Asian Journal of Applied Science and Technology (AJAST)

(Peer Reviewed Quarterly International Journal) Volume 2, Issue 4, Pages 240-245, Oct-Dec 2018

Appliance of Graph Theory to Compute the Shortest Path in Mixed Layer Height with K-Nearest Neighbor Graph (KNN-G)

E.Indhumathy[1] & Mr.T.Ramesh[2]

[1]Research Scholar, Department of Mathematics, Dr.SNS Rajalakshmi College of Arts and Science (Autonomous) (Affiliated to Bharathiar University), Reaccredited with 'A' Grade by NAAC, Coimbatore-641049, Tamil Nadu, India.
[2]Department of Mathematics, Dr.SNS Rajalakshmi College of Arts and Science (Autonomous) (Affiliated to Bharathiar University), Reaccredited with 'A' Grade by NAAC, Coimbatore-641049, Tamil Nadu, India.

Article Received: 12 June 2018 *Article Accepted: 24 October 2018* *Article Published: 05 December 2018*

ABSTRACT

The height of the atmospheric boundary layer or mixing layer is an important parameter for understanding the dynamics of the atmosphere and the dispersion of trace gases and air pollution. The height of the mixing layer (MLH) can be retrieved, among other methods, from Lidar or ceilometers backscatter data. These instruments use the vertical backscatter Lidar signal to infer MLHL, which is feasible because the main sources of aerosols are situated at the surface and vertical gradients are expected to go from the aerosol loaded mixing layer close to the ground to the cleaner free atmosphere above. Various Lidar/ceilometers algorithms are currently applied, but accounting for MLH temporal development is not always well taken care of. Graph theory is the study of graphs and which are mathematical structures used to model pair wise relations between objects. A graph in this context is made up of vertices, nodes, or points which are connected by edges, arcs, or lines. A graph may be undirected, meaning that there is no distinction between the two vertices associated with each edge, or its edges may be directed from one vertex to another. Complex networks, such as biological, social, and communication networks, often entail uncertainty, and thus, can be modeled as probabilistic graphs. Similar to the problem of similarity search in standard graphs, a fundamental problem for probabilistic graphs is to efficiently answer k-nearest neighbor queries (k-NN), which is the problem of computing the k closest nodes to some specific node. In this paper we introduce a framework for processing k-NN queries in probabilistic graphs. We propose novel distance functions that extend well-known graph concepts, such as shortest paths. In order to compute them in probabilistic graphs, we design algorithms based on sampling. During k-NN query processing we efficiently prune the search space using novel techniques.

Keywords: MLH, Atmospheric Boundary, KNN-G, Graph theory.

1. INTRODUCTION

Graphs can be used to model many types of relations and processes in physical, biological, social and information systems. Many practical problems can be represented by graphs. Emphasizing their application to real-world systems, the term network is sometimes defined to mean a graph in which attributes (e.g. names) are associated with the nodes and/or edges [2]. In computer science, graphs are used to represent networks of communication, data organization, computational devices, the flow of computation, etc. For instance, the link structure of a website can be represented by a directed graph, in which the vertices represent web pages and directed edges represent links from one page to another. A similar approach can be taken to problems in social media, travel, biology, computer chip design, mapping the progression of neuro-degenerative diseases, and many other fields. The development of algorithms to handle graphs is therefore of major interest in computer science. The transformation of graphs is often formalized and represented by graph rewrite systems. Complementary to graph transformation systems focusing on rule-based in-memory manipulation of graphs are graph databases geared towards transaction-safe, persistent storing and querying of graph-structured data [7].

The k-nearest neighbor graph (k-NNG) is a graph in which two vertices p and q are connected by an edge, if the distance between p and q is among the k-th smallest distances from p to other objects from P. The NNG is a special case of the k-NNG, namely it is the 1-NNG. k-NNGs obey a separator theorem: they can be partitioned into two sub graphs of at most n(d + 1)/(d + 2) vertices each by the removal of $O(k^{1/d}n^{1-1/d})$ points.

