

Efficient Algorithms for Pattern Mining in Spatiotemporal Data

Nagasaranya N

Asst. Prof., Dept. of MCA

Karpagam College of Engineering, Coimbatore - 641032

ABSTRACT

Spatio-temporal data is any information relating to space and time. It is continually updated data with 1TB/hr are greatly challenging our ability to digest the data. With that data, it is unable to gain exact information. Data mining models contains many statistical models such as regression models of various kinds, cluster analysis models, covariance analysis models, principle component analysis models, outlier detection models(temporal, spatial, non-spatial), trend detection models, partial least squares models(prediction) and multiple variant visualization models. Most of these models find applications in spatial data mining and pattern discovery.

Keywords

Spatio-Temporal Data, Clustering, Association Rule, Pattern Discovery

1. INTRODUCTION

Data mining is the analysis of observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner. Extracting interesting and useful patterns from spatio - temporal database is more difficult than extracting the corresponding patterns from traditional (fixed) numeric and categorical (clear) data due to the complexity of spatio - temporal data types, spatio - temporal relationships, and spatio - temporal autocorrelation. Spatio-temporal data is any information relating space and time.

Geospatial data is regularly updated data with 1TB/hr, are greatly challenging our ability to digest the data. With that data, it is unable to gain exact information. Here the data may lose. Geospatial data take a view of geospatial phenomena, here it captures the spatiality. If it evolves over time, it captures the spatiality as well as the temporality. From that we are able to know the process and events of spatiotemporal data. Knowledge of extracting spatio-temporal data gives better prediction of its process and events.

Finding the trend is most important task in spatiotemporal data. Particularly it is to analyze trajectories movement, animal movements, mating behavior, harvesting, soil quality changes, earthquake histories, volcanic activities and prediction etc. Spatial, temporal and/or spatio-temporal data mining looks for patterns in data using the spatiotemporal attributes. Trend discovery cannot be directly performed well. The proposed research performs trend discovery by combining the principles of clustering, association rule mining, generalization and characterization.

The significance of spatio-temporal data is continuously updated data with 1 TB/hr. It is terribly massive and huge dimensional geographic and spatio-temporal datasets. We ask that authors follow some simple guidelines. In essence, we ask you to make your paper look exactly like this document. The easiest way to do this is simply to download the template, and replace the content with your own material.

Meteorology

- Biology
- Crop sciences
- Forestry
- Medicine
- Geophysics
- Ecology
- Transportation, etc

And we can apply this proposed algorithm to any kind of data mining applications. Especially when we apply to spatiotemporal databases, it gives most significant results. When trend discovery algorithm is used, the rate of data retrieval is fast and results obtained were accurate.

The objective of this research is to develop a novel algorithm for trend discovery on multi dimensional Spatiotemporal databases. Objectives are,

Predict the knowledge

- Retrieve hidden information
- Discover Trend
- Future Usage

2. LITERATURE REVIEW

Number of papers has been reviewed for this study and that are categories into three major phases such as, Clustering in Spatio-temporal Data

- Outlier detection in Spatio-temporal Data
- Feature, Location and Pattern finding from Spatiotemporal Data

Based on study [1], several drawbacks have been found out in each and every phase. And we also compared the performance analysis in terms of accuracy, sensitivity, specificity, time and memory space.

2.1 Drawbacks of Existing Study

Clustering of spatio - temporal data is a difficult Problem that is compared to various fields and applications.

- The Major and common drawbacks are high dimensionality of data (2D or 3D), initial error propagation, high dimensional data complexity.
- Clustering of space and time related data, spatiotemporal clustering methods focus on the specific characteristics of distributions in 2-D or 3-D space, while general-purpose high-dimensional clustering method have limited power in recognizing spatiotemporal patterns that involve neighbors.
- In human computer interaction, the clustering techniques may not work properly to find out the complex patterns in huge volume of spatio-temporal data sets.

Outlier Detection in Spatio-temporal Data - The neighbors which occurs in spatio-temporal outliers are very conflict, even the non spatio-temporal values are normal for the rest of the objects of the same class.

- Spatio-temporal outlier detection may not properly work with non-spatial or non-temporal datasets. Even if the non-spatial attributes are separated, it unable to predict the outliers from spatio-temporal datasets.

Rule mining in spatio-temporal data - Drawbacks of existing Dependency Analysis algorithms are, while incorporating local search in high-dimensional spatio-temporal data, the performance is very low and it faces the diversity problem. In the optimization process, the existing algorithms are not able to work with partial optimism.

