Real-Time Traffic Updates in Moving Objects Databases

Goce Trajcevski, Ouri Wolfson,* Bo Xu and Peter Nelson
Dept. of Computer Science (M/c 152)
University of Illinois at Chicago
851 S. Morgan, Chicago, Il 60607
gtrajcev,wolfson,bxu,nelson@cs.uic.edu

Abstract

This work addresses the problem of updating Moving Objects Databases (MOD) using real-time traffice information. The motion of the object is represented by a trajectory, *which can be constructed using the available electronic maps and the information about traffic patterns. However, the statistical information about traffic patterns may change due to accidents, extreme weather conditions, road work, etc... In this work, we present a model for* updating *the trajectories of moving objects when unexpected traffic conditions occur.*

Many sites provide an up-to-date information about traffic conditions on major expressways. However, the unexpected traffic conditions may affect not only the vehicles on the expressways for which an on-line monitoring is provided, but may also have an effect on the streets near those expressways. We propose a model of this spill-over effect *and we utilize it in the process of* identifying *the trajectories which are affected by the abnormal traffic.*

Keywords: Moving Objects Databases, Updates, Real-Time Traffic

1. Introduction and Motivation

When constructing a *trajectory* (c.f. Section 2.) of a moving object, one needs to take into consideration different speed along a particular street-block/expressway, during, say, a day. The values of the speed along a given segment during different time periods (e.g. rush hour vs. late night) may be obtained, for example, by monitoring the speed patterns. We call this attribute a speed profile of a segment.

*Research supported by ARL Grant DAAL01-96-2-0003, NSF Grants ITR-0086144, CCR-9816633, CCR-9803974, IRI-9712967, EIA-0000516, INT-9812325

However, even if the speed profiles are considered, one may still experience unexpected variations due to accidents; road-works; bad weather; etc. In this case, some sort of a real-time information needs to be utilized by the Moving Objects Database (MOD) to update the motion plans of the objects whose trajectory is affected by the unexpected traffic conditions. Otherwise, the (location-time) information stored in the MOD will be inaccurate. Specifically, the answer to the query *"which objects will arrive at the destination before 4PM"*, will be incorrect.

Real-time traffic information is available for many metropolitan cities. The University of Illinois at Chicago *(www.ai.uic.edu)* maintains the information about traffic patterns on the major expressways around Chicagoland (I-55; I-290; I-90/94; I-294) which is updated every 2 minutes[1]. The sources of information are traffic sensors/ detectors which are mostly located on highways (e.g. toll booths) and intersections of major streets.

This kind of information can be used to detect an occurence of abnormal traffic conditions on given sections of a road network. In order to utilize this information, the MOD needs to: 1.) *Identify* the trajectories which are affected by the abnormal traffic; and 2.) *Update* the *(location-time)* information about their motion plans.

It is a common tendency that, upon an unexpected congestion on a particular route, some drivers choose an alternate route which is close to the initial (congested) one. This, in turn, will increase the traffic density on some neighbouring street – which will cause some of the drivers there to move a bit further away from the point where the abnormal traffic originated. The effect "recursively" repeats itself until, at some distance from the point of the abnormal traffic, the density is at its "normal" value for the particular time period (i.e. the effects of the abnormal traffic are negligible). We call this a *spill-over* effect of the abnormal traffic.

[1]NTUA maintains the real-time traffic information for Athens. The *Intelligent Transportation Systems* (ITS) *(www.itsonline.com)* has links to maps with the real-time traffic update for over 25 major cities throughout the world.

Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)

If a MOD is to be properly modified upon the occurence of the abnormal traffic conditions, the spill-over effect can not be neglected. Most of the reported approaches on re-routing of arterial traffic upon abnormal conditions [5, 9] are based on the results from queuing theory, analyzing different distributions for the arrival rate. However, to the best of our knowledge, the spill-over effects have not been considered when identifying the trajectories which are affected by abnormal traffic in MOD.

The main contributions of this work are as follows:

* We define a model of a *traffic incident* to represent the effects of abnormal traffic conditions and we demonstrate how to update a trajectory which is affected by the abnormal traffic.

* We also present a model for a spill-over effect and we demonstrate how to identify the trajectories which need to be updated.

The rest of the paper is structured as follows. In Section 2. we outline the model of a trajectory and its construction. Section 3 introduces the model of a traffic incident and shows how to update a particular trajectory which is affected by the abnormal traffic. Section 4. presents the spill-over effect and discusses the identification of affected trajectories in a MOD. In Section 5. we position our work with respect to the relevant literature, summarize and outline directions for future work.

