k\}$  and  $c_2 = \{f_1, \dots, f_{k-1}, f_{k+1}\}$  are size  $k$  MDCOPs, size  $k + 1$  pattern  $C_{k+1} = \{f_1, \dots, f_{k-1}, f_k, f_{k+1}\}$  will be generated by joining size  $k$  MDCOPs.

The  $gen\_co\_occ\_instance$  function does not miss any patterns. This function generates instances of candidate size  $k + 1$  MDCOPs by joining instances of size  $k$  MDCOPs if they are in the neighborhood distance and forming a clique.

The  $find\_spatial\_prevalent\_co\_occ$  function does not miss any patterns. It finds spatial prevalent patterns whose participation indices satisfy a given threshold.

The  $find\_time\_prevalence\_index$  function does not miss any patterns. This function calculates time prevalence indices of the patterns found in steps 4 through 8 and does not do any pruning.

The  $find\_time\_prevalent\_co\_occ$  function does not miss any MDCOPs. The function finds all the MDCOPs whose time prevalence indices are no less than a given threshold.  $\square$

**Theorem 5.2:** The FastMDCOP-Miner, MDCOP-Miner, and naive approach are correct. In other words,

if a MDCOP pattern  $P$  is returned by MDCOP-Miner and FastMDCOP-Miner algorithms, then  $P$  is a prevalent MDCOP.

*Proof:* The proof is easy to establish due to the pruning steps of  $find\_spatial\_prevalent\_co\_occ$ , and  $find\_time\_prevalent\_co\_occ$ , which weed out candidates not meeting the given thresholds.  $\square$

We also analyzed the algebraic cost models of the MDCOP mining algorithms MDCOP-Miner and FastMDCOP-Miner. The detail of this analysis can be found in the technical paper [6].

## 6 EXPERIMENTAL EVALUATION

In this section, we present our experimental evaluations of several design decisions and workload parameters on our MDCOP mining algorithms. We used a real-world training dataset and synthetic datasets. We evaluated the behavior of the FastMDCOP-Miner, MDCOP-Miner and naive approach to answer the following questions:

- What is the effect of the number of timeslots?
- What is the effect of the number of object-types?
- What is the effect of the spatial prevalence threshold?
- What is the effect of the time prevalence threshold?
- What is the effect of the number of noise instances?
- What is the effect of the average number of instances?

Fig. 6 shows the experimental setup to evaluate the impact of design decisions on the performance on the three algorithms. Experiments were conducted on an IBM Netfinity Linux Cluster, 2.6 GHz Intel Pentium 4 with 1.5 GB of RAM.

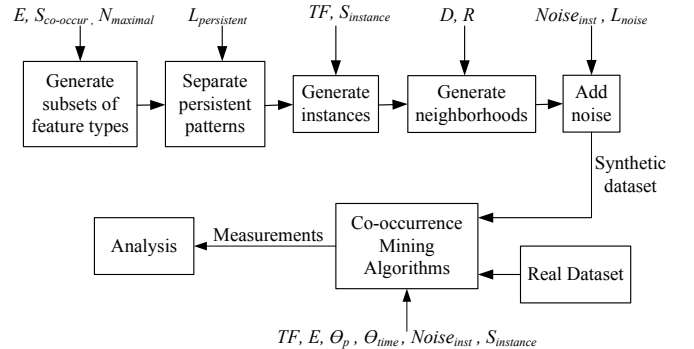


Fig. 6. Experimental setup and design

### 6.1 Datasets

#### 6.1.1 Real Dataset

The real dataset contains the location and time information of moving objects. It includes 15 time snapshots and 22 distinct vehicle types and their instances. The minimum instance number is 2, the maximum instance number is 78, and the average number of instances is 19.

