

Article

Minimizing Query Frequency to Bound Congestion Potential for Moving Entities at a Fixed Target Time [†]

William Evans *  and David Kirkpatrick * 

Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

* Correspondence: will@cs.ubc.ca (W.E.); kirk@cs.ubc.ca (D.K.)

[†] This paper is an extended version of our paper published in the International Symposium on Fundamentals of Computation Theory, Trier, Germany, 18–21 September 2023.

Abstract: Consider a collection of entities moving continuously with bounded speed, but otherwise unpredictably, in some low-dimensional space. Two such entities encroach upon one another at a fixed time if their separation is less than some specified threshold. Encroachment, of concern in many settings such as collision avoidance, may be unavoidable. However, the associated difficulties are compounded if there is uncertainty about the precise location of entities, giving rise to potential encroachment and, more generally, potential congestion within the full collection. We adopt a model in which entities can be queried for their current location (at some cost) and the uncertainty region associated with an entity grows in proportion to the time since that entity was last queried. The goal is to maintain low potential congestion, measured in terms of the (dynamic) intersection graph of uncertainty regions, at specified (possibly *all*) times, using the lowest possible query cost. Previous work in the same uncertainty model addressed the problem of minimizing the congestion potential of *point entities* using location queries of some bounded frequency. It was shown that it is possible to design query schemes that are $O(1)$ -competitive, in terms of worst-case congestion potential, with other, even clairvoyant query schemes (that exploit knowledge of the trajectories of all entities), subject to the same bound on query frequency. In this paper, we initiate the treatment of a more general problem with the complementary optimization objective: minimizing the *query frequency*, measured as the reciprocal of the minimum time between queries (granularity), while guaranteeing a fixed bound on congestion potential of *entities with positive extent* at one specified target time. This complementary objective necessitates quite different schemes and analyses. Nevertheless, our results parallel those of the earlier papers, specifically tight competitive bounds on required query frequency.

Keywords: data in motion; uncertain inputs; collision avoidance; online algorithms; competitive analysis



Citation: Evans, W.; Kirkpatrick, D. Minimizing Query Frequency to Bound Congestion Potential for Moving Entities at a Fixed Target Time. *Algorithms* **2024**, *17*, 246. <https://doi.org/10.3390/a17060246>

Academic Editors: Henning Fernau and Klaus Jansen

Received: 10 January 2024

Revised: 1 June 2024

Accepted: 1 June 2024

Published: 6 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

This paper addresses a fundamental issue in algorithm design, of both theoretical and practical interest: how to cope with unavoidable imprecision in data. We focus on a class of problems associated with location uncertainty arising from the motion of independent entities when location queries to reduce uncertainty are expensive. For concreteness, imagine a collection of robots following unpredictable trajectories with bounded speed. If an individual robot is not monitored continuously there is uncertainty, growing with the duration of unmonitored activity, concerning its precise location. This portends some risk of collision with neighboring robots, necessitating some perhaps costly collision avoidance protocol. Nevertheless, robots that are known to be well-separated at some point in time will remain free of collision in the near future. How then should a limited query budget be allocated over time to minimize the risk of collisions or, more realistically, help focus collision avoidance measures on robot pairs at serious risk of collision?

We adopt a general framework for addressing such problems, essentially the same as the one studied by Evans et al. [1] and by Busto et al. [2]. In this model, an entity may be queried at any time in order to reveal its true location but between queries, location uncertainty, represented by a region surrounding the last known location, grows. Our goal is to understand with what frequency such queries need to be performed, and which entities should be queried, in order to maintain a particular measure of the congestion potential of the entities, formulated in terms of the overlap of their uncertainty regions. We describe query schemes that ensure a specified bound on two measures of congestion potential at a specified target time. Our schemes are shown to be competitive, in terms of query granularity, with all other schemes that ensure the same bound.

While the problem of guaranteeing low congestion potential at a fixed target time is of interest in its own right, it also serves to set the stage for the more ambitious task of guaranteeing low congestion potential continuously (i.e., at all times). This task is taken up in a companion paper [3] where we present query schemes to maintain several measures of congestion potential that, over every modest-sized time interval, are competitive in terms of the frequency of their queries, with any scheme that maintains the same measure over that interval alone.

1.1. The Query Model

To facilitate comparisons with earlier results, we adopt much of the notation used by Evans et al. [1] and Busto et al. [2]. Let $\mathcal{E}$ be a set $\{e_1, e_2, \dots, e_n\}$ of (mobile) entities. Each entity e_i is modeled as a d -dimensional closed ball with fixed extent and bounded speed, whose position (center location) at any time is specified by the (unknown) continuous function ζ_i from $[0, \infty)$ (time) to $\mathbb{R}^d$. We take the entity radius to be our *unit of distance*, and take the time for an entity moving at maximum speed to move a unit distance to be our *unit of time*.

The n -tuple $(\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$ is called the $\mathcal{E}$ -*configuration* at time t . Entities e_i and e_j are said to *encroach* upon one another at time t if the distance $\|\zeta_i(t) - \zeta_j(t)\|$ between their centers is less than some fixed *encroachment threshold* Ξ . For simplicity we assume to start that the distance between entity centers is always at least 2, i.e., the *separation* $\|\zeta_i(t) - \zeta_j(t)\| - 2$ between entities e_i and e_j at time t , is always at least zero—so entities never properly intersect, and that the encroachment threshold is exactly 2 (i.e., we are only concerned with avoiding entity contact). The concluding section considers a relaxation (and decoupling) of these assumptions in which $\|\zeta_i(t) - \zeta_j(t)\|$ is always at least some positive constant ρ_0 (possibly less than 2), and the encroachment threshold Ξ is some constant at least ρ_0 .

We wish to maintain knowledge of the positions of the entities over time by making location queries to individual entities, each of which returns the exact position of the entity at the time of the query. A (*query*) *scheme* $\mathbb{S}$ is just an assignment of location queries to time instances. We measure the performance of a scheme over a specified time interval T as the minimum query *granularity* (the time between consecutive queries) over T .