- In that stage, the algorithm will easily affect the speed and direction of the particle and it may not work properly to do their remaining iterations.
- And the existing algorithms cannot able to solve the issues of huge dimensional, unsystematic datasets in the stage of particle search and moving object search.

Feature, Location and Pattern finding from Spatio – temporal Data - existing algorithms drawbacks, while discovering the trend in spatio-temporal data are as follow,

- The anomalies will occur in the existing system, while it concentrates the graph alteration. Here the information may loss badly. So the necessities of the table scan may increase more.
- In generalization process face problem to identify the description of data while doing the path relation and their hierarchy.
- Spatio-temporal dependency process faces several problems while finding the dependency of large spatio-temporal data.

To overcome these drawbacks, we propose new algorithms for discovering the best trend in spatio-temporal data sets.

2.2 Problem Definition

The performance of trend discovery analysis is hindered in spatio-temporal data due to the following reason:

- Clustering of massive spatio-temporal data is cumbersome. Features of the data are often preselected, yet the properties of different features and feature combinations are not well investigated in the huge spatio-temporal data sets. Finding appropriate features to form a cluster group is essential for better search.
- Cluster accuracy is less, so it requires deviation and outlier detection analysis for high dimensional spatio-temporal clustered data.
- Rule generation accuracy of spatio-temporal data is less, while it predicts the value of one attribute with the help of another attribute value over a period of time. So it requires optimization process for rule generation.
- To know about the compact description of data, it needs better generalization and characterization method.

Trend discovery of spatio-temporal data accuracy is very less, while it's going for the huge dimensional data sets.

3. DATABASE DESCRIPTION

For our implementation work we have taken the spatiotemporal moving object trajectory high dimensional data sets. Trucks dataset consists of 276 trajectories of 50 trucks delivering concrete to several construction places around the Athens metropolitan area in Greece for 33 distinct days [8].

Datasets consist of 1,12,203 data. The structure of each record is as follows:

obj-id, traj-id, date(dd/mm/yyyy), time(hh:mm:ss), lat, lon, x, y

where (lat, lon) is in WGS84 reference system and (x, y) is in GGRS87 reference system.

3.1 Finding the Latitude and Longitude Points

The LATLON system uses angular measurements. Earth's latitude various between -90° south to $+90^\circ$ north in the units of 1 degree, and is different for other planets, moons. When higher precision is needed, we express in it in terms of minutes, seconds, and decimal seconds.

The formula to convert angles in degrees to radians is,

$$\text{radians} = \frac{\pi}{180} + \left[\text{degree} + \frac{\text{minutes}}{60} + \frac{\text{seconds}}{3600} \right]$$

Longitudes on the other hand have a range of -180 East to $+180$ West. Here we always represent the latitude and longitude as pairs. Each part being suffixed by East, West, North and South. If the radians are 0 to ± 180 we omit East to West longitude.

Example,

LATLON coordinates of the points are (10.81 E, 79.17 N). Express (i) Both the decimal parts as minutes and seconds, (ii) Only the seconds as radians.

Solution:

We first convert the longitude multiply 0.81 with 60 to get 48.6, which gives 48 minutes. Next multiply 0.6 with 60 to get 36 as the seconds. Hence the longitude becomes $10^\circ 8' 36''$.

To convert the latitude we multiply 0.17 with 60 to get 10.2, which give 10 minutes. Next multiply 0.2 with 60 to get 12 as the seconds. Hence the latitude becomes $79^\circ 10' 12''$.

3.2 Pseudo Code

Step 1: Get the latitude and longitude position

Step 2: First convert the longitude position into minutes, seconds and decimal seconds.

Step 3: To convert the value into minutes and seconds we go for the multiplication process with 60.

Step 4: Multiply the first value into 60 to get the minute and with that reminder of minutes we get the seconds by multiplying 60.

Step 5: Do the same process for latitude too.

Step 6: Finally we get the latitude and longitude in terms of minutes and seconds.

Step 7: To repeat the step until all data have been converted into minutes and seconds.

Step 8: By using that value we easily find the exact trajectory of one particular object with a help of our proposed algorithm.

