

## PREDICT THE LOCATION OF MOVING OBJECTS USING MINING ASSOCIATION RULES OF MOVEMENT PATTERNS

NGUYEN TIEN PHUONG, DANG VAN DUC

*Institute of Information Technology - VAST; E-mail: {[phuongnt](mailto:phuongnt@ioit.ac.vn); [dvdudc](mailto:dvdudc@ioit.ac.vn)}@ioit.ac.vn*

**Tóm tắt.** Dịch vụ trên cơ sở vị trí địa lý (LBS) là dịch vụ thông tin giải trí, được truy cập bởi các thiết bị di động qua mạng điện thoại và được cung cấp nhờ tận dụng vị trí của thiết bị. LBS sử dụng thông tin vị trí của người dùng để đưa ra các dịch vụ khác nhau theo ngữ cảnh. Rất nhiều tình huống trong thực tế đòi hỏi LBS phải có khả năng dự đoán vị trí sắp tới của các đối tượng chuyển động. Đã có khá nhiều nghiên cứu về dự đoán vị trí của đối tượng chuyển động trong tương lai. Một số tập trung theo hướng lô-gíc mờ, xác suất thống kê hay kết hợp nhiều phương pháp. Mục tiêu của bài báo này là giới thiệu khung quy trình cho mô hình hóa và trích chọn mẫu hình, mà rất cần thiết trong việc mô hình hóa thiết kế cơ sở dữ liệu các đối tượng chuyển động. Đồng thời bài báo còn đưa ra một phương pháp dự đoán vị trí của đối tượng chuyển động bằng cách khai phá luật kết hợp của các mẫu hình di chuyển. Thực nghiệm của chúng tôi đã cho thấy phương pháp đề xuất cho kết quả chính xác hơn khi sử dụng kết hợp với phương pháp dự đoán theo hàm chuyển động.

**Từ khóa.** CSDL đối tượng chuyển động, hệ thống tin địa lý, dịch vụ trên cơ sở vị trí địa lý, luật kết hợp, quỹ đạo, mẫu hình di chuyển.

**Abstract.** A location-based service is an information and entertainment service, accessible with mobile devices through the mobile network and utilizing the ability to make use of the geographical position of the mobile device. Location-Based Services report the current location of the user. In order to support context-prediction and proactive devices, location-based services must be able to predict locations. In the world, there are many research issues of predicting the location of the moving objects. Some issues towards studying fuzzy logic, statistical probability or combining different methods. The goal of this paper is to introduce a framework for pattern extraction and modeling, which helps to design model for moving object databases. This paper also proposes a method to predict the location of moving objects using mining association rules of movement patterns. The experiments are given to show that the proposed method is more accurate when used in combination with the motion function prediction method.

**Key words.** MODB, GIS, LBS, association rules, trajectory, movement pattern.

### 1. INTRODUCTION

A location-based service (LBS) is an information and entertainment service, accessible with mobile devices through the mobile network. It makes use of the geographical position of the mobile device. LBS can be used in a variety of contexts, such as health, work, personal life, etc. LBS includes services to identify a location of a person or object, like discovering the

nearest banking cash machine or the whereabouts of a friend or an employee. LBS services include parcel tracking and vehicle tracking services. In the tracking LBS applications, moving objects use e-services that involve location information. The object discloses their positional information (position, speed, velocity, etc.) to the services, which in turn use this and other information to provide specific functionality.

With the advances in mobile communication and positioning technology, large amounts of moving objects data (location data) from various types of devices, such as GPS equipped mobile phones or vehicles with navigational equipment, have been collected. Then LBS applications send this location data to the server for processing. In order to support context-prediction and proactive devices, LBS applications must be able to predict location of moving objects for high quality services, such as traffic flow control or location-aware advertising, etc.

In the real world, there are many research issues of predicting the location of the moving objects. Some issues towards researching statistical probability [1, 2], fuzzy logic [3], motion function [4, 5] or combining different methods [6, 7].

The goal of this paper is to introduce a framework for pattern extraction and modeling, which helps to design model for moving object databases. This paper also proposes a method to predict the location of moving objects using mining association rules of movement patterns. This paper is organized as follows. In Section 2 we present an overview of the association rules and some definitions of trajectory. Movement patterns leading to our proposed method by using mining association rules of movement pattern are given in Section 3. In Section 4 we present the experiment results. The experiments show that the proposed method is more accurate when used in combination with the motion function prediction method.

## 2. BACKGROUND AND RELATED WORK

### 2.1. Motion Function Prediction

In recent years, predictive query processing has been paid a great amount of attention by the spatio-temporal database. For efficient query processing, various access methods have been proposed such as the time-parameterized method and its variations [6, 7], and dual transformation techniques. Despite of the variety of index structures, all of them estimated objects' future locations by motion functions.