At any time $t \geq 0$, let $p_i^{\mathbb{S}}(t)$ denote the time, prior to t , that entity e_i was last queried; we define $p_i^{\mathbb{S}}(0) = -\infty$. The *uncertainty region* of e_i at time t , denoted $u_i^{\mathbb{S}}(t)$, is defined as the ball with center $\zeta_i(p_i^{\mathbb{S}}(t))$ and radius $1 + t - p_i^{\mathbb{S}}(t)$; note that $u_i^{\mathbb{S}}(0)$ is unbounded. (We omit $\mathbb{S}$ when it is understood and the dependence on t when t is fixed.) Figure 1 illustrates the uncertainty regions of four unit-radius entities shown at their most recently known locations, four, three, two and one time unit in the past, respectively.

The set $U(t) = \{u_1(t), \dots, u_n(t)\}$ is called the (*uncertainty*) *configuration* at time t . Entity e_i is said to *potentially encroach* upon entity e_j in configuration $U(t)$ if $u_i(t) \cap u_j(t) \neq \emptyset$ (that is, there are potential locations for e_i and e_j at time t such that $e_i \cap e_j \neq \emptyset$).

In this way, any configuration U gives rise to an associated (symmetric) *potential encroachment graph* PE^U on the set $\mathcal{E}$. Note, that by our assumptions above, the potential encroachment graph associated with the initial uncertainty configuration $U(0)$ is complete.

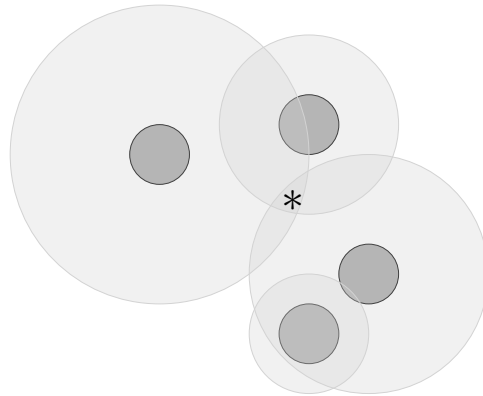


Figure 1. Uncertainty regions (light grey) of four unit-radius entities (dark grey) with uncertainty ply three (witnessed by point *).

We define the following notions of *congestion potential* (called *interference potential* in [2]) in terms of configuration U and the graph PE^U .

- The **(uncertainty) max-degree** (hereafter **degree**) of the configuration U is given by $\delta^U = \max_i \{\delta_i^U\}$ where δ_i^U is defined as the degree of entity e_i in PE^U (the maximum number of entities e_j , including e_i , that potentially encroach upon e_i in configuration U).
- The **(uncertainty) ply** ω^U of configuration U is the maximum number of uncertainty regions in U that intersect at a single point. This is the largest number of entities in configuration U whose mutual potential encroachment is witnessed by a single point.
- The **(uncertainty) thickness** χ^U of configuration U is the chromatic number of PE^U . This is the size of the smallest decomposition of $\mathcal{E}$ into independent sets (sets with no potential for encroachment) in configuration U .

The configuration illustrated in Figure 1 has uncertainty degree four and uncertainty ply three (witnessed by point *).

Note that $\omega^U \leq \chi^U \leq \delta^U$, so upper bounds on uncertainty degree, and lower bounds on uncertainty ply, apply to all three measures. As we will see, even when we seek to minimize congestion at one fixed target time, the query frequency required to guarantee that the uncertainty degree does not exceed some fixed value x can exceed the query frequency required to guarantee that uncertainty ply does not exceed x , by a factor that is $\Theta(x)$.

The assumption that entities never properly intersect is helpful since it means that if uncertainty regions are kept sufficiently small, uncertainty ply can be kept to at most two. Similarly, for x larger than some dimension-dependent sphere-packing constant, it is always possible to maintain uncertainty degree at most x , using sufficiently high query frequency.

1.2. Related Work

One of the most widely-studied approaches to computing functions of moving entities uses the *kinetic data structure* model which assumes precise information about the future trajectories of the moving entities and relies on elementary geometric relations among their locations along those trajectories to certify that a combinatorial structure of interest, such as their convex hull, remains essentially the same. The algorithm can anticipate when a relationship will fail, or is informed if a trajectory changes, and the goal is to update the structure efficiently in response to these events [4–6]. Another less common model assumes that the precise location of *every* entity is given to the algorithm periodically. The goal again is to update the structure efficiently when this occurs [7–9].

More similar to ours is the framework introduced by Kahan [10,11] for studying data in motion problems that require repeated computation of a function (geometric structure) of data that are moving continuously in space where data acquisition via queries is costly. There, location queries occur simultaneously in batches, triggered by requests (Kahan refers

to these requests as “queries”) to compute some specified function at some time (unknown in advance) rather than by a requirement to maintain a structure or property at all times. Performance is compared to a “lucky” algorithm that queries the minimum amount to calculate the function. Kahan’s model and use of competitive evaluation are common to much of the work on query algorithms for uncertain inputs (see Erlebach and Hoffmann’s survey [12]).

As mentioned, our model is essentially the same as the one studied by Evans et al. [1] and by Busto et al. [2], both of which focus on point entities. Like the current paper, paper [1] contains strategies whose goal is to guarantee competitively low congestion potential, compared to any other (even clairvoyant) scheme, at one specified target time. It provides precise descriptions of the impact of this guarantee using several measures of initial location knowledge and available lead time before the target. The other paper [2] contains a scheme for guaranteeing competitively low congestion potential at *all* times. For this more challenging task, the scheme maintains congestion potential over time that is within a constant factor of that maintained by any other scheme over modest-sized time intervals. All of these results dealt with *optimizing congestion potential measures subject to fixed query frequency*.

In this paper, we consider the complementary problem: *optimizing query frequency required to guarantee fixed bounds on congestion*. These two problems are fundamentally different: being able to optimize congestion using fixed query frequency provides little insight into how to optimize query frequency to maintain a fixed bound on congestion. In particular, even for stationary entities, a small change in the congestion bound can lead to an arbitrarily large change in the required query frequency. Our frequency optimization involves maximizing the minimum query granularity which requires more than just minimizing the number of queries made over a specified interval.

