MINIMIZING INTERFERENCE POTENTIAL AMONG MOVING ENTITIES

Daniel Busto

Ericsson, Montreal, Canada

Will Evans

David Kirkpatrick

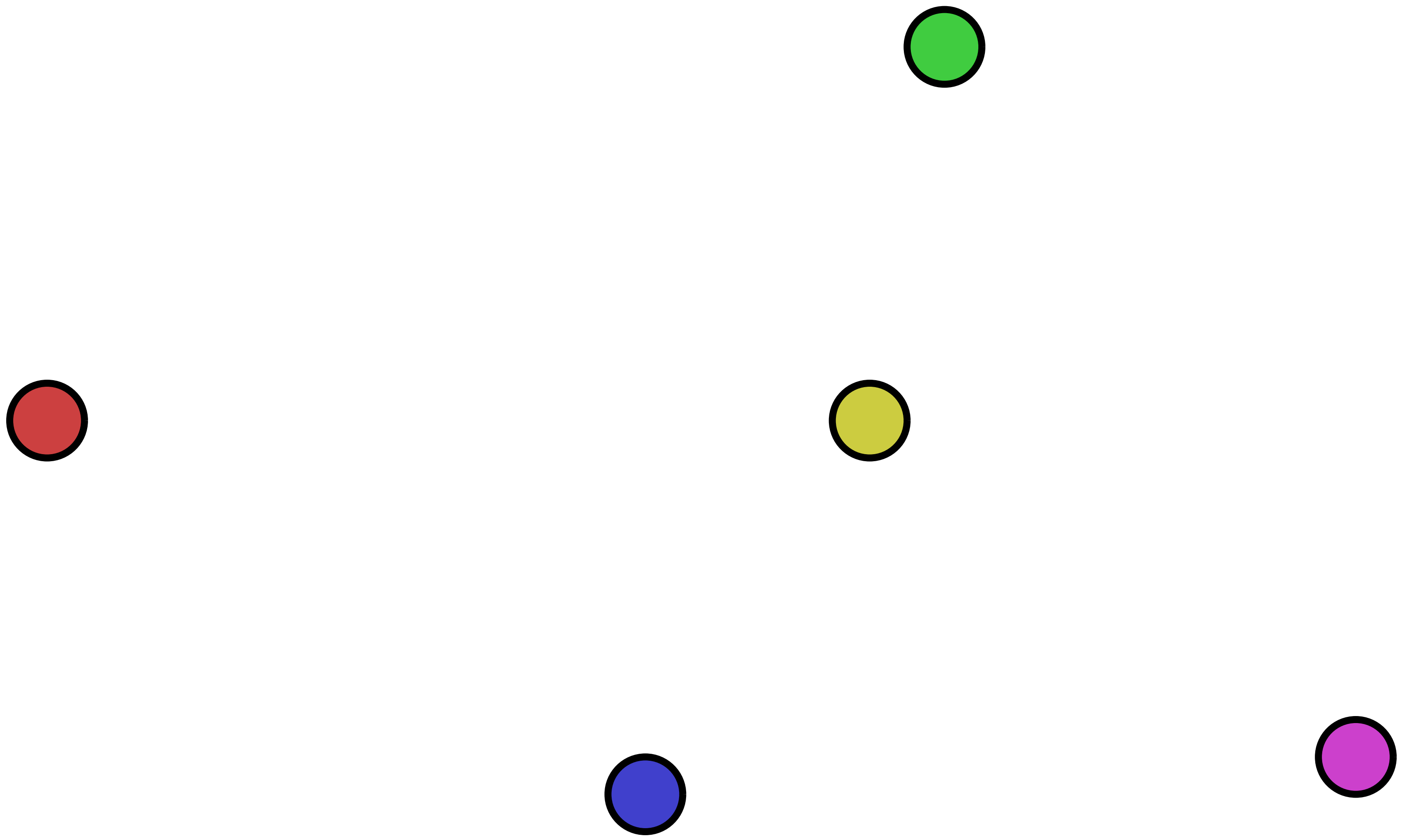
University of British Columbia, Canada

Moving entities

- Airplanes
- Smart phones
- Mobile transmitters

Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.

Suppose we know their positions now.

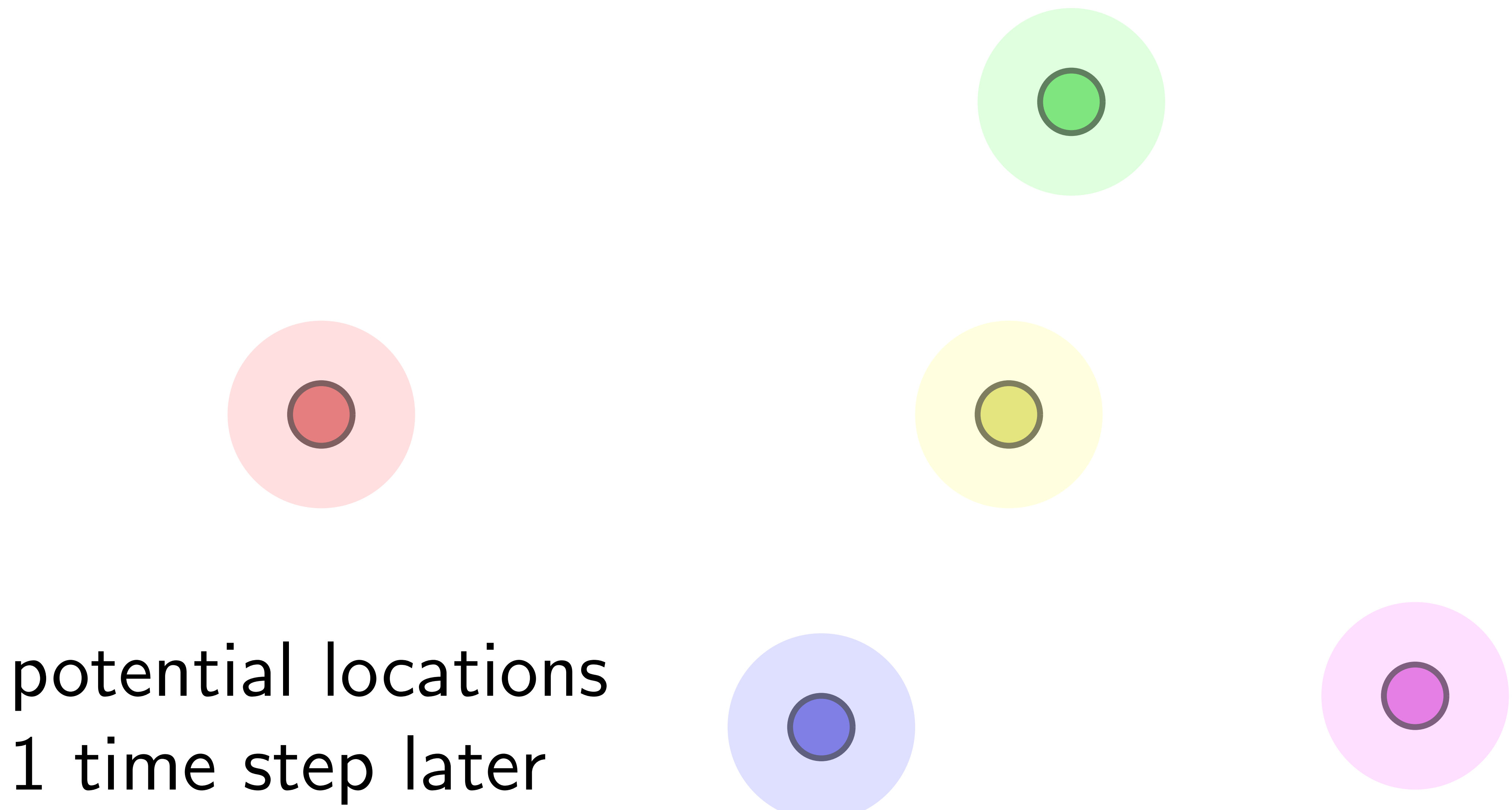


Moving entities

- Airplanes
- Smart phones
- Mobile transmitters

Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.

Suppose we know their positions now.

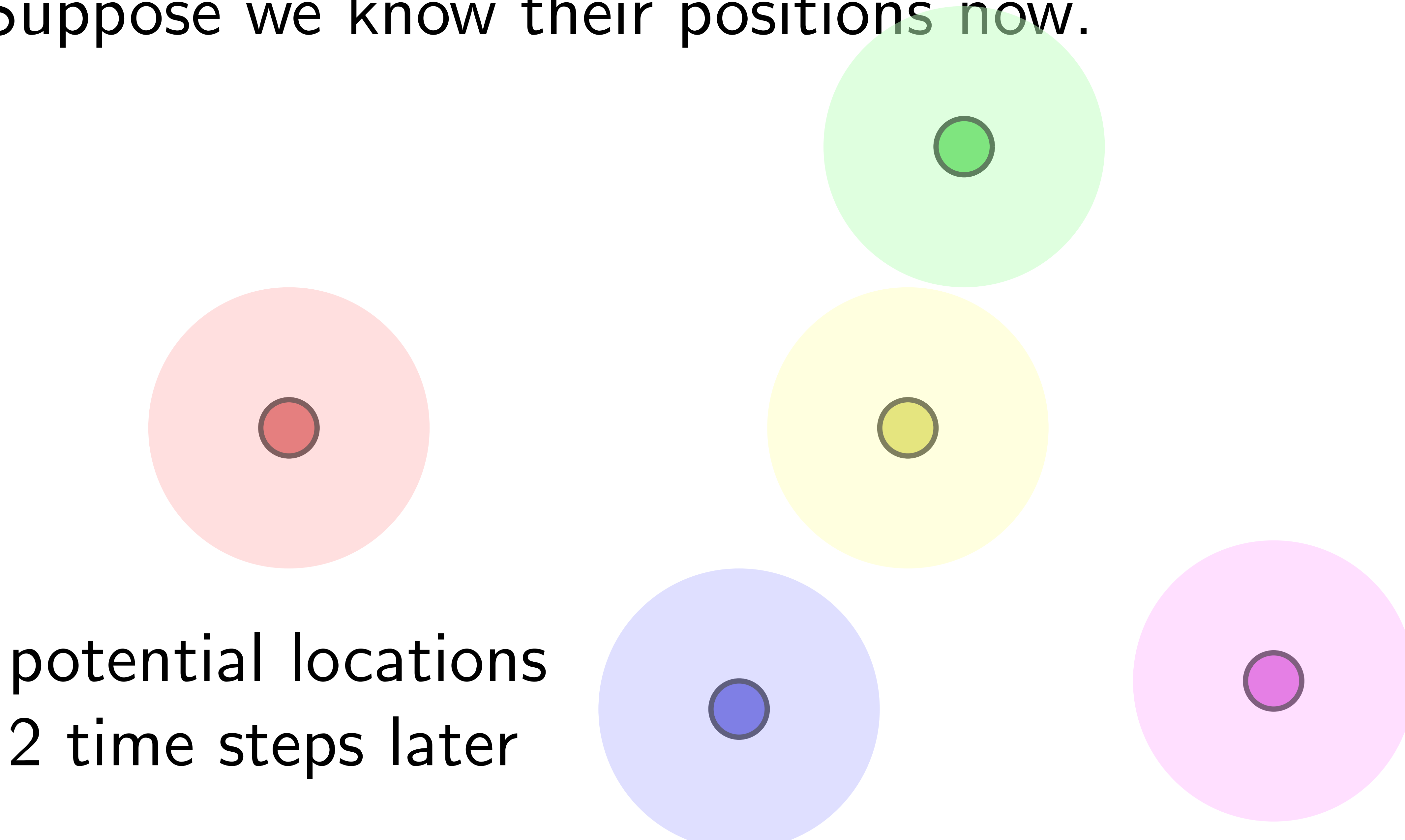


Moving entities

- Airplanes
- Smart phones
- Mobile transmitters

Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.

Suppose we know their positions now.

