An analytic solution to the alibi query in the bead model for moving object data

Walied Othman and Bart Kuijpers *

Theoretical Computer Science Group
Hasselt University & Transnationale Universiteit Limburg, Belgium

Abstract. Moving objects produce trajectories, which are stored in databases by means of finite samples of time-stamped locations. When also speed limitations in these sample points are known, beads [1, 6, 9] can be used to model the uncertainty about the object's location in between sample points.
In this setting, a query of particular interest, that has been studied in the literature of geographic information systems (GIS), is the *alibi query*. This boolean query asks whether two moving objects can have physically met. This adds up to deciding whether the necklaces of beads of these objects intersect. This problem can be reduced to deciding whether two beads intersect. Since, existing software to solve this problem fails to answer this question within a reasonable time, we propose an analytical solution to the alibi query, which can be used to answer the alibi query in constant time, a matter of milliseconds or less, for two single beads and in time proportional to the product of their lengths for necklaces of beads.

1 Introduction and summary

The research on spatial databases, which started in the 1980s from work in geographic information systems, was extended in the second half of the 1990s to deal with spatio-temporal data. In this field, one particular line of research, started by Wolfson, concentrates on *moving object databases* (MODs) [2, 12], a field in which several data models and query languages have been proposed to deal with moving objects whose position is recorded at discrete moments in time. Some of these models are geared towards handling uncertainty that may come from various sources (measurements of locations, interpolation, ...) and several query formalisms have been proposed [5, 11]. For an overview of models and techniques for MODs, we refer to the book by Güting and Schneider [2].

In this paper, we focus on the trajectories that are produced by moving objects and which are stored in a database as a collection of tuples $(\mathtt{t}_i, \mathtt{x}_i, \mathtt{y}_i)$, $i = 0, ..., N$, i.e., as a finite sample of time-stamped locations in the plane. These samples may have been obtained by GPS-measurements or from other location aware devices.

* Contact author. Address: Hasselt University, dept. WNI, 3590 Diepenbeek, Belgium; phone: +32 11 26 82 45; fax: +32 11 26 82 99; mail: `Bart.Kuijpers@uhasselt.be`.

Dagstuhl Seminar Proceedings 07212
Constraint Databases, Geometric Elimination and Geographic Information Systems
http://drops.dagstuhl.de/opus/volltexte/2007/1286

One particular model for the management of the uncertainty of the moving object's position in between sample points is provided by the *bead* model. In this model, it is assumed that besides the time-stamped locations of the object also some background knowledge, in particular a (e.g., physically or law imposed) speed limitation v_i at location $(\mathsf{x}_i, \mathsf{y}_i)$ is known. The bead between two consecutive sample points is defined as the collection of time-space points where the moving objects can have passed, given the speed limitation (see Figure 2 for an illustration). The chain of beads connecting consecutive trajectory sample points is called a *lifeline necklace* [1]. Whereas beads were already conceptually known in the time geography of Hägerstrand in the 1970s [3], they were introduced in the area of GIS by Pfoser [9] and later studied by Egenhofer and Hornsby [1, 4], Miller [6], and in a query language context by the present authors [5].

In this setting, a query of particular interest that has been studied, mainly by Egenhofer and Hornsby [1, 4], is the *alibi query.* This boolean query asks whether two moving objects, that are given by samples of time-space points and speed limitations, can have physically met. This question adds up to deciding whether the necklaces of beads of these moving objects intersect or not. This problem can be considered solved in practice, when we can efficiently decide whether two beads intersect.

Although approximate solutions to this problem have been proposed [1], also an exact solution is possible. We show that the alibi query can be formulated in the constraint database model by means of a first-order query [5, 8]. It is well-known that first-order constraint queries can be effectively evaluated and there exists implementations of quantifier-elimination algorithms for first-order logic over the real numbers that can be used to evaluate queries [8]. Experiments with software packages such as QEPCAD [10] and Mathematica [7] on a variety of beads show that deciding if two concrete beads intersect can be computed on average in 2 minutes (running Windows XP Pro, SP2, with a Intel Pentium M, 1.73GHz, 1GB RAM). This means that evaluating the alibi query on the lifeline necklaces of two moving objects that each consist of 100 beads would take around $100 \times 100 \times 2$ minutes, which is almost two weeks. Clearly, such an amount of time is unacceptable.

Another solution within the range of constraint databases is to find a formula, in which the apexes and limit speeds of two beads appear as parameters, that parametrically expresses that two beads intersect. We call this problem the *parametric alibi query.* A quantifier-free formula for this parametric version could, in theory, also be obtained by eliminating one block of three existential quantifiers using existing quantifier-elimination packages. We have attempted this approach using Mathematica and QEPCAD, but after several days of running (with the above processor) described above, we have interrupted the computation, without successful outcome. It is known that these implementations fail on complicated, higher-dimensional problems. The benefit of having a quantifier-free first-order formula that expresses that two beads intersect is that the alibi query on two beads can be answered in constant time. The problem of deciding whether two lifeline necklaces intersect can then be done in time proportional to the product of the lengths of the two necklaces of beads.

The main contribution of this paper is the description of an analytic solution to the alibi query. We give a quantifier-free formula, that contains square roots, however, and that expresses the (non-)emptiness of the intersection of two parametrically given beads. Although, in a strict sense, this formula cannot be seen as quantifier-free first-order formula (due to the roots), it still gives the above mentioned complexity benefits. At the basis of our solution is a geometric theorem that describes three exclusive cases in which beads can intersect. These three cases can then be transformed into an analytic solution that can be used to answer the alibi query on the lifeline necklaces of two moving objects in less than a minute. This provides a practical solution to the alibi query.

This paper is organized as follows. In Section 2, we describe a model for trajectory (or moving object) databases with uncertainty using beads. In Section 3, we discuss the alibi query. An analytic solution to this query is given in Section 5.

2 A model for moving object data with uncertainty

In this paper, we consider moving objects in the two-dimensional (x, y)-space $\mathbf{R}^2$ and describe this movement in the (t, x, y)-space $\mathbf{R} \times \mathbf{R}^2$, where t is time. Although it is more traditional to speak about moving object databases, we will use the term trajectory databases, to emphasize this particular aspect of a moving object.

2.1 Trajectories and trajectory samples

Moving objects, which we assume to be points, produce a special kind of curves, which are parameterized by time and which we call *trajectories*. More formally, a *trajectory* T is the graph of a mapping $I \subseteq \mathbf{R} \to \mathbf{R}^2 : t \mapsto \alpha(t) = (\alpha_x(t), \alpha_y(t))$, i.e., $T = \{(t, \alpha_x(t), \alpha_y(t)) \in \mathbf{R} \times \mathbf{R}^2 \mid t \in I\}$, where I is the *time domain* of T.

In practice, trajectories are only known at discrete moments in time. This partial knowledge of trajectories is formalized in the following definition. If we want to stress that some t, x, y-values (or other values) are constants, we will use sans serif characters.

Definition 1. A *trajectory sample* is a finite set of time-space points $\{(\mathsf{t}_0, \mathsf{x}_0, \mathsf{y}_0), (\mathsf{t}_1, \mathsf{x}_1, \mathsf{y}_1), ..., (\mathsf{t}_N, \mathsf{x}_N, \mathsf{y}_N)\}$, on which the order on time $\mathsf{t}_0 < \mathsf{t}_1 < \cdots < \mathsf{t}_N$, induces a natural order. $\square$

For practical purposes, we may assume that the $(\mathsf{t}_i, \mathsf{x}_i, \mathsf{y}_i)$-tuples of a trajectory sample contain rational values.

A trajectory T, which contains a trajectory sample $\{(\mathsf{t}_0, \mathsf{x}_0, \mathsf{y}_0), (\mathsf{t}_1, \mathsf{x}_1, \mathsf{y}_1), ..., (\mathsf{t}_N, \mathsf{x}_N, \mathsf{y}_N)\}$, i.e., $(\mathsf{t}_i, \alpha_x(\mathsf{t}_i), \alpha_y(\mathsf{t}_i)) = (\mathsf{t}_i, \mathsf{x}_i, \mathsf{y}_i)$ for $i = 0, ..., N$, is called a *geospatial lifeline* for this trajectory sample [1]. A common example of a lifeline, is the reconstruction of a trajectory from a trajectory samples by linear interpolation [2].

2.2 Modeling uncertainty with beads

Often, in practical applications, more is known about trajectories than merely some sample points $(\mathsf{t}_i, \mathsf{x}_i, \mathsf{y}_i)$. For instance, background knowledge like a physically or law imposed speed limitation v_i at location $(\mathsf{x}_i, \mathsf{y}_i)$ might be available. Such speed limits that hold between two consecutive sample points, can be used to model the uncertainty of a moving object's location between sample points.

More formally, we know that at a time t, $\mathsf{t}_i \leq t \leq \mathsf{t}_{i+1}$, the object's distance to $(\mathsf{x}_i, \mathsf{y}_i)$ is at most $\mathsf{v}_i(t - \mathsf{t}_i)$ and its distance to $(\mathsf{x}_{i+1}, \mathsf{y}_{i+1})$ is at most $\mathsf{v}_i(\mathsf{t}_{i+1} - t)$. The object is therefore somewhere in the intersection of the disc with center $(\mathsf{x}_i, \mathsf{y}_i)$ and radius $\mathsf{v}_i(t - \mathsf{t}_i)$ and the disc with center $(\mathsf{x}_{i+1}, \mathsf{y}_{i+1})$ and radius $\mathsf{v}_i(\mathsf{t}_{i+1} - t)$. The geometric location of these points is referred to as a *bead* [1] and defined, for general points $p = (t_p, x_p, y_p)$ and $q = (t_q, x_q, y_q)$ and speed limit $v_{\max}$ as follows.

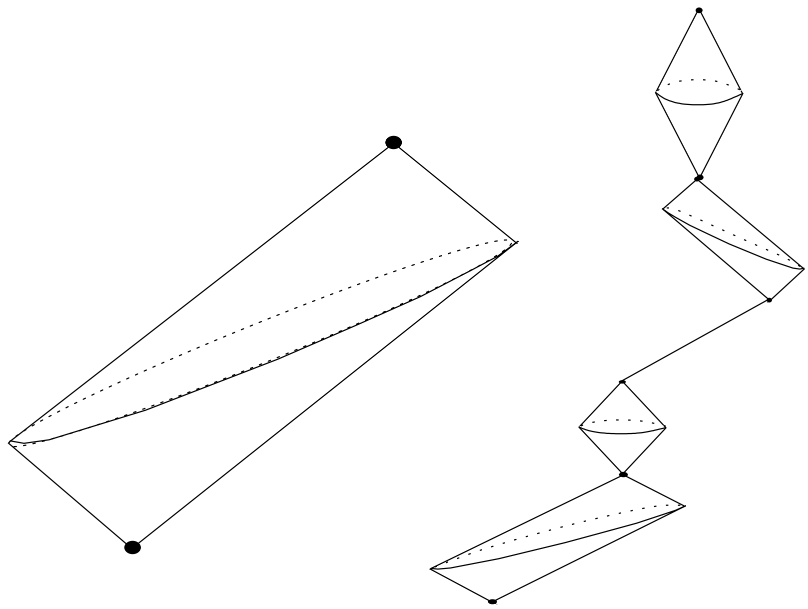

Fig. 1. A bead and a lifeline necklace.

