International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.3, 2015

RESEARCH ARTICLE

ISSN: 2321-7758

COMPUTING DIMENSIONAL SPACE OF SCATTER DIAGRAM FOR SPATIOTEMPORAL DATASET

G.SHEEBA[1], D.RENUKA DEVI[2], R.UTHRADEVI[3]

[1,3]PG scholar, Department of CSE, Rnganayagi varatharaj College of Engineering, Salvarpatti, TamilNadu

[2]Assistant Professor, Department of CSE, Rnganayagi varatharaj College of Engineering, Salvarpatti, TamilNadu

Article Received:05/05/2015 | Article Revised on:14/05/2015 | Article Accepted on:18/05/2015

ABSTRACT

Scientific simulation is a most challenging task in, which the basic components of large systems are treated as classical entities (i.e., particles) the structure of the system, movement and function are explore by molecular simulation (MS) data, which analyzing the behavior of natural systems. Data generated by such simulations impose great challenges to database storage and query processing. One of the queries against particle simulation data is spatial distance histogram (SDH), which asks for the distances of all pairs of particles in the simulated system. In previous work, the computation time of an SDH query using brute force method is quadratic. Often, such queries are executed continuously over certain time periods, increasing the computation time. The objective of our project is to propose highly efficient approximate algorithm to compute SDH, the key idea of our algorithm is to derive statistical distribution function from the spatial characteristics of particles. In order to analyze the MS data, it requires estimating the probability distribution of that individual node. However, to generate the particle spatial distributions, it is infeasible to evaluate for all possible data. Instead, it only focuses the data based on the spatial uniform distribution. Correlation Coefficient function is used to identify the similarity between the nodes. In this paper we identify Pearson Correlation Coefficient for all pairs, it also exhibits Euclidean Distance between two nodes. Upon organizing the data into a region based structure such as uniform region and non uniform region, for the spatial characteristics of particles are used to reduce the time consumption. The accuracy and efficiency of the proposed algorithm is further improved by implementing the above algorithm in Graphics Processing Units (GPUs).

Keywords: Spatial Distance Histogram, Probability Distribution Function, Euclidean Distance Calculation, Correlation Coefficient.

I. INTRODUCTION

The advancement of computer simulation systems and experimental devices has yielded large volume of scientific data. This imposes great strain on the data management software, in spite of effort made to deal with such large amount of data using database management systems (DBMS). Data in scientific databases is generally accessed through high-level analytical queries, which are much more complex to compute in comparison to simple

aggregates. Many of these queries are composed of few frequently used analytical routines which usually take super linear time to compute using brute-force methods. Hence, the scientific database systems need to be able to efficiently handle the computation of such analytical queries. This paper presents our work related to such type of a query that is very important for the analysis of *molecular simulation* (MS) data. Molecular (or particle) simulations are simulations of complex physical, chemical or biological structures done on computers. They are extensively used as a basic research tool for analyzing the behavior of natural systems under experimental framework [4], [5]. The number of particles involved in MSs is large, oftentimes counting millions. The brute-force method for SDH computation calculates the distances between all pairs of particles and updates the relevant buckets of the histogram. The main idea in such approach is to process all the particles in each node of the tree as a whole. This is an obvious improvement in terms of time over the brute-force method which builds the histogram by computing particle-to-particle distances separately. A quad-tree data structure is presented in my previous work [6]. Although the quad-tree based approach is an improvement over the brute-force method for SDH computation.

Objectives: The objective of my project is to propose highly efficient approximate algorithm to compute SDH. The key idea of this algorithm is to derive statistical distribution function from the spatial characteristics of nodes.

A. Problem Statement

The SDH is the distances between all particles in the system. It can be formally described as follows: given the coordinates of N particles and a user-defined distance w, we need to compute the number of particle-to-particle distances falling into a series of ranges.

B. Contributions and roadmap

We claim the following contributions via this work:

Large scale biological structures are represented using all the individual atoms. Data is stored in single or multiple trajectory databases containing time frames. Each frame is a sequential list of atoms with their positions, velocities, perhaps forces, masses, and types. Dataset is very large: millions of atoms, tens of thousands of frames.