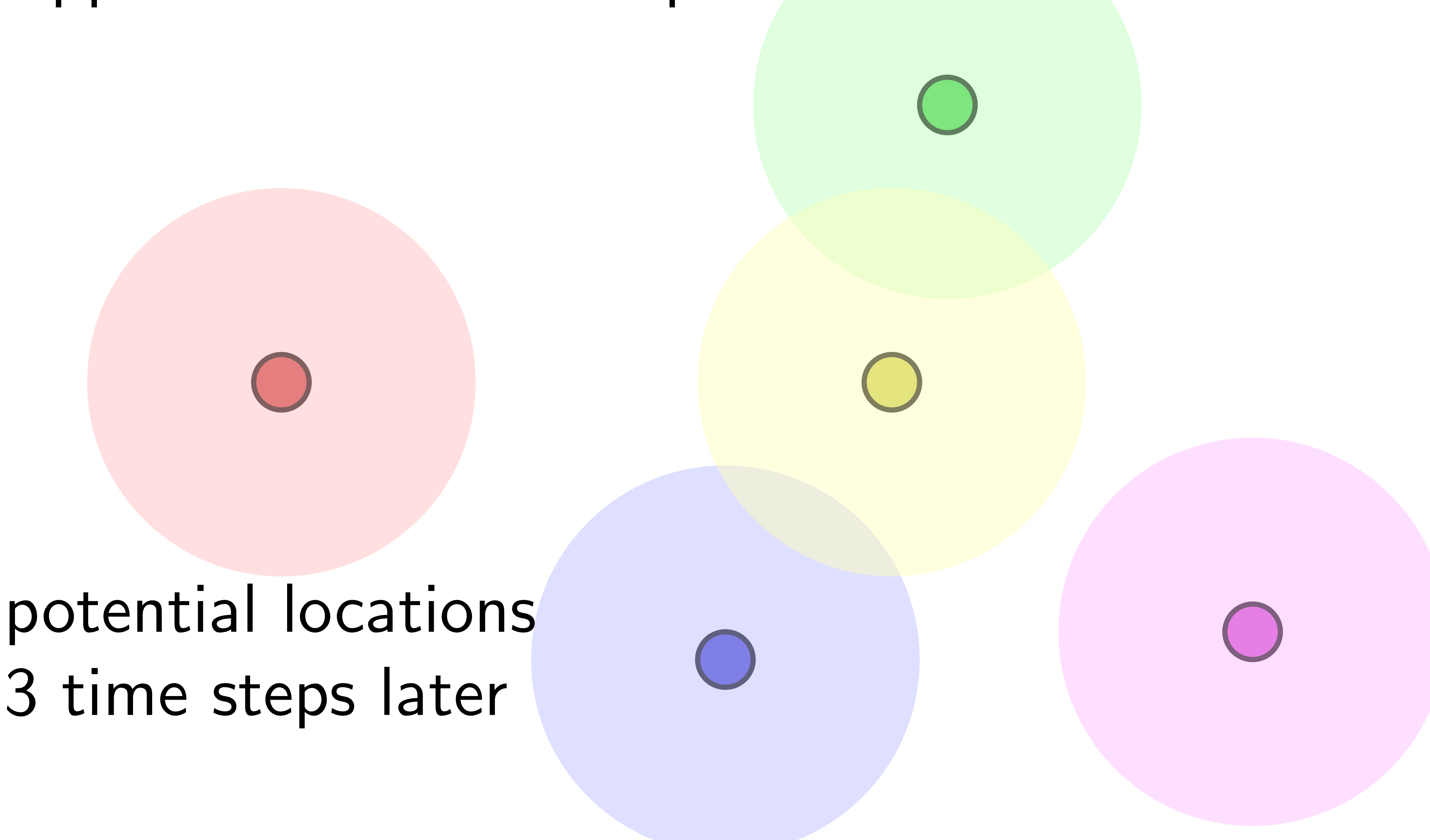


Moving entities

- Airplanes
- Smart phones
- Mobile transmitters

Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.

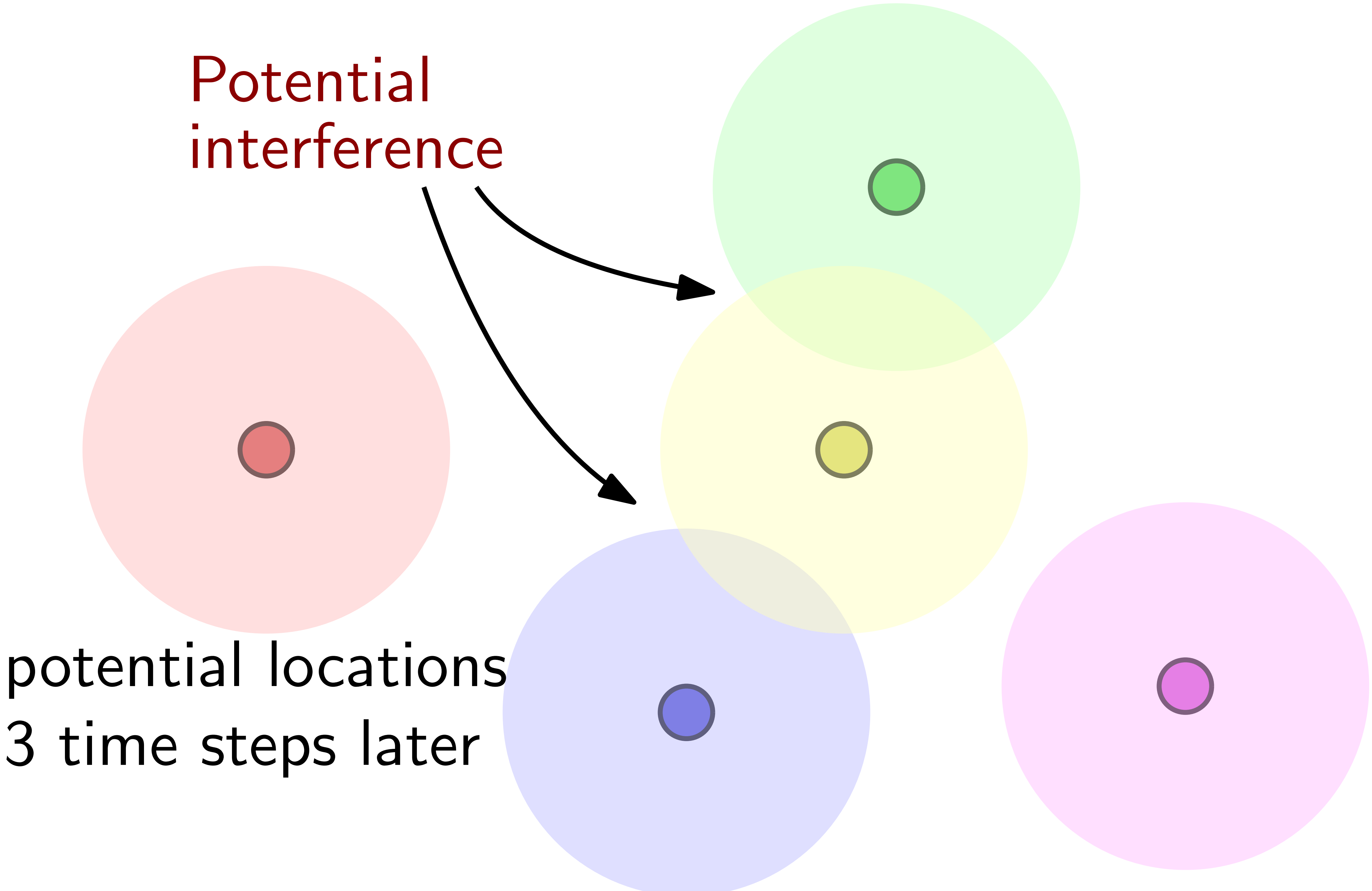
Suppose we know their positions now.



Moving entities

- Airplanes
- Smart phones
- Mobile transmitters

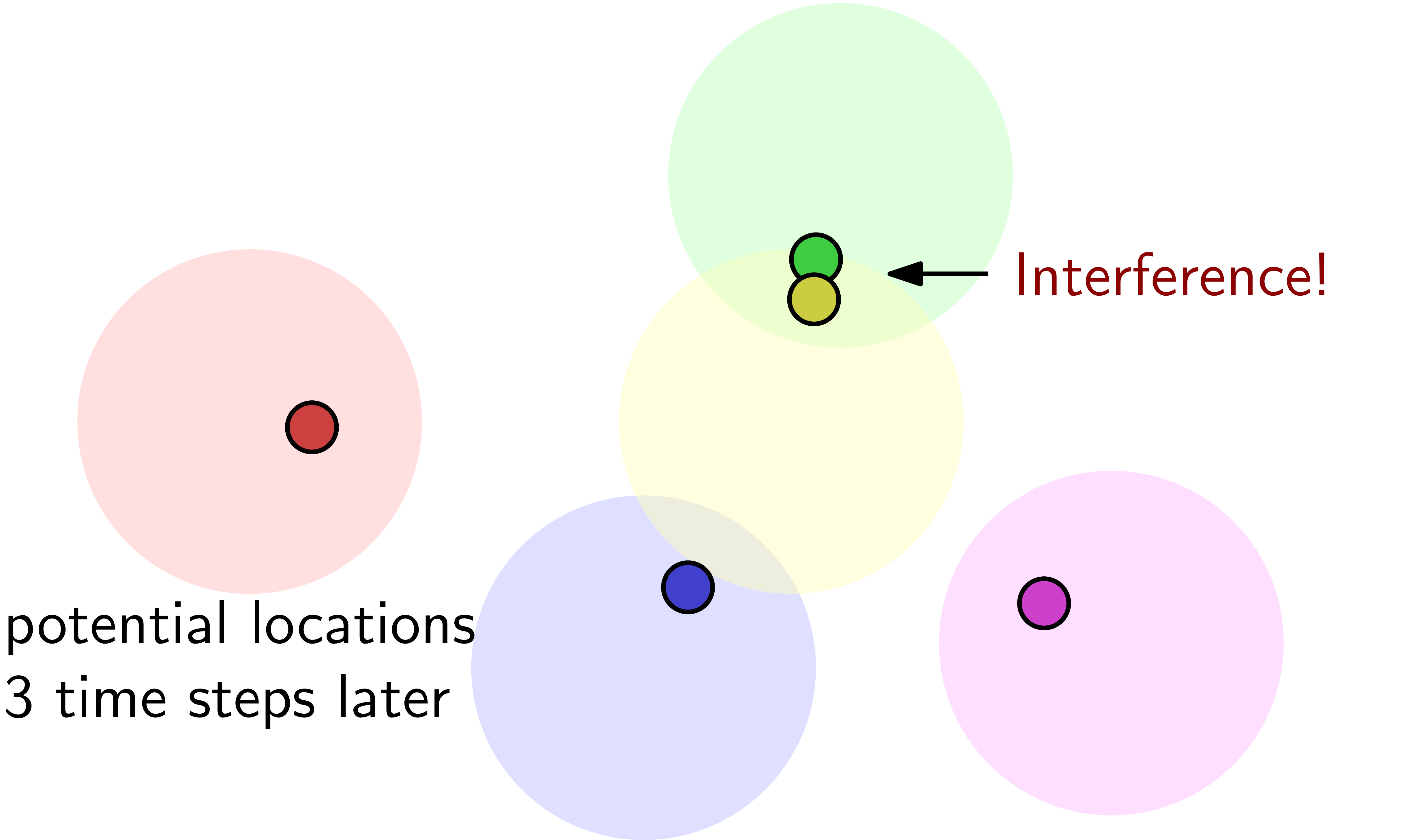
Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.



Moving entities

• Airplanes • Smart phones • Mobile transmitters

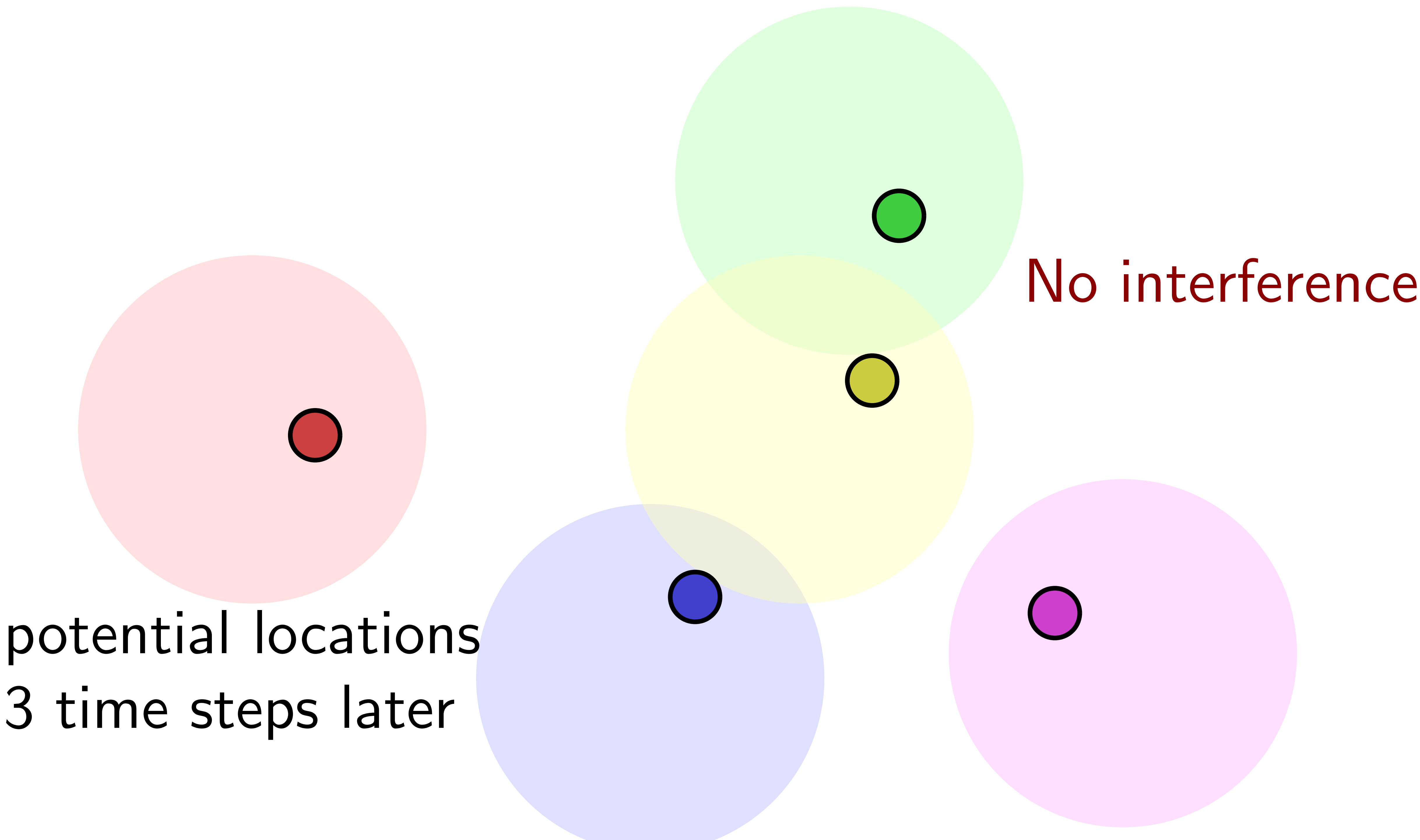
Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.