Another special case is the (n − 1)-NNG. This graph is called the farthest neighbor graph (FNG). In theoretical discussions of algorithms a kind of general position is often assumed, namely, the nearest (k-nearest) neighbor is unique for each object. In implementations of the algorithms it is necessary to bear in mind that this is not always the case.

In this method, various Lidar/ceilometers algorithms are currently applied, but accounting for MLH temporal development is not always well taken care of. As a result, MLHL retrievals may jump between different atmospheric layers, rather than reliably track true MLH development over time [3]. This hampers the usefulness of MLHL time series, e.g. for process studies, model validation/verification and climatology. Here, we introduce a new method “pathfinder”, which applies graph theory to simultaneously evaluate time frames that are consistent with scales of MLH dynamics, leading to coherent tracking of MLH. ‘Neighborhoods Method’ (NM) i.e. Nearest Neighbor Graph theory uses novel search space reduction techniques and has a theoretical quadratic speed-up making it practically faster (by an order of magnitude) than recent branch-and-bound exhaustive search solutions. This graph theory helps to find the shortest path efficiently [9].

2. RELATED WORK

There are a number of papers that use hill-climbing or kNN graphs for nearest neighbor search, but to the best of our knowledge, using hill-climbing on k-NN graphs is a new idea. Baltink, H. K assumes that each point (e.g., an image) is specified as a collection of components (e.g., objects). Each point has the form of Xi = (V1,...,Vm), where each Vj is an object and can take values from a finite set (e.g., a set of squares of different sizes). The objective is to find the point in the dataset that has the closest configuration to the query Q [4]. Beyrich, F says Xi and Xj are neighbors if one can be converted to the other by changing the value of one of its variables. Then several heuristics to perform hill-climbing on such a graph are proposed [5]. Dijkstra’s, E aim at minimizing the number of distance computations during the nearest neighbor search [7]. A k-NN graph is built from dataset points and when queried with a new point, the graph is used to estimate the distance of all points to the query, using the fact that the shortest path between two nodes is an upper bound on the distance between them.

Using the upper and lower bound estimates, Wauben W eliminate points that are far away from the query point and exhaustively search in the remaining dataset and define a visibility graph and then perform nearest neighbor search by a greedy routing over the graph. This is a similar approach to our method, with two differences [6]. First, J. ACM search over the visibility graph, while we search on the k-NN graph. k-NN graphs are popular data structures that are used in outlier detection, VLSI design, pattern recognition and many other applications [8]. The second difference is that Lifshits make the following strong assumption about the dataset.

LSH uses several hash functions of the same type to create a hash value for each point of the dataset. Each function reduces the dimensionality of the data by projection onto random vectors. The data is then partitioned into bins by a uniform grid. Since the number of bins is still too high, a second hashing step is performed to obtain a smaller hash value. At query time, the query point is mapped using the hash functions and all the data points that are in the same bin as the query point are returned as candidates. The final nearest neighbors are selected by a linear search through candidate data points.

A KD-tree partitions the space by hyper planes that are perpendicular to the coordinate axes. At the root of the tree a hyper plane orthogonal to one of the dimensions splits the data into two halves according to some splitting value. Each half is recursively partitioned into two halves with a hyper plane through a different dimension. Partitioning stops after log n levels so that the bottom of the tree each leaf node corresponds to one of the data points. The splitting values at each level are stored in the nodes. The query point is then compared to the splitting value at each node while traversing the tree from root to leaf to find the nearest neighbor. Since the leaf point is not necessarily the nearest neighbor, to find approximate nearest neighbors, a backtrack step from the leaf node is performed and the points that are closer to the query point in the tree are examined. In our experiments, instead of simple backtracking, we use Best Bin First (BBF) heuristic M. Bern to perform the search faster [9]. In BBF one maintains a sorted queue of nodes that have been visited and expands the bins that are closer to query point first. Further, we use the randomized KD-tree, where a set of KD-trees are created and queried instead of a single tree. In each random KD-tree, the data points are rotated randomly, so that the choice of axes affects the resulting points less. At query time, the same rotation is applied to the query point before searching each tree. The union of the points returned by all KD-trees is the candidate list. Similar to LSH, the best nearest neighbors are selected using linear search in the candidate list.