4. CONTRIBUTION OF THIS RESEARCH

Phase 1 gives the solutions to the challenges in segmenting the spatio - temporal data by Proposed PANN algorithm

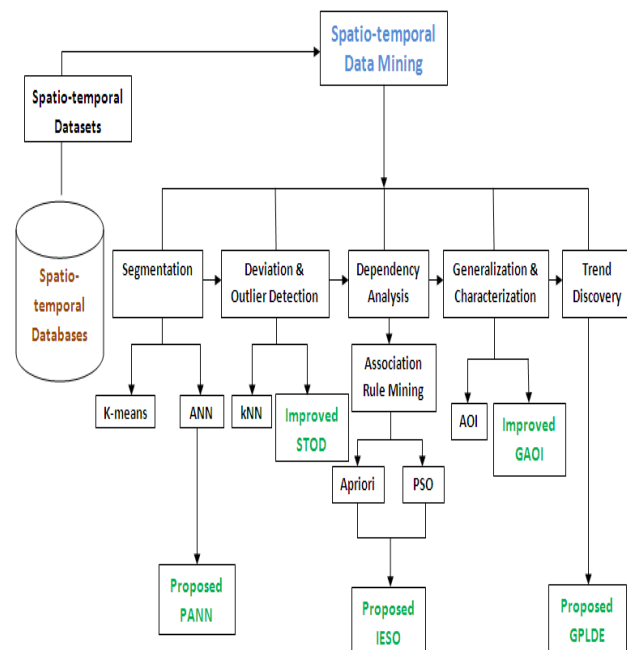
Phase 2 removes the deviations and outliers of spatio - temporal data accurately by improved STOD Technique

Phase 3 generates the rules for dependency analysis of spatio - temporal data by proposed IESO technique

Phase 4generalizes and characterizes the spatio – temporal data by improved GAOI technique

Phase 5 Finally the Proposed GPLDE algorithm is developed for the further trend discovery of spatio - temporal data.

Fig 1 shows that the entire framework of proposed research.



ANN – Artificial Neural Network PANN – Principle Artificial Neural Network OD – Outlier Detection
STOD - Spatio-temporal Outlier Detection PSO – Particle Swarm Optimization IESO – Intelligent
Swarm Ant Optimization AOI – Attribute Oriented Induction GAOI – Generalized Attribute Oriented
Induction GPLDE – Gaussian Pairwise Linear Difference Estimator

Fig 1: Entire Framework of Proposed Research

5. MATERIALS AND METHODS

5.1 Clustering in Spatio-temporal Data

Clustering of spatio-temporal data are the very difficult problem when compared to traditional and categorical data sets because of the high dimensionality, more complexity, initial error propagation, etc. So we propose PANN clustering techniques over spatio-temporal data.

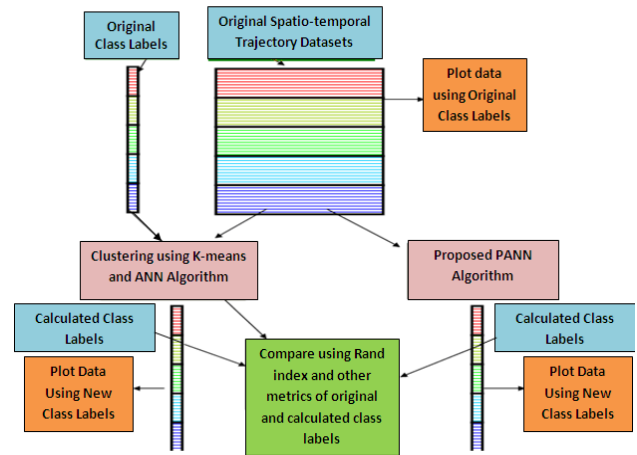


Fig 2: Work flow of Proposed PANN Algorithm

Fig 2, shows that the overall work flows of the proposed

PANN algorithm. Here we are applying several clustering algorithm like K-means, ANN, respectively in spatio-temporal trajectory data sets.

Steps to follow Proposed PANN Algorithm

Step1: Prepare $D_n \in C_c$ (Spatio-temporal moving object trajectory data)

Step2: Reduce the Dimension of input Data using Principal Component Analysis. This will give a reduced d dimensional data.

Step3: Create a neural network with D number of inputs and C number of outputs. Step4: Train the network with N Number of records.

Step4: Test the Network with the Training data or Testing Data and Find the new Class label L2

Step5: Compare the original Class labels L1 and calculated class label L2 with Rand Index and Other Metrics and evaluate the accuracy of Clusters.

Step6: Repeat the above steps for Different values of N for Studying the performance with different number of records.