Moving entities

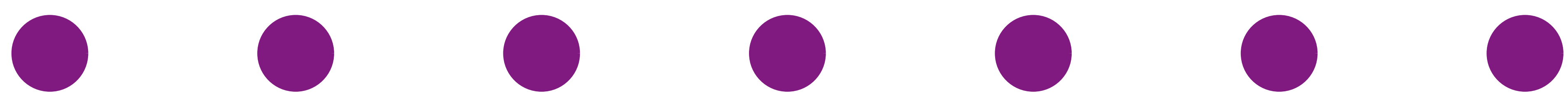
- Airplanes
- Smart phones
- Mobile transmitters

Equi-sized balls in $\mathbb{R}^d$ that move with bounded speed.



Goal: Minimize potential interference

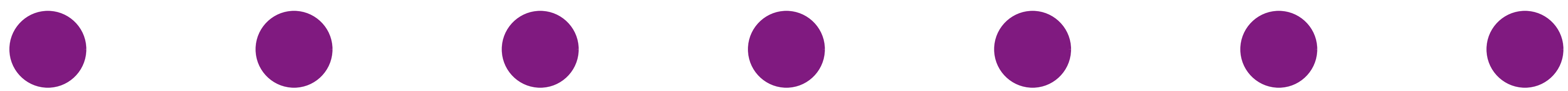
How: Query one entity per time step for its position



Goal: Minimize potential interference

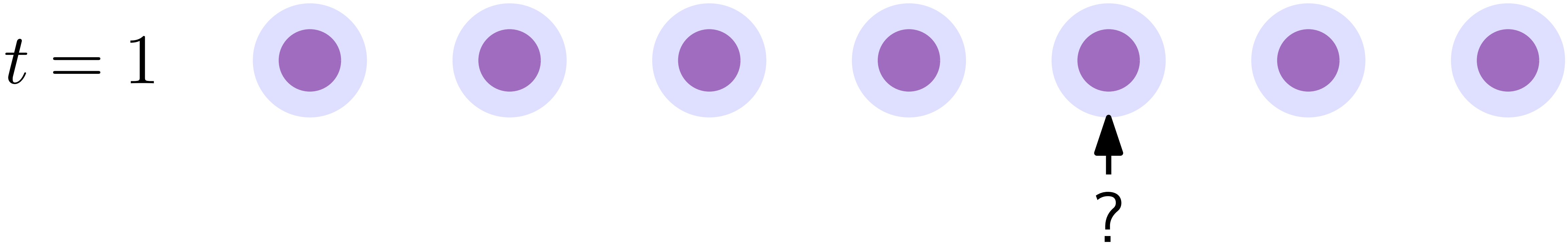
How: Query one entity per time step for its position

$t = 0$



Goal: Minimize potential interference

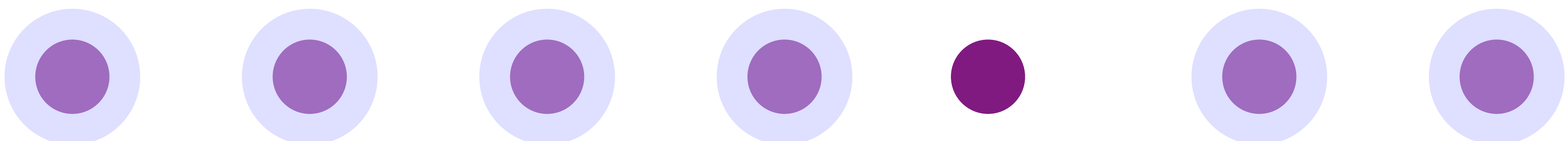
How: Query one entity per time step for its position