1.3. Our Results

The overarching goal of this line of research is to formulate efficient query schemes that, for all possible collections of moving entities, maintain fixed bounds on congestion potential measures continuously (i.e., at all times). Naturally, for many such collections, the required query frequency changes over time as entities cluster and spread, so efficient query schemes need to adapt to changes in the configuration of entities. While such changes are continuous and bounded in rate, they are only discernible through queries to individual entities, so entity configurations are *never* known precisely; future configurations are of course entirely hidden. In this latter respect, our schemes and the competitive analysis of their efficiency using as a benchmark a *clairvoyant scheme* that bases its queries on full knowledge of all entity trajectories (and hence, all future configurations), resembles familiar scenarios that arise in the design and analysis of on-line algorithms.

Our goal in this paper is to show how to optimize query frequency to guarantee low congestion potential at one fixed target time, say time $\tau > 0$ in the future, starting from a state of complete uncertainty of entity locations. In a companion paper [3] we turn our attention to the optimization of query frequency to guarantee low congestion potential continuously. Motivation for restricting attention to a fixed target time comes in part from the desire to prepare for a computation, at some known time in the future, whose efficiency depends on this low congestion potential (see [2,13] for examples). As we will see in Section 4, a modification of our fixed-target schemes also plays an important role in the efficient initialization of query schemes that optimize queries to guarantee low congestion potential continuously, from some point in time onward. Our fixed-target schemes also provide an informative contrast to continuous optimization schemes, highlighting the unavoidable extra cost of maintaining low congestion continuously.

We begin by describing a query scheme to achieve uncertainty ply at most x at one fixed target time in the future, reminiscent of the objective in [1]. The detailed description and analysis of our fixed target time scheme shows that uncertainty ply at most x can be achieved using query granularity that is at most a factor $\Theta(x)$ smaller than that used by

any, even clairvoyant, query scheme to achieve the same goal. A similar, but more intricate scheme and analysis establishes the same result for uncertainty degree. An example shows that the competitive factor for both congestion potential measures is asymptotically optimal in the worst case. Nevertheless, if we relax our objective, allowing instead uncertainty degree at most $x + \Delta$, where $1 \leq \Delta \leq x$, the competitive factor $\Theta(x)$ drops to $\Theta(\frac{x}{1+\Delta})$. Again, this competitive factor is shown to be asymptotically optimal in the worst case. This analysis of a query scheme that solves a slightly relaxed optimization, relative to a clairvoyant scheme that solves the un-relaxed optimization, foreshadows similar analyses of our schemes for continuous-time query optimization [14].

In the concluding discussion, we describe modifications to our model that make our query optimization framework even more broadly applicable.

2. Geometric Preliminaries

In any $\mathcal{E}$ -configuration $Z = (z_1, z_2, \dots, z_n)$ and for any positive integer x , we call the separation between e_i and its x th closest neighbour (not including e_i) its x -separation, and denote it by $\sigma_i^Z(x)$. We call the closed ball with radius (called the x -radius of e_i) $r_i^Z(x) = \sigma_i^Z(x) + 1$ and center z_i , the x -ball of e_i , and denote it by $B_i^Z(x)$ (cf. Figure 2). We will omit Z when the configuration is understood. Note that, for all entities e_i and e_j ,

$$\sigma_j^Z(x) \leq \|z_j - z_i\| + \sigma_i^Z(x), \quad (1)$$

since the ball with radius $\|z_j - z_i\| + \sigma_i^Z(x)$ centered at z_j contains the ball with a radius $\sigma_i^Z(x)$ centered at z_i (by the triangle inequality).

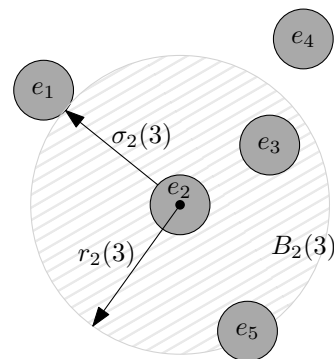


Figure 2. A configuration of five unit radius entities. The 3-ball $B_2(3)$ of entity e_2 is shown shaded.

We have assumed that entities do not properly intersect. Define $c_{d,x}$ to be the smallest constant such that a unit-radius d -dimensional ball B can have x disjoint unit-radius d -dimensional balls (not including itself) with separation from B at most $c_{d,x}$. Thus, $\sigma_i^Z(x) \geq c_{d,x}$.

Let X_d be the largest value of x for which $c_{d,x} = 0$ (e.g., $X_2 = 6$). Clearly, if $x \leq X_d$, there are entity configurations Z with $\sigma_i^Z(x) = 0$. Thus, for such x , obtaining uncertainty degree at most x , even at one specified target time, might be impossible, in general, for any query scheme. On the other hand, if $x > X_d$ then $c_{d,x} > 0$, and therefore, it suffices to query all n entities in the time interval $[\tau - c_{d,x}/2, \tau)$, using granularity $c_{d,x}/(2n)$, in order to obtain congestion less than x at time τ .

Remark 1. Hereafter we will assume that x , our bound on congestion potential, is greater than X_d .

3. Query Optimization to Obtain Bounded Congestion at a Fixed Target Time

Suppose our goal for a given entity set $\mathcal{E}$, is to optimize queries to guarantee low congestion potential at some fixed target time $\tau > 0$ in the future. If granularity is not an issue, $O(|\mathcal{E}|)$ queries suffice, provided they are made sufficiently close to the target time.

Maximizing the minimum query granularity is less straightforward. In fact, it is NP-hard, even in the case where all entities are stationary (this follows directly from [1] [Theorem 17]). Nevertheless, it is clear that any reasonable query scheme using minimum query granularity γ , that guarantees a given measure of congestion potential at most x at the target time τ , will not query any entity more than once within the final n queries. Thus, any such optimal query scheme determines a radius $1 + k_i\gamma$ for each entity $e_i \in \mathcal{E}$, where (i) $k_1, k_2, \dots, k_n$ is a permutation of $1, 2, \dots, n$, and (ii) the uncertainty configuration, in which entity e_i has an uncertainty region u_i with center $\zeta_i(\tau - k_i\gamma)$ and radius $1 + k_i\gamma$, has the given congestion measure at most x . For any measure, we associate with $\mathcal{E}$ an *intrinsic fixed-target granularity*, defined to be the largest γ for which these conditions are satisfiable.