Definition 2. The *bead* with origin $p = (t_p, x_p, y_p)$, destination $q = (t_q, x_q, y_q)$, with $t_p \leq t_q$, and maximal speed $v_{\max} \geq 0$ is the set of all points $(t, x, y) \in \mathbf{R} \times \mathbf{R}^2$ that satisfy the following constraint formula[1]

$$\begin{aligned}\Psi_{\mathsf{B}}(t, x, y, t_p, x_p, y_p, t_q, x_q, y_q, v_{\max}) := {} & (x - x_p)^2 + (y - y_p)^2 \leq (t - t_p)^2 v_{\max}^2 \\ & \wedge\ (x - x_q)^2 + (y - y_q)^2 \leq (t_q - t)^2 v_{\max}^2\ \wedge\ t_p \leq t \leq t_q\end{aligned}$$

[1] Later on, this type of formula's will be refered to as $\mathsf{FO}(+, \times, <, 0, 1)$-formulas.

We denote this set by $\mathsf{B}(p, q, v_{\max})$ or $\mathsf{B}(t_p, x_p, y_p, t_q, x_q, y_q, v_{\max})$. $\square$

Figure 2 illustrates the notion of bead in time-space. Whereas a continuous curve connecting the sample points of a trajectory sample was called a geospatial lifeline, a chain of beads connecting succeeding trajectory sample points is called a *lifeline necklace* [1].

2.3 Trajectory databases

We assume the existence of an infinite set $\mathsf{Labels} = \{\mathsf{a}, \mathsf{b}, ..., \mathsf{a}_1, \mathsf{b}_1, ..., \mathsf{a}_2, \mathsf{b}_2, ...\}$ of *trajectory labels*, that serve to identify individual trajectory samples. We now define the notion of trajectory database.

Definition 3. A *trajectory (sample) database* is a finite set of tuples $(\mathsf{a}_i, \mathsf{t}_{i,j}, \mathsf{x}_{i,j}, \mathsf{y}_{i,j}, \mathsf{v}_{i,j})$, with $i = 1, ..., r$ and $j = 0, ..., N_i$, such that $\mathsf{a}_i \in \mathsf{Labels}$ cannot appear twice in combination with the same t-value, such that $\{(\mathsf{t}_{i,0}, \mathsf{x}_{i,0}, \mathsf{y}_{i,0}), (\mathsf{t}_{i,1}, \mathsf{x}_{i,1}, \mathsf{y}_{i,1}), ..., (\mathsf{t}_{i,N_i}, \mathsf{x}_{i,N_i}, \mathsf{y}_{i,N_i})\}$ is a trajectory sample and such that the $\mathsf{v}_{i,j} \geq 0$. $\square$

3 Trajectory queries and the alibi query

3.1 Trajectory queries

A *trajectory database query* has been defined as a partial computable function from trajectory databases to trajectory databases [5]. Often, we are also interested in queries that express a property, i.e., in boolean queries. More formally, we can say that a *boolean trajectory database query* is a partial computable function from trajectory databases to $\{\mathsf{true}, \mathsf{false}\}$.

When we say that a function is computable, this is with respect to some fixed encoding of the trajectory databases (e.g., rational numbers are represented as pairs of natural numbers in bit representation).

3.2 A constraint-based query language

Several languages have been proposed to express queries on moving object data and trajectory databases (see [2] and references therein). One particular language for querying trajectory data, that was recently studied in detail by the present authors, is provided by the formalism of constraint databases. This query language is a first-order logic which extends first-order logic over the real numbers with a predicate S to address the input trajectory database. We denote this logic by $\mathsf{FO}(+, \times, <, 0, 1, S)$ and define it as follows.

Definition 4. The language $\mathsf{FO}(+, \times, <, 0, 1, S)$ is a two-sorted logic with *label variables* $a, b, c, ...$ (possibly with subscripts) that refer to trajectory labels and *real variables* $x, y, z, ..., v, ...$ (possibly with subscripts) that refer to real numbers. The atomic formulas of $\mathsf{FO}(+, \times, <, 0, 1, S)$ are

– $P(x_1, ..., x_n) > 0$, where P is a polynomial with integer coefficients in the real variables $x_1, ..., x_n$;
– $a = b$; and
– $S(a, t, x, y, v)$ (S ia a 5-ary predicate).

The formulas of $\mathsf{FO}(+, \times, <, 0, 1, S)$ are built from the atomic formulas using the logical connectives $\wedge, \vee, \neg, ...$ and quantification over the two types of variables: $\exists x$, $\forall x$ and $\exists a$, $\forall a$. □

The label variables are assumed to range over the labels occurring in the input trajectory database and the real variables are assumed to range over $\mathbf{R}$. The formula $S(a, t, x, y, v)$ expresses that a tuple (a, t, x, y, v) belongs to the input trajectory database. The interpretation of the other formulas is standard.

For example, the $\mathsf{FO}(+, \times, <, 0, 1, S)$-sentence

$$\exists a \exists b (\neg(a = b) \wedge \forall t \forall x \forall y \forall v S(a, t, x, y, v) \leftrightarrow S(b, t, x, y, v))$$

expresses the boolean trajectory query that says that there are two identical trajectories in the input database with different labels.

When we instantiate the free variables in a $\mathsf{FO}(+, \times, <, 0, 1, S)$-formula $\varphi(a, b, ..., t, x, y, ...)$ by concrete values $\mathsf{a}, \mathsf{b}, ..., \mathsf{t}, \mathsf{x}, \mathsf{y}, ...$ we write $\varphi[\mathsf{a}, \mathsf{b}, ..., \mathsf{t}, \mathsf{x}, \mathsf{y}, ...]$ for the formula we obtain.

3.3 The alibi query

The *alibi query* is the boolean query which asks whether two moving objects, say with labels a and a', that are available as samples in a trajectory database, can have physically met. Since the possible positions of these moving objects are, in between sample points, given by beads, the alibi query asks to decide if two lifeline necklaces of a and a' intersect or not.

More concretely, if the trajectory a is given in the trajectory database by the tuples $(\mathsf{a}, \mathsf{t}_0, \mathsf{x}_0, \mathsf{y}_0, \mathsf{v}_0), ..., (\mathsf{a}, \mathsf{t}_N, \mathsf{x}_N, \mathsf{y}_N, \mathsf{v}_N)$ and the trajectory a' by the tuples $(\mathsf{a}', \mathsf{t}'_0, \mathsf{x}'_0, \mathsf{y}'_0, \mathsf{v}'_0), ..., (\mathsf{a}', \mathsf{t}'_M, \mathsf{x}'_M, \mathsf{y}'_M, \mathsf{v}'_M)$, then a has an alibi for not meeting a' if for all i, $0 \leq i \leq N-1$ and all j, $0 \leq j \leq M-1$,

$$B(\mathsf{t}_i, \mathsf{x}_i, \mathsf{y}_i, \mathsf{t}_{i+1}, \mathsf{x}_{i+1}, \mathsf{y}_{i+1}, \mathsf{v}_i) \cap B(\mathsf{t}_j, \mathsf{x}_j, \mathsf{y}_j, \mathsf{t}_{j+1}, \mathsf{x}_{j+1}, \mathsf{y}_{j+1}, \mathsf{v}_j) = \emptyset. \qquad (\dagger)$$

We remark that the alibi query can be expressed by a formula in the logic $\mathsf{FO}(+, \times, <, 0, 1, S)$, which looks as follows. To start, we denote the subformula

$$S(a, \mathsf{t}_1, \mathsf{x}_1, \mathsf{y}_1, \mathsf{v}_1) \wedge S(a, t_2, x_2, y_2, v_2) \wedge \\ \forall t_3 \forall x_3 \forall y_3 \forall v_3 (S(a, t_3, x_3, y_3, v_3) \rightarrow \neg t_1 < t_3 < t_2),$$

that expresses that (t_1, x_1, y_1) and (t_2, x_2, y_2) are consecutive sample points on the trajectory a by $\sigma(a, t_1, x_1, y_1, v_1, t_2, x_2, y_2, v_2)$.

The alibi query on a and a' is then expressed as $\varphi_{\rm alibi}[\mathsf{a},\mathsf{a}'] =$

$$\begin{aligned}&\neg\exists t_1\exists x_1\exists y_1\exists v_1\exists t_2\exists x_2\exists y_2\exists v_2\exists t_1'\exists x_1'\exists y_1'\exists v_1'\exists t_2'\exists x_2'\exists y_2'\exists v_2'\\ &\quad(\sigma(\mathsf{a},t_1,x_1,y_1,v_1,t_2,x_2,y_2,v_2)\wedge\sigma(\mathsf{a}',t_1',x_1',y_1',v_1',t_2',x_2',y_2',v_2')\wedge\\ &\quad\exists t\exists x\exists y(t_1\le t\le t_2\wedge t_1'\le t\le t_2'\wedge\\ &\qquad(x-x_1)^2+(y-y_1)^2\le(t-t_1)^2v_1^2\wedge(x-x_2)^2+(y-y_2)^2\le(t_2-t)^2v_1^2\wedge\\ &\qquad(x-x_1')^2+(y-y_1')^2\le(t-t_1')^2v_1'^2\wedge(x-x_2')^2+(y-y_2')^2\le(t_2'-t)^2v_1'^2)).\end{aligned}$$

It is well-known that $\mathsf{FO}(+,\times,<,0,1,S)$-expressible queries can be evaluated effectively on arbitrary trajectory database inputs [8, 5]. Briefly explained, this evaluation can be performed by (1) replacing the occurrences of $S(\mathsf{a},t,x,y,v)$ by a disjunction describing all the sample points belonging to the trajectory sample a; the same for a'; and (2) eliminating all the quantifiers in the obtained formula. In concreto, using the notation from above, each occurrence of $S(\mathsf{a},t,x,y,v)$ would be replaced in $\varphi_{\rm alibi}[\mathsf{a},\mathsf{a}']$ by $\bigvee_{i=0}^{N-1}(t=\mathsf{t}_i\wedge x=\mathsf{x}_i\wedge y=\mathsf{y}_i\wedge v=\mathsf{v}_i)$, and similar for a'. This results in a (rather complicated) first-order formula over the reals $\tilde\varphi_{\rm alibi}[\mathsf{a},\mathsf{a}']$ in which the predicate S does not occur any more. Since first-order logic over the reals admits the elimination of quantifiers (i.e., every formula can be equivalently expressed by a quantifier-free formula), we can decide the truth value of $\tilde\varphi_{\rm alibi}[\mathsf{a},\mathsf{a}']$ by eliminating all quantifiers from this expression. In this case, we have to eliminate one block of existential quantifiers.