2. Representing the Motion of a Moving Object

Commonly, a moving object is represented as a point and its motion is represented as some *(location,time)* information (c.f. [12, 21]):

• *A* trajectory *of a moving object is a piece-wise linear function $f : T \rightarrow (x, y)$, represented as a sequence of points (x_1, y_1, t_1), (x_2, y_2, t_2), ..., (x_n, y_n, t_n) $(t_1 < t_2 < ... < t_n)$. For a given a trajectory Tr, its projection on the X-Y plane is called the* route *of Tr.*

Thus, the object is at (x_i, y_i) at time t_i, and during each segment $[t_i, t_{i+1}]$, the object moves along a straight line from (x_i, y_i) to (x_{i+1}, y_{i+1}), and at a constant speed. The expected location of the object at any time $t \in [t_i, t_{i+1}]$ $(1 \leq i < n)$ is obtained by a linear interpolation between (x_i, y_i) and (x_{i+1}, y_{i+1}). An illustration of trajectory and its route is shown in Figure 1.

Note that a trajectory can represent both the past and future motion of objects. Given the *start_point* and *start_time*, and the *destination_point* (plus, possibly, a sequence of other "to-be-visited" points), the trajectory consists of a set of points which is a superset of the given ("to-be-visited") points. This corresponds to a *motion plan* of the moving object and it is constructed based on the assumption that in between the points, the object will move along the shortest path. In order to explain how we construct the trajectory, we need to define an *electronic map* (map):

• *A* map *is a graph, represented as a relation where each tuple corresponds to a block with the following attributes:*
– Polyline: *Each block is a polygonal line segment. Polyline gives the sequence of the endpoints: (x_1, y_1), (x_2, y_2), ..., (x_n, y_n)*
– Length: *Length of the block.*
– Fid: *The block* id *number.*
– Speed_Profile: *Typical drive time from one end of the block to the other, in minutes, for each period of a day..*
Plus, among others, a set of geo-coding attributes which enable translating between an (x,y) *coordinate and an address, such as* "1030 North State St."*:*
(e.g. – L_f_add: *Left side from street number).*

Such maps are provided by, among the others, Geographic Data Technology[2] Co. An intersection of two streets is the endpoint of the four block-polylines. Thus, each map is an undirected graph, with the tuples representing edges of the graph.

An external routine, available in most Geographic Information Systems, (given a priori) computes the shortest cost (distance or travel-time) path in the map graph. Given the *start_time*, we compute the trajectory by computing for each straight line segment the time at which the object will arrive to the end point of the segment. For this purpose, the only relevant attributes are *Polyline* and *Speed_profile*.

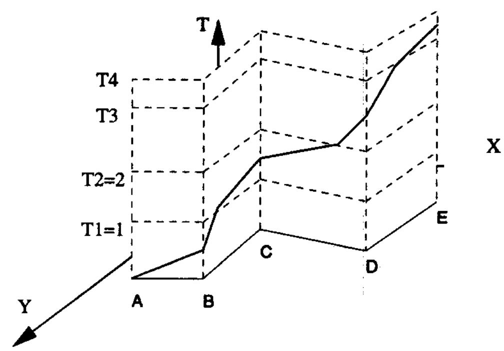


Figure 1. Trajectory and its construction based on a route and speed profiles

Note that we need to use a *time-dependent shortest path* algorithm, as presented in [2]. This is, essentially, an A^* [10] based extension of Dijkstra's algorithm, where the cost of an edge in a graph depends on the start time to travel along that edge (details are given in [20]). As a consequence, it may be a case that an object traveling along a line segment will "enter" two different speed-profile peri-

[2] *(www.geographic.com)* Note that the GDT maps do not provide a *speed_profile* but they provide a single *Drive_time* attribute. However, as we explained, the values of *Speed_profile* can be obtained by monitoring the traffic.

Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)
1529-4188/02 $17.00 © 2002 **IEEE**

ods. Thus, although the route consists of one segment, the trajectory between the route end-points may have more than one segment. This is illustrated in Figure 1. Observe that while traveling along the route segments $\overline{BC}$ and $\overline{DE}$, the object enters two different time periods for the speed profile.

Finally, let us observe that a trajectory can be constructed based on past motion. Specifically, consider a set of 3D points $(x_1, y_1, t_1), (x_2, y_2, t_2), \ldots, (x_n, y_n, t_n)$ which were transmitted by a moving object periodically, during its past motion (e.g. using an on-board GPS to detect location at given time points). One can construct a trajectory by first "snapping" the points on the road network, then simply connecting the snapped points with the shortest path on the map.