It is not hard to see that, by projecting the current uncertainty regions to the target time (assuming no further queries), some entities can be declared “safe” (meaning their projected uncertainty regions cannot possibly contribute to making a congestion measure, for itself or any other entity, exceed x at the target time). This idea is exploited in query schemes that query entities in rounds of geometrically decreasing duration, following each of which a subset of such “safe” entities are set aside with no further attention, until no “unsafe” entities remain.

3.1. The Fixed-Target-Time-ply (FTT-ply) Query Scheme

This query scheme shows that, for any Δ , $0 \leq \Delta \leq x$, uncertainty ply at most $x + \Delta$ can be guaranteed at a fixed target time using minimum query granularity that is at most $\Theta(\frac{x}{1+\Delta})$ smaller than that used by any (even clairvoyant) query scheme that guarantees uncertainty ply at most x . Assuming no prior knowledge about entity locations, we treat the uncertainty regions of all entities as being unbounded at time 0. Hence, none of the entities are $(x + \Delta)$ -ply-safe to start (assuming $x + \Delta < n$). Thus any scheme, including a clairvoyant scheme, must query all but x of the entities at least once in order to avoid ply greater than x at the target time. The FTT-ply[$x + \Delta$] scheme starts by querying all entities in a single round using query granularity $\frac{\tau}{2n}$, which is $O(1)$ -competitive, assuming $n - (x + \Delta) = \Omega(x + \Delta)$, with what must be conducted by any other scheme.

At this point, the FTT-ply[$x + \Delta$] scheme identifies the set of n_1 entities that are not yet $(x + \Delta)$ -ply-safe (the *unsafe survivors*). All other entities are set aside and attract no further queries.

The scheme then queries, in a second round, all n_1 survivors using query granularity $\frac{\tau}{4n_1}$. In general, after the r th round, the scheme identifies n_r unsafe survivors which, assuming $n_r > 0$, continue into an $(r + 1)$ st round using granularity $\frac{\tau}{2^{r+1}n_r}$. The r th round completes at time $\tau - \tau/2^r$. Furthermore, all entities that have not been set aside have a projected uncertainty region whose radius is in the range $(1 + \tau/2^r, 1 + \tau/2^{r-1}]$. Note that, by our assumption, $x > X_d$ and hence, $c_{d,x} > 0$. Thus, since the projected uncertainty region radius of all entities is less than $1 + c_{d,x}/2$, by the time $r > \lg(\tau/c_{d,x}) + 2$, no unsafe survivors remain after by the end of round $\lg(\tau/c_{d,x}) + 2$.

Theorem 1. For any Δ , $0 \leq \Delta \leq x$, the FTT-ply[$x + \Delta$] query scheme guarantees uncertainty ply at most $x + \Delta$ at target time τ and uses minimum query granularity over the interval $[0, \tau]$ that is at most a factor $\Theta(\frac{x}{1+\Delta})$ smaller than the intrinsic fixed-target granularity for guaranteeing uncertainty ply at most x .

Proof. We claim that any query scheme $\mathbb{S}$ that guarantees uncertainty ply at most x at time τ must use at least $\Theta(\frac{n_r(1+\Delta)}{x+\Delta})$ queries after the start of the r th query round of the FTT-ply[$x + \Delta$] query scheme; any fewer queries would result in one or more entities having ply greater than x at the target time.

To see this observe first that each of the n_r unsafe survivors is either queried by $\mathbb{S}$ after the start of the r th query round or has its projected uncertainty ply reduced to at most x , by queries to at least $1 + \Delta$ of its projected uncertainty neighbors after the start of the r th query round. Assuming that fewer than $n_r/2$ unsafe survivors are queried by $\mathbb{S}$ after the

start of the r th query round, we argue that at least $\frac{n_r(1+\Delta)}{2 \cdot 4^d(x+\Delta)}$ queries must be made after the start of the r th query round to reduce the projected uncertainty ply of the remaining unsafe survivors to some value at most x .

Note that any query after the start of the r th round to an entity set aside in an earlier round cannot serve to lower the projected uncertainty ply of any of the n_r unsafe survivors. Furthermore, any query to one of the survivors of the $(r-1)$ st round can serve to decrease by one the projected uncertainty ply of at most $4^d(x+\Delta)$ of the unsafe survivors whose projected uncertainty ply is at least $x+\Delta$. (This follows because (i) the projected uncertainty regions of all survivors are within a factor of 2 in size, and (ii) any collection of $4^d \hat{x}$ unit radius balls that are all contained in a ball of radius 4, must have ply at least $\hat{x}$.) Thus any scheme that guarantees uncertainty ply at most x at time τ must make at least $\frac{n_r(1+\Delta)}{2 \cdot 4^d(x+\Delta)}$ queries after the start of the r th query round.

Since query scheme $\mathbb{S}$ must use at least $\frac{n_r(1+\Delta)}{2 \cdot 4^d(x+\Delta)} \geq \frac{n_r}{4^{d+1}} \frac{1+\Delta}{x}$ queries over the interval $[\tau - \tau/2^{r-1}, \tau]$, it follows that our query scheme is $\Theta(\frac{x}{1+\Delta})$ -competitive, in terms of minimum query granularity, with any, even clairvoyant, query scheme that guarantees uncertainty ply at most x at the target time. $\square$

3.2. The Fixed-Target-Time-Degree (FTT-Degree) Query Scheme

This somewhat more involved query scheme shows that for any Δ , $0 \leq \Delta \leq x$, uncertainty degree at most $x+\Delta$ can be guaranteed at a fixed target time using minimum query granularity that is at most $\Theta(\frac{x}{1+\Delta})$ smaller than that used by any query scheme that guarantees uncertainty degree at most x . As before, since the projected uncertainty regions of all entities are unbounded at time 0, any scheme, including a clairvoyant scheme, must query all but x of the entities at least once in order to avoid degree (and also ply) greater than x at the target time. The FTT-degree $[x+\Delta]$ scheme starts by querying all entities in a single round using query granularity $\frac{\tau}{2^n}$, which is $O(1)$ -competitive, assuming $n - (x+\Delta) = \Omega(x+\Delta)$, with what must be conducted by any other scheme.