Our proposed framework has also worked with k-means and k-means based ANN clustering. Pseudo code for proposed PANN algorithm is shown in fig 3.

```

1. Prepare  $D_n \in C_c$ 
2. Define N as number of records;
   For each N and the resulting combined (union) spatial dataset ;
   Do PCA of spatial data to reduce dimensionality, and compute the  $score_{pca}(N)$ ;
   End
   Pick N which produces maximal  $score_{pca}(N)$ .
3. ActivateLayer(input, output)
   foreach d inputneuron
       calculate  $output_d$ 
   foreach j hiddenneuron
       calculate  $output_j$ 
   foreach c hiddenneuron
       calculate  $output_c$ 
   output = (output_c)
4. Output Error
   Error() {
       foreach input in InputSet
           Errorinput =  $\sum_c output_{neuron} (target_c - output_c)^2$ 
       return Average(Errorinput, InputSet)
   }
5. Train the NetworkN
6. find Class Label L2
7. compare (L1, L2)
8. repeat N
    
```

Fig 3: Pseudo Code for Proposed PANN (Principle Artificial Neural Network) Algorithm

5.2 Deviation and Outlier Detection

A spatial outlier is a spatially referenced object whose non-spatial attribute values are significantly different from the values of its neighborhood.

```

1. Given a spatial data set  $X = \{x_1, x_2, \dots, x_n\}$ , an attribute function  $f$ , a number  $k$  of nearest neighbors, and an expected number  $m$  of spatial outliers. For each spatial point  $x_i$ , compute the  $k$  nearest neighbor
set  $NN_k(x_i)$ , the neighborhood function

$$g(x) = \frac{1}{k} \sum_{x \in NN_k(x_i)} f(x)$$
 and
the comparison function

$$h_i = h(x_i) = \frac{f(x_i)}{g(x_i)}$$

2. Let  $h_q$  or  $h^{-1} q$  denote the maximum of  $h_1, h_2, \dots, h_n, h^{-1}, h_2^{-1}, \dots, h_n^{-1}$ . For a given threshold  $\theta$ , if  $h_q$  or  $h_q^{-1} > \theta$  treat  $x_q$  as an S-outlier
3. Update  $f(x_q)$  to be  $g(x_q)$ . For each spatial point  $x_i$  whose  $NN_k(x_i)$  contains  $x_q$ , update  $g(x_i)$  and  $h_i$ .
4. Repeat steps 2 and 3 until either the threshold condition is not met or the total number of S-outliers equals  $m$ .
    
```

Fig 4: Pseudo Code for improved STOD (Spatio-temporal Outlier Detection) Algorithm

Identification of spatial outliers leads to the discovery of unexpected, interesting, and useful spatial patterns. Major drawback of existing methods is that, the normal objects tend to be falsely detected as spatial outliers when their neighborhood contains true spatial outliers. In this paper, we propose a suite of spatial outlier detection algorithms to overcome the disadvantage. We formulate the spatial outlier detection problem in a general way and design algorithms to detect accurate spatial outliers. After the process of clustering, outlier detection is made.

Because clustered data must have some outliers. Detecting and removing the outliers from the data is said to be an outlier detection analysis. Fig 3 shows, the Pseudo code of Spatiotemporal Outlier Detection (STOD) Algorithm.

5.3 Dependency Analysis

PSO is first designed for simulating social behavior by Kennedy, Eberhart and Shi [3,4,5]. It represents the movements of organisms in a bird flock or fish school.

The algorithm was designed to perform the optimization. A survey of PSO application is described by Poli [6] [7]. The overall framework of the proposed IESO algorithm is shown in Fig 5. The proposed IESO algorithm gives efficient and promising results to do rule generation in Spatio-temporal data sets.