We can however simplify the quantifier-elimination problem. It is easy to see, looking at (†) above, that $\neg\tilde\varphi_{\rm alibi}[\mathsf{a},\mathsf{a}']$ is equivalent to

$$\bigvee_{i=0}^{N-1}\bigvee_{j=0}^{M-1}\psi_{alibi}[\mathsf{t}_i,\mathsf{x}_i,\mathsf{y}_i,\mathsf{v}_i,\mathsf{t}_{i+1},\mathsf{x}_{i+1},\mathsf{y}_{i+1},\mathsf{v}_i,\mathsf{t}_j',\mathsf{x}_j',\mathsf{y}_j',\mathsf{v}_j',\mathsf{t}_{j+1}',\mathsf{x}_{j+1}',\mathsf{y}_{j+1}',\mathsf{v}_j'],$$

where the restricted alibi-query formula $\psi_{alibi}(t_i,x_i,y_i,v_i,t_{i+1},x_{i+1},y_{i+1},v_i,t_j',x_j',y_j',v_j',t_{j+1}',x_{j+1}',y_{j+1}',v_j')$ abbreviates the formula

$$\begin{gathered}\exists t\exists x\exists y(t_i\le t\le t_{i+1}\wedge t_j'\le t\le t_{j+1}'\wedge(x-x_i)^2+(y-y_i)^2\le(t-t_i)^2v_i^2\wedge\\ (x-x_{i+1})^2+(y-y_{i+1})^2\le(t_{i+1}-t)^2v_i^2\wedge\\ (x-x_j')^2+(y-y_j')^2\le(t-t_j')^2v_j'^2\wedge(x-x_{j+1}')^2+(y-y_{j+1}')^2\le(t_{j+1}'-t)^2v_j'^2)\end{gathered}$$

that expresses that two beads intersect.

So, the instantiated formula $\psi_{alibi}[\mathsf{t}_i,\mathsf{x}_i,\mathsf{y}_i,\mathsf{v}_i,\mathsf{t}_{i+1},\mathsf{x}_{i+1},\mathsf{y}_{i+1},\mathsf{v}_i,\mathsf{t}_j',\mathsf{x}_j',\mathsf{y}_j',\mathsf{v}_j',\mathsf{t}_{j+1}',\mathsf{x}_{j+1}',\mathsf{y}_{j+1}',\mathsf{v}_j']$ expresses (†). To eliminate the existential block of quantifiers ($\exists t\exists x\exists y$) from this expression, existing software-packages for quantifier elimination, such as QEPCAD [10] and Mathematica [7] can be used. We used mathematica to decide if several beads intersected. The computation of $\psi_{alibi}[0,0,0,1,2,2,\sqrt{8},0,3,3,1,2,2,2]$ took 6 seconds,$\psi_{alibi}[0,0,0,1,2,2,\sqrt{8},0,3,4,1,2,2,2]$ took 209 seconds and $\psi_{alibi}[0,0,0,1,-1,-1,1,0,1,1,2,-1,1,2]$ took 613 seconds. Say this quantifier elimination can be computed on average in about 2 minutes (running Windows XP Pro, SP2, with a Intel Pentium M, 1.73GHz, 1GB RAM). This means that evaluating the alibi query on the lifeline necklaces of two moving objects that each consist of 100 beads would take around $100 \times 100 \times 2$ minutes, which is almost two weeks. Clearly, such an amount of time is unacceptable.

There is a better solution, however, which we discuss next, that can decide if two beads intersect or not in a couple of milliseconds or less.

3.4 The parametric alibi query

The uninstantiated formula $\psi_{alibi}(t_i, x_i, y_i, v_i, t_{i+1}, x_{i+1}, y_{i+1}, v_i, t'_j, x'_j, y'_j, v'_j, t'_{j+1}, x'_{j+1}, y'_{j+1}, v'_j)$ can be viewed as a parametric version of the restricted alibi query, where the free variables are considered parameters. This formula contains three existential quantifiers and the existing software-packages for quantifier elimination could be used to obtain a quantifier-free formula $\tilde{\psi}_{alibi}(t_i, x_i, y_i, v_i, t_{i+1}, x_{i+1}, y_{i+1}, v_i, t'_j, x'_j, y'_j, v'_j, t'_{j+1}, x'_{j+1}, y'_{j+1}, v'_j)$ that is equivalent to ψ_{alibi}. The formula $\tilde{\psi}_{alibi}$ could then be used to straightforwardly to answer the alibi query in time linear in its size, which is independent of the size of the input and therefore constant. We have tried to eliminate the existential block of quantifiers $\exists t \exists x \exists y$ from ψ_{alibi} using Mathematica and QEPCAD. After several days of running on the configuration described above, we have interrupted the computation.

The main contribution of this paper is a the description of a quantifier-free formula equivalent to $\psi_{alibi}(t_i, x_i, y_i, v_i, t_{i+1}, x_{i+1}, y_{i+1}, v_i, t'_j, x'_j, y'_j, v'_j, t'_{j+1}, x'_{j+1}, y'_{j+1}, v'_j)$. This is not a quantifier-free first-order formula in a strict sense, since it contains root expressions. However, it serves the purpose of efficiently answering the alibi query. It answers the alibi query on the lifeline necklaces of two moving objects that each consist of 100 beads in less than a minute. This description of this quantifier-free formula is the subject of the next section.

4 Preliminaries on the geometry of beads

Before, we can give an analytic solution to the alibi query and prove its correctness, we need to introduce some terminology concerning beads.

4.1 The geometry of beads

Various geometric properties of beads have already been described [1, 5, 6]. Here, we need some more definitions and notations to describe various components of a bead. These components are illustrated in Figure 2. In this section, let $p = (t_p, x_p, y_p)$ and $q = (t_q, x_q, y_q)$ be two time-space points, with $t_p \leq t_q$ and let $v_{\max}$ be a positive real number.

The set of the two apexes of $\mathsf{B}(p, q, v_{\max})$ is denotes $\tau\mathsf{B}(p, q, v_{\max})$, i.e., $\tau\mathsf{B}(p, q, v_{\max}) = \{p, q\}$. The bead $\mathsf{B}(p, q, v_{\max})$ is the intersection of two cones: its *bottom cone* is the set of all points (t, x, y) that satisfy

$$\Psi_{\mathsf{C}^-}(t, x, y, t_p, x_p, y_p, v_{\max}) := (x - x_p)^2 + (y - y_p)^2 = (t - t_p)^2 v_{\max}^2 \ \wedge \ t_p \leq t$$

and is denoted by $\mathsf{C}^-(p, v_{\max})$ or $\mathsf{C}^-(t_p, x_p, y_p, v_{\max})$; and its *upper cone* is the set of all points (t, x, y) that satisfy

$$\Psi_{\mathsf{C}^+}(t, x, y, t_q, x_q, y_q, v_{\max}) := (x - x_q)^2 + (y - y_q)^2 = (t_q - t)^2 v_{\max}^2 \ \wedge \ t \le t_q$$

and is denoted by $\mathsf{C}^+(q, v_{\max})$ or $\mathsf{C}^+(t_q, x_q, y_q, v_{\max})$.

We call the topological border of the bead $\mathsf{B}(p, q, v_{\max})$ its *mantel* and denote it by $\partial\mathsf{B}(p, q, v_{\max})$. It can be easily verified that the mantel consists of the set of points (t, x, y) that satisfy

$$\begin{gathered}\Psi_\partial(t, x, y, t_p, x_p, y_p, t_q, x_q, y_q, v_{\max}) := t_p \le t \le t_q \wedge \\ \big(2x(x_p - x_q) + x_q^2 - x_p^2 + 2y(y_p - y_q) + y_q^2 - y_p^2 \le v_{\max}^2 2t(t_p - t_q) + t_q^2 - t_p^2 \wedge \\ (x - x_p)^2 + (y - y_p)^2 = (t - t_p)^2 v_{\max}^2 \ \vee \ (x - x_q)^2 + (y - y_q)^2 = (t_q - t)^2 v_{\max}^2 \\ \wedge \ 2x(x_p - x_q) + x_q^2 - x_p^2 + 2y(y_p - y_q) + y_q^2 - y_p^2 \ge v_{\max}^2 \left(2t(t_p - t_q) + t_q^2 - t_p^2\right)\big) .\end{gathered}$$

The first conjunction describes the lower half of the mantel and the second conjunction describes the upper half of the mantel. The upper and lower half of the mantel are separated by a plane. The intersection of this plane with the bead is an ellipse, and the border of this ellipse is what we will refer to as the *rim* of the bead. We denote the rim of the bead $\mathsf{B}(p, q, v_{\max})$ by $\rho\mathsf{B}(p, q, v_{\max})$ and remark that it is described by the formula

$$\begin{gathered}\Psi_\rho(t, x, y, t_p, x_p, y_p, t_q, x_q, y_q, v_{\max}) := \\ (x - x_p)^2 + (y - y_p)^2 = (t - t_p)^2 v_{\max}^2 \ \wedge \ t_p \le t \le t_q \ \wedge \\ 2x(x_p - x_q) + x_q^2 - x_p^2 + 2y(y_p - y_q) + y_q^2 - y_p^2 = v_{\max}^2 \left(2t(t_p - t_q) + t_q^2 - t_p^2\right) .\end{gathered}$$

The plane in which the rim lies splits the bead into an *upper-half bead* and a *bottom-half bead*. The *bottom-half bead* is the set of all points (t, x, y) that satisfy

$$\begin{gathered}\Psi_{\mathsf{B}^-}(t, x, y, t_p, x_p, y_p, t_q, x_q, y_q, v_{\max}) := \\ (x - x_p)^2 + (y - y_p)^2 \le (t - t_p)^2 v_{\max}^2 \ \wedge \ t_p \le t \le t_q \ \wedge \\ 2x(x_p - x_q) + x_q^2 - x_p^2 + 2y(y_p - y_q) + y_q^2 - y_p^2 \le v_{\max}^2 \left(2t(t_p - t_q) + t_q^2 - t_p^2\right)\end{gathered}$$

and is denoted by $\mathsf{B}^-(t_p, x_p, y_p, t_q, x_q, y_q, v_{\max})$.

The upper bead is the set of all points (t, x, y) that satisfy

$$\begin{gathered}\Psi_{\mathsf{B}^+}(t, x, y, t_p, x_p, y_p, t_q, x_q, y_q, v_{\max}) := \\ (x - x_q)^2 + (y - y_q)^2 \le (t_q - t)^2 v_{\max}^2 \ \wedge \ t_p \le t \le t_q \ \wedge \\ 2x(x_p - x_q) + x_q^2 - x_p^2 + 2y(y_p - y_q) + y_q^2 - y_p^2 \ge v_{\max}^2 \left(2t(t_p - t_q) + t_q^2 - t_p^2\right)\end{gathered}$$

and is denoted by $\mathsf{B}^+(t_p, x_p, y_p, t_q, x_q, y_q, v_{\max})$.