At this point, the FTT-degree $[x+\Delta]$ scheme identifies two sets of entities (i) the n_1 entities that are not yet $(x+\Delta)$ -degree-safe (the *unsafe survivors*), and (ii) the m_1 entities that are $(x+\Delta)$ -degree-safe and whose projected uncertainty region intersects the projected uncertainty region of one or more of the unsafe survivors (the *safe survivors*). All other entities are set aside and attract no further queries.

The scheme then queries, in a second round, all $n_1 + m_1$ survivors using query granularity $\frac{\tau}{4(n_1+m_1)}$. In general, after the r th round, the scheme identifies n_r unsafe survivors and m_r safe survivors, which, assuming $n_r + m_r > 0$, continue into an $(r+1)$ st round using granularity $\frac{\tau}{2^{r+1}(n_r+m_r)}$. The r th round completes at time $\tau - \tau/2^r$. Furthermore, all entities that have not been set aside have a projected uncertainty region whose radius is in the range $(1 + \tau/2^r, 1 + \tau/2^{r-1}]$.

Theorem 2. For any Δ , $0 \leq \Delta \leq x$, the FTT-degree $[x+\Delta]$ query scheme guarantees uncertainty degree at most $x+\Delta$ at target time τ and uses minimum query granularity over the interval $[0, \tau]$ that is at most a factor $\Theta(\frac{x}{1+\Delta})$ smaller than the intrinsic fixed-target granularity for guaranteeing uncertainty degree at most x .

Proof. We claim that any query scheme $\mathbb{S}$ that guarantees the uncertainty degree at most x at time τ must use at least $\Theta(\frac{(n_r+m_r)(1+\Delta)}{x+\Delta})$ queries after the start of the r th query round of the FTT-degree $[x+\Delta]$ query scheme; any fewer queries would result in one or more entities having a degree greater than x at the target time.

To see this, observe first that each of the n_r unsafe survivors must be *satisfied*; they are either queried by $\mathbb{S}$ after the start of the r th query round or have their projected uncertainty degree reduced below x by at least $1+\Delta$ queries to their projected uncertainty neighbors after the start of the r th query round. Assuming that fewer than $n_r/2$ unsafe survivors

are queried by $\mathbb{S}$ after the start of the r th query round, we argue that at least $\frac{n_r(1+\Delta)}{2 \cdot 4^d(x+\Delta)}$ queries must be made after the start of the r th query round to reduce below x the projected uncertainty degree of the remaining unsafe survivors.

Note, that any query after the start of the r th round to an entity set aside in an earlier round cannot serve to lower the projected uncertainty degree of any of the n_r unsafe survivors. As in the proof of Theorem 1, any query to one of the survivors of the $(r-1)$ st round can serve to decrease by one the projected uncertainty degree of at most $4^d(x+\Delta)$ of the unsafe survivors whose uncertainty degree is at most $x+\Delta$. Thus, any scheme that guarantees uncertainty degree at most x at time τ must make at least $\frac{n_r(1+\Delta)}{2 \cdot 4^d(x+\Delta)}$ queries after the start of the r th query round.

Similarly, observe that each of the m_r safe survivors must have each of its unsafe neighbors satisfied in the sense described above. But, since the projected uncertainty regions of all survivors are within a factor of 2 in size, each query that serves to lower the projected uncertainty degree of an unsafe neighbor of some safe survivor e_i must be to an entity e_j that has the projected uncertainty region of e_i in its projected uncertainty *near-neighborhood* (the ball centered at z_j , whose radius is nine times the projected uncertainty radius of e_j). But e_j has at most $18^d(x+\Delta)$ such safe near-neighbours, since any collection of $18^d x$ unit radius balls that are all contained in a ball of radius 18, must have a degree (and also ply) at least $\hat{x}$.

It follows that, even if a query to e_j lowers the projected uncertainty degree of all of the unsafe neighbors of e_i , a total of at least $\frac{m_r(1+\Delta)}{18^d(x+\Delta)}$ queries must be made after the start of the r th query round by any scheme that guarantees uncertainty degree at most x at time τ .

Thus, query scheme $\mathbb{S}$ must use at least

$$\max \left\{ \frac{n_r(1+\Delta)}{2 \cdot 4^d(x+\Delta)}, \frac{m_r(1+\Delta)}{18^d(x+\Delta)} \right\} \geq \frac{n_r + m_r}{2 \cdot (18^d) x + \Delta} \geq \frac{n_r + m_r}{4 \cdot (18^d)} \frac{1+\Delta}{x}$$

queries over the interval $[\tau - \tau/2^{r-1}, \tau]$. It follows that our query scheme is $\Theta(\frac{x}{1+\Delta})$ -competitive, in terms of minimum query granularity, with any, even clairvoyant, query scheme that guarantees the uncertainty degree at most x at the target time. $\square$

The competitive factor in both of the preceding theorems is worst-case optimal. Specifically, the following example demonstrates that, for $0 \leq \Delta < x$ degree at most x can be guaranteed at a fixed target time by a clairvoyant scheme that uses query granularity, yet any non-clairvoyant scheme that guarantees ply at most $x+\Delta$ at the target time must use the query granularity that is $O(\frac{x}{1+\Delta})$.

Example 1. Imagine a configuration involving two collections A and B each with $(x+1+\Delta)/2$ point entities located in $\mathbb{R}^1$, on opposite sides of a point O . At time 0 all of the entities are at distance $x+3+3\Delta$ from O , but have unbounded uncertainty regions. All entities begin by moving towards O at unit speed, but at time $x+1+\Delta$ a subset of $1+\Delta$ entities in both A and B (the special entities) change direction and move away from O at unit speed, while all of the others carry on until the target time $x+3+3\Delta$ when they simultaneously reach O and stop (cf. Figure 3).

To avoid an uncertainty degree greater than x at the target time a clairvoyant scheme needs only to (i) query all entities (in arbitrary order) up to time $x+1+\Delta$, and then (ii) query just the special entities (in arbitrary order) in the next $2(1+\Delta)$ time prior to the target, using query granularity 1, since doing so will leave the uncertainty regions of the $(x+1+\Delta)/2$ entities in A disjoint from the uncertainty regions of the $1+\Delta$ special entities in B , and vice versa.