The motion functions can be divided into two types:

- (1) Linear models that assume an object following linear movements.
- (2) Non-linear models that consider not only linearity but also non-linear motions. Given an object's location  $l_0$  at time  $t_0$  and its velocity  $v_0$ , the linear models estimate the object's future location at time  $t_q$  by using the formula:

$$l(t_q) = l_0 + v_0 \times (t_q - t_0),$$

where  $l$  and  $v$  are  $d$ -dimensional vectors.

The non-linear models capture the object's movements by more sophisticate mathematical formulas. Thus, their prediction accuracies are higher than those of the linear models.

Motion function prediction methods in moving objects database cannot forecast locations accurately if the query time is far away from the current time. This is because all of these prediction methods are based on motion parameters, which may not be of much assistance for a distant time prediction.

Our approaching method is mining association rules of movement patterns. So we will introduce association rules and movement patterns in the next sub-session.

## 2.2. Association rules

In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using different measures of interestingness. Based on the concept of strong rules, Rakesh Agrawal et al. [9] introduced association rules for discovering regularities between products in large-scale transaction data recorded by point-of-sale (POS) systems in supermarkets.

The association rules problem is as follows:

Let  $I = \{i_1, i_2, \dots, i_n\}$  be a set of literals call items. Let  $D$  be a set of all transactions where each transaction  $T$  is a set of items such that  $T \subseteq I$ . Let  $X, Y$  is a set of items such that  $X, Y \subseteq I$ . An association rule is an implication in the form  $X \Rightarrow Y$ , where  $X \subset I$ ,  $Y \subset I$ ,  $X \cap Y = \emptyset$ .

*Support or prevalence.* The rule  $X \Rightarrow Y$  holds with support  $s$  if  $s\%$  of transactions in  $D$  contain both  $X$  and  $Y(X \cup Y)$ . Rules that have a  $s$  greater than a user-specified support is said to have minimum support.

*Confidence or predictability.* The rule  $X \Rightarrow Y$  holds with confidence  $c$  if  $c\%$  of the transactions in  $D$  that contain  $X$  also contain  $Y$ . Rules that have a  $c$  greater than a user-specified confidence is said to have minimum confidence.

*Expected predictability.* This is the frequency of occurrence of the item  $Y$ . So the difference between expected predictability and predictability (confidence) is a measure of the change in predictive power due to the presence of  $X$ . Usually, the algorithms only provide rules with support and confidence greater than the threshold values established.

The Apriori algorithm starts counting the number of occurrences of each item to determine the large itemsets, whose supports are equal or greater than the minimum support specified by the user. There are algorithms that generate association rules without generating frequent itemsets. Some of them simplifying the rule set by mining a constraint rule set, that is a rule set containing rules with fixed items as consequences.

Generally, an association rules mining algorithm contains the following steps:

- The set of candidate  $k$ -itemsets is generated by 1-extensions of the large  $(k - 1)$ -itemsets generated in the previous iteration.
- Supports for the candidate  $k$ -itemsets are generated by a pass over the database.
- Itemsets that do not have the minimum support are discarded and the remaining itemsets are called large  $k$ -itemsets.

This process is repeated until no more large itemsets are found.

Next, some concepts of trajectory and movement patterns will be provided. Applying the mining association rules on the movement patterns can be used to predict the location of moving objects in the distant time future.

## 2.3. Trajectory and movement patterns

For the near future prediction, some techniques assume the predictive trajectories of an object can be represented by some mathematical formulas based on its recent movements

[4]. However, in real world, the object’s movements are more complicated than what the mathematical formulas can represent. Their movements may be affected by flood or traffic jams for vehicles, turbulence places for aircraft, and so on.

In fact, the object’s movements follow some patterns in many applications. People go to work every weekday along similar routes, public transportation is governed by time schedules and destinations, and animals annually migrate to reproduce or seek warmer climates. These patterns can provide reasonable predictions if they satisfy some query conditions.

In order to extract movement patterns from trajectory data, we have some definitions similar to the one presented in [10].

**Definition 1.** [Stop] A stop  $s$  is a semantically important part of a trajectory which is considered that the object has not effectively moved. A stop is represented by a spatial feature type in the geographic space and a non-empty time interval.

In the definitions, stops are interesting places specified according to the application. For instance, a traffic light may be considered a stop in a transportation management application, but probably not in a tourism application.

**Definition 2.** [Move] A move  $s_1 \rightarrow s_2$  is the part of a trajectory that has a time interval and is delimited by two consecutive stops  $s_1$  and  $s_2$ , where consecutive stops by definition must have non-overlapping time intervals.