4.2 The intersection of two cones

Let $\mathsf{C}^-(t_1, x_1, y_1, v_1)$ and $\mathsf{C}^-(t_2, x_2, y_2, v_2)$ be two bottom cones. A bottom cone, e.g., $\mathsf{C}^-(t_1, x_1, y_1, v_1)$, can be seen as a circle in 2-dimensional space (x, y)-space with center (x_1, y_1) and linearly growing radius $(t - t_1)v_1$ as $t_1 \le t$.

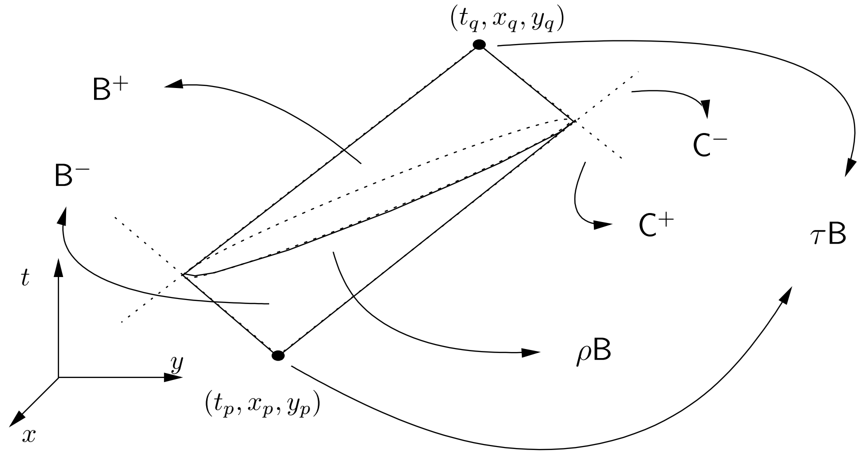

Fig. 2. A dissection of the bead $\mathsf{B}(t_p, x_p, y_p, t_q, x_q, y_q, v_{\max})$.

Let us assume that the apex of neither of these cones is inside the other cone, i.e., $(x_1-x_2)^2+(y_1-y_2)^2 > (t_1-t_2)^2v_1^2 \vee t_1 < t_2$ and $(x_1-x_2)^2+(y_1-y_2)^2 > (t_1-t_2)^2v_2^2 \vee t_2 < t_1$. This assumption implies that at t_1 and t_2 neither radius is larger than or equal to the distance between the two cone centers. So, at first the two circles are disjoint and after growing for some time they intersect in one point. We call the first (in time) time-space point where the two circles touch in a single point, and thus for which the sum of the two radii is equal to the distance between the two centers the *initial contact* of the two cones $\mathsf{C}^-(t_1, x_1, y_1, v_1)$ and $\mathsf{C}^-(t_2, x_2, y_2, v_2)$. It is the unique point (t, x, y) that satisfies the following formula

$$\begin{aligned}\Psi_{IC^-}(t,x,y,t_1,x_1,y_1,v_1,t_2,x_2,y_2,v_2) &:= t_1 \le t \wedge t_2 \le t \wedge \\ (x-x_1)^2+(y-y_1)^2 = (t-t_1)^2v_1^2 &\wedge (x-x_2)^2+(y-y_2)^2 = (t-t_2)^2v_2^2 \wedge \\ &((t-t_1)v_1+(t-t_2)v_2)^2 = (x_1-x_2)^2+(y_1-y_2)^2.\end{aligned}$$

The initial contact of two cones $\mathsf{C}^+(t_1, x_1, y_1, v_1)$ and $\mathsf{C}^+(t_2, x_2, y_2, v_2)$ is given by the formula $\Psi_{IC^+}(t,x,y,t_1,x_1,y_1,v_1,t_2,x_2,y_2,v_2)$ that we obtain from Ψ_{IC^-} by replacing in $t_1 \le t \wedge t_2 \le t$ by $t \le t_1 \wedge t \le t_2$. We denote the singleton sets containing the initial contacts by $\mathsf{IC}(\mathsf{C}^-(t_1,x_1,y_1,v_1), \mathsf{C}^-(t_2,x_2,y_2,v_2))$ and $\mathsf{IC}(\mathsf{C}^+(t_1,x_1,y_1,v_1), \mathsf{C}^+(t_2,x_2,y_2,v_2))$.

From the last equation in of the system in Ψ_{IC^-} and Ψ_{IC^+}, we easily obtain $t = \frac{\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}+t_1v_1+t_2v_2}{v_1+v_2}$. To compute the other two coordinates (x, y) of the initial contact, we observe that for in the plane of this time value t, it is on the line segment bounded by (x_1, y_1) and (x_2, y_2) and that its distance from (x_1, y_1) is $v_1(t-t_1)$ and its distance from (x_1, y_1) is $v_2(t-t_2)$. We can conclude that the initial contact has (x, y)-coordinates $(x_1, y_1)+v_1(t-t_1)\frac{(x_2-x_1, y_2-y_1)}{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}}$. This means that we can give more explicit descriptions to replace Ψ_{IC^-} and Ψ_{IC^+}.

5 An analytic solution to the alibi query

In this section, we first describe our solution to the alibi query. Next, we prove its correctness and transform it into an analytic solution and finally we show how to construct a true quantifier-free formula out of our solution.

5.1 Geometric outline of the solution

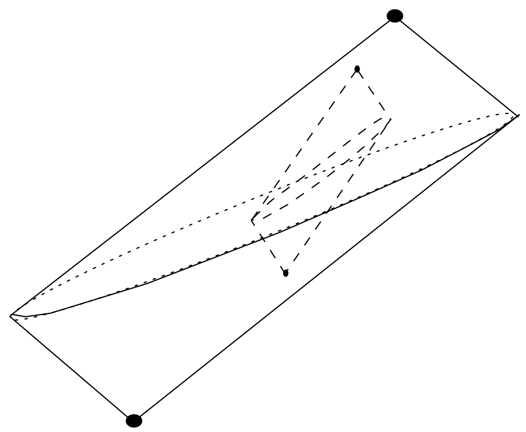

Fig. 3. One in another.

This solution we are about to present was born out of the following observations. The two main distinctions of intersection you can make is that either a bead is entirely contained in another or not. In the latter case their mantels must intersect. To rule out that the first case occurs it is enough to verify that there is an apex that is not contained in the other bead. If an apex of a bead is inside another then we have intersection and know what we wanted to know but that does not mean the entire bead is contained in the other. For ease of exposition we eliminate this first case by verifying that none of the apexes are inside the other bead and move on to the next case.

Next we assume there is no apex contained in another bead. If one assumes there still is an intersection that means the mantels must intersect, as we prove in lemma 1. The idea is to find a special point that is easily computable and always in the intersection.

Consider two cones $\mathsf{C}^-(t_1, x_1, y_1, v_1)$ and $\mathsf{C}^-(t_2, x_2, y_2, v_2)$ where none of the apexes is inside the other cone. One such special point is the initial contact $\mathsf{IC}(\mathsf{C}^-(t_1, x_1, y_1, v_1), \mathsf{C}^-(t_2, x_2, y_2, v_2))$. However, this point can not be guaranteed to be in the intersection if two mantels intersect, as we will show in the following example. Consider two cones $\mathsf{C}^-(0, 0, 0, 1)$ and $\mathsf{C}^-(0, 2, 0, 1)$. The intersection will be a hyperbola in the plane $x = 1$ with equation $t^2 - y^2 = 1$. The

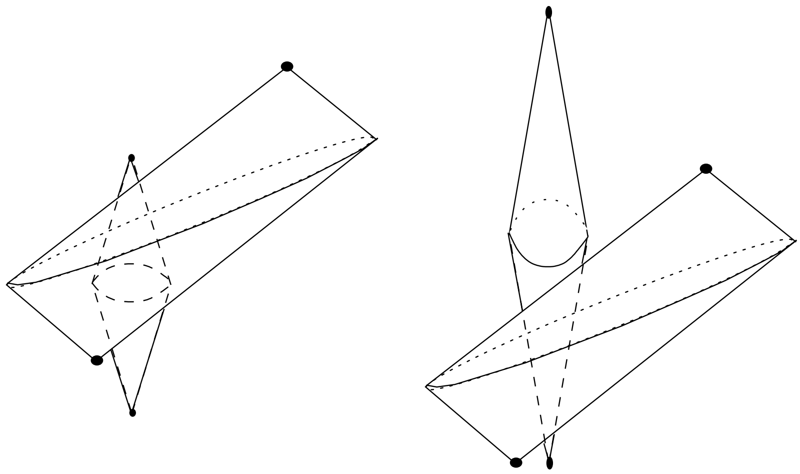

Fig. 4. Clean cut between cones.

initial contact is the point $(1, 0, 1)$. The idea is to cut this point out of the intersection as follows. Suppose a bead has apexes, $(0, 0, 0)$ and (a, b, c) and speed 1. The plane in which it rim lies is given by $-2ax+a^2-2by+b^2+2ct-c^2 = 0$. This plane cuts the plane $\alpha \leftrightarrow x = 1$ in a line given by $-2by + 2ct - 2a + a^2 - c^2 = 0$. Clearly we can choose (a, b, c) such that the line contains the points $\left(\frac{\sqrt{5}}{2}, 1, \frac{1}{2}\right)$ and $\left(\sqrt{2}, 1, 1\right)$. Everything below this line will be part of the first bead and the second cone, but the initial contact is situated above the line, effectively cutting it out of the intersection. All this is illustrated in figure 5.

Notice how the plane in which the rim lies and the rim itself is the evil do-er. If neither rim intersects the mantel of the other bead, then the intersection of mantels is the same as an intersection of cones. In which case the initial contact will not be cut out and can be used to determine if there is intersection in this manner.

Using contraposition on the statement in the previous paragraph we get: if there is an intersection and no initial contact is in the intersection then a rim must intersect the other bead's mantel.

To verify intersection with the apexes and initial contacts is straightforward. Verifying if a rim intersects a mantel results in solving a quartic polynomial equation in one variable and verifying the solution in a single inequality in which the no variable appears with a degree higher than one.

5.2 Outline of the solution

Suppose, for the remainder of this section, we wish to verify if the beads $\mathsf{B}_1 = \mathsf{B}(t_1, x_1, y_1, t_2, x_2, y_2, v_1)$ and $\mathsf{B}_2 = \mathsf{B}(t_3, x_3, y_3, t_4, x_4, y_4, v_2)$ intersect. Moreover, we assume the beads are non-empty, i.e. $(x_2 - x_1)^2 + (y_2 - y_1)^2 \leq (t_2 - t_1)^2 v_1^2$ and $(x_4 - x_3)^2 + (y_4 - y_3)^2 \leq (t_4 - t_3)^2 v_2^2$.