3. METHODOLOGY

The atmospheric boundary layer is the lowest part of the atmosphere where most of the interactions between surface and atmosphere take place. Knowledge of the processes and mechanisms in this layer is essential in meteorology and climate science. The height of the boundary layer, or mixing layer height (MLH), is an important parameter; it affects, for example, near-surface air quality, since it limits the volume of air into which pollutants are emitted, mixed and dispersed, and is therefore crucial in modelling pollution, smog and dispersion of greenhouse gases. Since the height of the mixed layer is determined by surface fluxes that drive turbulent processes (Stull, 1988), it is one of the parameters that can be used to test model representation of the energy balance against observations. MLH observations, therefore, are of key importance when testing models for a realistic representation of the atmosphere ranging from short timescales (weather forecasting) to long timescales (climate change) [1].

Here, we describe the new algorithm, pathfinder, which is used to track the development of the MLH during the day, based solely on single wavelength backscatter lidar data. The path finder algorithm is based on graph theory and the algorithm published by K-Nearest Neighbor graph for finding the shortest path in graphs, and therefore inherently takes temporal development into account.

The problem of finding shortest distance between source and destination are formulated as follows:

$$minimize \sum_{e_{ij} \in E} c_{ij} f_{ij} \qquad (1)$$

subject to
Flow Conservation Constraints

$$\sum_{v_j \in V} f_{ij} - \sum_{v_k \in V} f_{ki} = \begin{cases} D, & i = s \\ 0, & i \neq s \; or \; t \;, \forall v_i \in V \\ -D, & i = t \end{cases} \qquad (2)$$

Capacity Constraints

$$f_{ij} \leq u_{ij}, \forall e_{ij} \in E \qquad (3)$$

Other Link Constraints

$$\frac{1}{D} w_{ij}^{l} f_{ij} \leq l, \forall e_{ij} \in E, l \in \bar{l} \qquad (4)$$

Path Constraints

$$\frac{1}{D} \sum_{e_{ij} \in E} w_{ij}^{p} f_{ij} \leq p, \forall p \in \bar{p} \qquad (5)$$

Existential Constraints

$$f_{ij} \in \{0, D\}, \forall e_{ij} \in E, D > 0. \qquad (6)$$

Average path length is a concept in network topology that is defined as the average number of steps along the shortest paths for all possible pairs of network nodes. It is a measure of the efficiency of information or mass transport on a network. Average path length is one of the three most robust measures of network topology, along with its clustering coefficient and its degree distribution. Some examples are: the average number of clicks which will lead you from one website to another or the number of people you will have to communicate through, on an average, to contact a complete stranger. It should not be confused with the diameter of the network, which is defined as the longest geodesic, i.e., the longest shortest path between any two nodes in the network.

In the final step, KNN shortest path algorithm is applied to select the optimal MLH path. This method efficiently determines the path with the lowest total cost originating from a specific vertex in the first time step to one of the vertices in the last time step satisfying the above-mentioned conditions. Theoretically, any data set of arbitrary length can be translated into a graph and analyzed simultaneously. However, even with the relative efficiency of KNN algorithm, this is unpractical and would lead to long processing times [8]. Therefore, it was decided to split data into multiple time windows and apply the method to these windows separately. This improves the processing times substantially, making the method available for near-real-time MLH tracking.