TABLE 1  
Synthetic Dataset Generation Parameters

Parameter	Definition	Experiments					
		Syn-1	Syn-2	Syn-3	Syn-4	Syn-5	Syn-6
$E$	Number of object types	200	100-400	200			
$S_{instance}$	Average number of co-occurrence instances*	10				5-20	
$N_{maximal}$	Number of maximal co-occurrence patterns*	10					
$S_{co-occur}$	Minimum co-occurrence pattern size*	4					
$L_{persistent}$	Ratio of persistent patterns over transitory patterns*	0.5	0.8	0.5	0.8		
$L_{noise}$	Ratio of noise object-types over number of object-types	0.25					
$Noise_{inst}$	Number of noise instances	1000			1000-5000		1000
$TF$	Number of timeslots	10-50	20	50	20		
$D$	Spatial framework size ( $D \times D$ )	$10^6$	$10^3$	$10^6$	$10^3$		
$R$	Spatial Neighborhood relationship	10					

\* : For initial co-occurrence patterns

### 6.1.2 Synthetic Dataset Generation

To evaluate the performance of the algorithms, spatio-temporal datasets were generated based on the spatial data generator proposed by Huang et. al. [13]. Synthetic datasets were generated for spatial frame size  $D \times D$  (the first part of Fig. 6). For simplicity, the datasets were divided into regular grids whose side lengths had neighborhood relationship  $R$ . First, subsets of object-types were generated using the parameters average co-occurrence size  $S_{co-occur}$  and number of maximal patterns  $N_{maximal}$ . Object-types and sizes of each pattern were chosen randomly. The generated patterns were then divided into two categories – persistent patterns and transitory patterns – using the persistent pattern ratio  $L_{persistent}$ . Persistent patterns are ones whose time prevalences are strong over time, while transitory patterns are ones whose spatial prevalences are strong at a specific time slot. Next, instances of the patterns were generated based on the average number of co-occurrence instances  $S_{instance}$ . Instances were chosen at randomly located grid cells. This process was applied for each time slot  $TF$ . Finally, using the parameters number of noise instances  $Noise_{inst}$  and ratio of noise objects over number of features  $L_{noise}$ , noise object and their instances were generated and added to the dataset.

The parameters of the synthetic dataset generator and their definitions are listed in Table 1.

## 6.2 Experiment Results for Real Datasets

### 6.2.1 Effect of Number of Time Slots

We evaluated the effect of the number of time slots on the execution time of the MDCOP algorithms using the real dataset. The participation index, time prevalence index, and distance were set at 0.2, 0.8, and 150m respectively. Experiments were run for a minimum of 1 time slot and a maximum of 14 time slots. Results showed that the FastMDCOP-Miner requires less execution time than the MDCOP-Miner and naïve approaches, since it prunes out time non-prevalent MDCOPs as early as possible (Fig. 7a). As the number of time slots increases, the ratio of the increase in execution time is smaller for FastMDCOP-Miner than for the other approaches.

Fig. 7b shows the number of generated size 2 and size 3 instances for algorithms. The FastMDCOP-Miner generates fewer patterns due to its early pruning strategy. The MDCOP-Miner and naïve approaches generate the same number of size 2 instances.

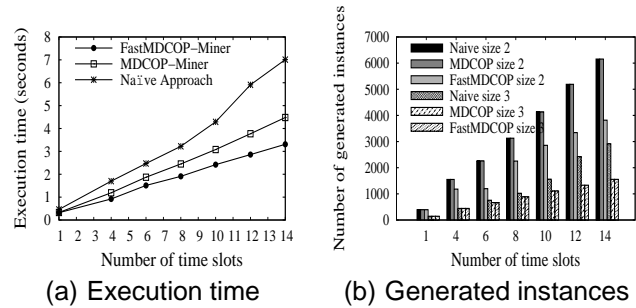


Fig. 7. Effect of number of time slots using real dataset

### 6.2.2 Effect of Number of Object-types

We evaluated the effect of the number of object-types on the execution time of the algorithms using the real dataset. The participation index, time prevalence index, number of time slots and distance were set at 0.2, 0.8, 15, and 150m respectively. Results showed that the FastMDCOP-Miner outperforms the other approaches as the number of object-types increases (Fig. 8a-b). It is observed that the increase in execution time for the naïve approach is bigger than that of the MDCOP-Miner and the FastMDCOP-Miner as the number of object-types increases for datasets.