On the other hand, to avoid ply $x+1+\Delta$ at the target time any non-clairvoyant scheme must query at least one of the special entities (in either A or B) in the last $2(1+\Delta)$ time before the target. Since special and non-special entities are indistinguishable before this time interval, at least $x/2+1$ entities in at least one of A or B must be queried in the last $2(1+\Delta)$ time before the target in order to be sure that at least one special entity is queried late enough to confirm that its uncertainty region will not contain O at the target time. This requires query granularity at most $\frac{4(1+\Delta)}{x+2}$. Thus, in the

worst case every scheme that achieves uncertainty ply at most $x + \Delta$ at the target time needs to use at least a factor $\Theta(\frac{x}{1+\Delta})$ smaller query granularity in some instances than the best query scheme for achieving uncertainty degree at most x at the target time on those same instances.

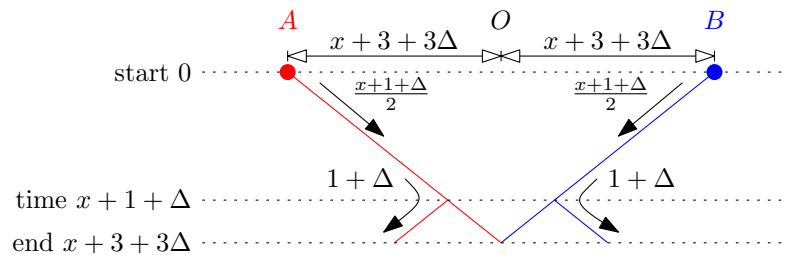


Figure 3. Illustration of Example 1. The trajectories of entities in A are in red. Those of entities in B are in blue.

Theorems 1 and 2 speak to the query frequency requirements for bounding congestion at a fixed target time, measured in terms of ply or degree individually. This leaves open the question of how these measures relate to one another. The following example demonstrates that in some cases the granularity required to bound congestion degree by $x + \Delta$ can be a factor $\Theta(\frac{x}{1+\Delta})$ smaller than that required to bound congestion ply by x .

Example 2. The example involves two clusters A and B of $(x + 1 + \Delta)/2$ point entities separated by distance $4(1 + \Delta)$. To maintain the uncertainty ply at most x it suffices to query $1 + \Delta$ entities in both clusters once every $2(1 + \Delta)$ steps, which can be achieved with query frequency one. Since the uncertainty regions associated with queried entities in cluster A never intersect the uncertainty regions associated with queried entities in cluster B, the largest possible ply involves entities in one cluster (say A) together with unqueried entities in the other cluster (B), for a total of x .

On the other hand, to maintain degree at most $x + \Delta$ no uncertainty region can be allowed to have radius $4(1 + \Delta)$. Thus, all $x + 1 + \Delta$ entities need to be queried with frequency at least $1/(4(1 + \Delta))$, giving a total query demand of $x + 1 + \Delta$ over any time interval of length $4(1 + \Delta)$.

Nevertheless, bounding congestion degree at a fixed target time cannot be too much worse than bounding congestion ply.

Theorem 3. The $\text{FTT-degree}[x + \Delta]$ scheme uses a query granularity that is at most a factor $\frac{x^2}{1+\Delta}$ smaller than the best, even clairvoyant, scheme that guarantees ply at most x .

Proof. In the $\text{FTT-degree}[x + \Delta]$ scheme, m_r (the number of safe survivors in round r) is $O(n_r(x + \Delta))$. Thus, the “extra” queries (to handle the safe survivors) are at most a factor $x + \Delta$ more numerous than the queries to handle the unsafe survivors. Suppose that $\text{FTT-ply}[x + \Delta]$ is modified so that the queries in round r occur with granularity $\frac{\tau}{2^{r+1}n_r}$ (i.e., half of their previous granularity), completing at the midpoint of the round, and that $\text{FTT-degree}[x + \Delta]$ is modified so that the queries in round r occur with granularity $\frac{\tau}{2^{r+1}(n_r+m_r)}$ (i.e., half of their previous granularity), starting at the midpoint of the round. Since the queries of $\text{FTT-degree}[x + \Delta]$ in round r now occur after the corresponding queries in $\text{FTT-ply}[x + \Delta]$, it is straightforward to see that the unsafe survivors in round r of $\text{FTT-degree}[x + \Delta]$ are no more numerous than the unsafe survivors in round r of $\text{FTT-ply}[x + \Delta]$. It follows that the granularity of the queries in $\text{FTT-degree}[x + \Delta]$ is no more than a factor $\Theta(x + \Delta)$ smaller than that of $\text{FTT-ply}[x + \Delta]$. $\square$

4. Towards Query Optimization to Maintain Bounded Congestion Continuously

What can be said about the relationship between the problem of maintaining bounded congestion at a fixed target time and the more general problem of maintaining bounded congestion at all times?

The following example shows that there are entity configurations, even ones that do not change over time, for which the query granularity required to maintain bounded congestion at all times is significantly smaller than that which suffices to obtain the same bounded congestion at a fixed target time.

Example 3. Consider a collection of $n/2$ well-separated pairs, where the i -th pair has separation $4i - 1$. Uncertainty ply (and degree) can be kept at one at a deadline n time units from the start, by a scheme that queries entities with granularity one in decreasing order of their separation. On the other hand to maintain degree/ply one continuously, the i -th entity pair must be queried at least once every $4i - 1$ steps, so over any time interval of length n , $\Omega(n \ln n)$ queries are required.

Our fixed target time results focus on the congestion of uncertainty regions of entities at one time τ . To describe continuously maintaining bounded congestion, we need a notation that describes an $\mathcal{E}$ -configuration and its uncertainty at any time t . Let $Z(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$ denote such a configuration at time t and let $B_i(x, t)$, $\sigma_i(x, t)$, and $r_i(x, t)$ be the corresponding x -ball, x -separation, and x -radius of entity e_i in $Z(t)$.

The continuous strategy described in our companion paper [3] repeatedly queries individual entities just in time to maintain reasonably accurate information about their x -separation; that is, the perception of their x -separation is close to their true x -separation. Relying on the accuracy of this perception, the continuous strategy is able to schedule queries in a way that simultaneously (i) maintains the desired bound on congestion, and (ii) sustains the accurate perception of x -separation over time.