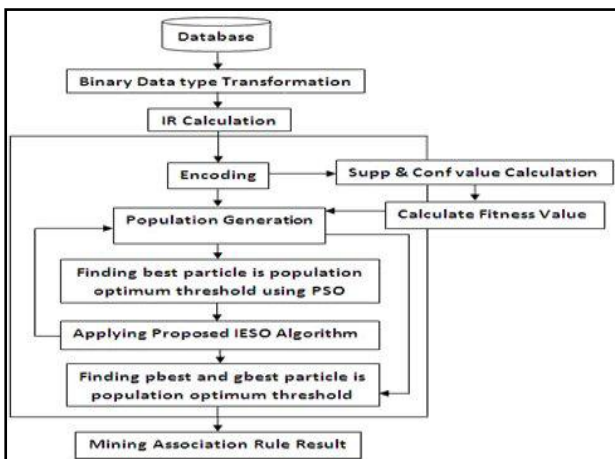


Fig 5: Work flow of proposed IESO algorithm

```

for each particle j = 1, ..., s do:
    initialize
    base attractiveness = τ, and visibility, η
    τ = tabu list, nm = next state to move
    a = each ant completed a solution
    parameters = ω, ρ, φg
    xi = particle position (i=1, ..., n)
    u = uniformly random vector with boundaries,
    where bu, bv are upper and lower boundaries
    of the search space
    pi = particle's best position (i=1, ..., n)
    r = random number, where rp, rg is global and
    local best solution
    find the global best solution
    if (f(pi) < f(g)) then
        update xi = pi
        update vi
        vi = u(-|bu-bv|, |bu-bv|)
        until xi met g
    repeat
    for each particle j = 1, ..., s do:
        for each dimension d = 1, ..., n do:
            rp, rg = u(0,1)
            update vi,d
            vi,d ← ω vi,d + φp rp (pi,d - xi,d) + φg rg (gd - xi,d)
            update xi
            xi ← xi + vi
            if (f(xi) < f(pi)) do:
                update pi = xi
            if (f(pi) < f(g))
                update g = pi
            best solution g
        find pheromone search
        for j < Iteration.Max do:
            for each ant do:
                choose probabilistically nm into
            for each ant τ = nm
                repeat until a
            end
            for each a do:
                update τ = a
            end
            if (pi < g)
                save g = pi
            best solution g
        end
    end
    end
    
```

Fig 6: Pseudo code of Proposed IESO (Intelligent Evolutionary Swarm Ant Optimization) Algorithm

The pseudo code for the proposed IESO algorithm is presented in fig 6.

Steps to follow IESO Algorithm:

- Step 1. Input data are taken from the clustered data output [1]
- Step 2. Transforming the data into binary form and calculate the IR
- Step 3. Applying the rule generation using IESO algorithm
- Step 4. Finding Support and confidence value
- Step 5. Finding the best fitness obtained by the particle in the iteration using IESO algorithm
- Step 6. Finally we get Mining Association rule result

5.4 Trend Discovery

The proposed GPLDE algorithm to solve the anomalies in the rules-based AOI uses Generalization Dependency Graph (GDG) in spatio-temporal rule generated data sets. This work introduces a rule-based Attribute-Oriented (AO) Induction method of rule-based concept hierarchies that can be constructed from generalization rules. Based on analyzing some major previous approaches such as rule-based AO induction with backtracking, path-id based AO induction and acyclic graph based AO induction, we propose a GAOI approach to facilitate induction on the rule based case that can avoid a problem of anomaly and overcome disadvantages of existing methods.

Our approach is different from the previous ones as follows:

we solve anomalies to tackle the information loss by altering on the graph, so we can avoid the requirement of table scans in order to backtrack the tuples in the prime relation to the initial tuples. In the generalization process, we also use generalization rule sets instead of path relations or hierarchies. Each rule set corresponds to one hierarchy for each attribute.

Experimental studies show that the proposed GPLDE algorithm is efficient and suitable for providing condensed and qualified information.

Workflow of Proposed GPLDE Algorithm for Trend Discovery

Fig 7 shows the workflow of the proposed GPLDE algorithm.

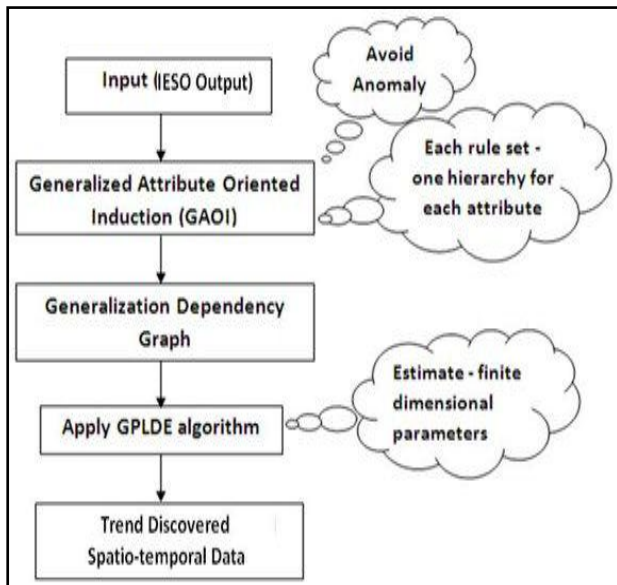


Fig 7: Work Flow of Proposed GPLDE Algorithm

After the rule generated by the proposed IESO to find the trend discovery the proposed GAOI and GPLDE algorithms were applied. In the first step, GAOI is used to avoid the anomaly and each rule set is related to one hierarchy for each attribute. So it is well knowledgeable approach to do attribute value detection process. Then the generalized dependency graph was drawn from these values. Here GPLDE estimates the finite dimension parameter for the datasets. The trend discovery results of the proposed approaches were more promising and significant when compared with other existing algorithms.