Goal: Minimize potential interference

How: Query one entity per time step for its position

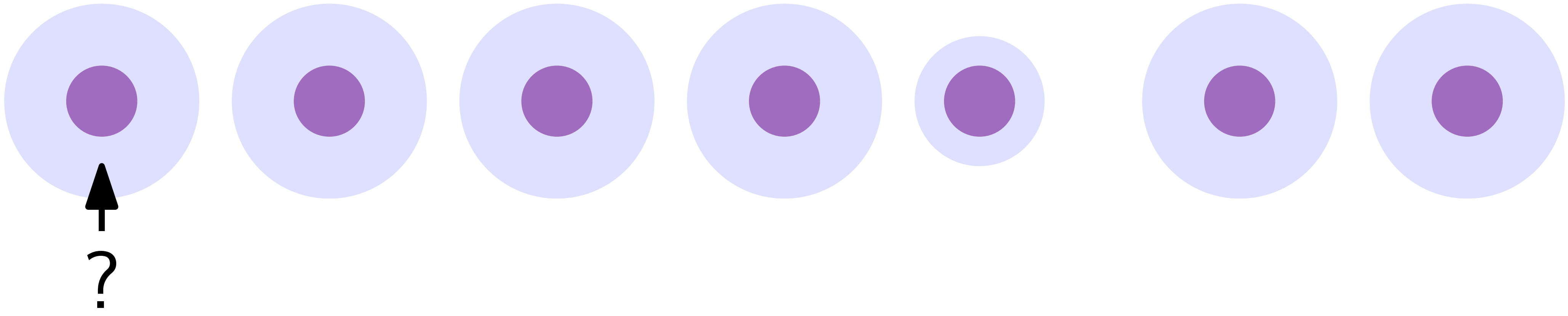
$t = 1$



Goal: Minimize potential interference

How: Query one entity per time step for its position

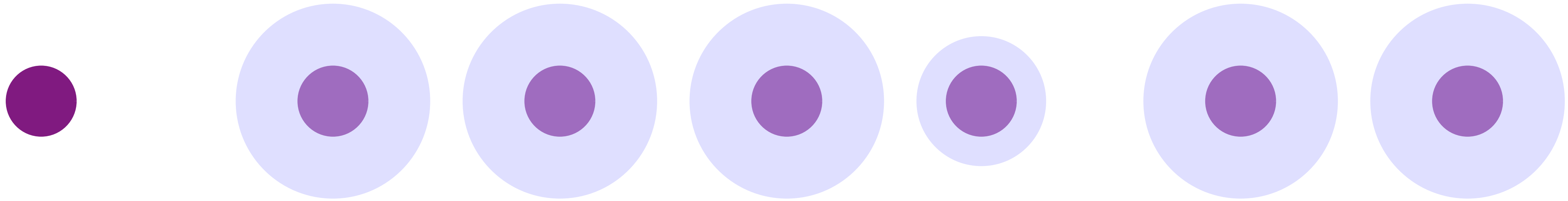
$t = 2$



Goal: Minimize potential interference

How: Query one entity per time step for its position

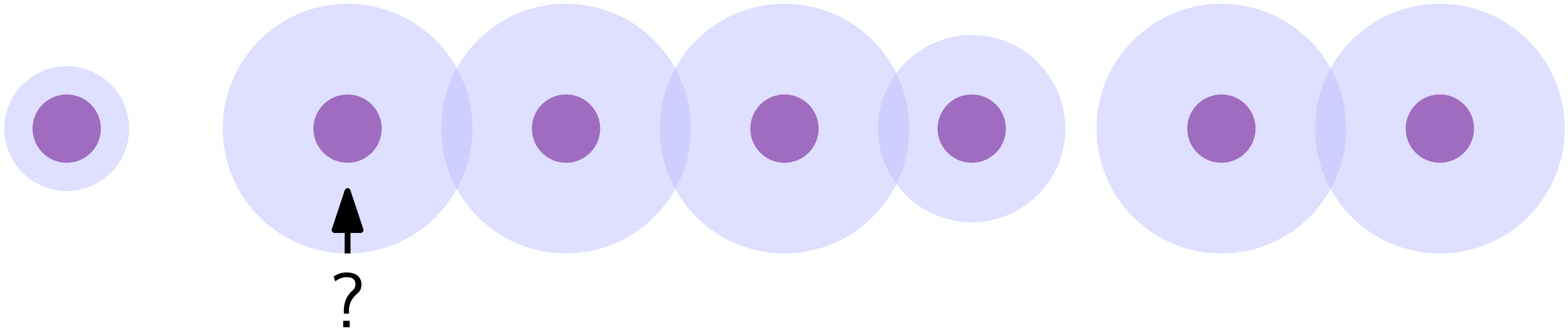
$t = 2$



Goal: Minimize potential interference

How: Query one entity per time step for its position

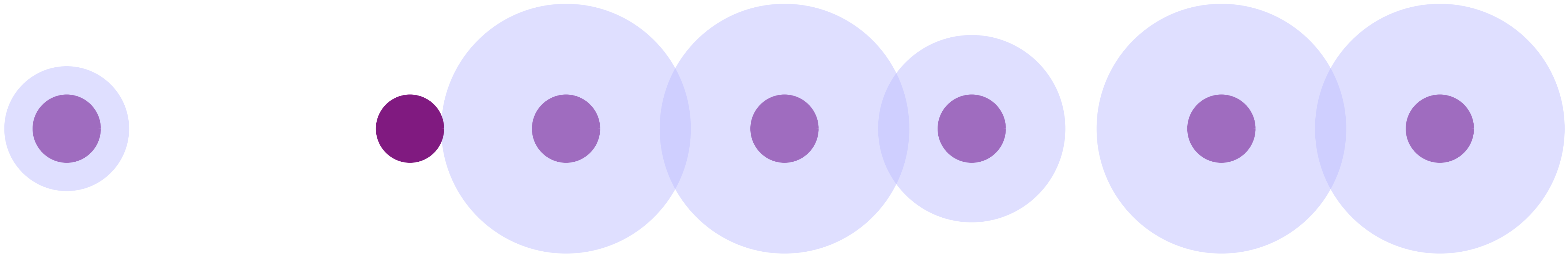
$t = 3$



Goal: Minimize potential interference

How: Query one entity per time step for its position

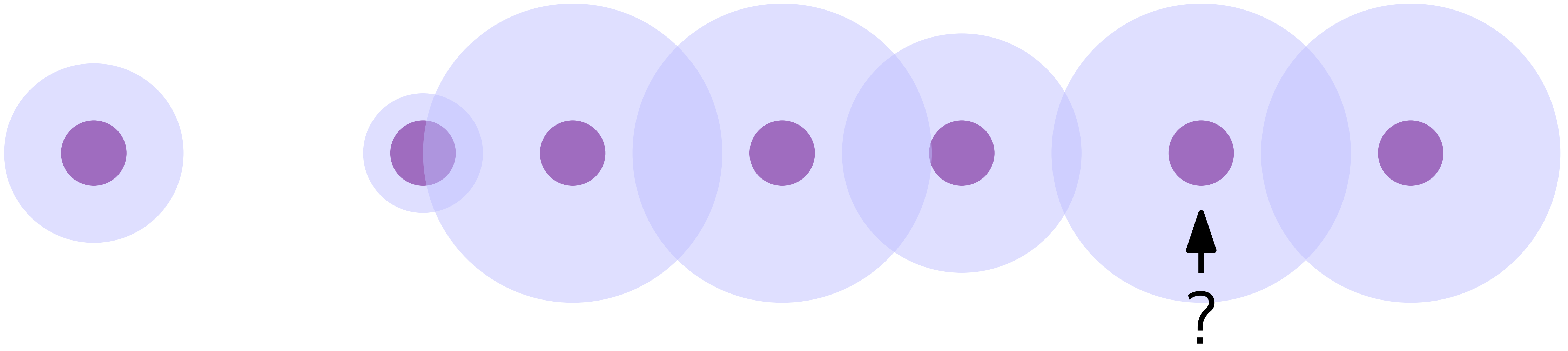
$t = 3$



Goal: Minimize potential interference

How: Query one entity per time step for its position

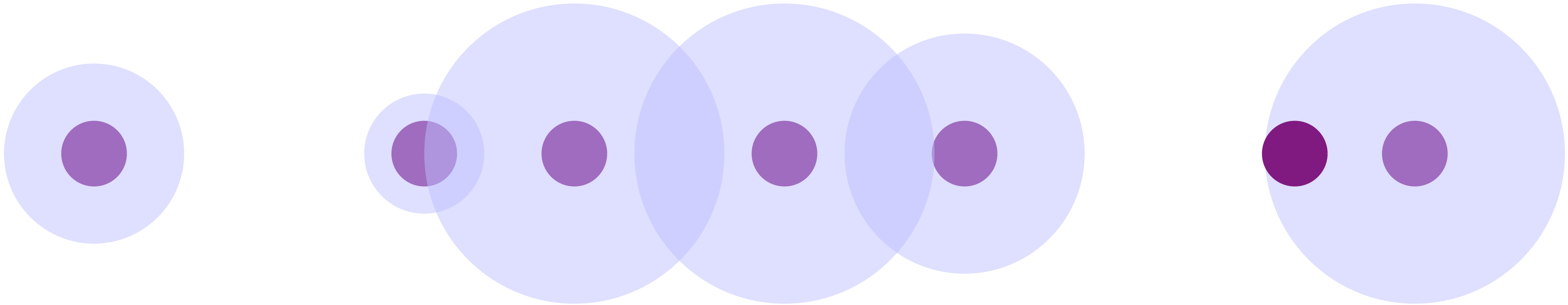
$t = 4$



Goal: Minimize potential interference

How: Query one entity per time step for its position

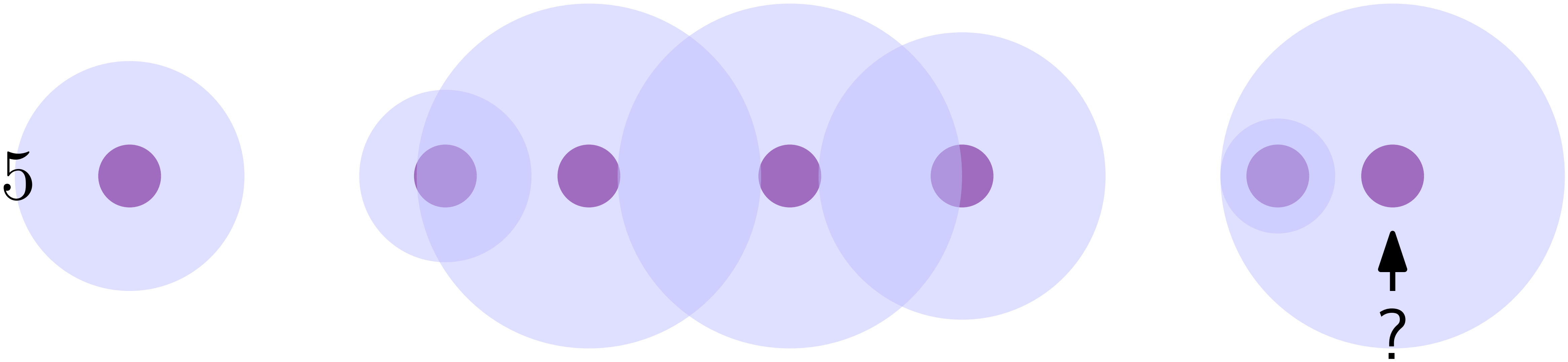
$t = 4$



Goal: Minimize potential interference

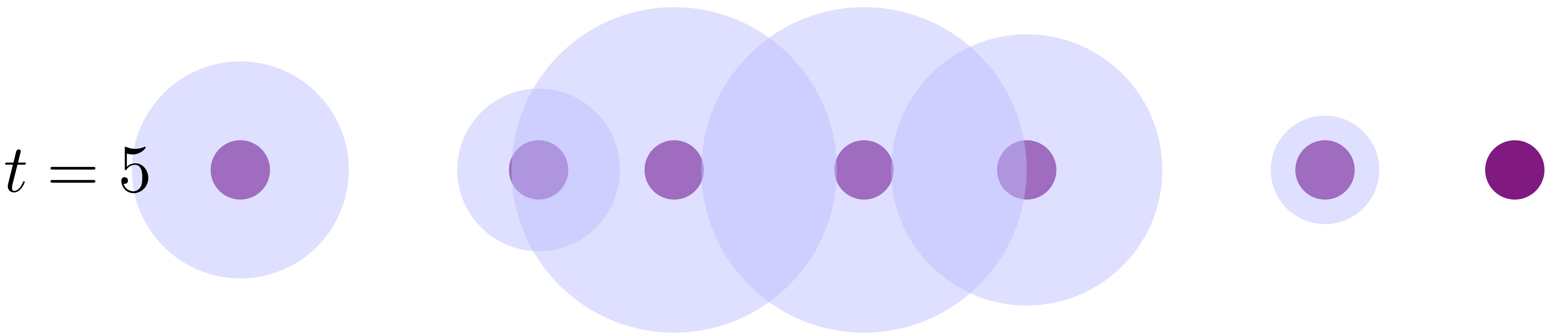
How: Query one entity per time step for its position

$t = 5$



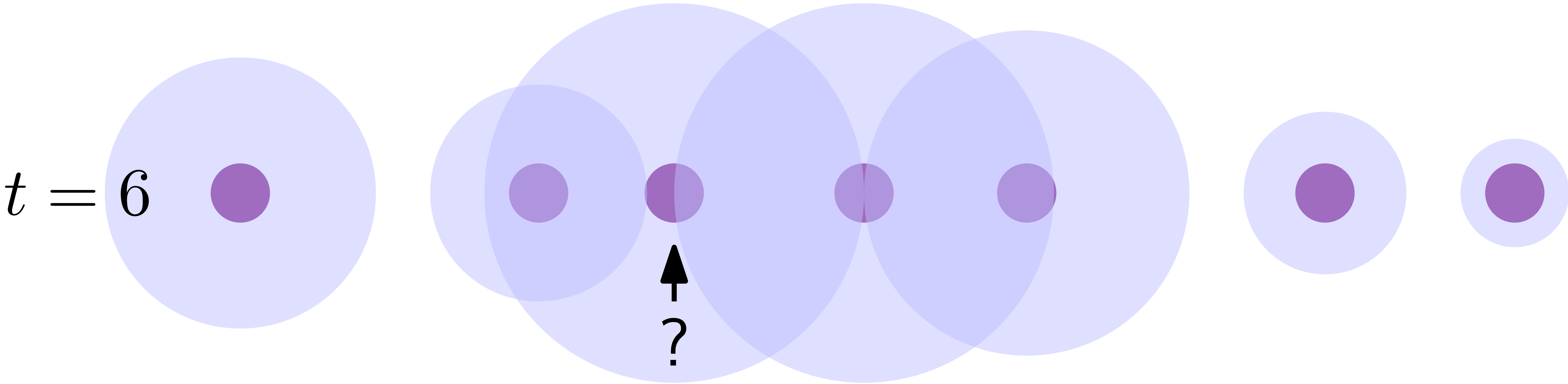
Goal: Minimize potential interference

How: Query one entity per time step for its position



Goal: Minimize potential interference

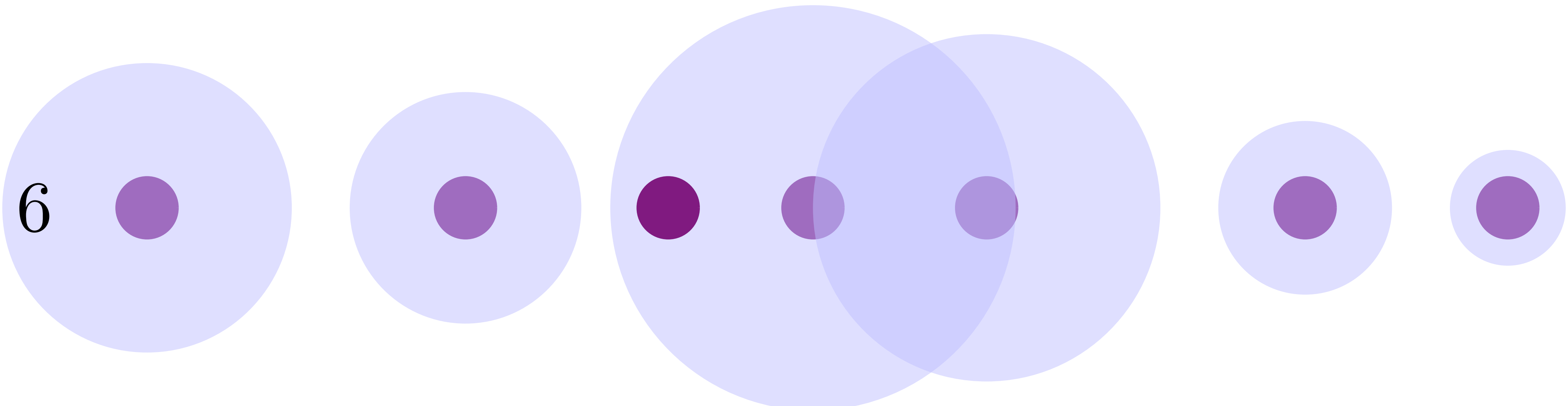
How: Query one entity per time step for its position



Goal: Minimize potential interference

How: Query one entity per time step for its position

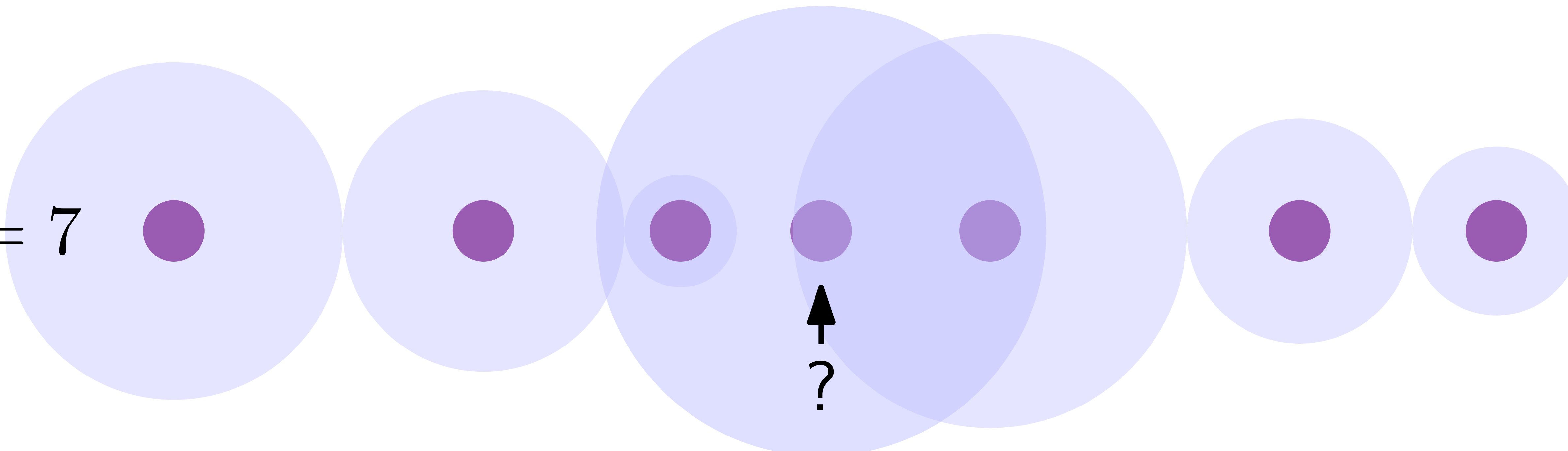
$t = 6$



Goal: Minimize potential interference

How: Query one entity per time step for its position

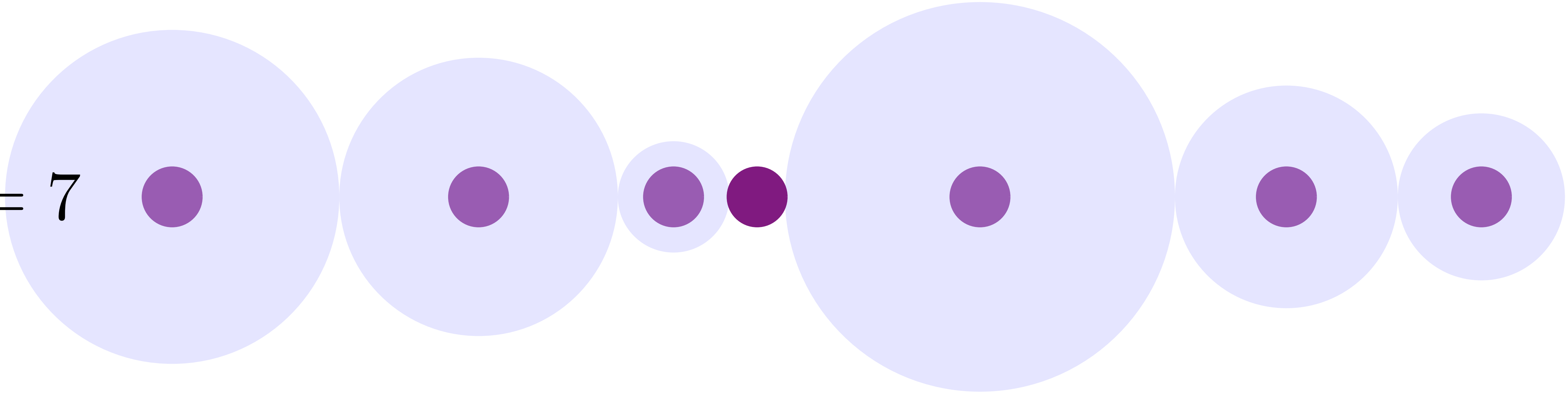
$t = 7$



Goal: Minimize potential interference

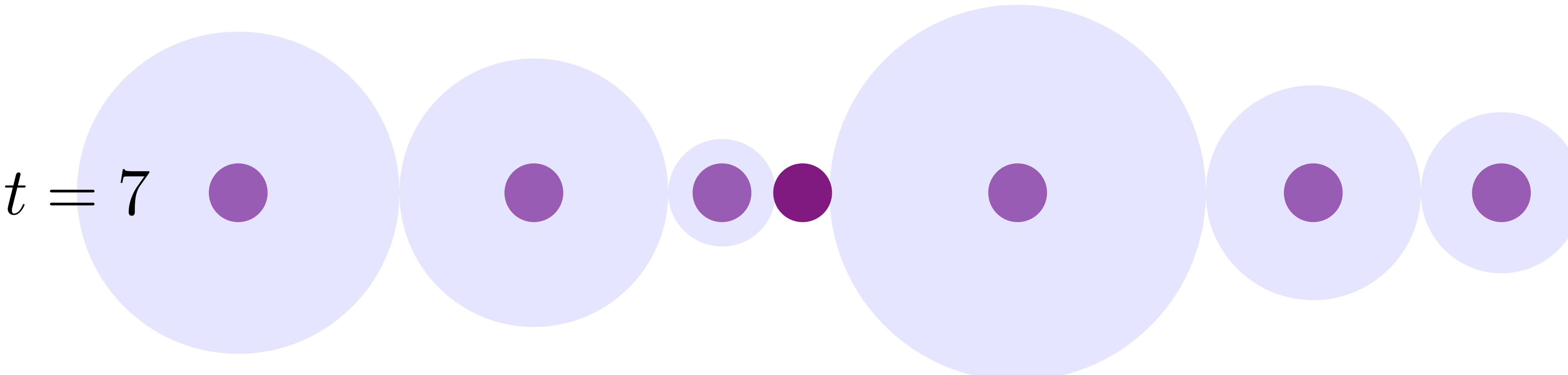
How: Query one entity per time step for its position