Similarly methodology in road map it contains large number of cities, represent by using their attributes value such as latitude and longitude. Spatial Distance Histogram is the approach, which specify the distance between two nodes, avoid the calculation of pair wise distances. Similar methodology in other sciences: astronomy, material science, civil engineering. Distance histogram is an important query in simulation databases.

C. Overview

This paper presents a *highly efficient* algorithm for processing SDH of large-scale MS data with improved efficiency and accuracy over existing solutions. To achieve this, the algorithm takes advantage of the two types of uniformity widely such as uniform and non uniform region. Once we identify these regions (using the $\chi 2$ test), we derive the Probability Distribution Functions (PDFs) of the distances between all pairs of these regions. Exploiting this property makes algorithm running time independent of the SDH bucket width w – such dependency is the main drawback of existing algorithms. On the other hand, working with the PDFs of distance distribution guarantees very little error will be made.

We continue this paper by formally provide a brief survey on related work in Section II; we introduce our proposed system in Section III; Section IV presents the results obtained through extensive experiments; and conclude this paper by Section V; Finally, which we also discuss our future work in Section VI.

II. RELATED WORK

The efficient handling of spatiotemporal data is an increasing demand of modern DBMS, motivate both location based services and telecommunication the idea of computing a 2-Dimensional space. Each cell of the node is associated with a list. Assuming in particular that an object O_1 enters cell C_m at time t_u the pair (O_1,t_u) is inserted into a list associated with cell C_m. Each such list is ordered by object ID. This approach is called list solution[3]. This indexing structure enables the efficient maintenance of versions of the lists that are created by update operations. Since the versions correspond to time instants, are finally achieved by the time dimension into a persistent indexing structure. A set of such predicates forms a buffer query or a spatiotemporal Pattern (STP) Query with time. In the more general

case of an STP query, the temporal dimension is introduced via the relative order of the spatial predicates (STP queries with order). Therefore, the efficient processing of a spatiotemporal predicate is crucial for the efficient implementation of more complex queries of practical interest. In this paper[2], data is organized into a Quad-tree based data structure. The spatial locality of each node (at given time) in the tree is acquired to determine the particle distribution. Similarly, the temporal locality of particles (between consecutive time periods) in each node is also acquired.

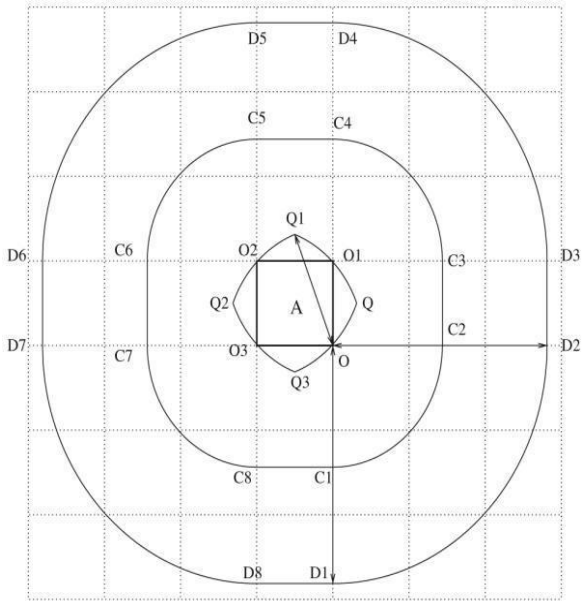

Fig. 1. Density Map

Analysis of scientific spatiotemporal data often involves computing functions of all point-to-point interactions. The main idea of this approach is, it first divides the simulated space into a grid, each cell of which records the number of data points in it. A density map is a conceptual data structure that divides the simulation space into a grid of small cells (or regions) of equal size. The region is a cube in 3D and square in 2D. Each cell of the grid is further divided into four equal sized cells to generate a density map of higher resolution. Such a grid is known as density map and density maps. These algorithms adopt a recursive tree traversing strategy to process point to point distances in the visited tree nodes. Therefore organize all point coordinates into a point region Quad-tree with each node representing a cell (square for 2D data and cube for 3D) in space. Point counts of each cell are cached in the corresponding tree node. Those with zero point count are removed from the tree. The height of the tree (denoted as H) is determined in a way such that the average number of points in all possible leaf nodes is no smaller than a predefined threshold β.

$$H = \log_2 d \frac{N}{\beta}$$

To be specific, d is the number of dimensions and 2^d is essentially the maximal degree of tree nodes. If a Quad-Tree is implemented using links, most of the memory will be taken up by the links.

