World Applied Sciences Journal 6 (10): 1377-1383, 2009
ISSN 1818-4952

An Algebraic Approach to Extend Spatial Operations to Moving Objects

[1,2]*Farid Karimipour,* [1]*Mahmoud R. Delavar and* [2]*Andrew U. Frank*

[1]Department of Surveying and Geomatics Engineering,
College of Engineering, University of Tehran, Tehran, Iran
[2]Institute for Geoinformation and Cartography, Vienna University of Technology,
Gusshausstr. 27-29, A-1040 Vienna, Austria

Abstract: Early Geospatial Information Systems (GIS) dealt with static objects. There is much demand, however, to include temporal objects in these systems. Many have studied this problem and suggested technical solutions for different spatial operations. A common shortcoming is that the extension techniques are highly dependent on the specific case studies and cannot be generalized. In this paper, we propose studying spatial operations via their dimension-independent properties. This research intends to construct a mathematical framework that contains primitives for different operations. The framework will be independent of the space in which the operations are applied using algebraic structures-and more specifically category theory-that ignore those properties of operations which depend on the objects they are applied to. Implementations for some case studies are presented.

Key words: Spatial operations . temporal GIS . algebraic structures . category theory and functor . functional programming languages

INTRODUCTION

Geospatial Information Systems (GIS) manage spatial data and provide the users with the required information for spatial decision making [1]. Early GISs dealt with static objects. To deal with processes in the real world phenomena, however, moving objects must be supported as well. This extension has a wide range of requirements, from data storage and data structure considerations to visualization strategies [2, 3, 4, 5].

Extension of spatial operations to moving objects has been the subject of many studies, each has developed a technical solution to extend a spatial operation with least increase in complexity and speed [6, 7, 8, 9]. Although there are some successful results for this aim, a common shortcoming is that the extension techniques are dependent on the specific case studied. It has resulted in developments which cannot be generalized. The main reason is that the extension techniques are highly dependent on the specific case studies, resulted in developments which cannot be generalized [10, 11]. For example, while there are some solutions for Delaunay Triangulation of moving points [12, 13, 14], they are not directly usable for convex hull computation of moving points and so some other solutions were required [15, 16]. Following such approach, it is not likely to achieve multi-dimensional counterparts of all of the already implemented 2D spatial operations in this haphazard way in the near future.

In this paper, we propose studying spatial operations via their space-invariant properties. We build a foundation based on the dimension-independent properties of spatial operations, which can be extended to other multi-dimensional spaces by using the transformation between domains. The result is a generalized method to extend spatial operations.

In Section 2, we discuss the motivation of the research in more details. Section 3 describes the underlying mathematical concepts of the research. Section 4 explains the research methodology to extend static spatial operations to moving objects. In Section 5, steps of extending spatial operations for moving points based on the proposed approach are shown to extend a simple spatial operation, i.e., Euclidean distance between two points, to moving points. The implementation results for some sample spatial operations are presented in Section 6. Finally, Section 7 contains conclusions and remarks for further steps of the research.

Corresponding Author: Mr. Farid Karimipour, Department of Surveying and Geomatics Engineering, College of Engineering, University of Tehran, Tehran, Iran

SPATIAL OPERATIONS FOR MULTI-DIMENSIONAL OBJECTS

Intuitively, spatial operations represent different aspects from the same real world operation. Therefore, spatial operations have space-invariant properties based on which they can be describe. However, Frank (1999) believes that it has not yet been done for lack of efficient methods:

"A fundamental scientific question today is how to construct complex systems from simple parts. Science is very good at analyzing individual pieces of the puzzle. The combination of these pieces to form a whole is left as "a simple exercise for the reader"-and everybody knows from experience, that these simple exercises are not easy at all… The lack of efficient methods to deal with the combination problem is likely the main reason" [17].

The main deficiency of current researches is that they differentiate the same spatial operations in different spaces despite their unification in the real world. People do not think about the types of the values when doing an operation; they do the same for adding things, independent of what is added: sheep, matches, Roman or Arabic numbers [18]. This research claims that spatial operations represent different aspects from the same real world operation. Thus they have space-invariant properties with which the spatial operations can be described. However, such properties have not been considered in the current researches, because of the lack of efficient methods [17].