**Definition 3.** [Trajectory] Semantically a trajectory  $P$  is an ordered list of stops  $S$  and moves  $M$ .  $P$  is usually expressed as follows

$$P = \{(s_0, s_1, \dots, s_{n-1})\},$$

where  $s_i$  ( $0 = i < n$ ) is  $i^{th}$  object’s stop.

For example, in the Figure 1 below we have an object’s trajectory over the geographic space, where we can visually infer information like the geographic location (Hanoi).

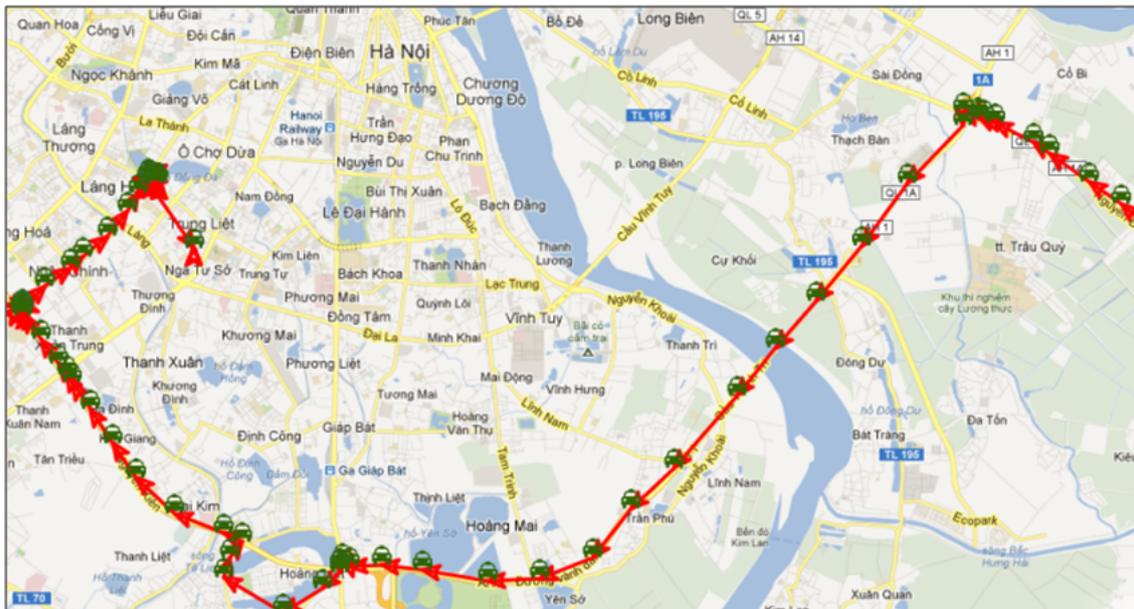


Figure 1. A trajectory with geographical information

**Definition 4.** [Movement's Support] Let  $X = \{P_1, P_2, \dots, P_n\}$  be a set of trajectories, where each  $P_i$  ( $0 < i \leq n$ ) is defined as in Definition 3 above.

Let  $A \rightarrow B$  be a move from a stop  $A$  to a consecutive stop  $B$ . We define the support of the move  $A \rightarrow B$ , denoted by  $\text{sup}(A \rightarrow B)$ , as the fraction of trajectories  $P_i$  of  $X$  in which the move  $A \rightarrow B$  occurs. More formally,

$$\text{sup}(A \rightarrow B) = \frac{|P \in X | A \rightarrow B \in P|}{|X|}$$

where  $|Z|$  is the number of elements in the set  $Z$ .

**Definition 5.** [Movement Pattern] A movement pattern is a move with support  $s$ , where  $s$  is higher than a given threshold, called  $\text{min sup}$ .

A movement pattern  $M : A \xrightarrow{s} B$  is a relationship between two spatial feature types  $A$  and  $B$  which has two main properties: direction and support. The direction of a movement pattern  $A \rightarrow B$  is a path from  $A$  to  $B$ , in this order, and the support  $s$  is the fraction of trajectories having this move.

**Definition 6.** [Trajectory Pattern] A trajectory pattern  $P$  is a special association rule of the form

$$P : A_{t_1} \wedge A_{t_2} \wedge \dots \wedge A_{t_m} \xrightarrow{s} B_{t_n},$$

where each  $A_{t_i}$  and  $B_{t_n}$  are consecutive stops with time constraint

$$t_1 < t_2 < \dots < t_m < t_n.$$

The left side  $A_{t_1} \wedge A_{t_2} \wedge \dots \wedge A_{t_m}$  is called the premise and the right side  $B_{t_n}$  the consequence. Confidence  $c$  means that when the premise occurs, the consequence will also occur with probability  $c$ .