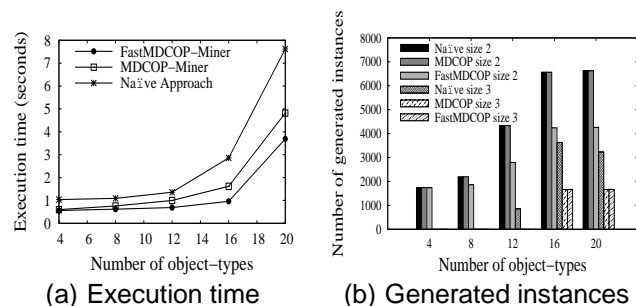


Fig. 8. Effect of number of object-types using real dataset

The execution times of the algorithms increases as the number of the object-types increases due to the increase in the number of join operations. The MDCOP-Miner and naïve approaches generate the same number of size 2 instances. In contrast, the FastMDCOP-Miner generates fewer size 2 instances due to its early pruning strategy (Fig. 8b).

### 6.2.3 Effect of the Time Prevalence Threshold

We evaluated the effect of the time prevalence threshold on the execution times of the MDCOP mining algorithms for the real dataset. The fixed parameters were participation index, number of time slots, and distance, and their values were 0.2, 15, and 150m respectively. For the naïve approach, the effective cost in execution time to generate spatial prevalent co-locations will be constant since it generates the same number of spatial prevalent patterns as the time prevalence index increases. In that case, the cost of the post-processing step will reflect the trend of the naïve approach. Experimental results show that the FastMDCOP-Miner is more computationally efficient than the other approaches (Fig. 9a-b). The execution times of the FastMDCOP-Miner and MDCOP-Miner decrease as the time prevalence threshold increases. It is also observed that the naïve approach is computationally more expensive as the time prevalence threshold decreases because of the increase in the number of MDCOPs to be discovered.

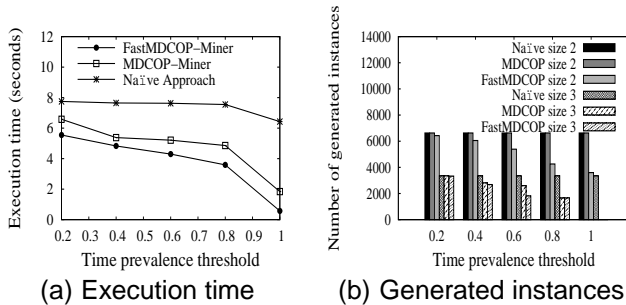


Fig. 9. Effect of the time prevalence threshold using real dataset

### 6.2.4 Effect of the Spatial Prevalence Threshold

We evaluated the effect of the spatial prevalence threshold on the execution times of the MDCOP algorithms. The fixed parameters were time prevalence index, number of time slots, and distance, with values of 0.2, 15, and 100m respectively. Fig. 10a shows the execution times of the algorithms and Fig. 10b shows the number of generated size 2 and 3 instances for the algorithms. FastMDCOP-Miner and MDCOP-Miner do not generate more than size 3 instances for a spatial prevalence threshold of greater than 0.2. The FastMDCOP-Miner outperforms the MDCOP-Miner and naïve approaches as the spatial prevalence threshold increases (Fig. 10a-b). The cost of the naïve approach will be higher than that of the FastMDCOP-Miner and MDCOP-Miner for low values of the spatial prevalence threshold.