In the following subsection, we outline how, once suitably initialized, the accurate perception of x -separation can be sustained using just-in-time queries based on perceived x -separation, leaving the details to the companion paper. This serves to motivate the second subsection in which we describe how the accurate perception of x -separation can be initialized at some time $t_0 > 0$, from a state of unbounded uncertainty of entity locations at time zero, using query granularity that is competitive with any other initialization scheme. This is achieved by a modified version of the FTT-degree[$x + \Delta$] scheme of Section 3, using higher query frequency and a more restrictive criterion than $(x + \Delta)$ -degree-safety.

4.1. Maintaining Accurate Perception of x -Separation

For any query scheme, the true location of a moving entity e_i at time t , $\zeta_i(t)$, may differ from its *perceived location*, $\zeta_i(p_i(t))$, its location at the time of its most recent query. Let $N_i(x, t)$ be e_i plus the set of x entities whose perceived locations at time t are closest to the perceived location of e_i at time t . The *perceived x -separation* of e_i at time t , denoted $\tilde{\sigma}_i(x, t)$, is the separation between e_i and its perceived x th-nearest-neighbour at time t , i.e., $\tilde{\sigma}_i(x, t) = \max_{e_j \in N_i(x, t)} \|\zeta_i(p_i(t)) - \zeta_j(p_j(t))\| - 2$. The *perceived x -radius* of e_i at time t , denoted $\tilde{r}_i(x, t)$, is just $1 + \tilde{\sigma}_i(x, t)$.

Since a scheme only knows the perceived locations of the entities, it is important that each entity e_i be probed sufficiently often that its perceived x -separation $\tilde{\sigma}_i(x, t)$ closely approximates its true x -separation $\sigma_i(x, t)$ at all times t . The following proposition, whose proof appears in the companion paper [3] (see also [14], Lemma 9) asserts that once a close relationship between perception and reality has been established, it can be sustained by ensuring that the time between queries to an entity is bounded by some small fraction of its perceived x -separation.

Proposition 1. Suppose that for some t_0 and for all entities e_i ,

- (i) $\sigma_i(x, p_i(t_0))/2 \leq \tilde{\sigma}_i(x, p_i(t_0)) \leq 3\sigma_i(x, p_i(t_0))/2$, [perception is close to reality for e_i at time $p_i(t_0)$] and
- (ii) for any $t \geq t_0$, $t - p_i(t) \leq \frac{c_{d,x}}{1+c_{d,x}} \tilde{\sigma}_i(x, p_i(t))/12$ [all queries are conducted promptly based on perception].

Then for all entities e_i , $\sigma_i(x, t)/2 \leq \tilde{\sigma}_i(x, t) \leq 3\sigma_i(x, t)/2$, for all $t \geq p_i(t_0)$.

4.2. Initializing Accurate Perception of x -Separation

To obtain the preconditions of Proposition 1, we could assume that all entities are queried very quickly using low granularity for a short initialization phase. We next show how to use a modified version of the FTT scheme of Section 3 to obtain these preconditions using granularity that is competitive with any scheme that guarantees uncertainty degree at most x from time t_0 onward.

Lemma 1. *For any Δ , $0 \leq \Delta \leq x$, and any target time $t_0 \geq 0$, there exists an initialization scheme that guarantees*

- (i) $\sigma_i(x + \Delta, t_0)/2 \leq \tilde{\sigma}_i(x + \Delta, t_0) \leq 3\sigma_i(x + \Delta, t_0)/2$, and
- (ii) $t_0 - p_i(t_0) \leq \lambda_{d,x} \tilde{\sigma}_i(x + \Delta, p_i(t_0))/12$.

using minimum query granularity over the interval $[0, t_0]$ that is at most $\Theta(\frac{x+\Delta}{1+\Delta})$ smaller than the minimum query granularity, over the interval $[0, (a+1)t_0]$, used by any other scheme that guarantees uncertainty degree at most x in the interval $[t_0, (a+1)t_0]$, where $a = 64/(5\lambda_{d,x})$.

Proof. The FTT $[x + \Delta]$ scheme described in Section 3 is modified as follows: Instead of conducting each successive query round-robin within half of the time remaining to the target time, we use just a fraction $1/16$ of the time remaining. This means that with each successive round-robin phase, the time remaining to the target decreases by a factor $b = 15/16$.

We say that an entity is $(x + \Delta)$ -degree-super-safe at time $t_0 - b^s t_0$ (i.e., $b^s t_0$ units before the target time) if its projected uncertainty region at that time is separated by distance at least $ab^s t_0$ from the projected uncertainty regions of all but at most $x + \Delta - 1$ other entities (so that its $(x + \Delta)$ -separation at the target time is guaranteed to be at least $ab^s t_0$). This ensures that when e_i is declared $(x + \Delta)$ -degree-super-safe both the true and perceived $(x + \Delta)$ -separation of e_i at the target time are at least $ab^s t_0$ (no matter what further queries are performed).

Assuming that e_i is $(x + \Delta)$ -degree-super-safe at time $b^s t_0$ before the target but not at time $b^{s-1} t_0$ before the target, both the true and perceived $(x + \Delta)$ -separation of e_i at the target time are at most $(a/b + 4/b^2)b^s t_0$. Indeed, at time $b^{s-1} t_0$ before target, the separation of surviving projected uncertainty regions is at most $ab^{s-1} t_0$, and while the x -separation at the target time could be more than this, it cannot be more than four times the radius of any surviving uncertainty region (which is less than $b^{s-2} t_0$) plus $ab^{s-1} t_0$. Since $(a/b + 4/b^2)b^s t_0 < (16a/15 + 5)b^s t_0$, it follows that (i) $\tilde{\sigma}_i(x + \Delta, t_0) \leq \frac{16a/15+5}{a} \sigma_i(x + \Delta, t_0)$, and (ii) $t_0 - p_i(t_0) \leq \frac{16}{15a} \tilde{\sigma}_i(x + \Delta, p_i(t_0))$. Choosing a large enough ($a \geq 64/(5\lambda_{d,x})$ suffices) guarantees the desired properties.