**Definition 7.** [Distant Time Query] Given the object's recent locations, the current time  $t_c$ , and the query time  $t_q$ , a predictive query estimates the object's future position at  $t_q$ . A distant time query is a spatio-temporal predictive query satisfying  $t_q \geq t_c + d$ , where  $t_q$  represents the query time,  $t_c$  represents the current time, and  $d$  is a threshold of a distant time

$$t_q < T, \quad 0 < d < T.$$

The boundary between distant time and non-distant time is application-dependent.

#### *Mining Movement Pattern*

There are some methods for mining movement pattern.

- (1) Raw data transformation: In this approach, raw data is approximated and transformed into an analysis format (i.e., the motion matrix) optimized for pattern discovery.
- (2) Indexing: Kalniset et al. [11] employ a grid index  $G_t$  at each time point  $t$  for storing the data points at that time. The density-based clustering algorithm DBSCAN [8] is then applied on the grid index  $G_t$  in order to identify the clusters at time  $t$ .
- (3) The apriori approach: The Apriori approach has been applied to discover trajectory patterns efficiently. The idea is described in [12, 13] as below.

In Definition 3, an object's trajectory is typically represented as a sequence  $\{(s_0, s_1, \dots, s_{n-1})\}$ , where  $s_i$  ( $0 \leq i < n$ ) is  $i^{\text{th}}$  object's stop. The goal is discovering an object's periodic patterns

from its historical trajectory. Given  $T$ , which is the number of timestamps that a pattern may reappear, an object’s trajectory is decomposed into  $\frac{n}{T}$  sub-trajectories (figure 2).  $T$  is data-dependent and has no definite value. For example,  $T$  can be set to “a day” in vehicle monitor applications since many vehicles have daily patterns, while the behaviors of animals’ annual migration can be discovered by  $T$  as “a year”.

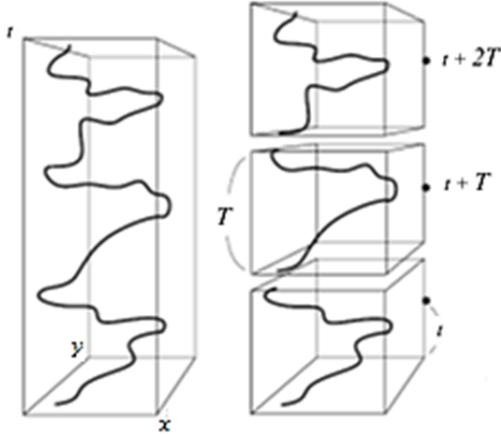


Figure 2. Trajectory decomposition

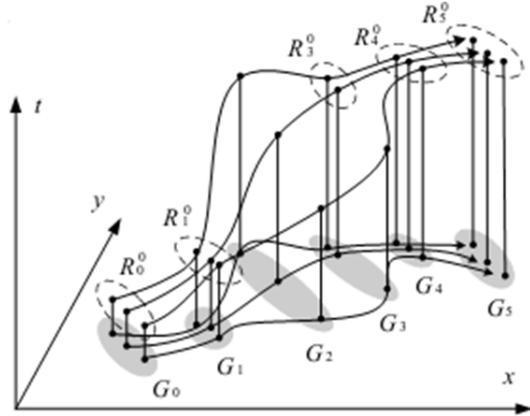


Figure 3. Sub-trajectories

All locations from  $\frac{n}{T}$  sub-trajectories which have the same time offset  $t$  of  $T(0 \leq t < T)$  will be gathered onto one group  $G_t$ .  $G_t$  represents all locations that the object has appeared at time offset  $t$ . A clustering method (we used DBSCAN algorithm in this paper) is applied to find dense clusters  $R_t$  in each  $G_t$ . The Figure 3 shows an example of the above concepts.

$R_t$  symbolizes the region inside of which the object may often appear at time offset  $t$ .  $R_t$  is called as a frequent region at  $t$ .

More than one frequent region at time offset  $t$  can exist. For example, An leaves home at 8h00 and goes to Cau Giay at 8h30 to go to her work every weekday. On most weekends, she passes by a market at 8h30 and goes to the Trung Nguyen Café at 10h00. In this case, there are two frequent regions at 8h30. To distinguish these frequent regions having the same time offset, we use to represent the  $j^{th}$  frequent region at time offset  $t$ .

