# The Truncated Tornado in TMBB: A Spatiotemporal Uncertainty Model for Moving Objects

Shayma Alkobaisi[1], Petr Vojtěchovský[2], Wan D. Bae[3],
Seon Ho Kim[4], Scott T. Leutenegger[4]

[1] College of Information Technology, UAE University, UAE
{shayma.alkobaisi}@uaeu.ac.ae,
[2] Department of Mathematics, University of Denver, USA
{petr}@math.du.edu,
[3] Department of Mathematics, Statistics and Computer Science,
University of Wisconsin-Stout, USA
{baew}@uwstout.edu,
[4] Department of Computer Science, University of Denver, USA
{seonkim, leut}@cs.du.edu

**Abstract.** The uncertainty management problem is one of the key issues associated with moving objects ($MO$s). Minimizing the uncertainty region size can increase both query accuracy and system performance. In this paper, we propose an uncertainty model called the $Truncated$ $Tornado$ model as a significant advance in minimizing uncertainty region sizes. The $Truncated$ $Tornado$ model removes uncertainty region sub-areas that are unreachable due to the maximum velocity and acceleration of the $MO$s. To make indexing of the uncertainty regions more tractable we utilize an approximation technique called $Tilted$ $Minimum$ $Bounding$ $Box$ ($TMBB$) approximation. Through experimental evaluations we show that $Truncated$ $Tornado$ in $TMBB$ results in orders of magnitude reduction in volume compared to a recently proposed model called the $Tornado$ model and to the standard "Cone" model when approximated by axis-parallel $MBB$.

## 1 Introduction

In recent years, there is an increasing number of location-aware spatiotemporal applications that manage continuously changing data. Tracking systems, mobile services and sensor-based systems now track millions of GPS and RFIDs that can report the positions of the moving objects. These applications require new strategies for modeling, updating and querying spatiotemporal databases.

To be able to answer location-based queries, it is necessary to maintain the locations of a large number of moving objects over time. It is infeasible to store the object's exact continuously changing location since this would require more updates than can be managed by the $MO$ database. This is a first cause of $MO$ location inaccuracy. A second cause is that devices are limited in ability to report accurate locations. As a result of these inaccuracies, $MO$ spatiotemporal data often require uncertainty management algorithms.

A common model of spatiotemporal query processing is to divide the query into a filtering step and a refinement step [3]. The performance of the filtering step is improved when there is a lower rate of false-hits, i.e., objects that are returned in the filtering step but subsequently are discarded by the refinement step. In spatiotemporal queries, minimizing the size of uncertainty regions and any used region approximations will result in improving the filtering step efficiency. Minimizing these regions and their approximations is the main goal of this work.

## 2 Related Work

Many uncertainty models for moving objects have been proposed based on the underlying applications. Uncertainty regions of moving objects in [10] are presented in 3D as cylindrical bodies which represent all the possible positions between two reported past locations. The authors in [7] proposed one of the most common uncertainty models showing that when the maximum velocity of an object is known, the uncertainty region between any two reported locations can be represented as an error ellipse. Another popular model is found in [5]. It represents the uncertainty region as an intersection of two half cones. Each cone constrains the maximum deviation from two known locations in one movement direction. Recently in [12], a non-linear extension of the funnel model [11], named $Tornado$ was presented. This higher degree model reduces the size of the uncertainty region by taking into account higher order derivatives, such as velocity and acceleration.

There is a lack of research in investigating the effect of different object approximations on the false-hit rate. In [3], the authors investigated six different types of static spatial objects approximations. Their results indicated that depending on the complexity of the objects and the type of queries, the approximations five-corner, ellipse and rotated bounding box outperform the axis-parallel bounding box. The authors in [1] presented $MBR$ approximations for three uncertainty region models, namely, the $Cylinder\ Model$, the $Funnel\ Model$ of $Degree$ 1 which is the $Cone$ model proposed in [5] and the $Funnel\ Model$ of $Degree$ 2 which is the $Tornado$ model presented in [12].