III. PROPOSED WORK

In this section, we introduce the main concepts of our proposed work. This project presents a highly efficient and practical algorithm for processing SDH of large-scale MS data with improved efficiency and accuracy over existing solutions. To achieve this, the algorithm takes advantage of the two types of uniformity widely present in MS data such as uniformity and non uniformity. The second type of uniformity data are used by the spatial distribution estimate by uniform probability distribution function. Because of this, there are many localized regions (call uniform regions) in the simulation space in which the particles are uniformly distributed so it reduce the time complexity when compared to the brute force approach Correlation Coefficient have to be computed. Correlation function is a statistical correlation between random variables at two different points in space or time, usually as a function of the spatial or temporal distance between the points. It used for measuring the similarity between the regions in the dataset. Based on this, we differentiate the uniform and non-uniform regions. Through this, we can evaluate the performance of existing and proposed system that means brute force and correlation function computation. Correlation function is used for identifying the similarities between the objects in the dataset. We will implement the correlation function called Pearson coefficient for identifying the closed regions. It achieves better accuracy and efficiency when compared with the existing system.

The major technical contributions presented here are:

1) Techniques to identify spatial uniformity within a frame and temporal uniformity among consecutive frames
2) An approximate approach to compute the SDH of large number of data frames by utilizing the above properties

A. Pearson Correlation Coefficient

It measure the strength and direction of the linear relationship between two variables, describing the direction and degree to which one variable is linearly related to another.

The Pearson Correlation Coefficient is a very helpful statistical formula that measures the strength between variables and relationships. In the field of statistics, this formula is often referred to as the Pearson R test. When conducting a statistical test between two variables, it is a good idea to conduct a Pearson correlation coefficient value to determine just how strong that relationship is between those two variables**.** Correlation computation function is used for measuring the similarity between the regions in the dataset. Based on this, it differentiates the uniform and non-uniform regions. Pearson Correlation Function is the measure of a correlation between two variables X and Y giving a value between +1 and -1 thus -1< r <1. This approach is often implemented by using correlation matrix function which computes Pearson Correlation Coefficient for all combinations. Is defined as the covariance of two variables, used to measure the strength of a linear association between two nodes.

Letter "r" and "ρ" is used to represent the Pearson Correlation Function:

$$r = \frac{n(\sum XY) - (\sum X)(\sum Y)}{\sqrt{[n\sum X^2 - (\sum X)^2][n\sum Y^2 - (\sum Y)^2]}}$$

Where X and Y represent the latitude and longitude values

Formula Description:

- N = Total number of nodes
- Σ XY = Sum of the product of paired nodes
- Σ X = Sum of X nodes
- Σ Y = Sum of Y nodes
- ΣX^2 = Sum of squared X nodes
- ΣY^2 = Sum of squared Y nodes

B. Region Classification

This test measure the similarities between two variables, where the value *r* = 1 means a perfect positive correlation and the value *r* = -1 means a perfect negative correlation. The correlation coefficient ranges from −1 to 1. All it contains a normally distributed values and there is no outliers in the data.

The Pearson correlation coefficient can take values from -1 to +1

- +1 denotes increasing relationship
- -1 denotes decreasing relationship
- 0 denotes related to each other

A value of 1 implies that a linear equation describes the relationship between *X* and *Y* perfectly, with all data points lying on a line for which *Y* increases as *X* increases. A value of −1 implies that all data points lie on a line for which *Y* decreases as *X* increases. A value of 0 implies that there is no linear correlation between the variables.