To prove this claim, we need a more abstract view that ignores those properties of operations which depend on the object they are applied to. It enables us to have abstract description of operations with known mappings to different multi-dimensional spaces so that they can be extended and combined to support a variety of multi-dimensional spaces. Frank (1999) introduces functional abstraction as a solution for such formal models:

"… A function square(x) can be used in various contexts with different values for x. The same concept can be applied at a higher level of abstraction. Algebras consist of several functions that can be named and have parameters. The parameters do not stand for concrete values as in procedures, but-a step more abstract-for types. They can be combined and the type parameters duly replaced by the actual parameters, much the same way as in the application of functions" [17].

The required abstraction is the subject of algebra which describes an abstract class of objects and their behaviors [19, 20]. Structure of operations in an algebra is independent of an implementation. Thus, behavior of many things can be described with the same algebra as long as their behavior is structurally equivalent.

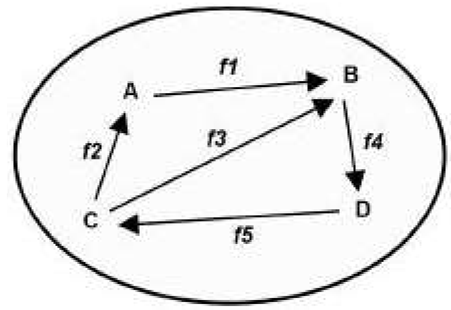


Fig. 1: A category with its objects and morphisms

Algebraic structures have different levels of abstraction: *set* and *group* are examples of algebraic structures. Somewhere at the top of the abstraction ladder, we reach *category* algebraic structure [20]. "Category theory gives a very high level abstract viewpoint: instead of discussing the properties of individual objects we directly address the properties of the operations" [19]. In this research we use categories and their transformations to extend static spatial operations to moving objects.

MATHEMATICAL BACKGROUND OF THE PROPOSED APPROACH

This section defines a "category" and their mappings, called "functor". A category is a collection of primitive element types, a set of operations upon those types and an operator algebra which is capable of expressing the interaction between operators and elements [21]. In the mathematical language, a category C is consists of a class of objects and a class of morphisms, which are functions between objects, with composition and identity properties as follows [20]:

$$\begin{aligned}&\forall A\in C\ \exists e_A: A\rightarrow A\\ &\ni[\forall f: A\rightarrow B,\ g: B\rightarrow C \Rightarrow e_A.f = f,\ g.e_A = g]\\ &\forall f: A\rightarrow B\ \ g: B\rightarrow C\ \exists h: A\rightarrow C \ni h = f.g\end{aligned} \tag{1}$$

Fig. 1 shows an example for a category with objects A to D and morphisms $f1$ to $f5$.

A *functor* between two categories associates elements (objects) and operations (morphisms) from one category to another that preserves the structure and operator algebra [21]. For example, in Fig. 2, functor **F** transforms the elements of the left category to the equivalent elements in the right category. A functor must preserve the structure of the category. It means that identity morphisms must be mapped to their associated identity morphisms and composition must be preserved. The fundamental laws guarantee these conditions:

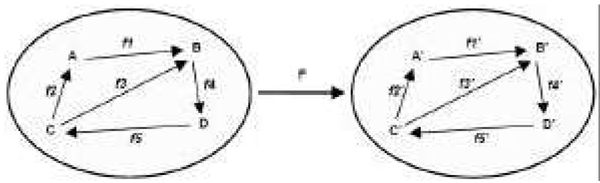


Fig. 2: Functor *F* transforms the elements of the left category to their equivalent elements in the right category

$$F(e_A) = e_{F(A)}$$
$$\forall f: A \to B \in P, g: B \to C \in P \Rightarrow$$
$$[F(f): F(A) \to F(B) \in Q, F(g): F(B) \to F(C) \in Q \qquad (2)$$