Research in the field of computational geometry has resulted in several object approximation solutions. In [9], the author was able to use the fact that a minimal area rectangle circumscribing a convex polygon has at least one side flush with an edge of the polygon to use the “rotating calipers” algorithm to find all minimal rectangles in linear time. In [6], O’Rourke presented the only algorithm for computing the exact arbitrarily-oriented minimum volume bounding box of a set of points in $R^3$ which runs in $\mathcal{O}(n^3)$. The authors in [2], proposed an efficient solution of calculating a $(1 + \epsilon)$-approximation of the non axis-parallel minimum-volume bounding box of $n$ points in $R^3$. The running time of their algorithm is $\mathcal{O}(nlogn + n/\epsilon^3)$.

## 3 The Truncated Tornado

| Notation | Meaning |
|---|---|
| $P_1$ | a reported position of a moving object |
| $P_2$ | a following reported position |
| $t_1$ | time instance when $P_1$ was reported |
| $t_2$ | time instance when $P_2$ was reported |
| $t$ | any time instance between $t_1$ and $t_2$ inclusively |
| $T$ | time interval between $t_2$ and $t_1$, $T = t_2 - t_1$ |
| $V_1$ | velocity vector at $P_1$ |
| $V_2$ | velocity vector at $P_2$ |
| $e$ | instrument and measurement error |
| $M_v$ | maximum velocity of an object |
| $M_a$ | maximum acceleration of an object |
| $t_{Mv}$ | time to reach maximum velocity |

**Table 1.** Notations used in this paper

Assuming maximum velocity $M_v$ and maximum acceleration $M_a$ of the moving object, the *Tornado* model [12] calculated the uncertainty region defining functions as shown in Fig. 1 (a). The dotted region is the uncertainty region defined by the *Tornado* model and the top and bottom intervals represent the instrument and measurement error associated with each reported position.

Let $displ_1$ and $displ_2$ be, respectively, a first-degree and second-degree displacement functions defined as follows:
$displ_1(V,t) = V \cdot t$ and $displ_2(V,a,t) = \int_0^t (V + a \cdot x)dx \approx V \cdot t + (a/2) \cdot t^2$, where $V$ is the current velocity of the moving object, $a$ is acceleration and $t$ is time. The future (past) position $f_{pos}$ ($p_{pos}$) of a moving object after (before) some time, $t$, can be calculated as follows:

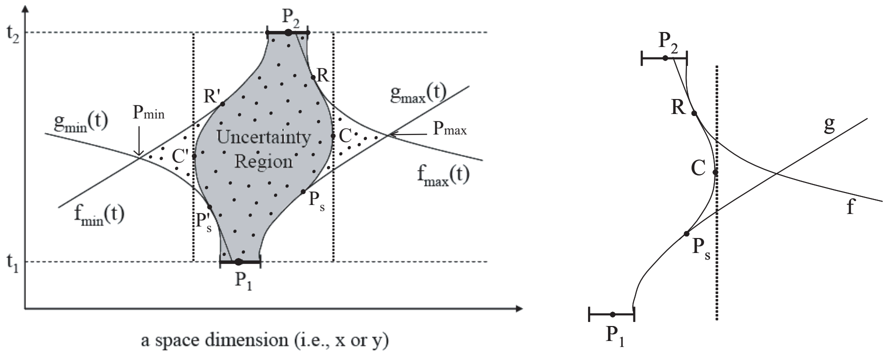
$$f\_pos(P_1, V_1, M_v, M_a, t) = \begin{cases} P_1 + D_2 + D_1 & \text{if } t_{Mv} < t \\ P_1 + D_2 & \text{otherwise} \end{cases}$$

$$p\_pos(P_2, V_2, M_v, M_a, t) = \begin{cases} P_2 - D_2 - D_1 & \text{if } t_{Mv} < t \\ P_2 - D_2 & \text{otherwise} \end{cases}$$