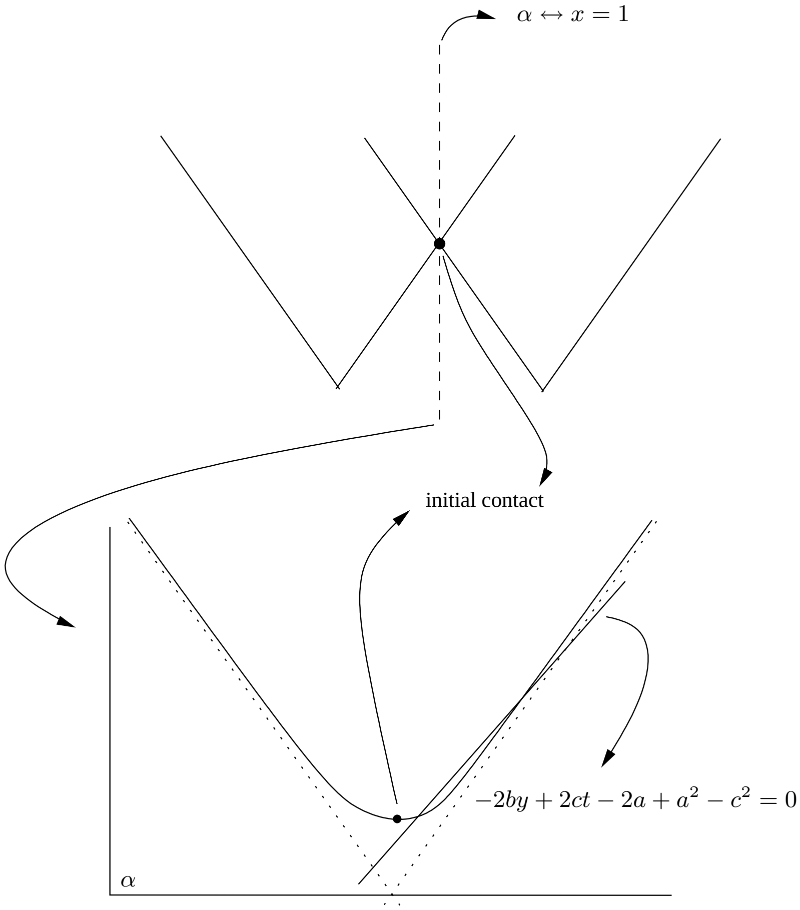

Fig. 5. The initial contact cut out.

We first observe that an intersection between beads can be classified into three, mutually exclusive, cases. The three cases then are:

(I) An apex of one bead is contained in the other, i.e.,

$$\tau\mathsf{B}_1 \cap \mathsf{B}_2 \neq \emptyset \text{ or } \mathsf{B}_1 \cap \tau\mathsf{B}_2 \neq \emptyset;$$

(II) Not **(I)**, but the rim of one bead intersects the mantel of the other, i.e.,

$$\rho\mathsf{B}_1 \cap \partial\mathsf{B}_2 \neq \emptyset \text{ or } \rho\mathsf{B}_2 \cap \partial\mathsf{B}_1 \neq \emptyset;$$

(III) Not **(I)** and not **(II)** and the initial contact of the upper or lower cones is in the intersection of the beads, i.e.,

$$\mathsf{IC}(\mathsf{C}_1^-, \mathsf{C}_2^-) \subset \mathsf{B}_1 \cap \mathsf{B}_2 \text{ or } \mathsf{IC}(\mathsf{C}_1^+, \mathsf{C}_2^+) \subset \mathsf{B}_1 \cap \mathsf{B}_2.$$

If none of these three cases occur then the beads do not intersect, as we show in the correctness proof below. First, we give the following straightforward lemma.

Lemma 1. *If* $\mathsf{B}_1 \cap \mathsf{B}_2 \neq \emptyset$, $\tau\mathsf{B}_1 \cap \mathsf{B}_2 = \emptyset$ *and* $\tau\mathsf{B}_2 \cap \mathsf{B}_1 = \emptyset$, *then* $\partial\mathsf{B}_1 \cap \partial\mathsf{B}_2 \neq \emptyset$.

Proof. We know that B_2 intersects B_1, that means there is a point p_1 in B_2, e.g. an apex of B_2, but not in B_1, also there is a point p_2 in B_2 and in B_1. The line segment bounded by p_1 and p_2 lies in B_2 since B_2 is convex and cuts the mantel of B_1 since p_2 is inside B_1 and p_1 is not. Let p be this point where the segment bounded by p_1 and p_2 intersects $\partial\mathsf{B}_1$. This point lies either on the upper-half bead B_1^+ or on the bottom-half bead B_1^-. Let r be the apex of this half bead. Since p is inside B_2 and r is not, the line segment bounded by p and r must cut $\partial\mathsf{B}_2$ in a point q. This point lies of course on $\partial\mathsf{B}_2$ and on $\partial\mathsf{B}_1$ since the line segment bounded by p and r is a part of $\partial\mathsf{B}_1$. Hence their mantels must have a nonempty intersection if the beads have a nonempty intersection and neither bead contains the apexes of the other. ◻

We show that if B_1 and B_2 intersect and neither (**I**), nor (**II**) occur, then (**III**) occurs.

Theorem 1. *If* $\mathsf{B}_1 \cap \mathsf{B}_2 \neq \emptyset$, $\tau\mathsf{B}_1 \cap \mathsf{B}_2 = \emptyset$, $\mathsf{B}_1 \cap \tau\mathsf{B}_2 = \emptyset$, $\rho\mathsf{B}_1 \cap \partial\mathsf{B}_2 = \emptyset$ *and* $\rho\mathsf{B}_2 \cap \partial\mathsf{B}_1 = \emptyset$, *then* $\mathsf{IC}(\mathsf{C}_1^-, \mathsf{C}_2^-) \subset \mathsf{B}_1 \cap \mathsf{B}_2$ *or* $\mathsf{IC}(\mathsf{C}_1^+, \mathsf{C}_2^+) \subset \mathsf{B}_1 \cap \mathsf{B}_2$.

Proof. Let us assume that the hypotheses of the statement of the theorem is true. It is sufficient to prove that either $\mathsf{C}_1^- \cap \mathsf{C}_2^- \subset \mathsf{B}_1^- \cap \mathsf{B}_2^-$ or $\mathsf{C}_1^+ \cap \mathsf{C}_2^+ \subset \mathsf{B}_1^+ \cap \mathsf{B}_2^+$. We will split the proof in two cases. From the fourth and fifth hypotheses it follows that either (1) $\rho\mathsf{B}_1 \subset \mathsf{B}_2$ or $\rho\mathsf{B}_2 \subset \mathsf{B}_1$; or (2) $\rho\mathsf{B}_1 \cap \mathsf{B}_2 = \emptyset$ and $\rho\mathsf{B}_2 \cap \mathsf{B}_1 = \emptyset$.

Case (1): We assume $\rho\mathsf{B}_2 \subset \mathsf{B}_1$ (the case $\rho\mathsf{B}_1 \subset \mathsf{B}_2$ is completely analogous). We prove $\mathsf{C}_1^- \cap \mathsf{C}_2^- \subset \mathsf{B}_1^- \cap \mathsf{B}_2^-$ (the case for upper cones is completely analogous).

Since $\rho\mathsf{B}_2 \subset \mathsf{B}_1$, we know that $\rho\mathsf{B}_2$ is inside C_1^-, and (t_3, x_3, y_3) is outside. We can show that $v_2 < v_1$. Consider the plane spanned by the two axis of symmetry of both C_1^- and C_2^-. Both C_1^- and C_2^- intersect this plane in two half lines each. Moreover, we know that C_1^- intersects the axis of symmetry of C_2^-. Let t_0 be the moment at which this happens. Obviously $t_0 > t_1$, but we know also know $t_0 > t_3$ since (t_3, x_3, y_3) is outside C_1^-. We have that $v_1(t_0 - t_1) = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}$. Since $\rho\mathsf{B}_2$ is inside C_1^- and (t_3, x_3, y_3) is outside, that means both half lines from C_2^- intersect the half lines from C_1^-. Let t_0' and t_0'' be the moments in time at which this happens and let $t_0' > t_0''$. We have again that $t_0' > t_1$ and $t_0' > t_3$. Then $v_1(t_0' - t_1) = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + v_2(t_0' - t_3)$ if and only if $v_1(t_0' - t_0) = v_2(t_0' - t_3)$. Since $t_0 > t_3$, we get $v_2 < v_1$. This is depicted in figure 6.

It follows that every straight half line starting in (t_3, x_3, y_3) on C_2^- intersects C_1^- between (t_3, x_3, y_3) and $\rho\mathsf{B}_2$, since $\rho\mathsf{B}_2$ is inside C_1^-, and (t_3, x_3, y_3) is outside. We also know that this line does not intersect C_1^- beyond $\rho\mathsf{B}_2$ since the cone C_2^- is entirely inside C_1^- beyond the rim $\rho\mathsf{B}_2$. Therefore, $\mathsf{C}_1^- \cap \mathsf{C}_2^- \subset \mathsf{B}_2^-$.

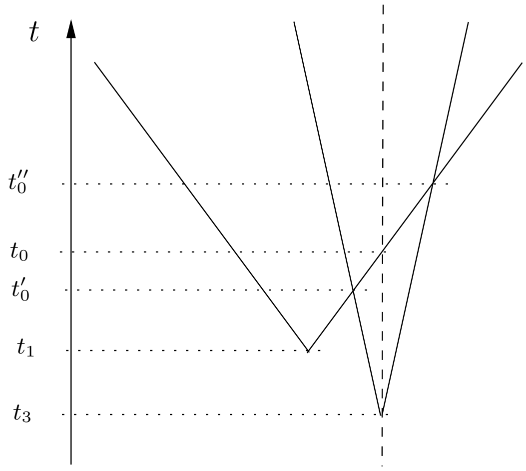

Fig. 6. Illustration to the proof.

Clearly, B_2^- intersects B_1^- since it can not intersect B_1^+. We know $\mathsf{C}_1^- \cap \partial\mathsf{B}_2^-$ is a closed continuous curve that lies entirely in C_1^-. This curve is also contained in B_1^-. Indeed, if we assume this is not the case, then it intersects the plane in which $\rho\mathsf{B}_1$ lies, and hence it intersects $\rho\mathsf{B}_1$ itself, contradicting the assumption $\rho\mathsf{B}_1 \cap \partial\mathsf{B}_2 = \emptyset$.

Case (2): Now assume $\rho\mathsf{B}_1 \cap \mathsf{B}_2 = \emptyset = \rho\mathsf{B}_2 \cap \mathsf{B}_1$. v_1 can not be equal to v_2, otherwise the depicted intersection can not occur. So suppose without loss of generality that $v_2 < v_1$. Now either B_2^- intersects both B_1^- and B_1^+ or B_2^+ intersects both B_1^- and B_1^+. These cases are mutually exclusive because of the following. If B_2^+ intersects B_1^+ then $\rho\mathsf{B}_2$ is inside C_1^+, likewise if B_2^- intersects B_1^- then $\rho\mathsf{B}_2$ is inside C_1^-. Hence $\rho\mathsf{B}_2 \subset \mathsf{B}_1$ which contradicts our hypothesis. If B_2^+ intersects B_1^- then $\rho\mathsf{B}_2$ must be outside C_1^- and thus B_2^- must be as well, hence B_2^- intersects neither B_1^- nor B_1^+. Likewise, if B_2^- intersects B_1^+ then B_2^+ can not intersect B_1.