3. Updating a Trajectory under Real – Time Traffic

In this section, we analyze the update of a particular trajectory due to abnormal traffic conditions. This occurs, for example, when there are accidents or road works along certain segments of a polyline in a given block(s). The effect of the abnormal traffic is that the speed in a particular block, say i, does not match the one for the j-th time period (i.e. $v_{current} = v'_{i,j} \neq v_{i,j}$).

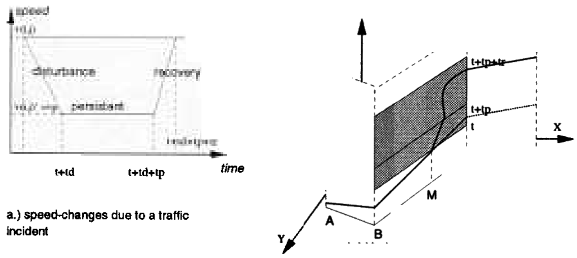


Figure 2. Effects of abnormal traffic conditions

As illustrated on part a.) of Figure 2, a *traffic incident* which occurs at time t has three parts:

• disturbance interval – *between* t *and* $t + t_d$*, during which the speed drops from its normal value* $v_{i,j}$ *to some value* v_p.

• persistant interval – *between* $t + t_d$ *and* $t + t_d + t_p$*, during which the speed is constant (* $= v_p \leq v_{i,j}$*).* and

• recovery interval – *during which the speed increases from* v_p *to its normal value for the current segment and the given time interval.*

We assume that the object is having a constant acceleration during the disturbance ($a_d = (v_p - v_{i,j})/t_p$) and the recovery ($a_r = (v_{i,j} - v_p)/t_r$) intervals. The values of each of t_r, t_p, t_d, as well as v_p depend on the type of the abnormality (e.g. light accident, heavy accident, road-work, etc ...) and are obtained from historic profiles. The site which monitors the traffic condition transmits the type of the abnormality to the MOD server.

Each of the intervals is illustrated on part a.) of Figure 2. Due to lack of space, we do not present all the formal details here (see [20]), however, we'd like to point out that the abnormality may spread over > 1 segment, in which case we need to modify (see below) all the affected segments of a given trajectory.

Note that, in practice, not all of the intervals may be significant. For example, an accident which blocks several lanes of the road happens instantaneously, persists until the emergency/ police vehicles arrive and the tow trucks clear the scene, and recovers gradually to the normal conditions,

Figure 2, part b.), illustrates the modifications of an affected trajectory for a scenario which corresponds 0-disturbance interval. The accident occured at time t, when the object was at location M along the segment $\overline{BC}$. The dotted polyline after M illustrates what would have been the normal trajectory of the moving object throughout the segments $\overline{MC}$ and $\overline{CD}$. Note how the object moves slower than normal during the persistent interval. During the recovery interval, the object accelerates, which results in an (inverse) parabola shape in the vertical plane. If the trajectory enters the affected traffic at point $M(x_M, y_M, t)$, then for its distance from M at time t_c we have: $d(t_c) = v \cdot (t_c - t)$ during the persistent and $d(t_c) = v \cdot (t_c - t) + a \cdot (t_c - t - t_p)^2/2$ during the recovery interval. In order to comply with the definition of a trajectory (c.f. Section 2.) we approximate the parabolic 3D shape of the recovery interval, with a straight 3D line segment between its end-points.

4. Detecting the Trajectories which are Affected by an Abnormal Traffic

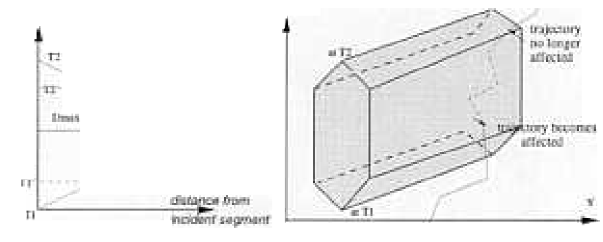
Given the road segments in which an incident occured at time t, we need to identify the trajectories which need to be updated. In order to determine if a given trajectory is affected by the incident, we need to check (details are given in [20]) if its route coincides with some of the line segments of the incident, sometime[3] between t and $t + t_d + t_p + t_r$.