Consider the example above, which An was at home  $R_0^0$  at time offset 0 and then in Cau Giay  $R_1^0$  at time offset 1. By examining her movement history, we can derive the probability that she would be at her work place  $R_2^0$  at time offset 2 is 0.5. We use an association rule  $R_0^0 \wedge R_1^0 \xrightarrow{0.5} R_2^0$  to represent the above knowledge. On the other hand, if she goes to a market ( $R_1^1$ ) at time offset 1 instead of Cau Giay she will be at Trung Nguyen Café ( $R_2^1$ ) with probability 0.4. This is represented as  $R_0^0 \wedge R_1^1 \xrightarrow{0.4} R_2^1$ . In this paper, we use this method for mining our movement pattern, which is described in the next section.

### 3. MINING ASSOCIATION RULES OF MOVEMENT PATTERNS TO PREDICT THE LOCATION OF MOVING OBJECTS

Considering a moving object continually over time, predictive queries ask for the object’s probable location at some future time. Most existing prediction methods in moving objects database cannot forecast locations accurately if the query time is far away from the current

time. This is because all of these prediction methods are based on motion parameters (location, velocity, acceleration...), which may not be of much assistance for a distant time prediction. For example, even if we know An is at home at 8h00, she is in the supermarket at 8h30 (current time), obviously we cannot identify her location at 11h00 based on these motion parameters. An object's recent movements are only helpful to predict near future locations.

The motion of the moving object is often more sophisticate to be represented by any mathematical function. Also, the motion is often affected by the weather and traffic conditions (rain, flood, traffic jam...). For example, you can see the Figure 4 below. Minh usually goes to work from A to B. Today it is traffic jam near B. Minh went to V at time  $t_2$  to bypass. Motion function method has predicted that Minh was in  $P_1$  at time  $t_2$ . The same at time  $t_5$ , Minh was not in  $P_2$ . He must be in B, on the way to his office.

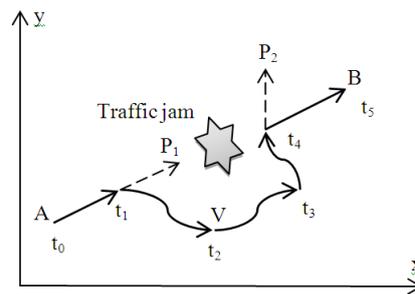


Figure 4. Incorrect prediction

In fact, the object's movements follow some patterns in many applications. These patterns can provide reasonable predictions if they satisfy some query conditions.

The knowledge of movement patterns between different places in the geographic space (e.g. home to office, touristic place A to touristic place B) may help the user to answer queries about moving objects or movement behavior. In order to capture and model such pattern relationships, data mining techniques play an essential role.

Usually, the pattern mining result is a set of mined patterns with their frequencies. The main problem in predicting the location of moving object based on the movement patterns is how to find the patterns over the object's historical movements. Some studies have shown that it could be done by mining patterns. However, in order to obtain the patterns, a large volume of an object's historical movements is required for a data mining process. It also implies that the number of patterns discovered could be very large. Therefore, the method of extracting and organizing these patterns to answer the predictive queries efficiently is important and needs to be addressed. In the next sub-section, we introduce a framework for pattern extraction and propose a method for mining movement patterns.

### 3.1. Pattern extraction framework

The Figure 5 below presents an overview of a pattern extraction framework that can be analyzed in three levels: data, pattern extraction and pattern modeling.

The data level has both trajectory data and GIS data, where stops are within spatial feature types defined in the geographic database.

At the pattern extraction level, the data are cleaned and converted into a standard format in preparation for the data mining. The data are also transformed to the input format required

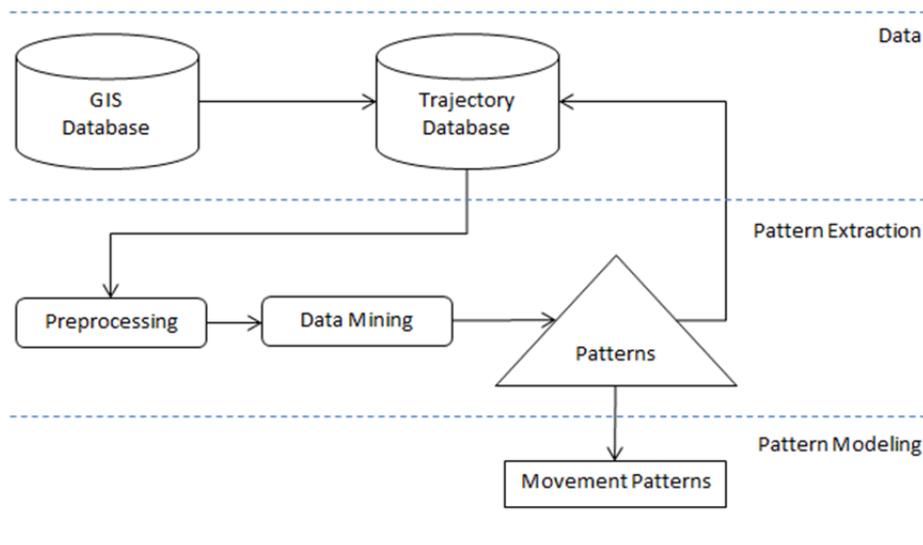


Figure 5. Framework for pattern extraction and modeling

by the data mining algorithm.