4. RESULTS AND DISCUSSION

Our experiments indicate that our distance functions outperform previously used alternatives in identifying true neighbors in real-world biological data. We also demonstrate that our algorithms scale for graphs with tens of millions of edges. Locations of strong gradients are connected under the condition that subsequent points on the path are limited to a restricted vertical range. The search is further guided by rules based on the presence of clouds and residual layers. After being applied to backscatter Lidar data from Cabauw, excellent agreement is found with wind profiler retrievals for a 12-day period in 2008 and visual judgment of Lidar data during a full year in 2010 (R2 D0.96). These values compare favorably to other MLHL methods applied to the same Lidar data set and corroborate more consistent MLH tracking by pathfinder. 'Neighborhoods Method' (NM) i.e. Nearest Neighbor Graph theory uses novel search space reduction techniques and has a theoretical quadratic speed-up making it practically faster (by an order of magnitude) than recent branch-and-bound exhaustive search solutions. This graph theory helps to find the shortest path efficiently. The results of Dijkstra's shortest path algorithm and K-Nearest Neighbor Graph theory are analyzed in table 1 and figure 1.

Method	Accuracy		
	K=1	K=2	K=3
Dijkstra's	89%	85%	90%
KNN - G	93%	97%	99%

Table No: 1 Testing the Accuracy

5. CONCLUSION

The pathfinder algorithm stores a full day of Lidar measurements arranged in a time–altitude matrix and subsequently divides the matrix into time windows of 15 min. These 15 min blocks are translated into graphs in which each individual data point represents a vertex. To estimate MLH exactly one altitude has to be selected in each time step. For the selection a certain cost is assigned to each vertex, which is inversely proportional to the gradient at the point in the graph. This way, the path with the lowest total cost will contain the maximum sum of strong gradients and will be a good estimate for the MLH. This method evaluates multiple time steps within a configurable time window simultaneously. Graph theory is applied together with KNN shortest path algorithm to imitate the continuous character of the MLH.

REFERENCES

[1] Angevine, W. M., White, A. B., and Avery, S. K.: Boundary layer depth and entrainment zone characterization with a boundary-layer profiler, Bound.-Lay. Meteorol., 68, 375–385, doi:10.1007/BF00706797, 1994.

[2] Apituley, A., Russchenberg, H., van der Marel, H., Boers, R., ten Brink, H., de Leeuw, G., Uijlenhoet, R., Arbresser-Rastburg, B., and Röckmann, T.: Overview Of Research And Networking With Ground Based Remote Sensing For Atmospheric Profiling At The Cabauw Experimental Site For Atmospheric Research (CESAR) – The Netherlands, in: Proceedings IGARSS 2008, Boston, Massachusetts, III, 903–906, 2008.

[3] Baars, H., Ansmann, A., Engelmann, R., and Althausen, D.: Continuous monitoring of the boundary-layer top with lidar, Atmos. Chem. Phys., 8, 7281–7296, doi:10.5194/acp-8-7281-2008, 2008.

[4] Baltink, H. K.: CESAR-database, available at: http://www.cesar-database.nl (last access: 25 May 2017), 2016.

[5] Beyrich, F.: Mixing-height estimation in the convective boundary layer using sodar data, Bound.-Lay. Meteorol., 74, 1–18, doi:10.1007/BF00715708, 1995.

[6] de Haij, M., Wauben, W., and Baltink, H. K.: Continuous mixing layer height determination using the LD-40 ceilometer: a feasibility study, Scientific report 2007-01, KNMI, De Bilt, 102 pp.,2007.

[7] Dijkstra, E. W.: A note on two problems in connexion with graphs, Numer. Math., 1, 269–271, doi:10.1007/BF01386390, 1959.

[8] An optimal algorithm for approximate nearest neighbor searching in fixed dimensions. J. ACM, 45:891–923, 1998.

[9] M. Bern. Approximate closest-point queries in high dimensions.Inf. Process. Lett., 45(2):95–99, 1993.