$D_1 = displ_1(M_v, t - t_{Mv})$ and $D_2 = displ_2(V, M_a, t_{Mv})$, where $t_{Mv}$ is the time the moving object needs to reach $M_v$. Notice that the above functions define the dotted uncertainty region in Fig. 1 (a). $g_{min}(t)$ and $g_{max}(t)$ are produced by the $f_{pos}$ function, and $f_{min}(t)$ and $f_{max}(t)$ are produced by the $p_{pos}$ function.

Objects moving with momentum cannot make extreme changes in their velocity. Hence they need some time to change their velocities from one direction to the opposite direction, thus, the right and left corners $P_{max}$ and $P_{min}$ shown in Fig. 1 (a) are impossible to be reached by the moving object unless we assume infinite acceleration. Our proposed *Truncated Tornado* model removes unreachable sub-areas of the uncertainty regions by calculating the furthest point an object can reach given its maximum acceleration.

Assuming that the two intervals and trajectories are as shown in Fig. 1 (b), we define the *Truncated Tornado* model as follows:



(a) The *Tornado* and *Truncated Tornado* models (b) Point of no safe return

**Fig. 1.** Calculating uncertainty regions of *Truncated Tornado*

We say that $P_s$ is the *point of no safe return* for $g$ if $P_s$ is the rightmost point on the trajectory $g$ such that when the object (car) starts changing direction at $P_s$ then it will touch the trajectory $f$ (at point $R$), i.e., the object is within the boundary of the maximum possible deviation.

Since any realizable trajectory between the two intervals must remain within the boundary defined by $f$ and $g$, it is clear that the point $C$ (which is the rightmost point on the decelerating trajectory started at $P_s$) can be used as a cut point for the right boundary of the $MBR$ encompassing the uncertainty region. This boundary is indicated in the figure by the dotted line.

The question is how to calculate the points $P_s$ and $C$. We show the case when both $f$, $g$ are parabolas. Upon turning the situation by 90 degrees counterclockwise, $f$, $g$ are parabolas given by $f(x) = ax^2 + b_1x + c_1$, $g(x) = ax^2 + b_2x + c_2$. (We use the same quadratic coefficient $a$ since the maximal acceleration $M_a$ is the same for $f$ and $g$.) Note that $b_1 \neq b_2$ since the parabolas $f$, $g$ are not nested.

Let $x_0$ be the $x$-coordinate of the point $P_s$. The deceleration trajectory started at $P_s$ is a parabola, and it can be given by $h(x) = -ax^2+ux+v$. To determine $u$, $v$ and $x_0$, we want $h$ to stay below $f$ at all times and hence $-ax^2+ux+v \leq ax^2 + b_1x + c_1$ for every $x$. Equivalently, $k(x) = 2ax^2 + (b_1 - u)x + (c_1 - v) \geq 0$ for every $x$. Since $h$ needs to touch $f$, we want $k$ to be a parabola that touches the $x$-axis. Equivalently, the discriminant $(b_1 - u)^2 - 4(2a)(c_1 - v)$ needs to be equal to 0. This yields

$$v = c_1 - (b_1 - u)^2/(8a) \tag{1}$$

Analogously, we want $h$ to stay below $g$ at all times and hence $-ax^2+ux+v \leq ax^2 + b_2x + c_2$ for every $x$. Equivalently, $k(x) = 2ax^2 + (b_2 - u)x + (c_2 - v) \geq 0$ for every $x$. Since $h$ needs to touch $g$, we want $k$ to be a parabola that touches the $x$-axis. Equivalently, the discriminant $(b_2 - u)^2 - 4(2a)(c_2 - v)$ needs to be equal to 0. This yields

$$v = c_2 - (b_2 - u)^2/(8a) \tag{2}$$

The parabola $h$ must satisfy both (1) and (2), therefore, we can set (1) = (2), eliminate $v$ from the equation, solve for $u$ and then substitute to find $v$. Solving for $u$ with the observation that $b_1 \neq b_2$ we get

$$u = 4a\frac{c_2 - c_1}{b_1 - b_2} + \frac{1}{2}(b_1 + b_2) \tag{3}$$

It is now easy to find the cut point $C$, as this is the vertex of the parabola $h$. Notice that we only need to calculate $u$ to find $C$ since $C = \frac{u}{2a}$.