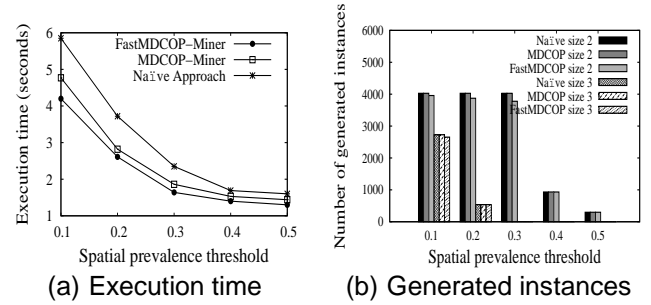


Fig. 10. Effect of the spatial prevalence threshold using real dataset

## 6.3 Experiment Results for Synthetic Datasets

### 6.3.1 Effect of Number of Time slots

We evaluated the effect of the number of time slots on the execution time of the algorithms using synthetic datasets. To generate the datasets, we used a framework size of  $10^6 \times 10^6$ , a square proximity neighborhood size of  $10 \times 10$ , a noise feature ratio of 0.25, a noise instance number of 1000, an average number of co-occurrence instances of 10, and a maximal co-occurrence pattern number of 10 (Table 1, column syn-1). The parameters used to generate the datasets are given in Table 1 column syn-1. The participation index, time prevalence index, and distance were set at 0.3, 0.9, and 10m respectively. Experiments were for a minimum of 10 time slots and a maximum of 50 time slots. The results showed that the FastMDCOP-Miner requires less execution time than the other approaches, since it prunes out time non-prevalent MDCOPs as early as possible (Fig. 11a-b). The generated size 2 and size 3 instances are given in Fig. 11b. The naïve approach generated up to size 7 spatial prevalent subsets before the post-processing step. In contrast, FastMDCOP-Miner and MDCOP-Miner generated up to size 4 subsets.

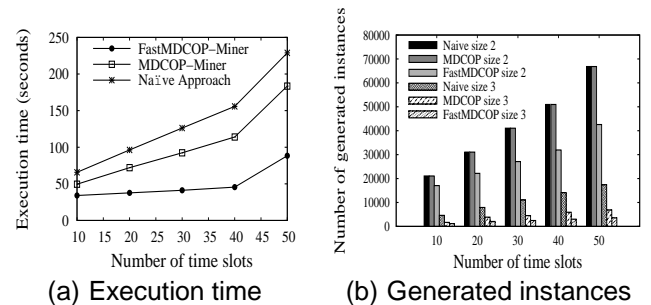


Fig. 11. Effect of number of time slots using synthetic dataset

The execution times of the algorithms increases as the number of time slots increases and the FastMDCOP-Miner outperforms the other approaches due to its early pruning strategy. As can be seen in Fig. 11b, the FastMDCOP-Miner algorithm examines fewer size 2 and 3 instances than the other approaches.

### 6.3.2 Effect of Number of Object-types

We evaluated the effect of the number of object-types on the execution time of the algorithms for synthetic datasets. The parameters used to generate the datasets are given in Table 1, column syn-2. The participation index, time prevalence index, number of time slots, and distance were set at 0.3, 0.8, 20, and 10m respectively. The FastMDCOP-Miner outperforms the MDCOP-Miner and naïve approaches as the number of object-types increases (Fig. 12a-b). The ratio of the increase in the execution time of the naïve approach is greater than that of the MDCOP-Miner and FastMDCOP-Miner as the number of object-types increases. Fig. 12b shows the number of generated size 2 and 3 instances for the algorithms. The execution times of the algorithms increases as the number of the object-types increases due to the increase in the number of join operations.

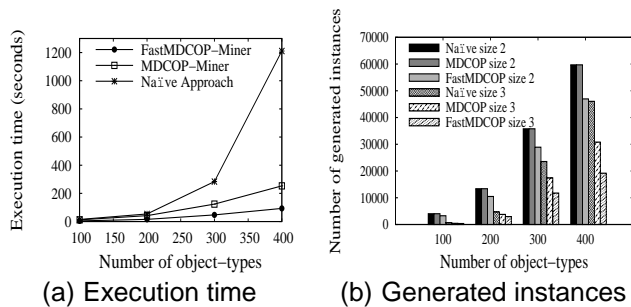


Fig. 12. Effect of number of object-types using synthetic dataset

### 6.3.3 Effect of the Time Prevalence Threshold

We evaluated the effect of the time prevalence threshold on the execution times of the algorithms for synthetic datasets. The parameters used to generate the datasets are given in Table 1 column syn-3. The fixed parameters were participation index, distance, and number of time slots, and their values were 0.4, 10m, and 50 respectively. Experimental results show that the FastMDCOP-Miner is more computationally efficient than the MDCOP-miner and naïve approaches (Fig. 13a).