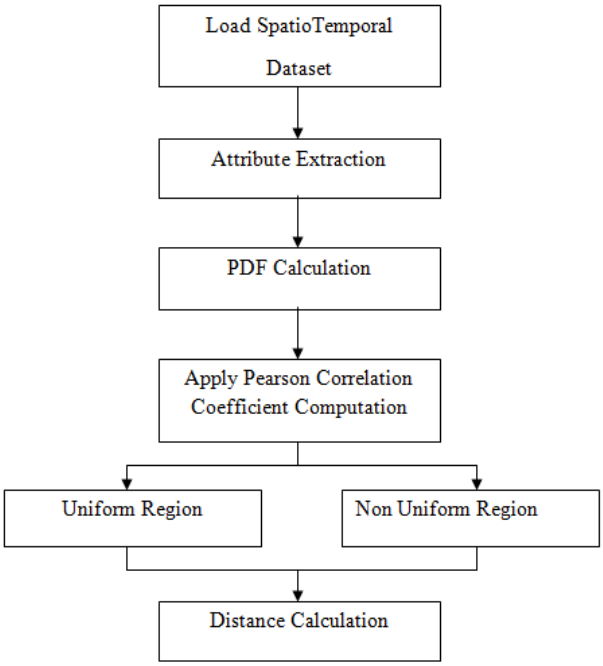

Fig. 2. System Design

C. PDF Estimations

First idea to address the aforementioned challenge is to take advantage of the spatial distribution of data points in the regions. PDF is an accurate description of the underlying distance distribution. Therefore, the main task of the proposed approach is to derive the PDF. However, to generate the particle spatial distributions, it is infeasible to test the MS dataset for all possible data distributions. Instead, it focus on testing if the data follows the most popular distribution in MS spatial uniform distribution. Once we identify these uniform regions and non uniform, regions (using the Pearson Correlation Coefficient function), we derive the Probability Distribution Functions (PDFs) for non uniform region particles can be done by this formula:

Probability Density Function

The Probability Density of X is:

$$f(x) = \frac{1}{b-a} \quad a < x < b$$

The distances are proportionally distributed into the few buckets based on the overlaps range u and v, such a primitive heuristic which is named PROP (Proportional) implicitly assumes that the distance distribution is uniform within u and v. uniform distribution or rectangular distribution, defined the each member of the family of all intervals of the same length on the distributions support on equally probable.

D. Spatial Distance Computation

In this, we are estimating the spatial distance among the regions, region positions, temporal calculation and spatial distance. In the region position, it estimating the x and y co-ordinates of the different regions. By using the SDH algorithm, here we estimate the spatial distance between the two regions. Then we compute the temporal estimation between the time estimations.

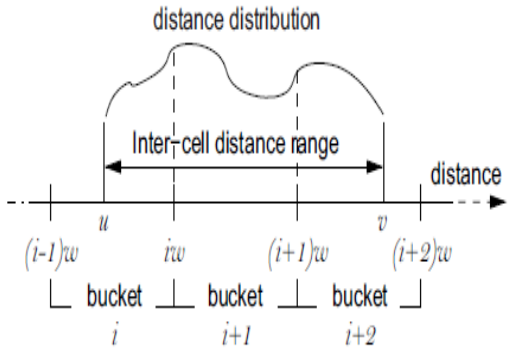

Fig. 3. Particle to Particle Distance Estimation

Fig..3 describes that the two cells have a distance range [u, v] which overlaps with three SDH buckets (i.e., from bucket i to i + 2). A critical observation here is: if we knew the probability distribution function (PDF) of the particle to particle distances between cells A and B, we can effectively distribute the actual number of distances node A, node B into the three overlapping SDH buckets.

$$dist(X,Y) = \sqrt{2n(1-r(X-Y)}$$

Specifically, the total number of node A, node B distances will be assigned to the buckets based on the probability of distance falling into each bucket according to the Z Normalized Euclidean distance.

E. Shortest Path Estimation

Approximate landmark based methods for point-to-point distance estimation in very large networks. These methods involve selecting a subset of nodes as landmarks and computing offline the distances from each node in the graph to those landmarks. At runtime, when the distance between a pair of nodes is needed, it can be estimated quickly by combining the precomputed distances. One way of finding the shortest path between two locations is *Dijkstra's algorithm,*. It finds the shortest path between two nodes of a weighted graph, trying out the most promising routes first. A weighted graph is simply a graph where each edge e is assigned a non-negative value called the *weight*, w(e), of the edge. A *path* is a sequence of vertices p = (v_1,..., v_n) such that v_i~ v_{i+1}. Set e_i = (v_i, v_{i+1}). The length of a path p is d (p) = &Sigma w(e_i). For convenience we will also write w(u, v) to denote the weight of the edge (u,v). In this, we are fetching the optimal and distance between two regions. we have to choose the optimal source and destination regions. After choosing the source and destination, we find the time taken to travel and find the distance between the source and destination.