RESEARCH METHODOLOGY TO EXTEND STATIC SPATIAL OPERATIONS TO MOVING OBJECTS

This section explains how to use the proposed approach to extend 2D spatial operations to 3D and moving objects. In our categorical approach, representations of spatial operations in static and moving spaces have equivalent structures. These spaces can be seen as categories with data types and primitive operations whose combination constructs more complex spatial operations. A spatial operation in the category of static objects based on the formal description and using the data types and primitive operations is extendable to the categories of moving objects with a relevant functor. The functor extends the used data types and primitive operations through a defined mapping between static and moving objects. Then the complex spatial operations are mapped automatically, because they are defined as a combination of data types and primitive operations independent of their implementations [10, 11, 22]. Fig. 3 shows the described methodology to extend static spatial operations to their moving counterparts.

USING THE PROPOSED APPROACH TO EXTEND EUCLIDEAN DISTANCE TO MOVING POINTS

This section shows how to use the proposed approach to extend the Euclidean distance between two points to moving points. For 2D static points, this operations is defined as:

$$\mathrm{dist}(p_1,p_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (3)$$

To implement this, the data types *Point* and *Floating number* and required and the primitive operations are *plus*, *subtract*, *square* and *square root*. In

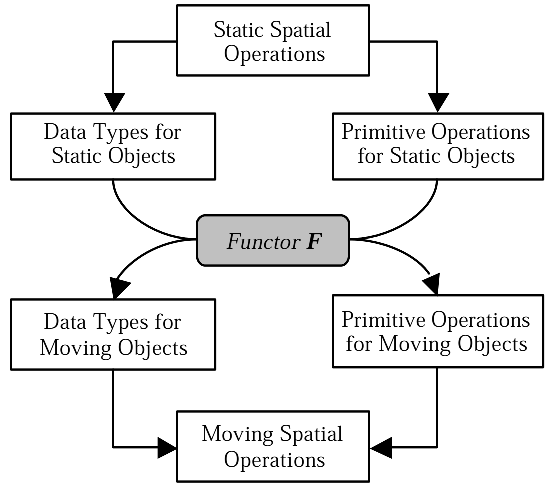


Fig. 3: Using functor *F* to extend static spatial operations to moving objects

the Cartesian coordinate system, a point is shown as a pair of Floating numbers:

$$Pt = (x, y) \qquad (4)$$

A moving point has a different position for any given time. In practice, the position of the point is known for a set of discrete time instances and for other time instances an interpolation mechanism is used. A moving point is modeled as a function of time, i.e., whose elements in the Cartesian coordinate system are time dependent:

$$MovingPt = (x(t), y(t)) \qquad (5)$$

To extend a data type to a moving data type, we define a functor *Moving*. It make its argument as a function of time:

$$Moving\ v = t \to v \qquad (6)$$

Application of this functor on the required data types for distance computation is as follows:

$$\begin{aligned} &MovingFloat = Moving\ (Float) \\ &MovingPt = Moving\ (Pt) \end{aligned} \qquad (7)$$

To extend an operation of static points to moving points, all of its arguments must become functions of time. We define different functors for functions with different number of arguments. For example, for operations with one and two arguments, *T1* and *T2* are developed:

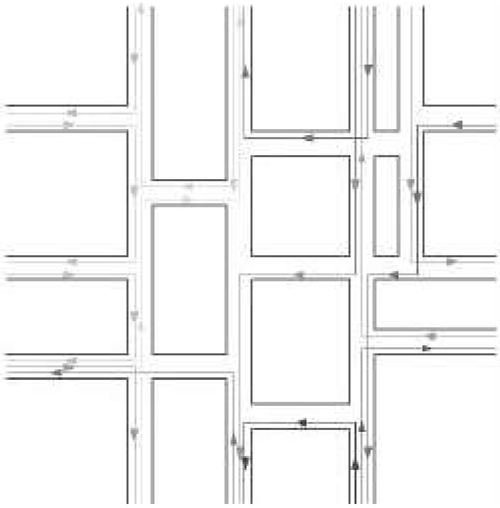


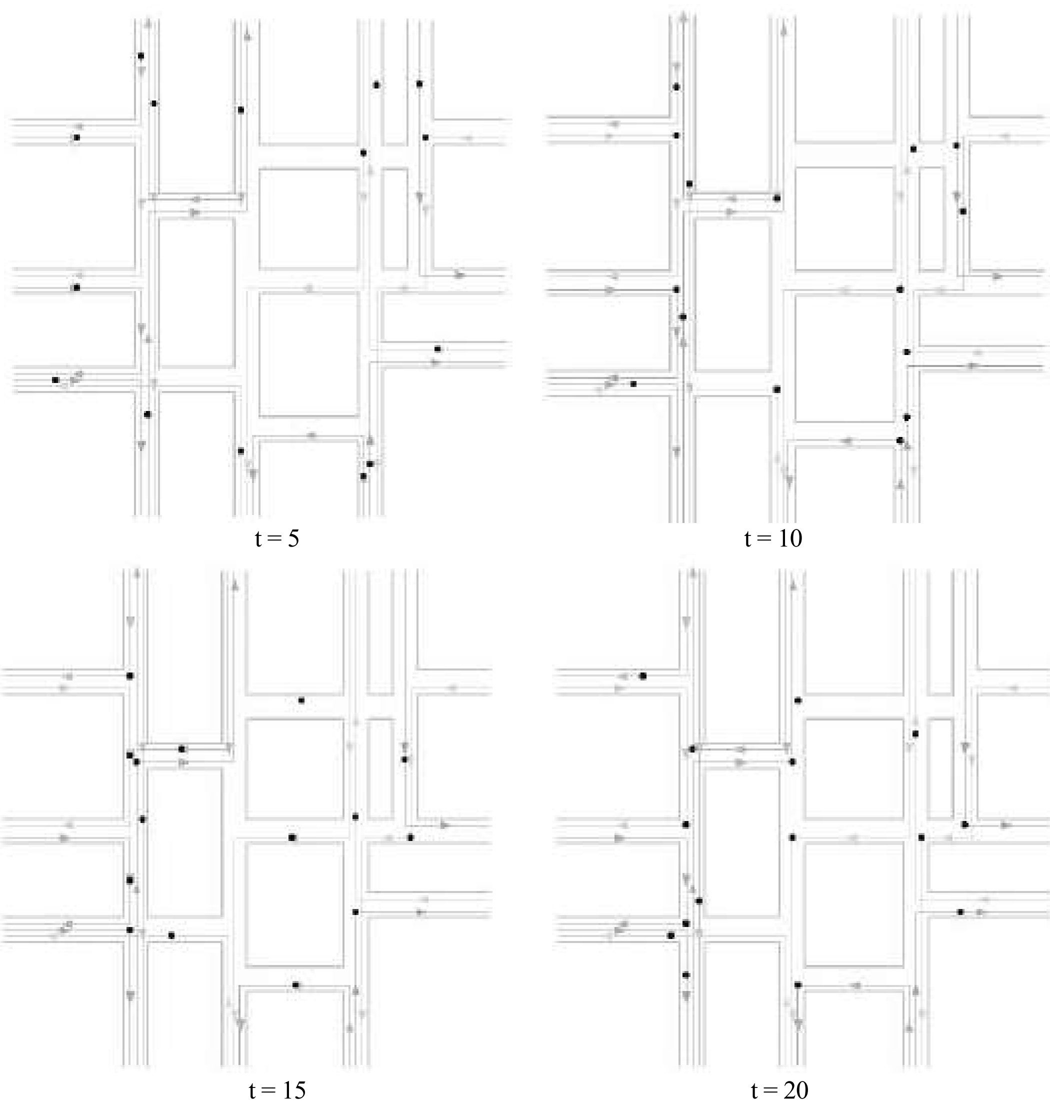
Fig. 4: The simulated street network

$$T1\,(f(a)) = f\,(a(t))$$
$$T2\,(f(a,b)) = f(a(t), b(t)) \qquad (8)$$

Note that these functors are independence of operations. Applying these functors on operations on static points provides us with the extended version of those operation for moving points. For example, extensions of the primitive operations of the distance computation to moving points are as follows:

$$(+) = T2\,(+)$$
$$(-) = T2\,(-)$$
$$\text{square} = T1\,(\text{square})$$
$$\text{sqrt} = T1(\text{sqrt}) \qquad (9)$$