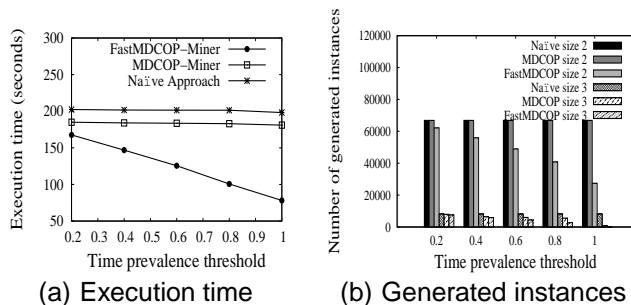


Fig. 13. Effect of the time prevalence threshold using synthetic dataset

The execution time of the FastMDCOP-Miner decreases as the time prevalence threshold increases. The

execution time of the naïve approach is almost constant since it does not prune time non-prevalent pattern before the post-processing step and it is computationally more expensive as the time prevalence threshold decreases because of the increase in the number of MDCOPs to be generated (Fig. 13b).

### 6.3.4 Effect of the Spatial Prevalence Threshold

We evaluated the effect of the spatial prevalence threshold on the execution times of MDCOP mining algorithms. To generate the dataset, we used a spatial framework size of  $10^6 \times 10^6$ , a square proximity neighborhood size of  $10 \times 10$ , an average number of co-occurrence instances of 10, a noise feature ratio of 0.25, a noise instance number of 1000, a maximal co-occurrence pattern number of 10, and a time slot number of 50 (Table 1, column syn-4). The parameters used to generate the datasets are given in Table 1 column syn-4. The value of the time prevalence index was set to 0.8. The FastMDCOP-Miner outperforms the other approaches (Fig. 14a-b). The cost of the naïve approach will be higher than the MDCOP-Miner and FastMDCOP-Miner for low values of the spatial prevalence threshold since the naïve approach tends to generate MDCOP non-prevalent patterns. For high values of the spatial prevalence index the cost of the algorithms are closer. The generated size 2 and size 3 instances are given in Fig. 14b. The naïve approach generates up to size 8 spatial prevalent subsets for spatial prevalence thresholds 0.2 and 0.3, before the post-processing step. In contrast, the FastMDCOP-Miner and MDCOP-Miner generate up to size 4 MDCOP prevalent subsets.

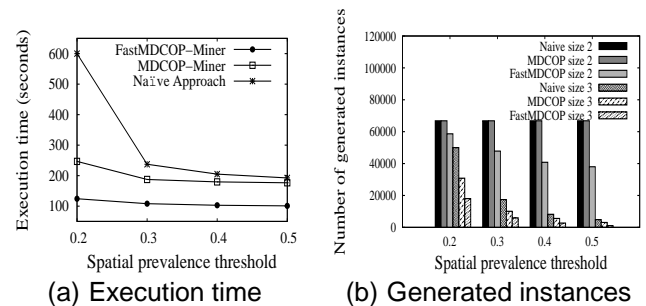


Fig. 14. Effect of the spatial prevalence threshold using synthetic dataset

### 6.3.5 Effect of the Number of Noise Instances

We evaluated the effect of the number of noise instances on the execution times of the MDCOP mining algorithms using the synthetic datasets. The parameters for data generation are listed in column syn-5 of Table 1. The time prevalence threshold, the spatial prevalence threshold, and distance were 0.3, 0.8, and 10m respectively. The FastMDCOP-Miner is more robust than the MDCOP-Miner and naïve approaches as the number of noise features increases (Fig. 15a).

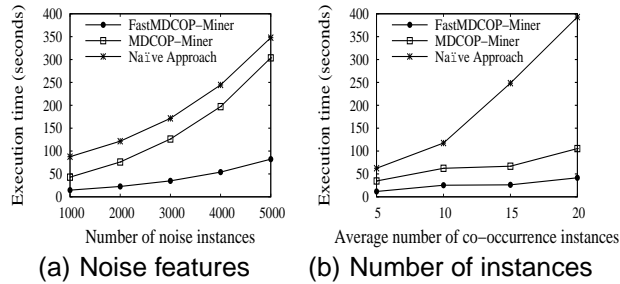


Fig. 15. Effect of noise and average number of instances using synthetic dataset

### 6.3.6 Effect of the Average Number of Instances

We evaluated the average number of total instances on the execution times of the MDCOP mining algorithms using synthetic datasets. The parameters for data generation are listed in column syn-6 of Table 1. The time prevalence threshold, the spatial prevalence threshold, and distance were 0.3, 0.8, and 10m respectively. The FastMDCOP-Miner algorithm outperformed the other approaches as the average number of total instances increases (Fig. 15b).