However, recall that (c.f. Section 1.) when the traffic density unexpectedly changes to some $\rho_{abnormal}$ due to an incident, some of the vehicles will move to the near-by routes, causing the *spill_over* effect. Hence, even though a particular object's route did not include the segment where the abnormal traffic occurred, its trajectory may still need to be updated. Since the traffic sensors/detectors are mostly

[3] One can use a 3D index for the filtering stage – so far, the STR tree [12] and the Octree- based one in [17] are closest to our model of trajectory).

Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA'02)
1529-4188/02 $17.00 © 2002 IEEE

available on expressways and major streets, we need a model to grasp the consequences of the traffic spill-over and identify all the trajectories which need to be updated.



a.) spreading of abnormal traffic on neighbouring streets.

b.) spill-over effect on a trajectory

Figure 3. Spill-over effects of abnormal traffic conditions

Figure 3 presents our model of the spreading of abnormal traffic conditions. Part a.) illustrates how far the traffic incident along a certain segment is spreading to the neighboring streets, as a function of time (we "reversed" the axes for compatibility with part b.) – the time is on the vertical axis). T_1' is the time when the spill_over reaches its maximum distance – D_{max}.

Now, as part b.) of Figure 3 illustrates, in order to check if a particular trajectory is affected by abnormal traffic conditions, it suffices to check if the trajectory intersects the 3D spill_over body (again, a 3D index can be utilized).

Let D_{max} denote the maximal distance from the incident segment which is affected during the time of the abnormal traffic conditions. Further, let i be the segment where the incident occurs, during the j-th period of the speed profile. If $v_{i,j}^a$ is the speed during the persistent time of the abnormal traffic, let $\delta = v_{i,j}^a / v_{i,j}$ denote the ratio of the abnormal vs. normal traffic. The spill-over effect, is propagated as a linear function of the distance from the segment i. If k is a segment which is at distance D_k from the segment i, with a normal speed $v_{k,j}$ during the j-th time interval of the speed profile, we have the following equation:

$$\frac{v_{k,j}^a}{v_{k,j}} = \frac{(D_{max} - D_k) \cdot \delta + D_k}{D_{max}}$$

The equation simply states that the closer certain street (k) is to the incident segment (i), the closer its *abnormal* vs. *normal* speed ratio is to the one on the incident segment itself (for the $j-$th interval of the speed profile). At the distance D_{max}, there is no abnormality (i.e. the ratio of *abnormal* vs. *normal* speed is 1).

5. Related Work and Concluding Remarks

The database community has been very active in researching various aspects of interest for the MOD: 1.) *Modeling and linguistic issues*: [16] introduced the MOST model for representing moving objects (similar to [15]) as a function of (location, velocity_vector). The underlying query language is nonstandard, based on the Future Temporal Logic (FTL). Similar issues are tackled in [23]. A trajectory model similar to ours is given in [24] and the work presents new operators for special cases of spatio-temporal range queries. The series of works [3, 4, 6] address the issues of modeling and querying moving objects by presenting a rich algebra of operators and a comprehensive framework of data types; 2.) *Uncertainty*: A formal quantitative approach to the uncertainty when modeling moving objects is presented in [11]. [25, 26] introduce a cost based approach to determine the size of the uncertainty for optimizing the communication cost and query imprecision. A model of uncertainty and its implication on the spatio-temporal queries is given in [21]; 3.) *Indexing*: A lot of work has been done on selecting an appropiate index for certain types of queries both in primal [12, 15, 13, 14, 17] and in dual space [1, 7, 8]. [18, 19] present specifications of *what* an indexing in MOD should consider.

However, to the best of our knowledge, none of the works has addressed the issue of identifying and updating the trajectories which are affected by abnormal traffic conditions, utilizing the available sources of real-time traffic information.

We presented a model of a trajectory and described its construction based on electronic map and statistical information about the speed profiles. In order to keep the trajectory up to date in the MOD server, we showed how to "react" to the abnormal changes in traffic conditions. We gave a model of how to update a given trajectory which is affected by abnormal traffic. Since the effects of abnormal traffic may not be localized on the road segment where an abnormality has occured, we presented a model of a spill-over effect. We also showed how to identify and update the all trajectories which are affected by the spill-over effect. Our results have been partially implemented in our (ongoing) DOMINO project [22].

Currently, we are investigating how to utilize triggers for updating the MOD upon a detection of abnormal traffic, and we are experimenting wigh indexing methods to be used in the filtering stage of detecting the affected trajectories. Another extension of our work, which is a long term goal, is the optimization of spatio-temporal query processing. Namely, in the presence of user-defined functions, the processing cost of the refinement stage of a given query is not negligible.

References

[1] A. K. Agarwal, L. Arge, and J. Erickson. Indexing moving points. In *19th ACM PODS Conference*, 2000.