Having extended the data types and the primitive operations to moving points, the combinatorial



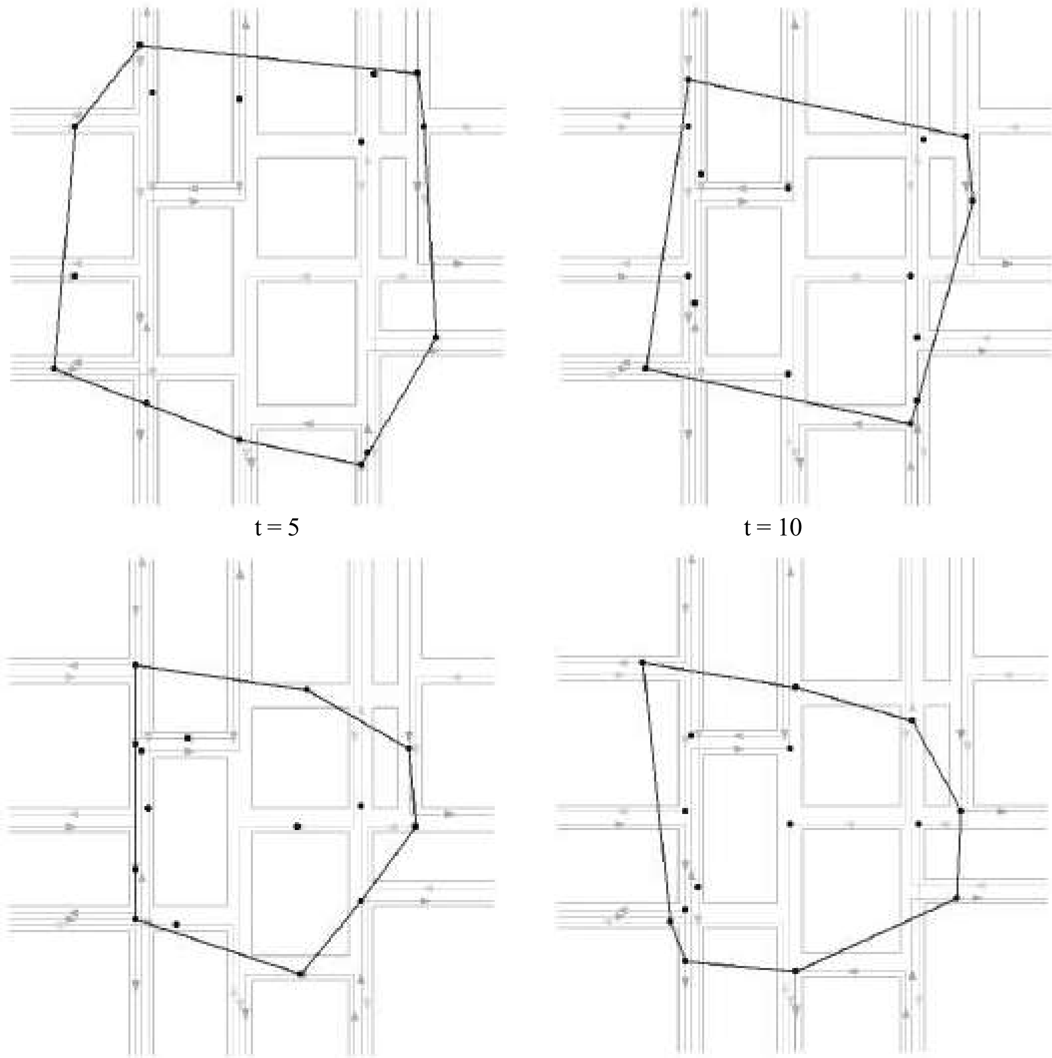
t = 5

t = 10

t = 15

t = 20

Fig. 5: The paths of the moving points on the street network for times 5, 10, 15 and 20



t = 5 t = 10 t = 15 t = 20

Fig. 6: Convex hull of the moving points for times 5, 10, 15 and 20

operations (e.g. distance computation) will be automatically extended, because they are defined based on the combination of the primitive operations, independent of any implementation.