## 7 CONCLUSIONS AND FUTURE WORK

We defined mixed-drove spatio-temporal co-occurrence patterns (MDCOPs) and the MDCOP mining problem and proposed a new monotonic composite interest measure which is the composition of distinct object-types, spatial prevalence measures, and time prevalence measures. We presented a novel and computationally efficient algorithm (the MDCOP-Miner) for mining these patterns. We also presented an improved MDCOP-Miner algorithm (the FastMDCOP-Miner) which prunes time non-prevalent patterns at an early stage and offers even greater computational efficiency than the MDCOP-Miner algorithm. We compared our algorithms with a naïve approach, which runs the spatial co-location mining algorithm at each time slot and then discovers MDCOPs using a post-processing step. We proved that the proposed algorithms are correct and complete in finding mixed-drove prevalent (i.e., spatial prevalent and time prevalent) MDCOPs. Our experimental results using a real and synthetic datasets provide further evidence of the viability of our approach.

For future work, we would like to explore the relationship between the proposed MDCOP interest measures and spatio-temporal statistical measures of interaction [3]. Another problem of interest is the characterization of the probability distribution of the proposed interest measure to help in making the choice of thresholds in the proposed measures. We plan to explore other potential interest measures for MDCOPs by evaluating similarity measures for tracks of moving objects. We plan to investigate new monotonic composite interest measures and develop other new computationally efficient algorithms for mining MDCOPs.

In the literature, there are also other studies that have focused on defining spatio-temporal patterns and algorithms [10], [12], [14], [16], [19], [26]. Laube et al. defined several spatio-temporal patterns, such as leadership and convergence [17]. Query processing algorithms have been proposed to extract such patterns [17]. We hope to extend our algorithm to mine these patterns.

## ACKNOWLEDGMENTS

This work was partially supported by the US Army Corps of Engineers under contract number W9132V-06-C-0011, the NSF grant 0431141, the NSF grant 0708604, the NSF grant 0713214, and the NGA grant. The authors would like to thank the members of the Spatial Database and Spatial Data Mining Research Group at the University of Minnesota for their comments. We would also like to express our thanks to Kim Koffolt for improving the readability of this paper.

## REFERENCES

- [1] A. Brix and P.J. Diggle, "Spatio-temporal Prediction for Log-Gaussian Cox Processes," *Journal of the Royal Statistical Society*, vol. 63, no. 10, pp. 823-841, 2001.
- [2] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," *Proc. 20th Int'l Conf. on Very Large Data Bases (VLDB'94)*, Santiago, Chile, 1994.
- [3] S. Banerjee, B.P. Carlin, and A.E. Gelfand, *Hierarchical Modeling and Analysis for Spatial Data*, CRC Press, 2003.
- [4] H. Cao, N. Mamoulis, and D.W. Cheung, "Discovery of Collocation Episodes in Spatiotemporal Data," *Proc. 6th IEEE Int'l Conf. on Data Mining (ICDM)*, pp. 823-827, Hong Kong, China, 2006.
- [5] M. Celik, S. Shekhar, J.P. Rogers, J.A. Shine, and J.S. Yoo, "Mixed-drove Spatio-temporal Co-occurrence Pattern Mining: A Summary of Results," *Proc. the 6th IEEE Int'l Conf. on Data Mining (ICDM)*, pp. 119-1287, Hong Kong, China, 2006.
- [6] M. Celik, S. Shekhar, J.P. Rogers, and J.A. Shine, "Mixed-drove Spatio-temporal Co-occurrence Pattern Mining" *University of Minnesota, Computer Science Dept. Tech. Report*, no. 08-015, 2008.
- [7] N.A.C. Cressie, *Statistics for Spatial Data*, Wiley, 1993.
- [8] C.W.C. Granger, "Time Series Analysis, Cointegration, and Applications," *Nobel Prize lecture, Department of Economics, University of California, San Diego. Paper 2004-02.*, 2004.
- [9] J. Gudmundsson and M.v. Kreveld, "Computing Longest Duration Flocks in Trajectory Data," *Proc. the 14th annual ACM Int'l Symp. on Geographic Information Systems (ACM-GIS)*, pp. 35-42, Virginia, USA, 2006.
- [10] J. Gudmundsson, M.v. Kreveld, and B. Speckmann, "Efficient Detection of Motion Patterns in Spatio-Temporal Data Sets," *Proc. the 12th annual ACM Int'l Workshop on Geographic Information Systems (ACM-GIS)*, pp. 250-257, Washington DC, USA, 2004.
- [11] R.H. Guting and M. Schneider, *Moving Object Databases*, Morgan Kaufmann, 2005.
- [12] M. Hadjieleftheriou, G. Kollios, P. Bakalov, and V.J. Tsotras, "Complex Spatio-Temporal Pattern Queries," *Proc. the 31st Int'l Conf. on Very Large Data Bases*, pp. 877-888, Trondheim, Norway, 2005.
- [13] Y. Huang, S. Shekhar, and H. Xiong, "Discovering Co-location Patterns from Spatial Datasets: A General Approach," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, vol. 16, no. 12, pp. 1472-1485, 2004.
- [14] P. Kalnis, N. Mamoulis, and S. Bakiras, "On Discovering Moving Clusters in Spatio-Temporal Data," *Proc. 9th Int'l Symp. on Spatial and Temporal Databases (SSTD)*, Angra dos Reis, Brazil, 2005.
- [15] M. Koubarakis, T.K. Sellis, A.U. Frank, S. Grumbach, R.H. Guting, C.S. Jensen, N.A. Lorentzos, Y. Manolopoulos, E. Nardelli, B. Pernici H.J. Schek, M. Scholl, B. Theodoulidis, and N. Tryfona *Spatio-Temporal Databases: The Chorochronos Approach*, LNCS 2520, Springer Verlag, 2003.