IV. EXPERIMENT RESULTS

The state of a system at a particular point in time is given below as a snapshot. Each diagram specifies the various process in project. Initially the spatial dataset is browsed from the system. It have attributes like latitude, longitude, and city names.

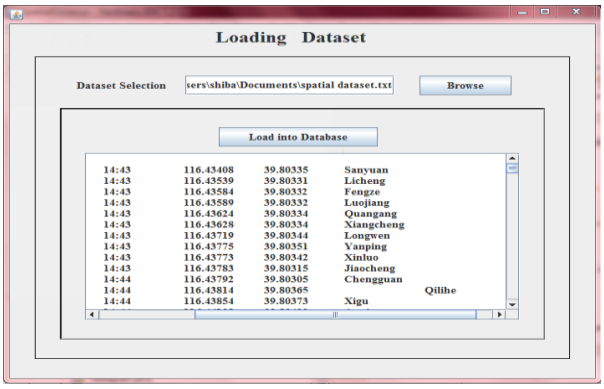

Fig.4. Spatiotemporal Dataset

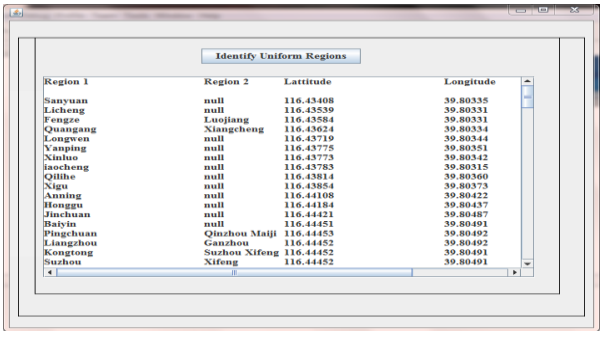

Fig. 5. Clustering the Region

Regions are clustered into two forms such as uniform region and non uniform region

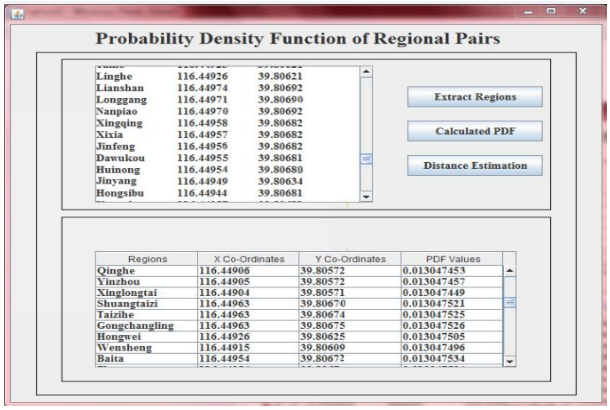

Fig. 6. PDF Estimation

After clustering the regions probability distribution function is calculated for all non uniform regions

Distance calculation Using Correlation Coefficient

latitude	longitude	place	distance
116.43408	39.80335	Sanyuan	0.08486472
116.43539	39.80331	Licheng	0.08486621
116.43584	39.80331	Fengze	0.0848667
116.43584	39.80331	Luojiang	0.0848667
116.43624	39.80334	Quangang	0.08486712
116.43624	39.80334	Xiangcheng	0.08486712
116.43719	39.80344	Longwen	0.08486805
116.43775	39.80351	Yanping	0.08486861
116.43773	39.80342	Xinluo	0.08486868
116.43783	39.80315	Jiaocheng	0.08486909
116.43792	39.80305	Chengguan	0.0848693
116.43814	39.8036	Qilihe	0.08486893
116.43854	39.80373	Xigu	0.084869236
116.44108	39.80422	Anning	0.0848715