IMPLEMENTATION RESULTS

The explained algorithm was developed to extend two sample spatial operations to moving points. The selected operations are convex hull computation and Voronoi Diagrams. The convex of a set of points is the smallest region that contains the points. The voronoi Diagram of a set of points partitions the space into a set of cells assigned to the points in a way that this point is the nearest point to all of the inside points of the cell. For more information about these operations, [23, 24, 25]. Implementation of the selected spatial operations needs more primitive operations for *Points*, e.g. ccw (counter-clock wise test for three points), sorting a list of points, etc.

A simulated transportation system, which was made of fifteen moving points, was selected as an example (Fig. 4 and 5). Results for the convex hull and Voronoi Diagrams of these moving points for times 5, 10, 15 and 20 are represented in Fig. 6 and 7, respectively.

CONCLUSION

Extending spatial operations to multi-dimensional objects is an essential advancement toward multi-dimensional GIS. Current approaches recommend particular technical solutions to extend a spatial operation to a new multi-dimensional space. What is

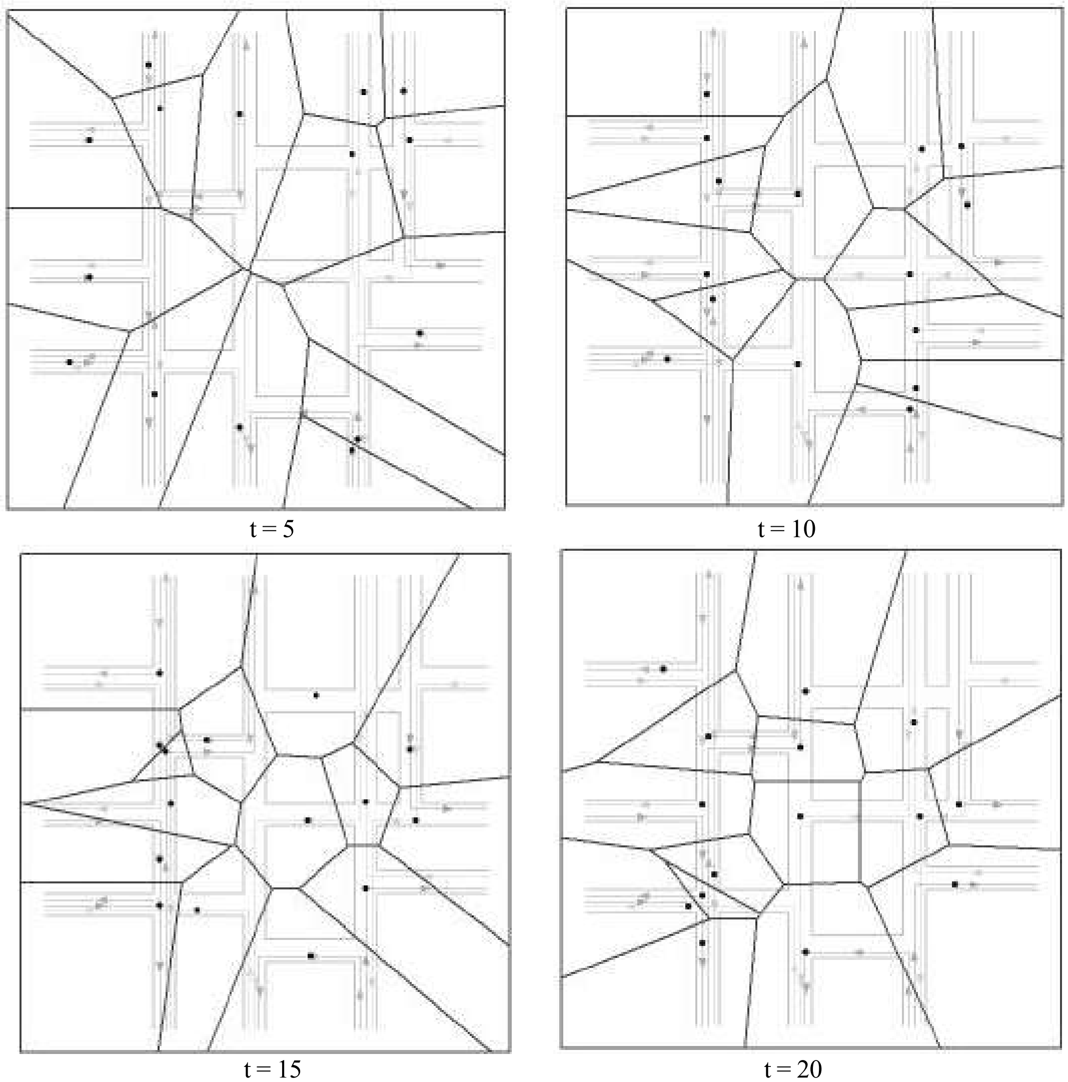


Fig. 7: Voronoi Diagrams of the moving points for times 5, 10, 15 and 20

reported here is the extension of spatial operations via their dimension-independent properties. This approach leads to a consistent solution toward a multi-dimensional GIS.

The achieved results to extend some selected spatial operations to moving points demonstrate the viability of the approach. Using the formalization of functions from category theory enabled us to implement the desired algorithm effectively. The same approach can be applied to extend other spatial operations, e.g., network analyses and simulation of pursuer-evader problem, which are the future work of this research.

Complexity and speed are factors used to evaluate the performance of an extension approach in current researches. However, the aim here is how to avoid recoding each spatial analysis for each dimension. Thus, the main concern of this research is on mathematical validation of the conceptual framework and investigation of its implementation issues. Nevertheless, the results show that the proposed approach has the minimum effect on complexity and time for applying the spatial operations on objects of higher dimensions.

REFERENCES

1. Burrough, P. and R. McDonnell, 1998. Principles of Geographical Information Systems. Oxford University Press.
2. Peuquet, D.J., 1999. Time in GIS and Geographical Databases, In: Longley, P.A., M.F. Goodchild, D.J. Maguire and D.W. D.W. Rhind (Eds.). Geographical Information System, Principals and Technical Issues (2nd Edn.), John Wiley & Sons, pp: 91-103.