To prove that if B_2^- intersects B_1^- then it also intersects B_1^+ and if B_2^- intersects B_1^+ then it also intersects B_1^- we proceed as follows. The case for B_2^+ is analogous. Suppose B_2^- intersects B_1^-, then $\mathsf{B}_2^- \cap \mathsf{B}_1^- \subset \mathsf{B}_1$, but $\rho\mathsf{B}_2$ is outside B_1, that means B_2^- must intersect B_1^+ since it can not intersect B_1^- anymore. This is the "what goes in must come out"-principle. Likewise, suppose B_2^- intersects B_1^+, then $\mathsf{B}_2^- \cap \mathsf{B}_1^+ \subset \mathsf{B}_1$, but (t_3, x_3, y_3) is outside B_1, that means B_2^- must intersect B_1^- since it can not intersect B_1^- anymore.

So suppose now that B_2^- intersects both B_1^- and B_1^+. The case for B_2^+ is completely analogous. If B_2^- intersects B_1^- that means $\rho\mathsf{B}_2$ is completely inside C_1^- and therefore that $\mathsf{C}_1^- \cap \mathsf{C}_2^- \subset \mathsf{B}_2^-$. We proceed like in the first case, we know that $\mathsf{C}_1^- \cap \mathsf{B}_2^-$ is a closed continuous curve. This curve lies entirely in C_1^-. If this curve is not entirely in B_1^- that means it intersects the plane in which $\rho\mathsf{B}_1$ lies, and hence intersects $\rho\mathsf{B}_1$ itself. But this is contradictory to the assumption that $\rho\mathsf{B}_1 \cap \partial\mathsf{B}_2 = \emptyset$. □

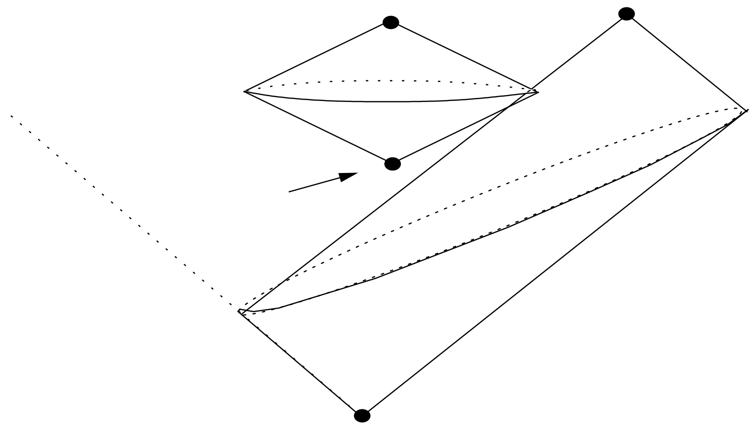

Fig. 7. Case **II** is not obsolete.

In theorem 1 we proved that if there is an intersection and neither rim cuts the other bead's mantel and neither apex of a bead is contained in the other then there must be an initial contact in the intersection. Visualizing how beads intersect might tempt one to think there is always an initial contact in the intersection. There exists a counterexample in which there is an intersection and no initial contact is in that intersection. That means case **II** is not obsolete. This situation is depicted in figure 7.

Observe that the bottom apex, marked with an arrow, of the bottom cone is inside the other bottom cone, that means there is no initial contact between the bottom cones. There does however exist an initial contact between the upper cones. By moving the bottom apex of the smallest bead into or out of the paper, you can position the plane in which its rim lies such that the initial contact is cut out. This concludes the outline.

5.3 A formula for Case (I)

Here we verify whether $\tau\mathsf{B}_1 \cap \mathsf{B}_2 \neq \emptyset$ or $\mathsf{B}_1 \cap \tau\mathsf{B}_2 \neq \emptyset$. To check if that is the case we merely need to verify if one of the apexes satisfies the set of equations of the other bead. In this way we obtain

$$\begin{aligned}&\Phi_{\mathbf{I}}\left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2\right) := \\ &\left(\Psi_{\mathsf{B}}\left(t_3, x_3, y_3, t_1, x_1, y_1, t_2, x_2, y_2, v_1\right) \vee \Psi_{\mathsf{B}}\left(t_4, x_4, y_4, t_1, x_1, y_1, t_2, x_2, y_2, v_1\right) \vee\right. \\ &\left.\Psi_{\mathsf{B}}\left(t_1, x_1, y_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2\right) \vee \Psi_{\mathsf{B}}\left(t_2, x_2, y_2, t_3, x_3, y_3, t_4, x_4, y_4, v_2\right)\right).\end{aligned}$$

For the following sections we assume that the apex sets of the beads are not singletons, i.e. $t_1 < t_2$ and $t_3 < t_4$.

5.4 A formula for Case (II)

Now assume that $\Phi_{\mathbf{I}}$ failed in the previous section. Note that we can always apply a speed-preserving [5] transformation to $\mathbf{R}\times\mathbf{R}^2$ to obtain easier coordinates. We can always find a transformation such that $(t'_1, x'_1, y'_1) = (0,0,0)$ and that the line-segment connecting (t'_1, x'_1, y'_1) and (t'_2, x'_2, y'_2) is perpendicular to the y-axis, i.e. $y'_2 = 0$. This transformation is a composition of a translation in $\mathbf{R}\times\mathbf{R}^2$, a spatial rotation in $\mathbf{R}^2$ and a scaling in $\mathbf{R}\times\mathbf{R}^2$ [5]. Let the coordinates without a prime be the original set, and let coordinates with a prime be the image of the same coordinates without a prime under an this transformation. Note that we do not need to transform back because the query is invariant under such transformations [5]. The following formula returns the transformed coordinates (t', x', y') of $(t, x, y,)$ given the points (t_1, x_1, y_1) and (t_2, x_2, y_2):

$$
\begin{aligned}
\varphi_A(t_1, x_1, y_1, t_2, x_2, y_2, t, x, y, t', x', y') &:= (y_2 \neq y_1 \ \wedge \\
&t' = (t - t_1)\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \ \wedge \ x' = (x - x_1)(x_2 - x_1) \\
&+(y - y_1)(y_2 - y_1) \ \wedge \ y' = (x - x_1)(y_1 - y_2) + (y - y_1)(x_2 - x_1)) \\
&\vee \ (y_2 = y_1 \ \wedge \ t' = (t - t_1) \ \wedge \ x' = (x - x_1) \ \wedge \ y' = (y - y_1))\,.
\end{aligned}
$$

The translation is over the vector $(-t_1, -x_1, -y_1)$, the rotation over minus the angle that $(t_2 - t_1, x_2 - x_1, y_2 - y_1)$ makes with the x-axis, and a scaling by a factor $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. Notice that the rotation and scaling only need to occur if y_2 is not already in place, i.e. if $(y_2 \neq y_1)$.

The formula $\psi_{crd}(t'_1, x'_1, y'_1, t_1, x_1, y_1, t'_2, x'_2, y'_2, t_2, x_2, y_2, t'_3, x'_3, y'_3, t_3, x_3, y_3, t'_4, x'_4, y'_4, t_4, x_4, y_4)$ is short for $\varphi_A(t_1, x_1, y_1, t_2, x_2, y_2, t_1, x_1, y_1, t'_1, x'_1, y'_1) \ \wedge \ \varphi_A(t_1, x_1, y_1, t_2, x_2, y_2, t_2, x_2, y_2, t'_2, x'_2, y'_2) \wedge \varphi_A(t_1, x_1, y_1, t_2, x_2, y_2, t_3, x_3, y_3, t'_3, x'_3, y'_3) \wedge \varphi_A(t_1, x_1, y_1, t_2, x_2, y_2, t_4, x_4, y_4, t'_4, x'_4, y'_4)$.

This transformation yields some simple equations for the rim $\rho\mathsf{B}_1$:

$$
\rho\mathsf{B}_1 \leftrightarrow \begin{cases} x^2 + y^2 = t^2 v_1^2 \\ 2x(-x'_2) + x'^2_2 = v_1^2(2t(-t'_2) + t'^2_2) \\ 0 \leq t \leq t'_2\ . \end{cases}
$$

Not only that, but with these equations we can deduce a simple parametrization in the x-coordinate for the rim,

$$
\rho\mathsf{B}_1 \leftrightarrow \begin{cases} t = \frac{2xx'_2 - x'^2_2 + v_1^2 t'^2_2}{2v_1^2 t'_2} \\ y = \pm\sqrt{v_1^2\left(\frac{2xx'_2 - x'^2_2 + v_1^2 t'^2_2}{2v_1^2 t'_2}\right)^2 - x^2} \\ 0 \leq t \leq t'_2\ . \end{cases}
$$

Note that this implies $t'_2 \neq 0$ and $v_1 \neq 0$. If $t'_2 = 0$ then B_1 is a point, hence degenerate. If $v_1 = 0$ then B_1 is a line segment, and again degenerate. Next we will inject these parameterizations in the constraints for $\partial\mathsf{B}_2^+$ and $\partial\mathsf{B}_2^-$ separately. The constraints for $\partial\mathsf{B}_2^-$ are

$$
\begin{cases} (x - x'_3)^2 + (y - y'_3)^2 = (t - t'_3)^2 v_2^2 \\ 2x(x'_3 - x'_4) + x'^2_4 - x'^2_3 + 2y(y'_3 - y'_4) + y'^2_4 - y'^2_3 \leq v_2^2\left(2t(t'_4 - t'_3) + t'^2_4 - t'^2_3\right) \\ t'_3 \leq t \leq t'_4\ . \end{cases}
$$

We will explain how to proceed to compute the intersection with $\partial\mathsf{B}_2^-$ and simply reuse formulas for intersection with $\partial\mathsf{B}_2^+$. First we insert our expressions for x and y in the first equation. This is equivalent to computing intersections of $\rho\mathsf{B}_1$ with C_1^-:

$$
\begin{aligned}
(x-x_3')^2 + \left(\pm\sqrt{v_1^2\left(\frac{2xx_2' - x_2'^2 + v_1^2t_2'^2}{2v_1^2t_2'}\right)^2 - x^2} - y_3'\right)^2 \\
= \left(\frac{2xx_2' - x_2'^2 + v_1^2t_2'^2}{2v_1^2t_2'} - t_3'\right)^2 v_2^2
\end{aligned}
$$

$$
\begin{aligned}
&\text{iff } \pm\left(2v_1^2t_2'\right)2y_3'\sqrt{v_1^2\left(2xx_2' - x_2'^2 + v_1^2t_2'^2\right)^2 - \left(2v_1^2t_2'\right)^2x^2} = \\
&\left(2xx_2' - x_2'^2 + v_1^2t_2'^2 - \left(2v_1^2t_2'\right)t_3'\right)^2 v_2^2 - \left(2v_1^2t_2'\right)^2(x-x_3')^2 \\
&-\left(2v_1^2t_2'\right)^2 y_3'^2 - \left(v_1^2\left(2xx_2' - x_2'^2 + v_1^2t_2'^2\right)^2 - \left(2v_1^2t_2'\right)^2x^2\right)
\end{aligned}
$$

$$
\begin{aligned}
\text{iff } \pm\left(2v_1^2t_2'\right)v_12y_3'\sqrt{x^24\left(x_2'^2 - v_1^2t_2'^2\right) + x4x_2'^2\left(v_1^2t_2'^2 - x_2'^2\right) + \left(v_1^2t_2'^2 - x_2'^2\right)^2} \\
= x^24x_2'^2\left(v_2^2 - v_1^2\right) + \\
x4\left(-x_2'^2v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right) + 2v_1^4t_2'^2x_3' + v_1^2x_2'\left(v_1^2t_2'^2 - x_2'^2\right)\right) \\
+\left(v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right)^2 - 4v_1^4t_2'^2\left(x_3'^2 + y_3'^2\right) - v_1^4\left(-x_2'^2 + v_1^2t_2'^2\right)\right) .
\end{aligned}
$$

By squaring left and right in this last expression we rid ourselves of the square root and obtain the following polynomial equation of degree four.