In the data mining step the user may define mining parameters such as minimum support, in order to extract only patterns that pass these constraints. Trajectory pattern relationships are identified in this step.

Based on the idea in Section 2, our approaching method is described in sub-section below.

### 3.2. Our approaching method

We use apriori like algorithm to discover trajectory patterns. Discovering the trajectory patterns is similar to mining association rules. Therefore, discovery process is divided into two main steps.

(1) Detect frequent regions. Some candidate trajectory patterns are generated, and their frequencies are counted by scanning the trajectory database.

In this step, the methods decompose the whole trajectory into  $n/T$  sub-trajectories and group all the locations  $G_t$  having the same time offset  $t$  in each sub-trajectory. After that, apply the density-based clustering algorithm DBSCAN [8] to find frequent regions for each time offset  $t$ .  $MinPts$  and  $Eps$  parameters of DBSCAN play the same role as support of mining frequent item sets.

(2) Derive trajectory patterns (corresponding to association rules). In this step, unpromising patterns (with low frequencies) are removed from consideration. The remaining candidates are combined together to form larger candidate patterns. This process is repeated until no further candidates are generated. Different kinds of patterns can be discovered considering stops and moves in trajectories. Such as,

- The most frequent stops during a certain period of time;
- Frequent stops that have a duration higher than a given threshold;
- Frequent moves at a certain time interval;
- Most frequent moves inside a certain region;

- Frequent moves that intersect a given spatial feature type.

In this paper, the objective is to find one specific kind of patterns from trajectory data: frequent moves between two stops. So we use the apriori like algorithm to generate trajectory patterns from the frequent regions discovered. The key idea of the algorithm is to generate candidate item sets of length  $k$  from candidate item sets of length  $k - 1$ . In this process, we remove some candidate item sets which are not necessary to predict as follows:

- Any patterns of predicting past or current positions from future movements are removed. In Definition 6, a trajectory pattern is a monotonically increasing sequence in terms of the time offset associated with each region. So all patterns which contradict this constraint are removed.

- Any patterns that have multiple items in their consequences are removed. According to the apriori algorithm, rules having more than one item in the consequence are created by rules having one item in the consequence.

The Apriori approach is able to reduce the search space significantly in the first few iterations, allowing trajectory patterns to be mined efficiently. The algorithm for trajectories based on original apriori algorithm is described as follows.

### Apriori trajectories algorithm

**Input:** Trajectory database  $D$  with the movement's *support* =  $\min\_sup$

**Output:** Frequent Regions

```

 $R_1 = find\_frequent\_1 - itemsets(D);$ 
for ( $k = 2; R_{k-1} \neq \emptyset; k++$ ) {
 $C_k = apriori\_gen(R_{k-1}, \min\_sup);$ 
  for each  $T_t \in D$  /*  $T_t$  is trajectory  $t^{th}$  */ {
     $C_t = subset(C_k, t);$  /* use  $t$ th sub-trajectory for candidate */
    for each candidate  $c \in C_t$ 
       $c.count++$ ; }
 $R_k = \{c \in C_k | c.count \geq \min\_sup\}$ 
}
return (Frequent Regions =  $\cup_k R_k$ );
Procedure apriori_gen( $R_{k-1} : frequent(k-1) itemsets; \min\_sup$ )
for each itemset  $R_1 \in R_{k-1}$ 
  for each itemset  $R_2 \in R_{k-1}$ 
    if ( $(R_1[1] = R_2[1]) \wedge (R_1[2] = R_2[2]) \wedge \dots \wedge (R_1[k-2] = R_2[k-2]) \wedge (R_1[k-1] <$ 
 $R_2[k-1])$ ) then {
       $c = R_1 \times R_2$  /* joint step: produce  $k$  candidate from  $k-1$  item sets */
      if has_infrequent_subset( $c, R_{k-1}$ ) then
        delete  $c$ ; /* removal if there are multiple entries in the result */
      else add  $c$  to  $C_k$ ; }
return  $C_k$ ;
Procedure has_infrequent_subset( $c : candidate k$ -itemset;  $R_{k-1} : frequent (k-1)$  item-sets)
for each  $(k-1)$ -subset  $s$  of  $c$ 

```

```

if  $s \notin R_{k-1}$  then
  return TRUE
return FARSE

```

## 4. EXPERIMENT

### 4.1. Experiment data

The dataset for our experiment is daily activities of an object with the corresponding time interval in three months (91 days). This object has some stops (frequent regions): Home, Café, Market, Office, and Laboratory. The moves in these trajectories are shown in Table 1, where each row is a move of one trajectory.