$t = 7$



Goal: Minimize potential interference

How: Query one entity per time step for its position



Related Goals:

Minimize queries to calculate some function of entities

Kahan '91, Erlebach&Hoffmann '15 (survey)

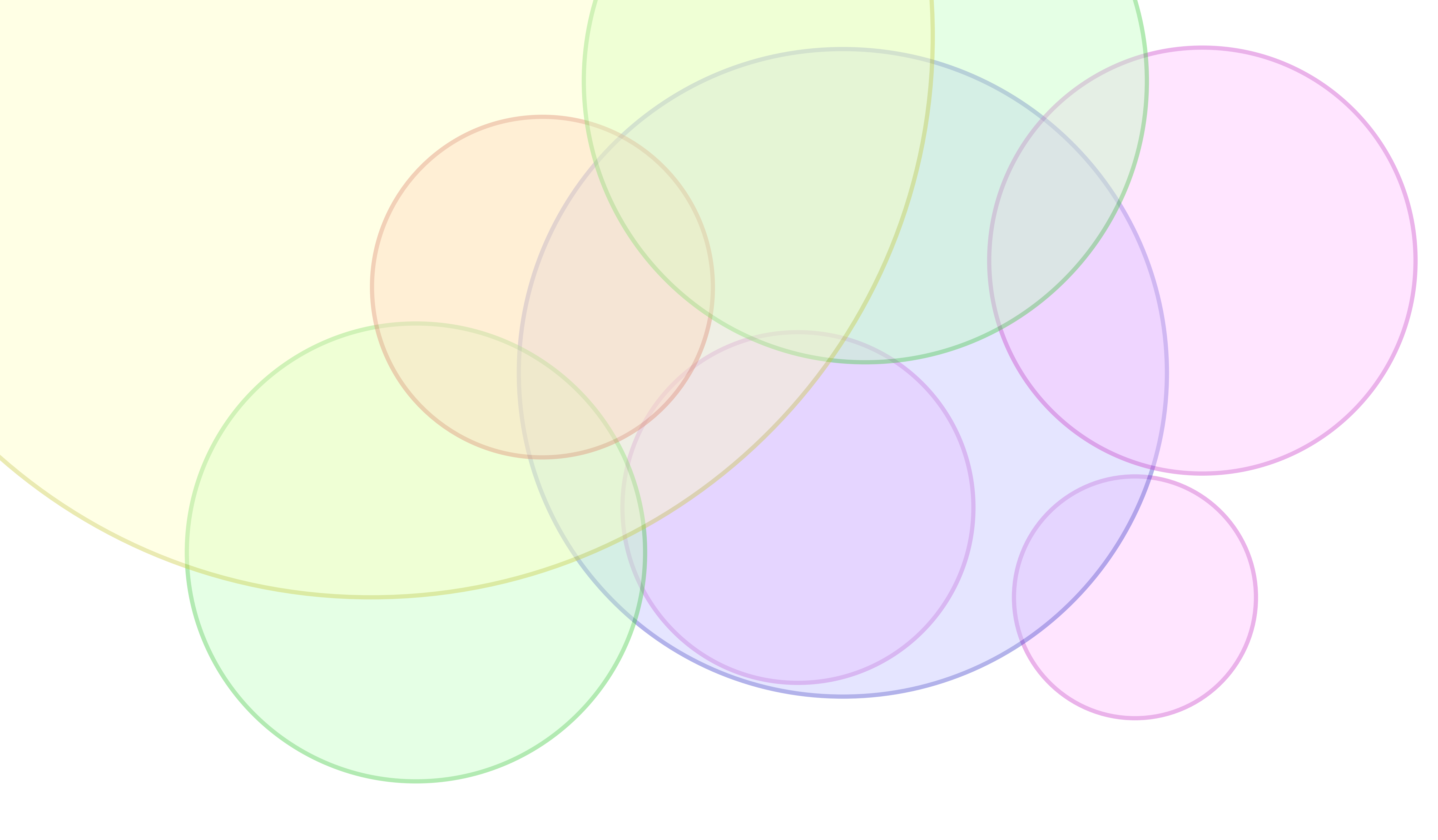
Query reveals partial information

Kirkpatrick '09 (hyperbolic dovetailing)

Measures of potential interference

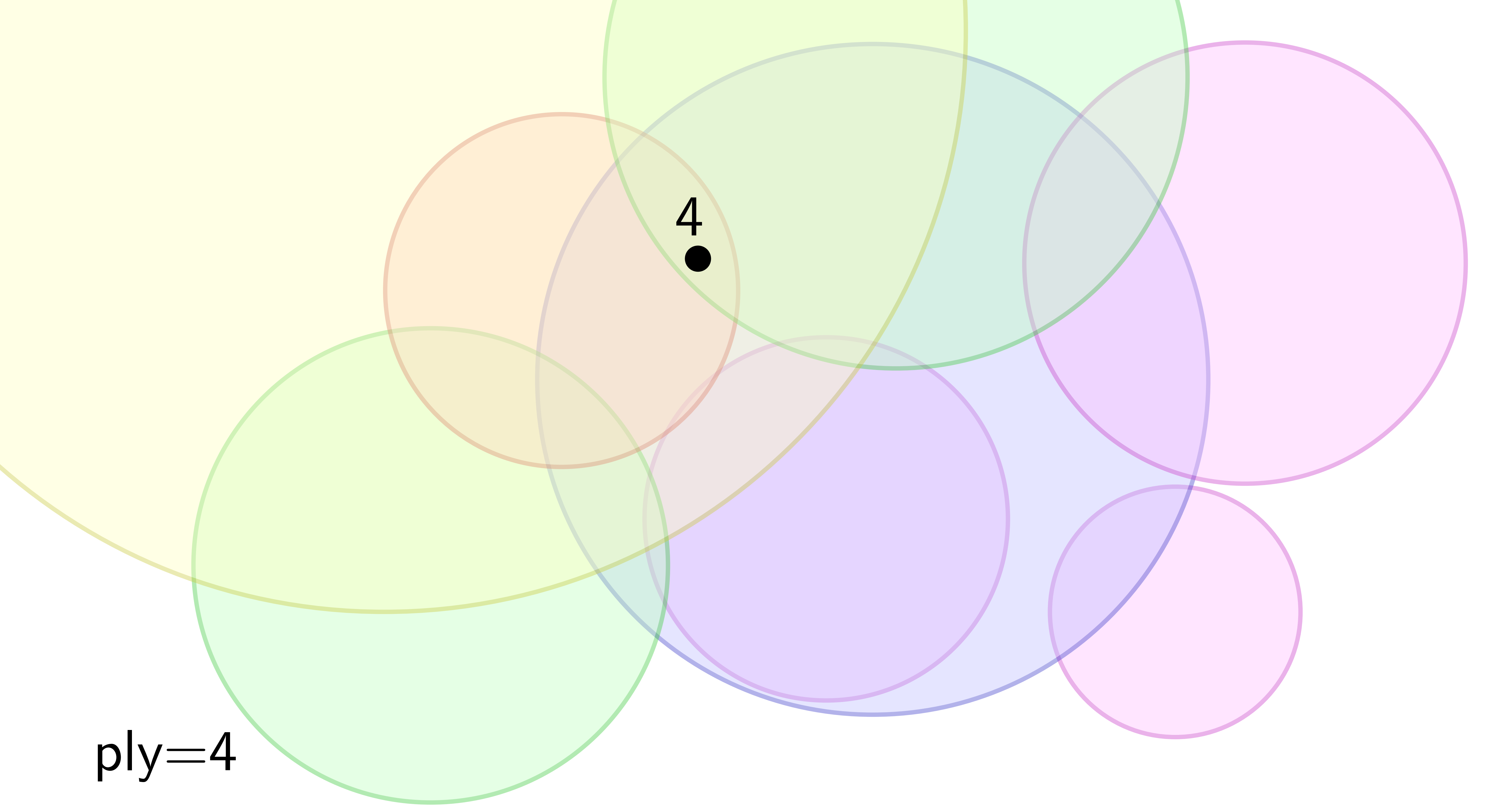
Measures of potential interference

Ply: max number of regions covering any point



Measures of potential interference

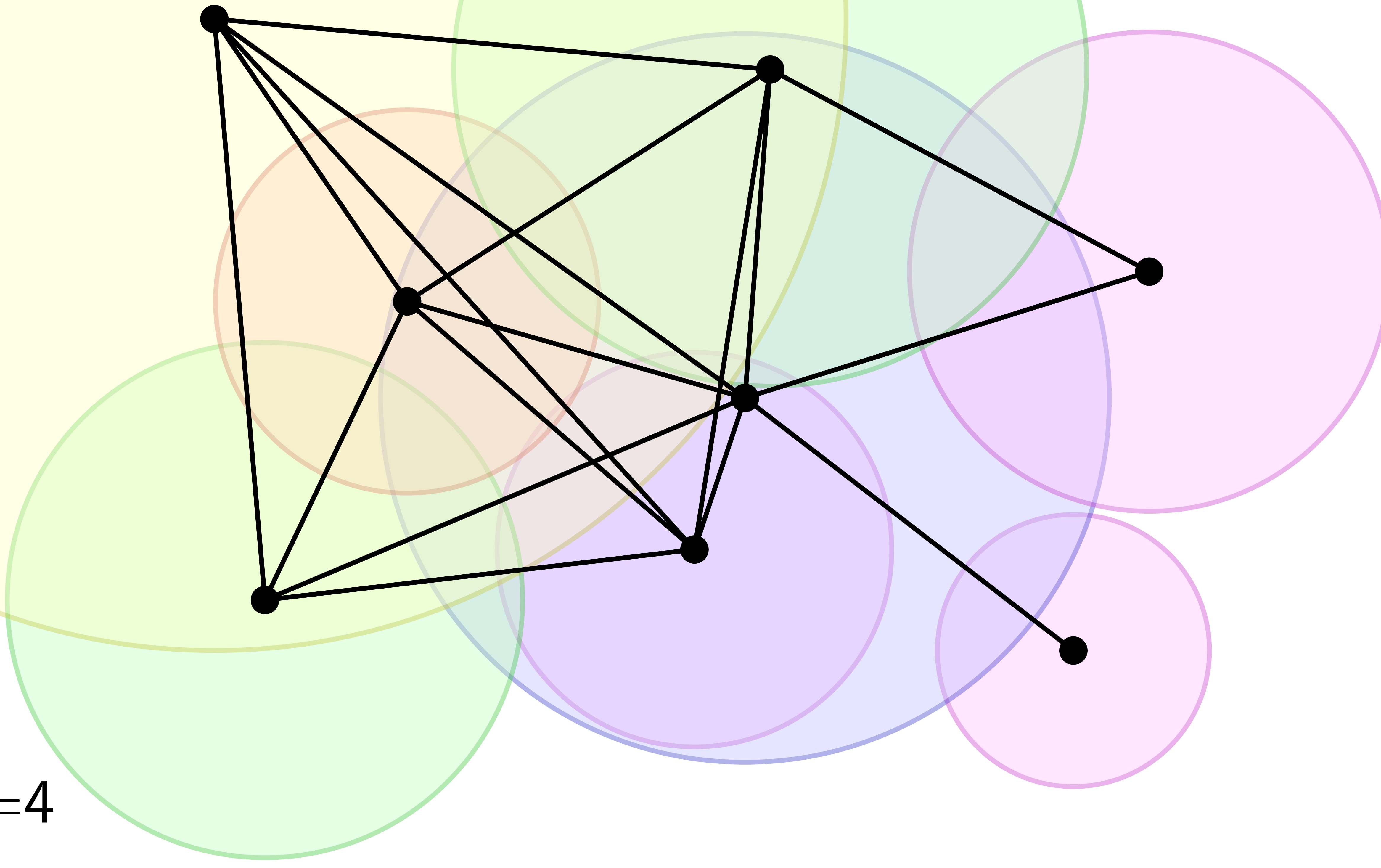
Ply: max number of regions covering any point $= 4$