Steps to follow GPLDE Algorithm:

```

Initialize
Dataset D, threshold  $\mu$ , ontology O
Preprocess (D,  $\mu$ , O)
discover_trends (db: set_of_objects; sel:
predicate; min_length, max_length: int;
minconf: real; att: attribute;
pred: neighborhood_relation;
focus:= select (db, sel);
graph:= get_ngraph (db, pred);

for each categorical attribute A:
If |distancevalues (A) | >  $\mu$  |D|
  Generalize (A, O, D)
  Compute and store frequency of values in A
else
  for each numerical attribute A:
  cluster the values of attribute A
  compute and store the frequency of values
of each cluster
do trend_analysis
for each object in focus do
  all_paths:= create_npaths ((object), graph,
similar_direction, min_length);
  local_trends:= empty_trend; trend_list:=
empty_list; correlation:= maxreal;
  slope:= maxreal;
  new_paths:= extend (all_paths, graph,
similar_direction, 1);
  current_length:= 2; current_trend:= no_trend; end
  while current_length < max_length and
  new_paths  $\neq$  empty and correlation > minconf
  do
    all_paths:= union (new_paths, all_paths);
    for each path in all_paths do
      last_object:= get_object (path, length
(path));
      attr_change:= get_value (object, attr);
      get_value (last_object, attr);
      distance:= dist (object, last_object);
      insert [attr_change, distance] into list
local_trends;
    end for each path in all_paths;
    perform_linear_regression (local_trends,
slope, correlation);
    new_paths:= extend (all_paths, graph,
similar_direction, 1);
    current_length:= current_length + 1;
    if correlation > minconf then
      current_trend:= [object, slope,
correlation];
    end if
  end while
  if current_trend  $\neq$  no_trend then
  insert current_trend into trend_list;
  end if
  end for
  return trend_list;
end
  
```

Fig 8: GPLDE pseudo code

The proposed Gaussian Pairwise Linear Difference Estimator (GPLDE) algorithm has several stages as

Step 1. Input data is taken from phase3 output

Step 2. The GAOI is to facilitate induction on the rule based system that avoid the problem of anomaly

Step 3. GAOI – each rule set corresponds to one hierarchy for each attribute

Step 4. GPLDE is attractive – estimating the finite-dimensional parameters

Step 5. Finally we discover the trend of spatial data

The pseudo code of proposed GPLDE algorithm is shown in fig 8.

6. CONCLUSION

Spatio – temporal trajectory data has been used in this research of which 80% is treated as training data and 20% is considered as testing data. Proposed PANN method is an influential tool for the clustering and improved STOD for outlier detection of multidimensional spatio – temporal data sets. And the improved GAOI and proposed GPLDE algorithms gives better performance in trend discovery of spatio-temporal trajectory data. Results and discussion of these proposed algorithms will be discussed in the next research paper.

7. REFERENCE

- [1] R. V. Nehme and E. A. Rundensteiner, “SCUBA: Scalable Cluster- Based Algorithm for Evaluating Continuous Spatio-temporal Queries on Moving Objects,” EDBT, 2006, pp.1001-1019.
- [2] Nanopoulos, Y. Theodoridis and Y. Manolopoulos, “C2P: Clustering based on closest pairs,” VLDB, 2001, pp.331-340.
- [3] J. Kennedy and R. Eberhart, “Particle Swarm Optimization,” Proceedings of IEEE International Conference on Neural Networks.IV, 1995.
- [4] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” Proceedings of IEEE International Conference on Evolutionary Computation, 1998, pp.69–73.
- [5] J. Kennedy, “The particle swarm: social adaptation of knowledge,” Proceedings of IEEE International Conference on Evolutionary Computation, 1997, pp.303–308.
- [6] R. Poli, “An analysis of publications on particle swarm optimization applications,” Technical Report CSM-469 (Department of Computer Science, University of Essex, UK), 2007.
- [7] R. Poli, “Analysis of the publications on the applications of particle swarm optimization,” Journal of Artificial Evolution and Applications, 2008, pp.1–10.
- [8] http://www.rtreportal.org/index.php?option=com_content&task=view&id=30&Itemid=43.