Fig. 7. Distance Calculation

Fig.7 shows the distance between selected source and destination

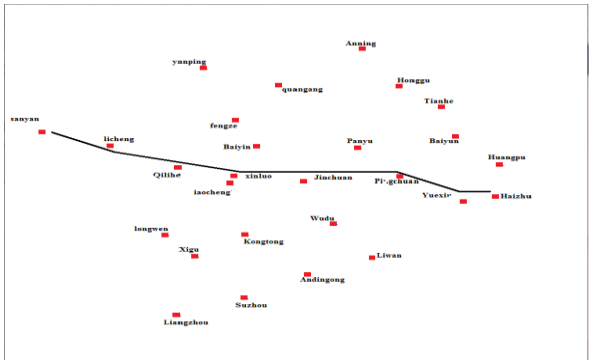

Fig. 8. Shortest Path Estimation

V. CONCLUSION

An efficient approximate solution to the spatial distance histogram query is provided in this project. SDH is one of the very important molecular simulation data analysis queries that is frequently applied to a collection of data frames. In previous work they use the point region Quad-tree to organize simulation data as we did in our previous work. However, Correlation Coefficient algorithm provides much higher efficiency and accuracy by taking advantage of the data locality and statistical data distribution properties. The algorithm presented makes it practically feasible to analyze data of large number of frames continuously. Its efficiency and accuracy are supported by mathematical analysis and extensive experimental results. The scientific data analysis can be performed in real time by using such modern hardware systems. landmark-based methods for point-to-point distance involve a subset of nodes as landmarks and computing offline the distances from each node in the graph to those landmarks. An immediate work of interest is to extend our algorithm to identify the intermediate path for the minimal distance. We are in the process of developing solutions for such problems and hope, with the success of such development, to enter an exciting era of computational science endeavours.

VI. FUTURE WORK

Finally this algorithm makes profit of the multi-core lateral processing character of GPUs. It grant a low-priced and low-power staging to improve efficiency as related to computer clusters. The upgrading of computer simulation scheme and experimental gadget has allowed large volume of scientific data.

REFERENCES

[1]. Vladimir Grupcev, Yongke Yuan, “Approximate Algorithms for Computing Spatial Distance Histograms with Accuracy Guarantees,” IEEE Trans Knowl Data Eng. 1; volume 25, issue 9, September, 2012

[2]. Yongke Yuan∗., “Distance histogram computation based on spatiotemporal uniformity in scientific data.” in International Journal of Engineering Trends and Technology (IJETT) – Volume 6 Number 1- Dec, 2013

[3]. G. Lagogiannis et. al., “A time efficient indexing scheme for complex spatiotemporal retrieval,” Springer, pp.113-133, 2012

[4]. Naveen Kumar et al, Density-Based Spatial Clustering International Journal of Computer Science and Mobile Computing, Vol.3 Issue.3, pg. 1004-1012, March- 2014

[5]. Mikhail Trifonov, Age-Related Changes in Probability Density Function of Pairwise Euclidean Distances between Multichannel

Journal of Biosciences and Medicines, 2, 19-23, 2014

[6]. Kurada Ramachandra Rao,Unsupervised Classification of Uncertain Data Objects in Spatial Databases Using Computational Geometry and Indexing Techniques International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 2, pp.806-814, Mar-Apr 2012

[7]. Karine Zeitouni, A Survey of Spatial Data Mining Methods Databases and Statistics Point of Views International Journal of Basic & Applied Sciences IJBAS-IJENS Vol:10 No:01, Oct 2013

[8]. Toyonori Munakataa and Kang Kim, M-body density functional theory and the generalized hypernetted-chain Equation Journal of Chemical Physics Volume 113, number 10 8, Sep 2013

[9]. A. Kumar et. al., "Computing Spatial Distance Histograms for Large Scientific Datasets," in Knowledge and Data Engineering, IEEE Transactions on, Jan 2014

[10]. I. Szapudi, Introduction to Higher Order Spatial Statistics in Cosmology. Lecture Notes in Physics, Springer Verlag, 2009, vol. 665

[11]. B. Hess et. al., "GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation," Journal of Chemical Theory and Computation, vol. 4, no. 3, March 2008

[12]. A. Gray et. al.,"N-body problems in statistical learning," in Advances in Neural Info. Processing Systems, 2001

[13]. J. Barnes et. al., "A Hierarchical O(N log N) Force-Calculation Algorithm," Nature, vol. 324, no. 4, 1986

[14]. L. Greengard et. al., "A Fast Algorithm for Particle Simulations," Journal of Computational Physics, vol. 135, no. 12, 1987

[15]. S. Chen et. al., "Performance analysis of a dual-tree algorithm for computing spatial distance histograms," VLDBJ, vol. 20, no. 4, 2011

[16]. H. Kaplan, "Persistent data structures," in Handbook on Data Structures and Applications. CRC Press, 2001