Finally, the reverse time problem (going from $P_2$ to $P_1$) is precisely the forward time problem: we are looking for a parabola that stays below and touches both $f$ and $g$, hence the reverse time parabola coincides with the forward time parabola.

The same technique needs to be applied to find the cut point $C'$ on the left boundary of the calculated $MBR$, i.e., the minimum extreme point of the uncertainty region (see Fig. 1 (a)). In this case, upon turning the situation by 90 degrees counterclockwise, we see that $f$, $g$ are parabolas given by $f(x) = -ax^2 + b_1x+c_1$, $g(x) = -ax^2+b_2x+c_2$ and $h$ is a parabola given by $h(x) = ax^2+ux+v$ and we need $h$ to stay above $f$ and $g$ at all times and touches them.

The uncertainty region example shown in Fig. 1 (a) is generated by this model when both $P_s$ and $R$ for the minimum and maximum calculations lie on curved part of $g$ and $f$, respectively. Obviously, there are three other cases that need to be considered when calculating $C$ and $C'$, depending on the locations of $P_s$, $R$ and $P_s'$, $R'$, respectively. We leave these cases to the reader.

## 4 The TMBB Approximation

The uncertainty regions of $MO$s are rather "tilted" in shape in which traditional (axis-parallel) Minimum Bounding Boxes $MBB$s are most likely not close to the optimal approximations of the regions. The advantage of *Truncated Tornado* can be strengthened by a more accurate approximation that takes the tilted shape of the regions into account and not only the extreme points of the uncertainty region. We investigate *Tilted Minimum Bounding Boxes* ($TMBB$s) as approximations of the uncertainty regions generated by *Truncated Tornado*. When compared with axis-parallel $MBB$s in 3D, $TMBB$s, which are minimum volume bounding boxes that relax the axis-aligned property of $MBB$s, generally allow geometries to be bounded more tightly with a fewer number of boxes [4].

To calculate $TMBB$ of the uncertainty region of Fig. 1 (a), we identify all the extreme points that need to be considered as follows:
$R$, $C$ and $P_s$ of the maximum direction (upper boundary) in the x-dimension need to be calculated and the corresponding y-values (at specific time instance when the x-values are calculated) are assigned to these points. Similarly, $R$, $C$ and $P_s$ in the y-dimension need to be calculated and the corresponding x-values (at specific time instance when the y-values are calculated) are assigned to these points. This results in 6 points calculated in 3D. The same calculation set needs to be done for the minimum direction (lower boundary) by calculating $R'$, $C'$ and $P'_s$ in both the x and y dimensions which results in 6 other points. The other extreme points are $P_1 - e$, $P_1 + e$, $P_2 - e$ and $P_2 + e$.

Given 6 points in 3D calculated for the upper boundary, 6 points in 3D calculated for the lower boundary and finally 4 points in 3D (2 top and 2 bottom), we calculate $TMBB$ enclosing the uncertainty region of $Truncated\ Tornado$ using the approximation method of [2].