3. Pfoser, D., 2000. Issues in the Management of Moving Point Objects. Ph.D. Thesis, Advisor: Jensen, C.S., Department of Computer Sceince. Faculty of Engineering and Sceince, Aalborg University, Aalborg, Denmark.
4. Raper, J., 2000. Multi-dimensional Geographic Information Science, Taylor & Francis.
5. Wolfson, O., B. Xu, S. Chamberlain and L. Jiang, 1998. Moving Objects Databases: Issues and Solutions. In Proceedings of the 10th International Conference on Scientific and Statistical Database Management, Capri, Italy, pp: 111-122.
6. Bajaj, C. and W. Bouma, 1990. Dynamic Voronoi Diagrams and Delaunay Triangulations. In Proceedings of the 2nd Annual Canadian Conference on Computational Geometry, Ottawa, Canada, pp: 273-277.
7. Hashemi-Beni, L., M.A. Mostafavi and M.L. Gavrilova, 2007. Moving Objects Management in a 3D Dynamic Environment. In Proceedings of the Geocomputation 2007, NUI Maynooth, Ireland, September 3-5, 2007.
8. Kallmann, M., H. Bieri and D. Thalmann, 2003. Fully Dynamic Constrained Delaunay Triangulations, In: Brunnett, G., B. Hamann, H. Mueller and L. Linsen (Eds.). Geometric Modelling for Scientific Visualization, pp: 241-257.
9. Mostafavi, M.A., 2002. Development of a Global Dynamic Data Structure, Ph.D. Thesis, Advisor: C. Gold, University of Laval, Laval, Canada.
10. Karimipour, F., A.U. Frank and M.R. Delavar, 2008. An Operation-Independent Approach to Extend 2D Spatial Operations to 3D and Moving Objects, In: Sammet, H., C. Shahabi and W.G. Aref (Eds.). Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS 2008), Irvine, CA, USA, November 5-7, 2008.
11. Karimipour, F., H. Rezayan and M.R. Delavar, 2006. Formalization of Moving Objects Spatial Analyses Using Algebraic Structures. In: Kuhn, W. and M. Rabaul (Eds.). Proceedings of Extended Abstracts of GIScience 2006, Münster, Germany, September 20-23, 2006, IfGI Prints, 28: 105-111.
12. Guibas, L.J., J.S. Mitchell and T. Roos, 1992. Voronoi Diagrams of Moving Points in the Plane. In Proceedings of the 17th International Workshop, June 17-19, 1991, pp: 113-125.
13. Mostafavi, M.A., C. Gold and M. Dakowiczb, 2003. Delete and Insert Operations in Voronoi/Delaunay: Methods and Applications. Journal of Computers and Geosciences, 29: 523-530.
14. Roos, T., 1991. Dynamic Voronoi Diagrams, Ph.D. Thesis, University of Wurzburg, Wurzburg, Germany.
15. Jacob, R., 2002. Dynamic Planar Convex Hull, Ph.D Thesis, Advisor: G. Brodal, Department of Computer Science, University of Aarhus, Denmark.
16. Razzazi, M.R. and A. Sajedi, 2003. Kinetic Convex Hull Maintenance Using Nested Convex Hulls, In Proceedings of the 19th European Workshop on Computational Geometry, Bonn, Germany, March 24-26, 2003, pp: 113-116.
17. Frank, A.U., 1999. One Step Up the Abstraction Ladder: Combining Algebras-From Functional Pieces to a Whole, In: C. Freksa and D. Mark (Eds.). Proceedings of the International Conference COSIT'99, Stade, Germany, August 25-29, 1999, Lecture Notes in Computer Science (LNCS), Springer-Verlag, 1661: 95-107.
18. Karimipour, F., M.R. Delavar and A.U. Frank, 2008. A Mathematical Tool to Extend 2D Spatial Operations to Higher Dimensions, In: Gervasi, O. *et al*. (Eds.). Proceedings of the International Conference on Computational Science and Its Applications (ICCSA 2008), Perugia, Italy, June 30-July 3, 2008, Lecture Notes in Computer Science (LNCS), Springer-Verlag, 5072: 153-167.
19. Frank, A.U., 2007. Practical Geometry-Mathematics for Geographic Information Systems. Unpublished Manuscript.
20. MacLane, S. and G. Birkhoff, 1999. Algebra (3rd Edn.). AMS Chelsea Publishing.
21. Lawvere, F.W. and S.H. Schanuel, 2005. Conceptual Mathematics: A First Introduction to Categories, Cambridge University Press.
22. Karimipour, F., 2005. Formalization of Spatial Analyses of Moving Objects Using Algebraic Structures. M.Sc. Thesis, Advisors: Delavar, M.R. and A.U. Frank, Department of Surveying and Geomatics Engineering, College of Engineering. University of Tehran, Tehran, Iran, (In Persian).
23. Berg, M.D., O. Cheong, M. Van-Kreveled and M. Overmars, 2008. Computational Geometry: Algorithms and Applications (3rd Edn.), Springer-Verlag.
24. Orourke, J., 1998. Computational Geometry in C (2nd Edn.), Cambridge University Press.
25. Okabe, A., B. Boots, K. Sugihara and S.N. Chiu, 2000. Spatial Tessellations: Concepts and Applications of Voronoi Diagrams (2nd Edn.), John Wiley.