The frequent moves between two stops considering the dataset shown in Table 1, are shown in Table 2, with their respective support.

Considering the dataset shown in Table 1 and  $\text{minsup} = 0.5$ , we obtain the moving patterns shown in table 3.

### 4.2. Distant time future queries

We use MonetDB [14] to store objects' movement data. MonetDB/SQL extends the SQL by adding and taking advantage of the self-defined functions. In MonetDB, we can write user-defined functions in PL/SQL or Java to provide functionality that is not available in SQL or SQL built-in functions. MonetDB/GIS provides simple GIS features extension to MonetDB/SQL.

Performing the queries on the computer (CPU 2GHz, 1GB RAM) with Windows 7 platform installed MONET-DB using MonetDB/SQL and MonetDB/GIS.

*WHERE will object X at given time?*

Example: The current time is 8h00 and Object X is at Home. Where will object X at 14h15?

```
LET given_time = time(14, 15) // at 14h15
```

```
SELECT o.obj_loc FROM tblMovingObjects WHERE o.id = 'X' AND time(o) = given_time o.obj_loc = "Office";
```

### 4.3. The experimental results when combined with motion function prediction methods

The dataset for this our experiment is daily activities of an object in 100 days. We used two methods to predict the future location of the object: motion function method and combining of motion function and movement pattern method. The accuracy is shown in the Figure 6 below.

In Figure 7, the average accuracy of prediction by motion function method (Average Schema, EWMA...) is about 82% and is represented by the blue line. Easy to find when combined with movement patterns method, the average accuracy of distant time queries is significantly improved (approximately 88%), represented by the red line.

*Table 1.* Daily activities of an object

Did	Tid	Stop 1	Stop 2	Time interval
1	1	Home	Office	08:12-08:27
1	2	Office	Home	18:03-19:05
2	1	Home	Café	07:50-08:12
2	2	Café	Laboratory	08:35-09:23
2	3	Laboratory	Office	11:04-11:15
2	4	Office	Home	12:03-12:15
2	5	Home	Office	14:08-15:07
2	5	Office	Home	18:58-20:03
3	1	Home	Laboratory	08:10-08:20
3	2	Laboratory	Café	08:35-09:20
3	3	Café	Office	09:35-09:40
3	4	Office	Home	10:40-18:15
4	1	Home	Laboratory	08:06-08:20
4	2	Laboratory	Home	17:03-18:20
5	1	Home	Café	07:40-08:03
5	2	Café	Laboratory	09:25-10:18
5	3	Laboratory	Home	18:27-19:05
6	1	Home	Café	08:05-08:15
6	2	Café	Market	09:38-10:09
6	3	Market	Home	11:05-11:37
7	1	Home	Café	08:10-08:19
7	2	Café	Home	10:55-11:25
8	1	Home	Office	11:07-11:20
8	3	Office	Home	17:06-18:12
9	1	Home	Café	07:55-08:33
9	2	Café	Laboratory	08:40-09:35
9	3	Laboratory	Home	18:25-19:38
10	1	Home	Laboratory	08:32-09:02
10	2	Laboratory	Office	11:03-16:27
10	3	Office	Home	17:28-18:15
...	...	...	...	...

## 5. CONCLUSION

In this paper, we have introduced a framework for pattern extraction and modeling, which helps to design model for moving object databases. We have also proposed a method to predict the location of moving objects using movement patterns. Movement patterns can be obtained by mining association rules. Our experiments have demonstrated that the proposed method is more accurate when used in combination with the motion function prediction method. This method is also good support for the direction prediction of the object at the moment after having extraordinary event as traffic jam, accident or passing the intersections.

Table 2. Move and respective support

Move	Sup
Home → Office	0.5
Office → Home	0.6
Home → Café	0.3
Café → Laboratory	0.4
Laboratory → Office	0.5
...	...

Table 3. Moving pattern with minimum support 0.5

Move	Sup
Home → Office	0.5
Office → Home	0.6
Laboratory → Office	0.5
...	...