- [16] P. Laube and S. Imfeld, "Analyzing Relative Motion within Groups of Trackable Moving Point Objects," *Proc. the Second Int'l Conf. on Geographic Information Science (GIScience)*, LNCS 2478, pp. 132-144. Springer, Berlin, 2002.
- [17] P. Laube, M.v. Kreveld, and S. Imfeld, "Finding REMO - Detecting Relative Motion Patterns in Geospatial Lifelines," *Proc. 11th Int'l Symp. on Spatial Data Handling*, London, pp. 201-214, Springer, Berlin Heidelberg New York, 2004.
- [18] J. Ma, D. Zeng, and H. Chen, "Spatial-temporal Cross-correlation Analysis: A New Measure and A Case Study in Infectious Disease Informatics," *Proc. the 2006 IEEE Int'l Conf. on Intelligence and Security Informatics*, pp. 542-547, San Diego, CA, 2006.
- [19] C. Mouza and P. Rigaux, "Mobility Patterns," *Geoinformatica*, vol. 9, no. 4, pp. 297-319, 2005.
- [20] B.D. Ripley, *Spatial Statistics*, Wiley, 1981.
- [21] O. Schabenberger and C.A. Gotway, *Statistical Methods for Spatial Data Analysis*, Chapman and Hall, 2004.
- [22] S. Shekhar, Y. Huang, and H. Xiong, "Discovering Spatial Colocation Patterns: A Summary of Results," *Proc. 7th Int'l Symp. on Spatial and Temporal Databases (SSTD)*, pp.236-256, L.A., CA, 2001.
- [23] SSTD06, "First Int'l Workshop on Spatial and Spatio-Temporal Data Mining," *In Conjunction with the 6th IEEE Int'l Conf. on Data Mining (ICDM)*, 2006.
- [24] J. Wang, W. Hsu, and M.L. Lee, "A Framework for Mining Topological Patterns in Spatio-temporal Databases," *Proc. ACM 14th Conf. on Information and Knowledge Management (CIKM)*, pp. 429 -436, Bremen, Germany, 2005.
- [25] W.W.S. Wei, *Time Series Analysis: Univariate and Multivariate Methods*, Addison Wesley, 2005.
- [26] H. Yang, S. Parthasarathy, and S. Mehta, "A Generalized Framework for Mining Spatio-temporal Patterns in Scientific Data," *Proc. the 11th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 716-721, Chicago, Illinois, USA, 2005.
- [27] J.S. Yoo and S. Shekhar, "A Partial Join Approach for Mining Co-location Patterns," *Proc. the 12th annual ACM Int'l Workshop on Geographic information systems (ACM-GIS)*, Washington D.C., USA, 2005.
- [28] J.S. Yoo and S. Shekhar, "A Joinless Approach for Mining Spatial Colocation Patterns," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, vol. 18, no. 10, 2006.
- [29] J.S. Yoo, S. Shekhar, and M. Celik, "A join-less Approach for Colocation Pattern Mining: A Summary of Results," *IEEE Int'l Conf. on Data Mining*, Houston, USA, 2005.
- [30] X. Zhang, N. Mamoulis, D.W. Cheung, and Y. Shou, "Fast Mining of Spatial Collocations," *Proc. ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 384-393, Seattle, WA, 2004.