ply$=$4

Measures of potential interference

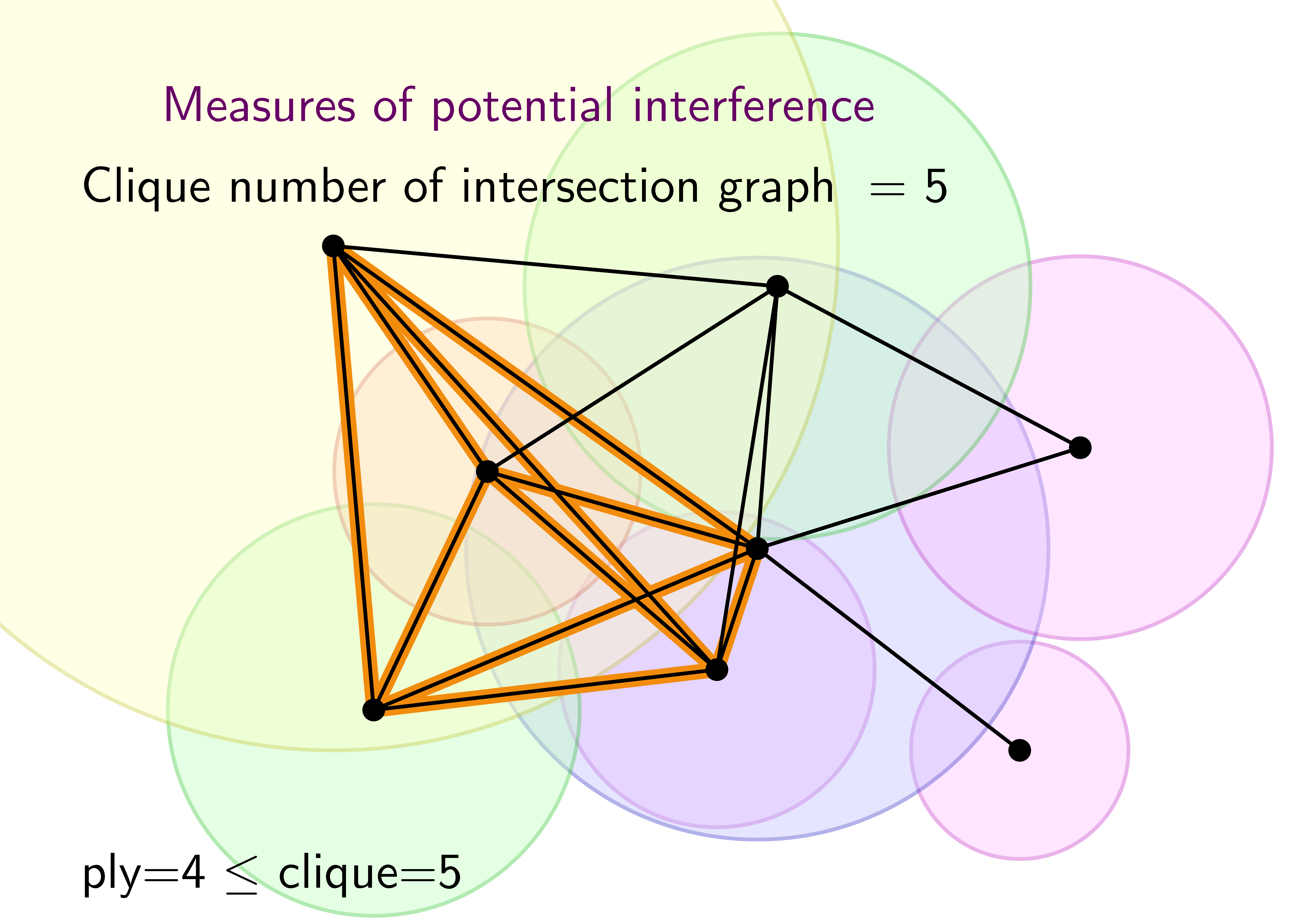
Clique number of intersection graph



ply=4

Measures of potential interference

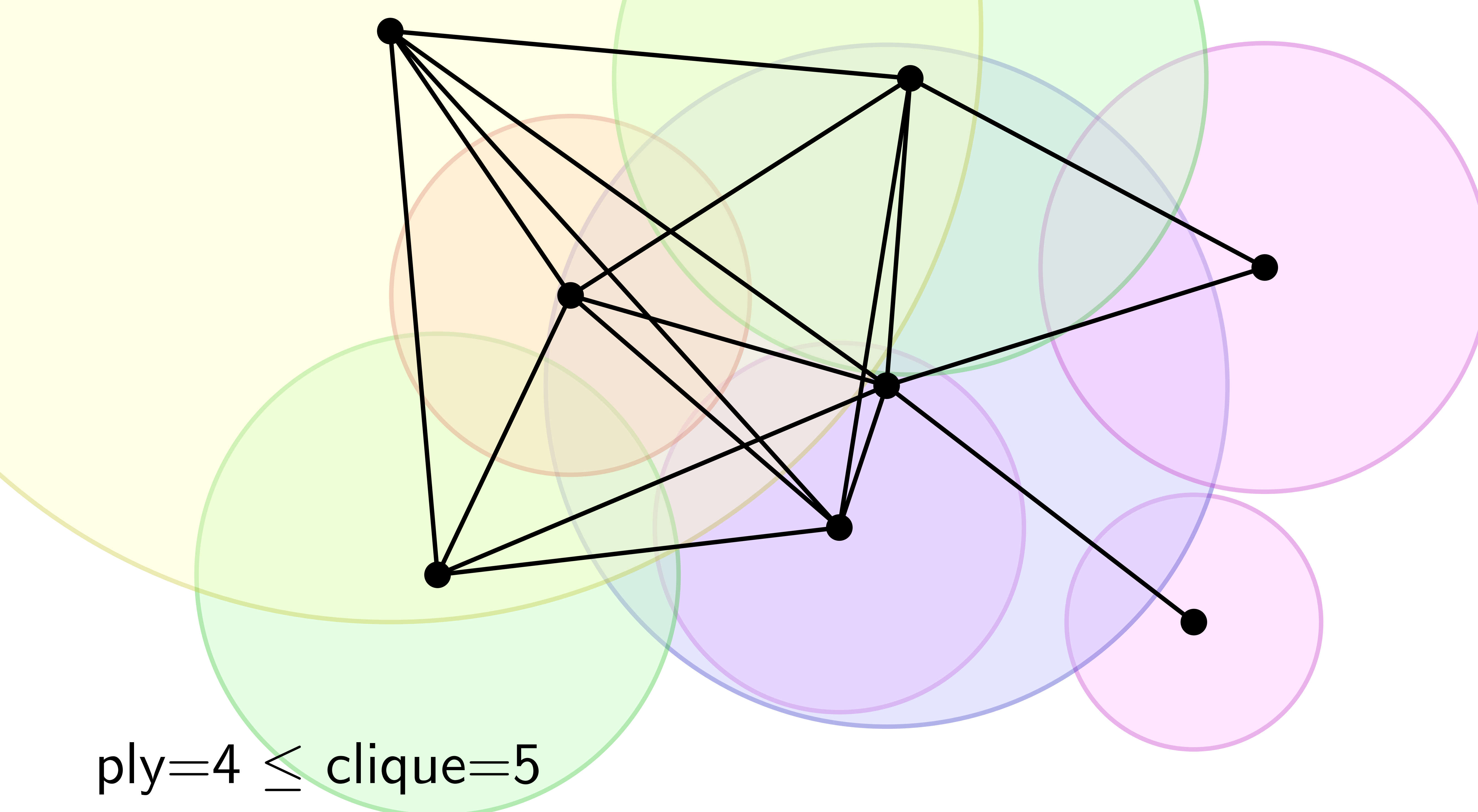
Clique number of intersection graph $= 5$