Notice that if B_1 is degenerate, i.e. $x_2'^2 = v_1^2t_2'^2$ or $v_1 = 0$, then the square root vanishes and the polynomial in ϕ_4 is the square of a polynomial of degree two, yielding to at most two roots and intersection points as we expect. So the following still works if one or both beads are degenerate,

$$
\begin{aligned}
&\phi_4(x, t_2', x_2', v_1, t_3', x_3', y_3', v_2) := \exists a\exists b\exists c\exists d\exists e\ ax^4 + bx^3 + cx^2 + dx + e = 0 \\
&\quad\wedge\ a = \left(4x_2'^2\left(v_2^2 - v_1^2\right)\right)^2\ \wedge\ b = -32x_2'^4v_2^2\left(v_2^2 - v_1^2\right)\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right. \\
&\quad\left.+2v_1^4t_2'^2x_3' + v_1^2x_2'\left(v_1^2t_2'^2 - x_2'^2\right)\right)\ \wedge\ c = 8\left(x_2'^2 - v_1^2t_2'^2\right)\left(-4v_1^4t_2'^2\left(x_3'^2 + y_3'^2\right) + \right. \\
&\quad\left.+v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right)^2 - v_1^4\left(-x_2'^2 + v_1^2t_2'^2\right)\right) + \left(8v_1^3t_2'y_3'\right)^2\left(x_2'^2 - v_1^2t_2'^2\right) \\
&\quad + \left(4\left(-x_2'^2v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right) + 2v_1^4t_2'^2x_3' + v_1^2x_2'\left(v_1^2t_2'^2 - x_2'^2\right)\right)\right)^2 \\
&\quad\wedge\ d = 8\left(-x_2'^2v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right) + 2v_1^4t_2'^2x_3' + v_1^2x_2'\left(v_1^2t_2'^2 - x_2'^2\right)\right) \\
&\quad\left(v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right)^2 - 4v_1^4t_2'^2\left(x_3'^2 + y_3'^2\right) - v_1^4\left(-x_2'^2 + v_1^2t_2'^2\right)\right) \\
&\quad + \left(8v_1^3t_2'y_3'\right)^2\left(4x_2'^2\left(v_1^2t_2'^2 - x_2'^2\right)\right)\ \wedge\ e = \left(8v_1^3t_2'y_3'\right)^2\left(v_1^2t_2'^2 - x_2'^2\right)^2 + \\
&\quad\left(v_2^2\left(-x_2'^2 + v_1^2t_2'^2 - 4v_1^4t_2'^2t_3'\right)^2 - 4v_1^4t_2'^2\left(x_3'^2 + y_3'^2\right) - v_1^4\left(-x_2'^2 + v_1^2t_2'^2\right)\right)^2 .
\end{aligned}
$$

The quantifiers we introduced here are only in place for esthetic considerations and can be eliminated by direct substitution.

Note that if $v_1 = v_2$ the degree is merely two. This can be solved in an exact manner using square roots [?] or Maple if you will. This gives us at most four values for x. Let

$$\phi_{roots}(x_a, x_b, x_c, x_d, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2)$$

be a formula that returns all four real roots, if they exist, that satisfy $\phi_4(x, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2)$. We substitute these values in the parameter equations of $\rho\mathsf{B}_1$. By substituting these in the last equation above we can determine the sign of the square root we need to take for y. A point (t, x, y) satisfies the following formula is a point on $\rho\mathsf{B}_1$, but instead of using the square root for y, we use an expression from above to get the correct sign for the square root:

$$\begin{aligned}\psi_\rho(t, x, y, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2) &:= t\left(2v_1^2 t'_2\right) = 2xx'_2 - x'^2_2 + v_1^2 t'^2_2 \ \wedge \\ &2y'_3\left(2v_1^2 t'_2\right)^2 y = \left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2 - \left(2v_1^2 t'_2\right) t'_3\right)^2 v_2^2 - \left(2v_1^2 t'_2\right)^2 (x - x'_3)^2 \\ &- \left(2v_1^2 t'_2\right)^2 y'^2_3 - \left(v_1^2\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2\right)^2 - \left(2v_1^2 t'_2\right)^2 x^2\right) \ \wedge \ \ 0 \le t \le t'_2 \ .\end{aligned}$$

The four roots give us four spatio-temporal points on $\rho\mathsf{B}_1 \cap \mathsf{C}_2^-$. In order for these points (t, x, y) to be in $\rho\mathsf{B}_1 \cap \partial\mathsf{B}_2^-$, they need to satisfy

$$\begin{aligned}&\psi_-(t, x, y, t'_3, x'_3, y'_3, t'_4, x'_4, y'_4, v_2) := \\ &2x(x'_3 - x'_4) + x'^2_4 - x'^2_3 + 2y(y'_3 - y'_4) + y'^2_4 - y'^2_3 \le v_2^2\left(2t(t'_3 - t'_4) + t'^2_4 - t'^2_3\right) \ .\end{aligned}$$

This formula returns true if (t, x, y) lies in the same half space as the bottom-half bead; The formula ψ_+ returns true if (t, x, y) lies in the same half space as the upper-half bead; $\psi_+(t, x, y, t'_3, x'_3, y'_3, t'_4, x'_4, y'_4, v_2) := \psi_-(t, x, y, t'_4, x'_4, y'_4, t'_3, x'_3, y'_3, v_2)$. By combining $\psi_\rho(t, x, y, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2)$ and $\psi_-(t, x, y, \hat{t}, \hat{x}, \hat{y}, \tilde{t}, \tilde{x}, \tilde{y}, v)$ we get a formula that decides their intersection for a parameter x:

$$\begin{aligned}&\psi_{\rho\cap\partial^\pm}(x, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2, \hat{t}, \hat{x}, \hat{y}, \tilde{t}, \tilde{x}, \tilde{y}, v) := \big(2x(\hat{x} - \tilde{x}) + \tilde{x}^2 - \hat{x}^2 \\ &+\tilde{y}^2 - \hat{y}^2\big)\, 2y'^2_3\left(2v_1^2 t'_2\right)^2 + 2y'_3\left(\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2 - \left(2v_1^2 t'_2\right) t'_3\right)^2 v_2^2 - \left(2v_1^2 t'_2\right)^2\right. \\ &\left.(x - x'_3)^2 - \left(2v_1^2 t'_2\right)^2 y'^2_3 - \left(v_1^2\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2\right)^2 - \left(2v_1^2 t'_2\right)^2 x^2\right)\right)(\hat{y} - \tilde{y}) \le \\ &v^2 2y'^2_3\left(2\left(2v_1^2 t'_2\right)\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2\right)(\hat{t} - \tilde{t}) + \left(2v_1^2 t'_2\right)\left(\tilde{t}^2 - \hat{t}^2\right)\right) \ \wedge \ 0 \le t'_2 \\ &\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2\right) \le 2v_1^2 t'^3_2 \wedge \left(\hat{t}\left(2v_1^2 t'^2_2\right) \le t'_2\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2\right) \le \tilde{t}\left(2v_1^2 t'^2_2\right)\right. \\ &\left.\vee \ \ \tilde{t}\left(2v_1^2 t'^2_2\right) \le t'_2\left(2xx'_2 - x'^2_2 + v_1^2 t'^2_2\right) \le \hat{t}\left(2v_1^2 t'^2_2\right)\right) \ .\end{aligned}$$

We are ready now to construct the formula that decides if $\rho\mathsf{B}_1$ and B_2^- have a nonempty intersection:

$$\begin{aligned}&\varphi_{\rho_1\cap\partial_2^-}(t'_2, x'_2, v_1, t'_3, x'_3, y'_3, t'_4, x'_4, y'_4, v_2) := \exists x \exists x_a \exists x_b \exists x_c \exists x_d \ \wedge \\ &\phi_{roots}(x_a, x_b, x_c, x_d, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2) \wedge (x = x_a \vee x = x_b \vee x = x_c \vee x = x_d) \\ &\wedge \ \psi_{\rho\cap\partial^\pm}(x, t'_2, x'_2, v_1, t'_3, x'_3, y'_3, v_2, t'_3, x'_3, y'_3, t'_4, x'_4, y'_4, v'_2) \ .\end{aligned}$$

The formula that decides if $\rho\mathsf{B}_1$ intersects $\partial\mathsf{B}_2^+$ looks strikingly similar:

$$\varphi_{\rho_1\cap\partial_2^+}(t_2',x_2',v_1,t_3',x_3',y_3',t_4',x_4',y_4',v_2) := \exists x\exists x_a\exists x_b\exists x_c\exists x_d \ \wedge$$
$$\phi_{roots}(x_a,x_b,x_c,x_d,t_2',x_2',v_1,t_4',x_4',y_4',v_2)\wedge(x=x_a\vee x=x_b\vee x=x_c\vee x=x_d)$$
$$\wedge \ \psi_{\rho\cap\partial^\pm}(x,t_2',x_2',v_1,t_4',x_4',y_4',v_2,t_4',x_4',y_4',t_3',x_3',y_3',v_2') \ .$$