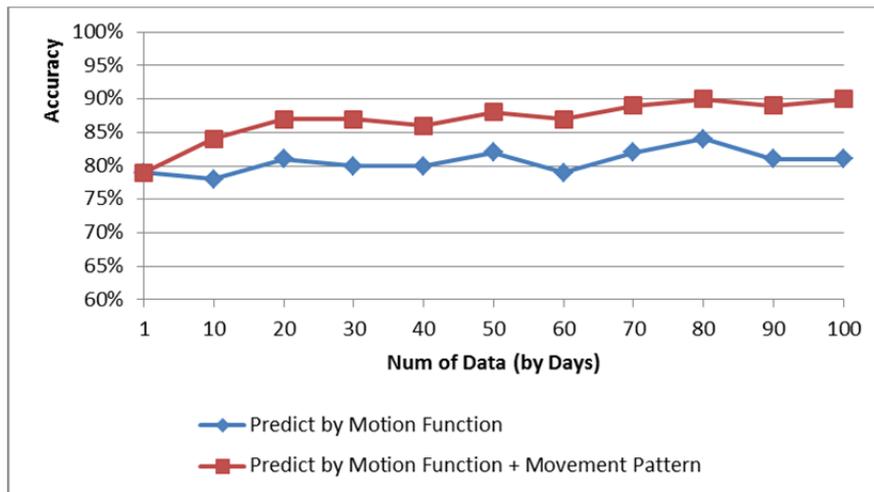


Figure 6. Compare the accuracy of predict by motion function method and combining of motion function and movement pattern method

Future works include the investigation the possibility of using this approach in real-time context. The methods for predicting future times also need to be studied, for example, the users want to know when the object will arrive at a given location.

### REFERENCES

[1] Craig Schlenoff, Raj Madhavan, Stephen Balakirsky, An approach to predicting the location of moving objects during on-road navigation, *18th International Joint Conference on Artificial Intelligence*, Mexico, August, 2003.

- [2] R. Madhavan, and C. Schlenoff, Moving object prediction and tracking for off-road autonomous navigation, *Proceedings of the SPIE Aerosense 2003 Conference*, Orlando, FL, 2003 (doi:10.1117/12.485771).
- [3] Xiaoguang Hong, Yan Yuan, Xinglei Hu, Shandong Univ., Jinan, Prediction of moving objects'  $k$ -nearest neighbor based on fuzzy-rough sets theory, *Fuzzy Systems and Knowledge Discovery*, IEEE Computer Society, 2007 (407–411).
- [4] J. M. Patel, Y. Chen, and V. P. Chakka, Stripes: an efficient index for predicted trajectories, *SIGMOD*, ACM New York, NY, USA, 2004 (635–646).
- [5] Y. Tao, C. Faloutsos, D. Papadias, and B. Liu, Prediction and indexing of moving objects with unknown motion patterns, *SIGMOD*, ACM New York, NY, USA, 2004 (611–622).
- [6] Dong-Oh Kim, Kang-Jun Lee, et.al., An efficient indexing technique for location prediction of moving objects, *Proc. of the 11th int'l conf., KES 2007 and XVII Italian Workshop on Neural Networks Conf. on Knowledge-Based Intelligent Information and Engineering Systems, Lecture Notes in Computer Science* **4693** (2007) 1–9.
- [7] Mikolaj Morzy, Mining frequent trajectories of moving objects for location prediction, *Proceeding of MLDM '07 Proc. of the 5th Int. Conf. on Machine Learning and Data Mining in Pattern Recognition, Lecture Notes in Computer Science* **4571** (2007) 667–680.
- [8] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, *Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD-96)*, AAAI Press., 1996 (226–231) (ISBN 1-57735-004-9).
- [9] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, *ACM Sigmod Int. Conf. on Management of Data*, ACM New York, NY, USA, 1993 (207–216).
- [10] Spaccapietra, et al., A conceptual view on trajectories, *Data & Knowledge Engineering* **65** (1) (April 2008) 126–146.
- [11] P. Kalnis, N. Mamoulis, S. Bakiras, On discovering moving clusters in spatio-temporal data, *Proc. of the Intl. Symposium on Spatial and Temporal Databases, Lecture Notes in Computer Science* **3633** (2005) 364–381.
- [12] Y. Wang, E.P. Lim, S.Y. Hwang, Efficient mining of group patterns from user movement data, *Data & Knowledge Engineering* **57** (3) (June 2006) 240–282.
- [13] Hoyoung Jeung et al., A hybrid prediction model for moving objects, data engineering, *IEEE 24th International Conference*, Cancun, Mexico, 2008.
- [14] MonetDB Introduction, <http://www.monetdb.org/Documentation/UserGuide>.

*Received on June 12, 2013*

*Revised on August 08, 2013*