## 5 Experiments

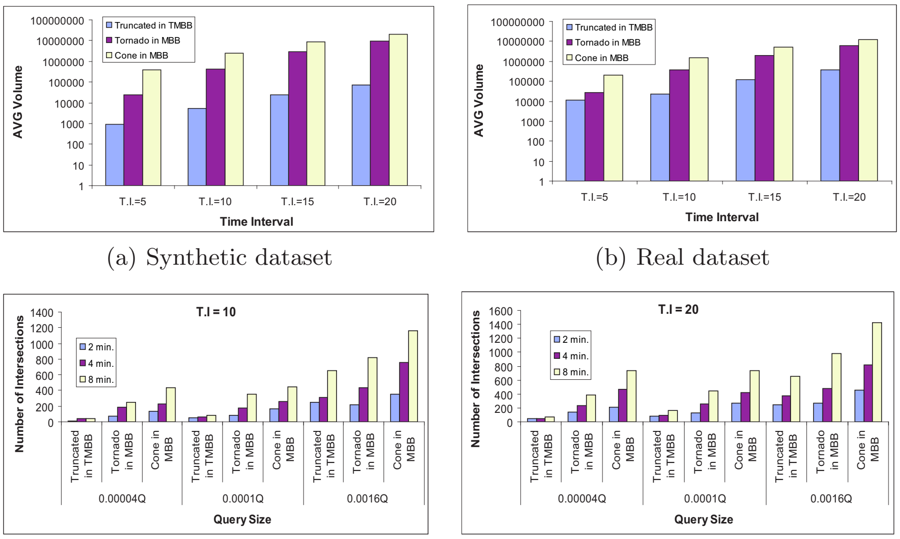
All velocities in this section are in $meters/second$ ($m/s$), all accelerations are in $meters/second^2$ ($m/s^2$) and all volumes are in $meters^2 \cdot second$ (volumes in 3D are generated by moving objects in 2D with time being the $3^{rd}$ dimension). Our synthetic datasets were generated using the "Generate_Spatio_Temporal_Data" ($GSTD$) algorithm [8].

| datasets | | reported records | | | parameters | |
|---|---|---|---|---|---|---|
| | | AVG Vel. | MAX Vel. | MAX Acc. | $M_v$ | $M_a$ |
| synthetic | Dataset | 17.76 | 20.61 | 6.41 | 55 | 8 |
| real | San Diego | 11.44 | 36.25 | 6.09 | 38.89 | 6.5 |

**Table 2.** Synthetic and real datasets and system parameters

Our synthetic dataset was generated by 200 objects moving with the velocity in the $x$ direction greater than the velocity in the $y$ direction with an average velocity of 17.76 m/s. Each object in the synthetic dataset reported its position and velocity every second for an hour. The real data set was collected using a GPS device while driving a car in the city of San Diego in California, U.S.A. The actual positions and velocities were reported every one second and the average velocity was 11.44 m/s. Although each moving object in both datasets reported its position every second, we used different time interval (T.I) values (e.g., T.I=10) to simulate various update frequencies. The range queries' sizes in our experiments are calculated using certain percentages of the universe area combined with specific time extents. Table 2 shows the datasets and system parameters used in our experiments.

We first compared the volume of $TMBB$s approximating the *Truncated Tornado* uncertainty regions to the volume of the axis-parallel $MBB$s of *Tornado* and *Cone* using the real and synthetic datasets. Fig. 2 (a) and (b) show the average volume of $TMBB$s generated by *Truncated Tornado* and the average volume of $MBB$s generated by the other two models using the synthetic and real datasets, respectively. *Truncated Tornado* combined with $TMBB$ resulted in an average reduction of 93% and 97% over the axis-parallel $MBB$ of *Tornado* and *Cone*, respectively, using the real dataset. The reduction when using the synthetic dataset was 99% over both *Tornado* and *Cone*.