Following the analysis of the FTT $[x + \Delta]$ scheme, if the s th query round uses q_s queries then any query that guarantees uncertainty degree at most x at time $t_0 + ab^s t_0/2 < (a+1)t_0$ must use at least $\Theta(\frac{q_s(1+\Delta)}{x+\Delta})$ queries between time $t_0 - b^{s-1} t_0$, the start of the s th query round, and time $t_0 + ab^s t_0/2$; otherwise, some entity that was not $(x + \Delta)$ -degree-super-safe at time $t_0 - b^{s-1} t_0$ would not be x -safe at time $t_0 + ab^s t_0/2$. It follows that this initialization scheme uses a minimum query granularity that is competitive to within a factor of $\Theta(\frac{x+\Delta}{1+\Delta})$ with the minimum granularity used by any other scheme that guarantees the uncertainty degree is at most x . $\square$

5. Discussion

To this point, we have assumed that the distance between entity centers is always at least 2 (i.e., entities never properly intersect), and that the encroachment threshold is exactly 2 (i.e., we are only concerned with avoiding entity contact). However, without changing the units of distance and time, we can model a collection of unit-radius entities, any pair of which possibly intersect but whose centers always maintain distance at least some positive constant $\rho_0 < 2$, by simply scaling the constant $c_{d,x}$ by $\rho_0/2$. Similarly (and simultaneously), we can model a collection of unit-radius entities with encroachment threshold $\Xi > 2$ by

(i) changing the *basic uncertainty radius* (the radius of the uncertainty region of an entity immediately after it has been queried) to $\Xi/2$ thereby ensuring that entities with disjoint uncertainty regions do not encroach on one another, and (ii) changing X_d to be the largest x such that $c_{d,x} \geq \Xi - 2$ since for x exceeding this changed X_d there can be at most $x - 1$ entities that are within the encroachment threshold of any fixed entity.

This relaxation of our basic assumptions makes it possible to additionally relax our assumption that location queries are answered exactly since potential errors in the response to location queries can be modeled by an increase in the basic uncertainty radius. Furthermore, it increases significantly the scope of applications of our results.

For example, recall the problem concerning collision avoidance mentioned in the introduction. Here, it would be useful to consider encroachment occurring well before contact (inviting the use of an encroachment threshold $\Xi > 2$). It follows from our results that, by achieving an uncertainty degree at most x at time τ , we obtain for each entity e_i a certificate identifying the, at most $x - 1$, other entities that could potentially encroach (in this more general sense) upon e_i at that time (those warranting more careful local monitoring).

An additional application, considered in [2], concerns entities that are mobile transmission sources with associated broadcast ranges that one would expect might sometimes properly intersect, where the goal is to minimize the number of broadcast channels at time τ so as to eliminate potential transmission interference. In this case, achieving uncertainty thickness at most x using minimum query frequency serves to obtain a fixed bound on the number of broadcast channels required at time τ , an objective that seems to be at least as well-motivated as optimizing the number of channels for a fixed query frequency (the objective in [2]).

Author Contributions: Both authors contributed to the investigation and writing of the presented research. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded in part by Discovery Grants from the Natural Sciences and Engineering Research Council of Canada.

Data Availability Statement: Data sharing is not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Evans, W.; Kirkpatrick, D.; Löffler, M.; Staals, F. Minimizing co-location potential of moving entities. *Siam J. Comput.* **2016**, *45*, 1870–1893. [CrossRef]
2. Busto, D.; Evans, W.; Kirkpatrick, D. Minimizing Interference Potential Among Moving Entities. In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), San Diego, CA, USA, 6–9 January 2019; pp. 2400–2418.
3. Evans, W.; Kirkpatrick, D. Frequency-Competitive Query Strategies to Maintain Low Congestion Potential Among Moving Entities. In Proceedings of the Workshop on Approximation and Online Algorithms, Amsterdam, The Netherlands, 7–8 September 2023; pp. 14–28. Journal version in preparation.
4. Guibas, L.J. Kinetic Data Structures: A State of the Art Report. In Proceedings of the Third Workshop on the Algorithmic Foundations of Robotics on Robotics: The Algorithmic Perspective, WAFR '98, Houston, TX, USA, 5–7 March 1998; pp. 191–209.
5. Basch, J.; Guibas, L.J.; Hershberger, J. Data Structures for Mobile Data. *J. Algorithms* **1999**, *31*, 1–28. [CrossRef]
6. Guibas, L.J.; Roeloffzen, M. Modeling Motion. In *Handbook of Discrete and Computational Geometry*; Toth, C.D., O'Rourke, J., Goodman, J.E., Eds.; CRC Press: Boca Raton, FL, USA, 2017; Chapter 53, pp. 1401–1420.
7. de Berg, M.; Roeloffzen, M.; Speckmann, B. Kinetic compressed quadrees in the black-box model with applications to collision detection for low-density scenes. In Proceedings of the European Symposium on Algorithms, Ljubljana, Slovenia, 10–12 September 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 383–394.
8. de Berg, M.; Roeloffzen, M.; Speckmann, B. Kinetic convex hulls, Delaunay triangulations and connectivity structures in the black-box model. *J. Comput. Geom.* **2012**, *3*, 222–249.
9. de Berg, M.; Roeloffzen, M.; Speckmann, B. Kinetic 2-centers in the black-box model. In Proceedings of the Symposium on Computational Geometry, Rio de Janeiro, Brazil, 17–20 June 2013; pp. 145–154.
10. Kahan, S. A Model for Data in Motion. In Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing, STOC '91, New Orleans, LO, USA, 5–8 May 1991; pp. 265–277.
11. Kahan, S. Real-Time Processing of Moving Data. Ph.D. Thesis, University of Washington, Seattle, WA, USA, 1991.

12. Erlebach, T.; Hoffmann, M. Query-competitive algorithms for computing with uncertainty. *Bull. Eur. Assoc. Theor. Comput. Sci.* **2015**, *2*, 116.
13. Löffler, M.; Snoeyink, J. Delaunay triangulation of imprecise points in linear time after preprocessing. *Comput. Geom. Theory Appl.* **2010**, *43*, 234–242. [[CrossRef](#)]
14. Evans, W.; Kirkpatrick, D. Frequency-Competitive Query Strategies to Maintain Low Congestion Potential among Moving Entities. *arXiv* **2023**, arXiv:2205.09243.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.