[2] S. E. Dreyfus. An appraisal of some shortest – path algorithms. *Operations Research*, 17(3), 1969.

[3] M. Erwig, M. Schneider, and R. H. Guting. Temporal and spatio – temporal datasets and their expressive power. Technical Report 225-12/1997, Informatik berichte, 1997.

[4] L. Forlizzi, R. H. Gutting, E. Nardelli, and M. Schneider. A data model and data structures for moving objects databases. In *ACM SIGMOD*, 2000.

[5] D. Getmann, K. H. Head, and P. B. Mirelmandi. The milos integrated freeway – surface street control strategy. INFORMS Online, 1997. Institute for Operation REsearch and MAnagement Sciences.

[6] R. H. Guting, M. H. Bohlen, M. Erwig, C. Jensen, N. Lorentzos, M. Schneider, and M. Vazirgiannis. A foundation for representing and queirying moving objects. *ACM TODS*, 2000.

[7] D. Kollios, D. Gunopulos, and V. J. Tsotras. On indexing mobile objects. In *18th ACM PODS Conference*, 1999.

[8] G. Kollios, D. Gunopulos, and V. J. Tsotras. Nearest neighbour queries in a mobile environment. In *STDBM*, 1999.

[9] T. Maze and A. Kamyab. Simulation and analysis of arterial traffic operations along us 61 corridor in burlington iowa. Center for Transporation Research and Education, Iowa State University, 1998. Final Report.

[10] N. J. Nilsson. *Problem – Solving methods in Artificial Intelligence*. McGraw Hill, 1971.

[11] D. Pfoser and C. Jensen. Capturing the uncertainty of moving objects representation. In *SSDB*, 1999.

[12] D. Pfoser, Y. Theodoridis, and C. Jensen. Novel approaches in query processing for moving objects databases. In *VLDB*, 2000.

[13] S. Saltenis and C. Jensen. R-tree based indexing of general spatio-temporal data. Technical Report TR-45, TimeCenter, 1999.

[14] S. Saltenis and C. Jensen. Indexing of moving objects for location-based services. Technical Report TR-63, TimeCenter, 2001.

[15] S. Saltenis, C. S. Jensen, S. T. Leutenegger, and M. A. Lopez. Indexing the positions of continuously moving objects. Technical Report TR - 44, TimeCenter, 1999.

[16] A. P. Sistla, O. Wolfson, S. Chamberlain, and S. Dao. Modeling and querying moving objects. In *13th Int'l Conf. on Data Engineering (ICDE)*, 1997.

[17] J. Tayeb, O. Ulusoy, and O. Wolfson. A quadtree – based dynamic attribute indexing method. *The Computer Journal*, 41(3), 1998.

[18] Y. Theodoridis, T. Sellis, A. N. Papadopoulos, and Y. Manolopoulos. Specifications for efficient indexing in spatiotemporal databases. In *IEEE SSDBM*, 1999.

[19] Y. Theodoridis, J. R. O. Silva, and M. A. Nascimento. On the generation of spatiotemporal datasets. In *6th Int'l symposium on Large Spatial Datasets*, 1999.

[20] G. Trajcevski, O. Wolfson, and B. Xu. Modeling and querying trajectories of moving objects with uncertainty. Technical Report UIC - EECS - 01 - 2, Dept. of EECS, Univ. of Illinois at Chicago, May 2001.

[21] G. Trajcevski, O. Wolfson, F. Zhang, and S. Chamberlain. The geometry of uncertainty in moving objects databases. In *EDBT*, 2002.

[22] G. Trajecevski, O. Wolfson, C. Hu, H. Lin, and F. Zhang. Managing uncertain trajectories of moving objects with domino. In *ICEIS*, 2002.

[23] M. Vazirgiannis, Y. Theodoridis, and T. Sellis. Spatiotemporal composition and indexing for large multimedia applications. *Multimedia systems Journal*, 6(4), 1998.

[24] M. Vazirgiannis and O. Wolfson. A spatiotemporal model and language for movign objects on road networks. In *MOBIDE*, 2001.

[25] O. Wolfson, S. Chamberlain, S. Dao, L. Jiang, and G. Mendez. Cost and imprecision in modeling the position of moving objects. In *14 Int'l Conf. on Data Engineering (ICDE)*, 1998.

[26] O. Wolfson, A. P. Sistla, S. Chamberlain, and Y. Yesha. Updating and querying databases that track mobile units. *Distributed and Parallel Databases*, 7, 1999.

IEEE COMPUTER SOCIETY