(a) Synthetic dataset

(b) Real dataset

(c) T.I = 10 (Real dataset)

(d) T.I = 20 (Real dataset)

**Fig. 2.** *Truncated* in $TMBB$ Vs. *Tornado* and *Cone* in $MBB$

Next, we generated and evaluated 5000 random queries to $TMBB$s and $MBB$s calculated in the previous result for the real dataset. Fig. 2 (c) and (d) show the number of intersecting $TMBB$s of *Truncated Tornado* and the number of intersecting $MBB$s of *Tornado* and *Cone*. $TMBB$s of *Truncated* resulted in much less number of intersections compared to $MBB$s of the other models since *Truncated Tornado* results in much smaller uncertainty regions compared to *Tornado* and *Cone*. Also, $TMBB$s result in significantly smaller average volumes compared to $MBB$s as they more accurately approximate the uncertainty regions. The reduction in the number of intersections of *Truncated Tornado* $TMBB$s was 42% over *Tornado* $MBB$s and 62% over *Cone* $MBB$s when T.I=10. When T.I=20, the reduction over *Tornado* $MBB$s was 47% and was 68% over *Cone* $MBB$s.

## 6 Conclusions

In this paper we proposed the *Truncated Tornado* model that minimizes the moving object uncertainty regions. The model takes advantage of the fact that changes in the velocities of moving objects that move with momentum are limited by maximum acceleration values. This fact is used to identify and eliminate unreachable object locations, thus significantly reducing uncertainty region size. We then showed how to combine this model with the *Tilted Minimum Bounding Box* (*TMBB*), in order to achieve another order of magnitude reduction in uncertainty region size when compared to approximation bounding via traditional *MBB*s. Experiments on both synthetic and real datasets showed an order of magnitude improvement over previously proposed uncertainty models in terms of I/O accesses.

## References

1. S. Alkobaisi, W. D. Bae, S. H. Kim, and B. Yu. MBR models for uncertainty regions of moving objects. In *Proceedings of Int. Conf. on Database Systems for Advanced Applications, LNCS 4947*, pages 126–140, 2008.
2. G. Barequet and S. Har-Peled. Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. *Journal of Algorithms*, 38(1):91–109, 2001.
3. T. Brinkoff, H.-P. Kriegel, and R. Schneider. Comparison of approximations of complex objects used for approximation-based query processing in spatial database systems. In *Proceedings of Int. Conf. on Data Engineering*, pages 40–49, 1993.
4. S. Gottschalk, M. C. Lin, and D. Manocha. OBB-tree: A hierarchical structure for rapid interference detection. In *Proceedings of ACM Siggraph*, pages 171–180, 1996.
5. K. Hornsby and M. J. Egenhofer. Modeling moving objects over multiple granularities. *Annals of Mathematics and Artificial Intelligence*, 36(1-2):177–194, 2002.
6. J. O'Rourke. Finding minimal enclosing boxes. *International Journal of Parallel Programming*, 14(3):183–199, 1985.
7. D. Pfoser and C. S. Jensen. Capturing the uncertainty of moving-objects representations. In *Proceedings of Int. Symposium on Advances in Spatial Databases*, pages 111–132, 1999.
8. Y. Theodoridis, J. R. O. Silva, and M. A. Nascimento. On the generation of spatiotemporal datasets. In *Proceedings of Int. Symposium on Advances in Spatial Databases*, pages 147–164, 1999.
9. G. T. Toussaint. Solving geometric problems with the rotating calipers. In *Proceedings of IEEE MELECON*, pages A10.02/1–4, 1983.
10. G. Trajcevski, O. Wolfson, K. Hinrichs, and S. Chamberlain. Managing uncertainty in moving objects databases. *ACM Trans. on Databases Systems*, 29(3):463–507, 2004.
11. B. Yu. A spatiotemporal uncertainty model of degree 1.5 for continuously changing data objects. In *Proceedings of ACM Int. Symposium on Applied Computing, Mobile Computing and Applications*, pages 1150–1155, 2006.
12. B. Yu, S. H. Kim, S. Alkobaisi, W. D. Bae, and T. Bailey. The Tornado model: Uncertainty model for continuously changing data. In *Proceedings of Int. Conf. on Database Systems for Advanced Applications, LNCS 4443*, pages 624–636, 2007.