**Mete Celik** received the B.S. degree in Control and Computer Engineering from Erciyes University, Kayseri, Turkey, in 1999 and the M.S. degree in Electrical Engineering from Erciyes University, Kayseri, Turkey, in 2001. He is currently working toward the Ph.D. degree in Computer Science at the University of Minnesota, Minneapolis, US. His research interests include spatial databases, spatial data mining, spatio-temporal data mining, and location-based services. He is a student member of the IEEE

Computer Society.



**Shashi Shekhar** received the B. Tech degree in Computer Science from the Indian Institute of Technology, Kanpur, India, in 1985, the M.S. degree in Business Administration and the Ph.D. degree in Computer Science from the University of California, Berkeley, CA, USA, in 1989. He is a McKnight Distinguished University Professor at the University of Minnesota, Minneapolis, MN, USA. His research interests include spatial databases, spatial data mining, geographic and information systems (GIS), and intelligent transportation systems. He is a co-author of a textbook on Spatial Databases (Prentice Hall, 2003, ISBN 0-13-017480-7 ), co-edited an Encyclopedia of GIS (Springer, 2008, isbn 978-0-387-30858-6 ) and has published over 200 research papers in peer-reviewed journals, books, conferences, and workshops. He is a co-Editor-in-Chief of Geo-Informatica: An Int'l Journal on Advance in Computer Sc. for GIS and has served on the editorial boards of IEEE Trans. on Knowledge and Data Engineering as well as the IEEE-CS Computer Science & Engineering Practice Board. He also served as a program co-chair of the ACM Int'l. Workshop on Advances in Geographic Information Systems, 1996. He is serving on the steering committee of the ACM Symp. on Geographic Information Systems and has served as a member of the mapping science committee of the Nat'l Research Council National Academy of Sciences (2004-9), as well as the Board of Directors of University Consortium on GIS (2003-2004). Dr. Shekhar received the IEEE Technical Achievement Award (2006), and is a Fellow of the IEEE Computer Society, as well as a member of the ACM.



**James P. Rogers** received a B.S. in Civil Engineering in 1983 from North Carolina State University, Raleigh, NC, an M.S. in Computer-Aided Design in 1990 from George Washington University, Washington, DC, and is currently working on his Ph.D. in Spatial Data Mining at George Mason University, Fairfax, VA. He is currently an Associate Technical Director for the U.S. Army Engineer Research and Development Center in Alexandria, VA and is the Principal Investigator for the Modelling of Spatio-temporal

Co-occurrence Patterns and Statistical Determination of Spatial Outliers basic research projects. He has published papers in peer-reviewed journals, conferences, and workshops.



**James A. Shine** received a B.A. in Mathematics and Chemistry in 1979 from the University of Virginia, Charlottesville, VA, an M.S. in Statistical Science in 1996 from George Mason University, Fairfax, VA, and a Ph.D. in Computational Statistics in 2003 from George Mason University. He is a Mathematician for the U.S. Army Topographic Engineering Center in Alexandria, VA. His research interests include spatial and spatio-temporal statistics, spatial and spatio-temporal data mining, and geographical evidential reasoning.

He has published over 40 papers in peer-reviewed journals, conferences, workshops and technical notes. He is a member of the American Statistical Association, the Institute for Mathematical Statistics, the Interface Foundation of North America, and the Washington Statistical Society.