The quantifiers introduced here can also be eliminated in a straightforward manner. Notice that ϕ_{roots} acts as a function rather than a formula that inputs $(t_2',x_2',v_1,t_4',x_4',y_4',v_2)$ to construct a polynomial of degree four and returns the four roots (x_a,x_b,x_c,x_d), if they exist, of that polynomial. The existential quantifier for the variable x is used to cycle through those roots to see if any of them does the trick. Finally we are ready to present the formula for case **II**:

$$\Phi_{\mathbf{II}}(t_1,x_1,y_1,t_2,x_2,y_2,v_1,t_3,x_3,y_3,t_4,x_4,y_4,v_2) :=$$
$$\neg\ \Phi_{\mathbf{I}}(t_1,x_1,y_1,t_2,x_2,y_2,v_1,t_3,x_3,y_3,t_4,x_4,y_4,v_2)\wedge$$
$$\exists t_1'\exists x_1'\exists y_1'\exists t_2'\exists x_2'\exists y_2'\exists t_3'\exists x_3'\exists y_3'\exists t_4'\exists x_4'\exists y_4'$$
$$\psi_{crd}(t_1',x_1',y_1',t_1,x_1,y_1,t_2',x_2',y_2',t_2,x_2,y_2,t_3',x_3',y_3',t_3,x_3,y_3,t_4',x_4',y_4',t_4,x_4,y_4)$$
$$\wedge\ \Big(\varphi_{\rho_1\cap\partial_2^-}(t_2',x_2',v_1,t_3',x_3',y_3',t_4',x_4',y_4',v_2')\ \vee$$
$$\varphi_{\rho_1\cap\partial_2^+}(t_2',x_2',v_1,t_3',x_3',y_3',t_4',x_4',y_4',v_2)\Big)\ \vee$$
$$\psi_{crd}(t_3',x_3',y_3',t_3,x_3,y_3,t_4',x_4',y_4',t_4,x_4,y_4,t_1',x_1',y_1',t_1,x_1,y_1,t_2',x_2',y_2',t_2,x_2,y_2)$$
$$\wedge\ \Big(\varphi_{\rho_1\cap\partial_2^-}(t_3',x_3',v_2,t_1',x_1',y_1',t_2',x_2',y_2',v_1')\ \vee$$
$$\varphi_{\rho_1\cap\partial_2^+}(t_3',x_3',v_2,t_1',x_1',y_1',t_2',x_2',y_2',v_1')\Big)\ .$$

The reader may notice that a lot of quantifiers have been introduced in the formula above. These quantifiers are merely there to introduce easier coordinates and can be straightforwardly computed (and eliminated) by the formula ψ_{crd} and hence the formula $\varphi_A(t_1,x_1,y_1,t_2,x_2,y_2,t,x,y,t',x',y')$. The latter actually acts like a function, parameterized by $(t_1,x_1,y_1,t_2,x_2,y_2)$, that inputs (t,x,y) and outputs (t',x',y').

5.5 A formula for Case (III)

Here we assume that both $\varphi_{\mathbf{I}}$ and $\varphi_{\mathbf{II}}$ failed. So there is no apex contained in the other bead and neither rim cuts the mantel of the other bead.

As we proved in theorem 1, the intersection between two half beads will reduce to the intersection between two cones and that means there is an initial contact that is part of the intersection. To verify if this is the case we compute the two initial contacts and verify if they are effectively part of the intersection.

Using the expression for the initial contact $\mathsf{IC}(\mathsf{C}_1^-,\mathsf{C}_2^-)$ we computed in 4.2 we can construct a formula that decides if it is part of $\mathsf{B}_1^-\cap\mathsf{B}_2^-$. We will recycle the formulas ψ_- from the previous section to construct an expression without the need for extra variables. The following formula that returns true if $\mathsf{IC}(\mathsf{C}_1^-, \mathsf{C}_2^-) = (t_0, x_0, y_0)$ satisfies $\psi_-(t_0, x_0, y_0, t', x', y', \hat{t}, \hat{x}, \hat{y}, v)$:

$$
\begin{aligned}
&\phi_-(t_1, x_1, y_1, v_1, t_3, x_3, y_3, v_2, t', x', y', \hat{t}, \hat{x}, \hat{y}, v) := \\
&\quad 2(x' - \hat{x}) \Big((x_1 v_2 + x_3 v_1) \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + v_1 \left((t_3 - t_1) v_2 \right) (x_3 - x_1) \Big) \\
&+ 2(y' - \hat{y}) \Big((y_1 v_2 + y_3 v_1) \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + v_1 \left((t_3 - t_1) v_2 \right) (y_3 - y_1) \Big) \\
&+ \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} (v_1 + v_2) \left(\hat{x}^2 - x'^2 + \hat{y}^2 - y'^2 \right) \le v^2 \Big(\left(\hat{t}^2 - t'^2 \right) (v_1 + v_2) \\
&+ 2 \Big(\sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + t_1 v_1 + t_3 v_2 \Big) \left(t' - \hat{t} \right) \Big) \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} \, .
\end{aligned}
$$

The following formula expresses that the time coordinate t_0 of $\mathsf{IC}(\mathsf{C}_1^-, \mathsf{C}_2^-)$ satisfies the constraints $t' \le t \le t''$ and $\hat{t} \le t \le \check{t}$:

$$
\begin{aligned}
&\psi_t \left(t_1, x_1, y_1, v_1, t_3, x_3, y_3, v_2, t', t'', \hat{t}, \check{t} \right) := \\
&\qquad t'(v_1 + v_2) \le \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + t_1 v_1 + t_3 v_2 \le t''(v_1 + v_2) \\
&\qquad \wedge \ \hat{t}(v_1 + v_2) \le \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + t_1 v_1 + t_3 v_2 \le \check{t}(v_1 + v_2) \, .
\end{aligned}
$$

Now, $\mathsf{IC}(\mathsf{C}_1^-, \mathsf{C}_2^-) \subset \mathsf{B}_1^- \cap \mathsf{B}_2^-$ iff $\psi_{\mathsf{IC}^-}(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2)$ where $\psi_{\mathsf{IC}^-}(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2) := \psi_t(t_1, x_1, y_1, v_1, t_3, x_3, y_3, v_2, t_1, t_2, t_3, t_4) \wedge \phi_-(t_1, x_1, y_1, v_1, t_3, x_3, y_3, v_2, t_1, x_1, y_1, t_2, x_2, y_2, v_1) \wedge \phi_-(t_1, x_1, y_1, v_1, t_3, x_3, y_3, v_2, t_3, x_3, y_3, t_4, x_4, y_4, v_2)$ and $\mathsf{IC}(\mathsf{C}_1^+, \mathsf{C}_2^+) \subset \mathsf{B}_1^+ \cap \mathsf{B}_2^+$ iff $\psi_{\mathsf{IC}^+}(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2) := \psi_t(t_2, x_2, y_2, v_1, t_4, x_4, y_4, v_2, t_1, t_2, t_3, t_4) \ \wedge \ \phi_-(t_2, x_2, y_2, v_1, t_4, x_4, y_4, v_2, t_2, x_2, y_2, t_1, x_1, y_1, v_1) \ \wedge \ \phi_-(t_2, x_2, y_2, v_1, t_4, x_4, y_4, v_2, t_4, x_4, y_4, t_3, x_3, y_3, v_2)$.

The formula that expresses the criterium for case two then looks as follows:

$$
\begin{aligned}
&\Phi_{\mathbf{III}} \left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2 \right) := \\
&\neg \, \Phi_{\mathbf{I}} \left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2 \right) \ \wedge \\
&(\psi_{\mathsf{IC}^-}(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2) \vee \\
&\ \psi_{\mathsf{IC}^+}(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2)) \ \ .
\end{aligned}
$$

5.6 The final solution

The final formula that decides if two beads, $\mathsf{B}_1 = \mathsf{B}(t_1, x_1, y_1, t_2, x_2, y_2, v_1)$ and $\mathsf{B}_2 = \mathsf{B}(t_3, x_3, y_3, t_4, x_4, y_4, v_2)$, do not intersect looks as follows

$$
\begin{aligned}
\psi_{alibi} \left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2 \right) := \neg \, ((t_1 < t_2 \ \wedge \ t_3 < t_4) \ \wedge& \\
(\Phi_{\mathbf{III}} \left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2 \right)& \\
\vee \, \Phi_{\mathbf{II}} \left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2 \right))& \\
\vee \, \Phi_{\mathbf{I}} \left(t_1, x_1, y_1, t_2, x_2, y_2, v_1, t_3, x_3, y_3, t_4, x_4, y_4, v_2 \right)) \, .&
\end{aligned}
$$

6 Conclusion

In this paper we proposed a method that decides if two beads have a non-empty intersection or not. Existing methods could achieve this already through means of quantifier elimination though not in a reasonable amount of time. Deciding intersection of concrete beads took of the order of minutes, while the parametric case could be measured at least in days. The parametric solution we laid out in this paper only takes a few milliseconds or less.

The solution we present is a first order formula containing square root-expressions. We claim that these can easily be disposed of using repeated squarings and adding extra conditions, thus obtaining a true quantifier free-expression for the alibi query.

Acknowledgments. This research has been partially funded by the European Union under the FP6-IST-FET programme, Project n. FP6-14915, GeoPKDD: Geographic Privacy-Aware Knowledge Discovery and Delivery, and by the Research Foundation Flanders (FWO-Vlaanderen), Research Project G.0344.05.

References

1. M. Egenhofer. Approximation of geopatial lifelines. In *SpadaGIS, Workshop on Spatial Data and Geographic Information Systsems*, 2003. Electr. proceedings, 4p.
2. R. Güting and M. Schneider. *Moving Object Databases.* Morgan Kaufmann, 2005.
3. T. Hägerstrand What about People in Regional Science? *Papers of the Regional Science Association* vol.24, 1970, pp.7-21.
4. K. Hornsby and M. Egenhofer. Modeling moving objects over multiple granularities. *Annals of Mathematics and Artificial Intelligence*, 36(1–2):177–194, 2002.
5. B. Kuijpers, W. Othman Trajectory Databases: Data Models, Uncertainty and Complete Query Languages. In *Database Theory - ICDT 2007, 11th International Conference, Barcelona, Spain, January 10-12, 2007, Proceedings*, volume 4353 of *Lecture Notes in Computer Science*, pages 224-238, 2007.
6. Harvey J. Miller A Measurement Theory for Time Geography *Geographical Analysis*, Vol. 37 Issue 1 Page 17, 2005
7. Mathematica *http://www.wolfram.com*
8. J. Paredaens, G. Kuper, and L. Libkin, editors. *Constraint databases.* Springer-Verlag, 2000.
9. D. Pfoser and C. S. Jensen. Capturing the uncertainty of moving-object representations. In *Advances in Spatial Databases (SSD'99)*, volume 1651 of *Lecture Notes in Computer Science*, pages 111–132, 1999.
10. QEPCAD *http://www.cs.usna.edu/ qepcad/B/QEPCAD.html*
11. J. Su, H. Xu, and O. Ibarra. Moving objects: Logical relationships and queries. In *Advances in Spatial and Temporal Databases (SSTD'01)*, volume 2121 of *Lecture Notes in Computer Science*, pages 3–19. Springer, 2001.
12. O. Wolfson. Moving ob jects information management: The database challenge. In *Proceedings of the 5th Intl. Workshop NGITS*, pages 75–89. Springer, 2002.