ply$=4 \leq$ clique$=5$

Measures of potential interference

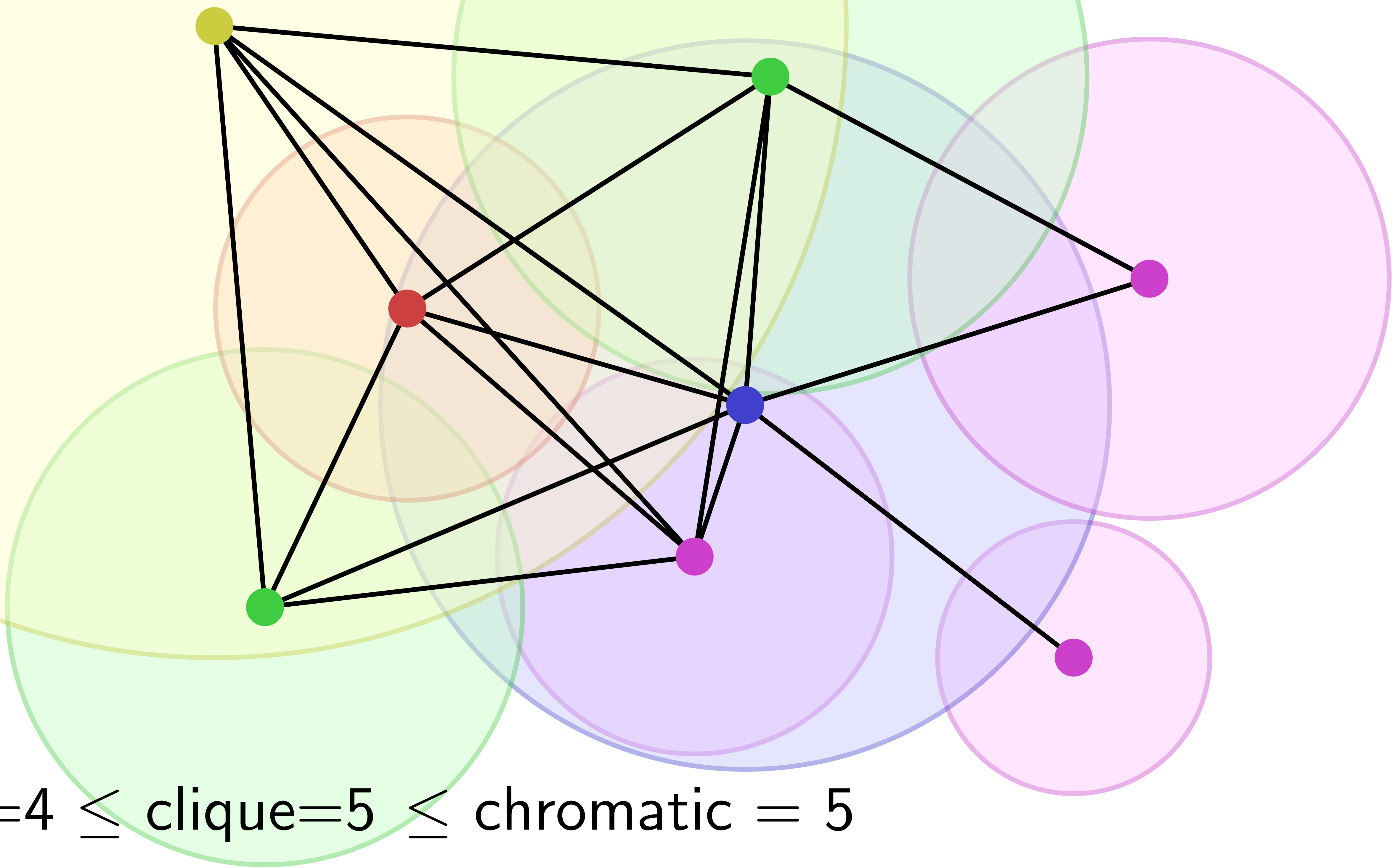
Chromatic number of intersection graph



ply=4 $\leq$ clique=5

Measures of potential interference

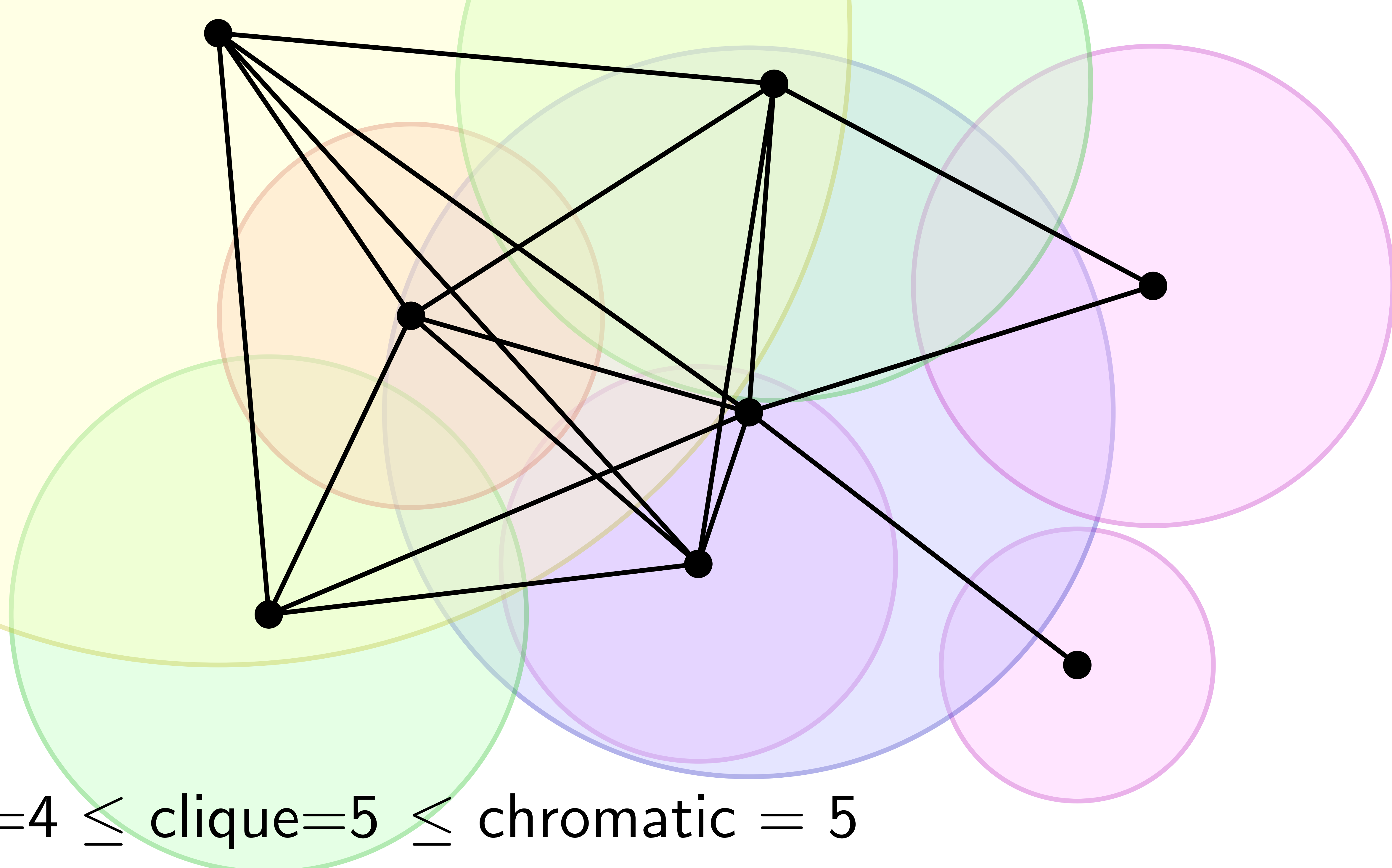
Chromatic number of intersection graph $= 5$



ply=4 $\leq$ clique=5 $\leq$ chromatic $= 5$

Measures of potential interference

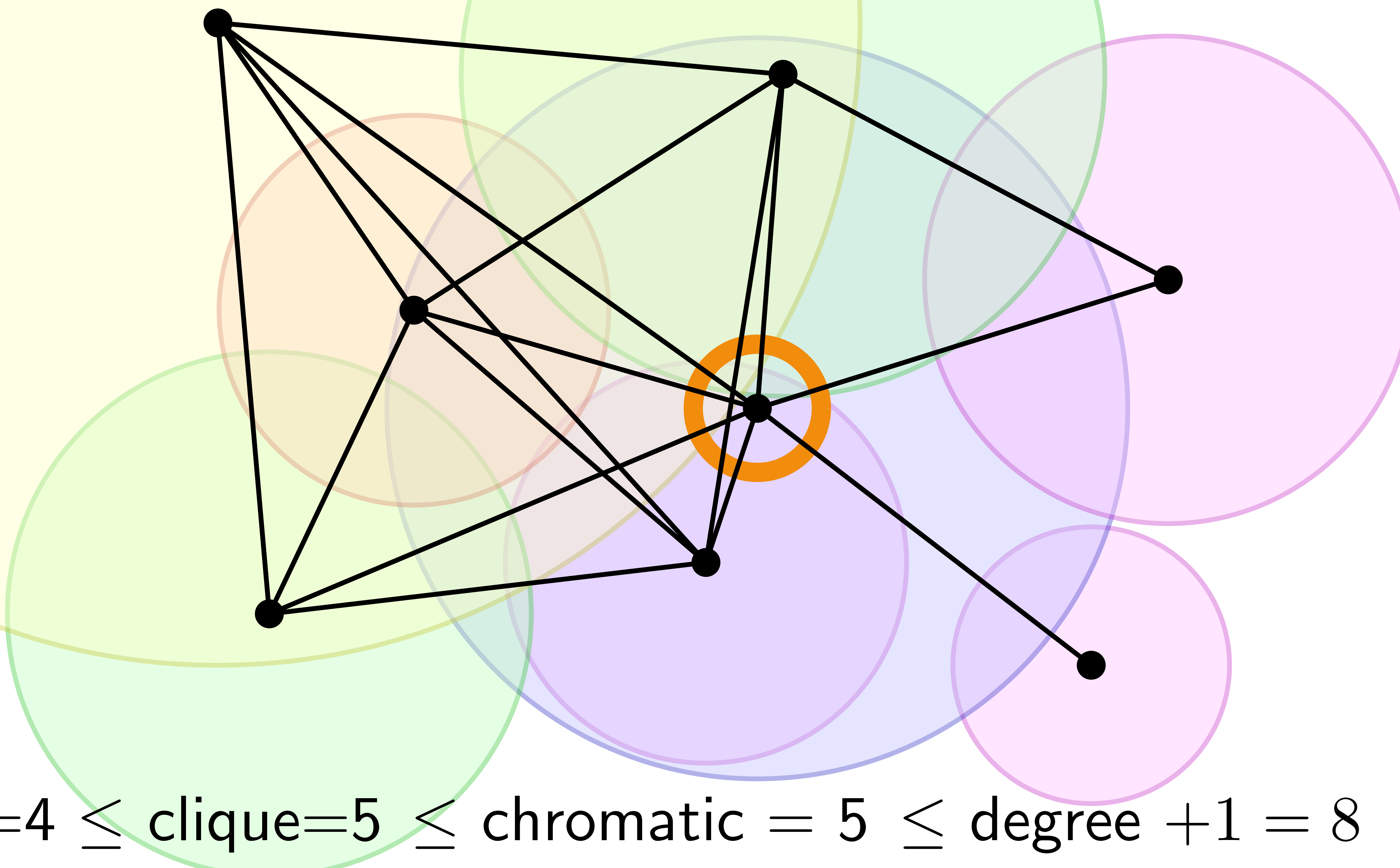
Max degree of intersection graph



ply=4 $\leq$ clique=5 $\leq$ chromatic = 5

Measures of potential interference

Max degree of intersection graph $= 7$



ply=4 $\leq$ clique=5 $\leq$ chromatic $= 5 \leq$ degree $+1 = 8$

Our main result

Theorem. *For* n *entities, the Adaptive Bucket Strategy keeps* **degree** $O(x^*)$ *during any time interval* $T = [a, b]$ *where* x^* *is the maximum* **ply** *observed by the best strategy over the interval* $[a - |T|, b]$, *provided* $|T| > c_d n$.

Query to reduce interference potential

at a target time [E,K,Löffler,Staals '16]

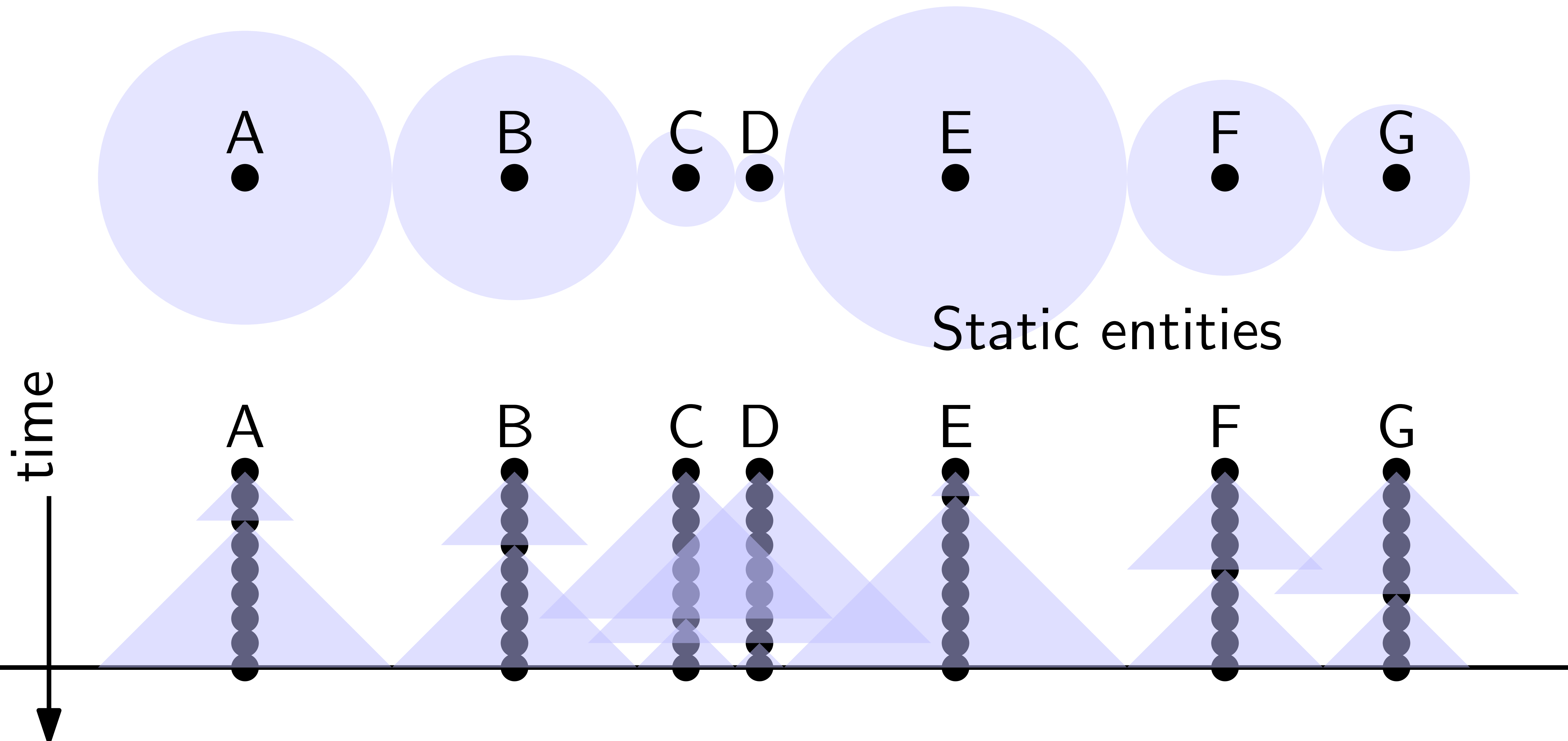
Static entities

Query to reduce interference potential

at a target time [E,K,Löffler,Staals '16]

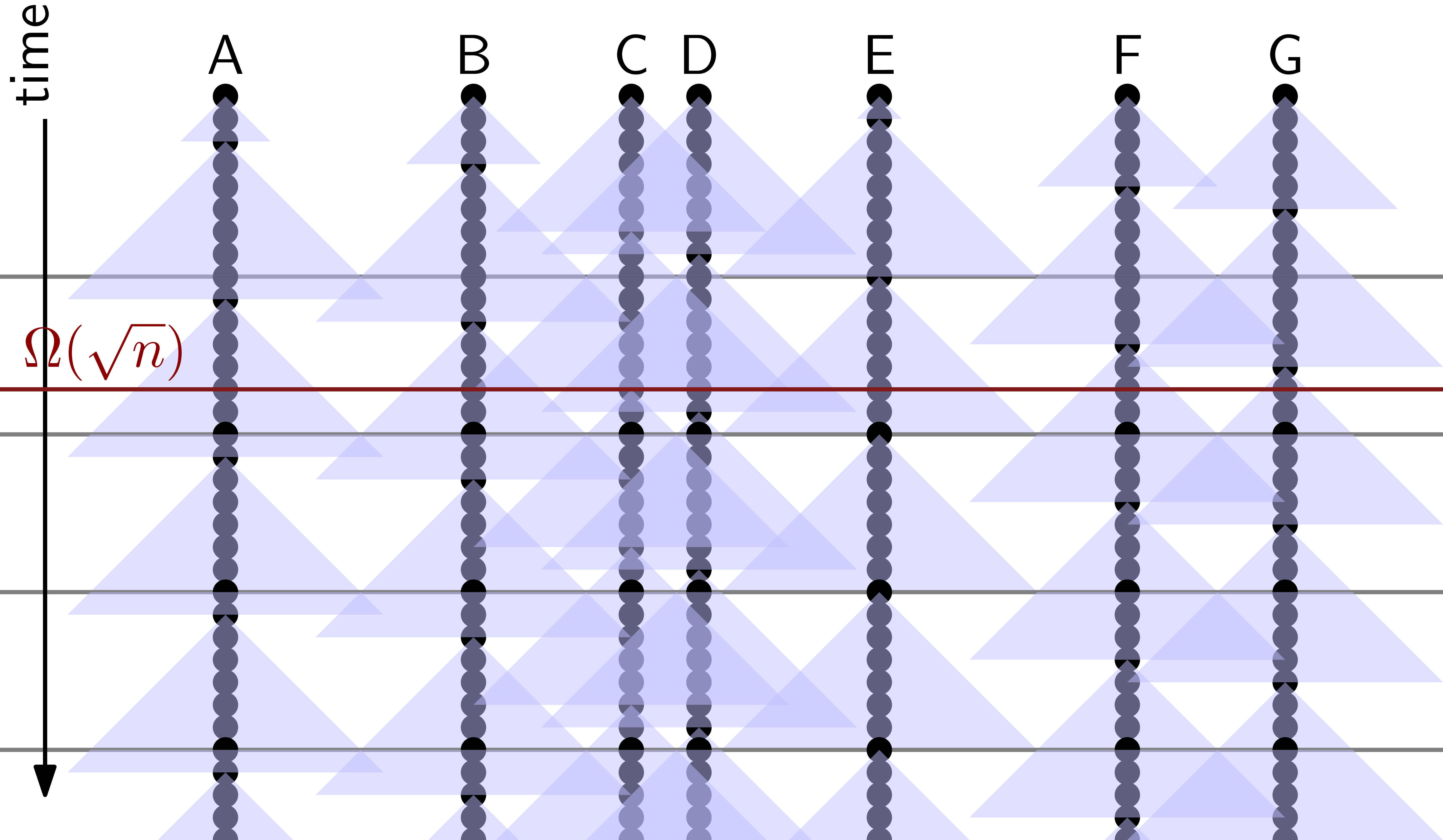
Optimal: Query EABFGCD

NP-complete even for static entities



Query to reduce interference potential

Repeat at a target time [E,K,Löffler,Staals '16]



Query to reduce interference potential

at **all** times [this paper]

Let's start with static point entities $e_1, \ldots, e_n$.

Query to reduce interference potential

at **all** times [this paper]

Let's start with static point entities $e_1, \ldots, e_n$.

	e_1	e_2	e_3	e_4	e_5	e_6	e_7
$r_i(1)$	11	7	3	3	8	7	7

x-**radius** $r_i(x)$ = distance from e_i to its x^{th}-nearest entity

To keep **degree** $< x$, must query e_i every $r_i(x)$ steps.

If $\underbrace{\sum_{i=1}^{n} \frac{1}{r_i(x)}}_{x\text{-density}} > 1$ this is impossible.

Observation

Unavoidable interference

The **congestion** of a set $\mathcal{E}$ of entities is

$$x_{\mathcal{E}} = \max\{x | \sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} > 1\}.$$

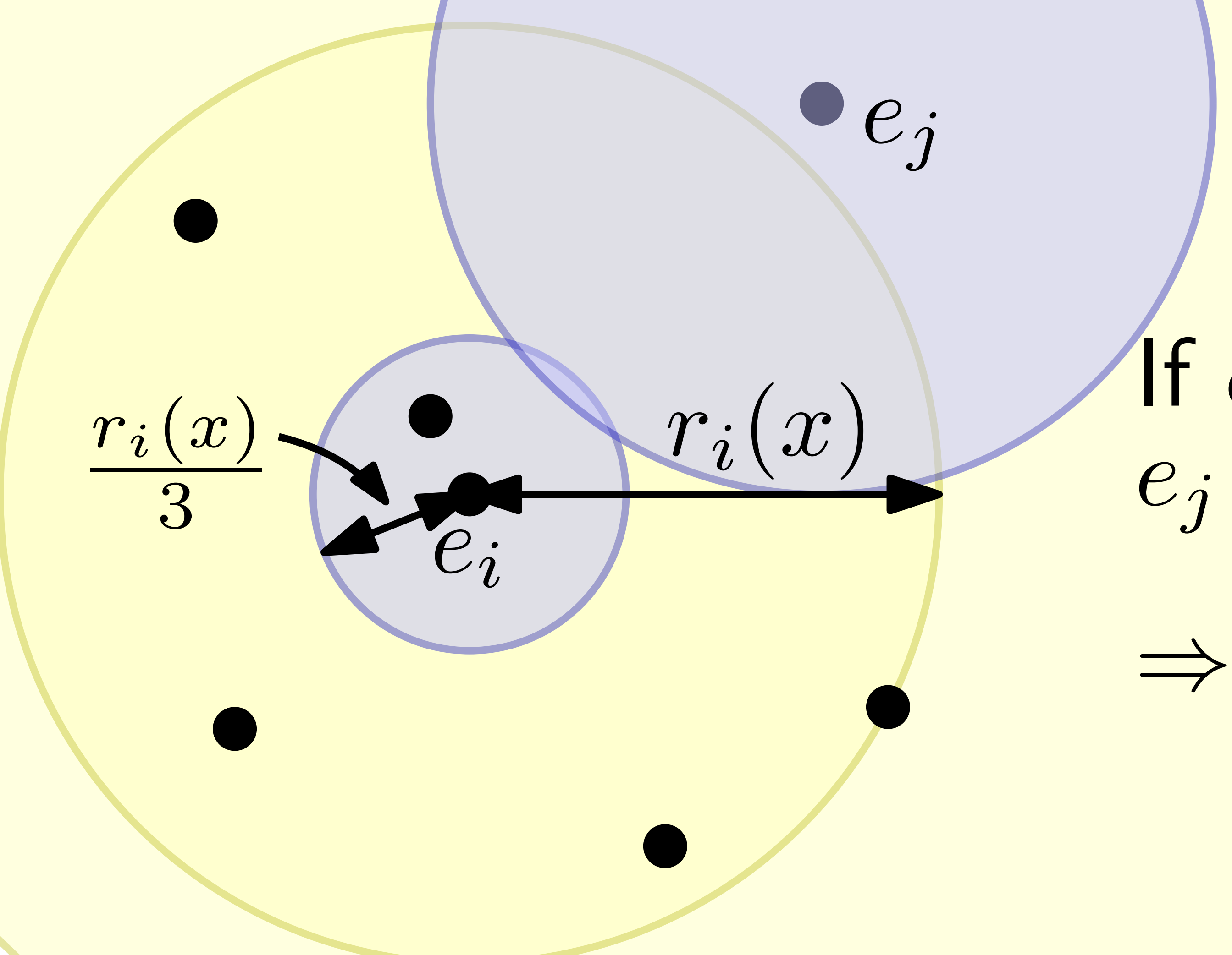
Theorem. *Any query strategy for n static entities $\mathcal{E}$ suffers **ply** $\Omega(x_{\mathcal{E}})$ at some time in any n consecutive time steps.*

Algorithm to avoid interference

Weighted Round Robin

Idea: Query entity e_i so its uncertainty region has radius $< r_i(k(x_{\mathcal{E}}+1))/3$.

Let $x = k(x_{\mathcal{E}}+1)$



If e_j's region intersects e_i's then e_j is nbr$_x$ of e_i.

$\Rightarrow$ max **degree** $\leq x$

Algorithm to avoid interference

Theorem. *Any query strategy for n static entities $\mathcal{E}$ suffers* **ply** $\Omega(x_{\mathcal{E}})$ *at some time in any n consecutive time steps.*

Theorem. *For static entities, Weighted Round Robin maintains* **degree** $O(x_{\mathcal{E}})$ *at all times.*

Algorithm to avoid interference

Theorem. *Any query strategy for n static entities $\mathcal{E}$ suffers* **ply** $\Omega(x_{\mathcal{E}})$ *at some time in any n consecutive time steps.*

Theorem. *For static entities, Weighted Round Robin maintains* **degree** $O(x_{\mathcal{E}})$ *at all times.*

Theorem. *For n static point entities $\mathcal{E}$, if $\omega_{\mathcal{E}}$ is the minimum ply achievable at some target time, $x_{\mathcal{E}} \in O(\omega_{\mathcal{E}} \log^d(n/\omega_{\mathcal{E}}))$ and for some such $\mathcal{E}$ this is tight.*

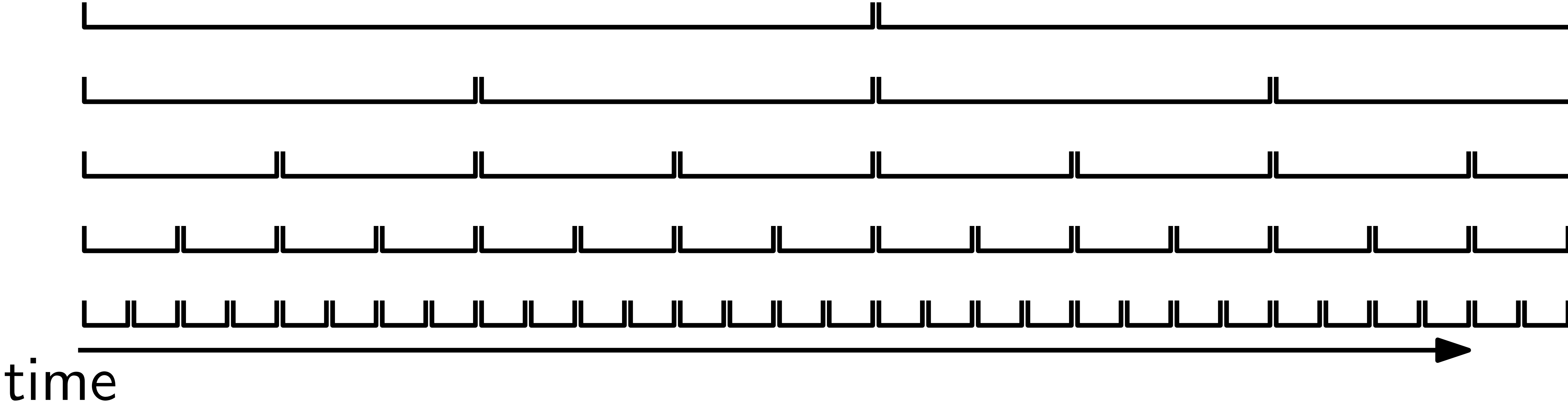
Dynamic entities

Complications due to movement:

1. Strategy knows **perceived** (last queried) location rather than true (current) location
2. Congestion $x_{\mathcal{E}}$ varies over time

Bucket Strategy for fixed tolerance x

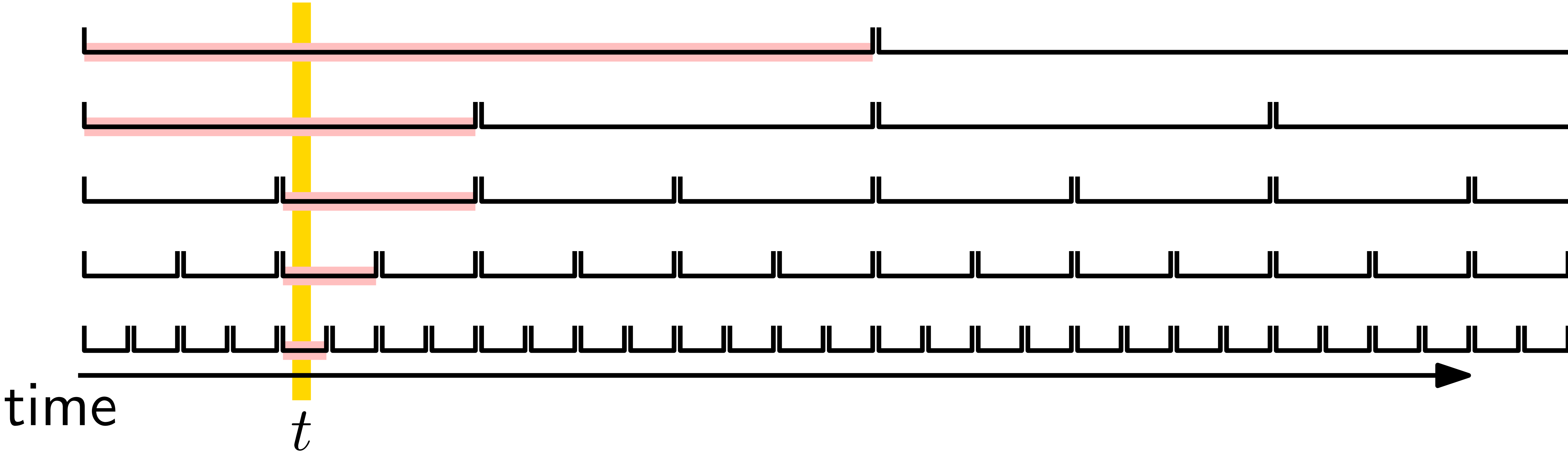
Idea: Schedule e_i for a future time interval (**bucket**) based on perceived x-radius $\widetilde{r}_i(x,t)$ when queried.



Bucket Strategy for fixed tolerance x

Idea: Schedule e_i for a future time interval (**bucket**) based on perceived x-radius $\widetilde{r}_i(x,t)$ when queried.

Repeat for all time t
  Query entity e_i from shortest active bucket
  Move e_i to next bucket of length 2^b
    where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



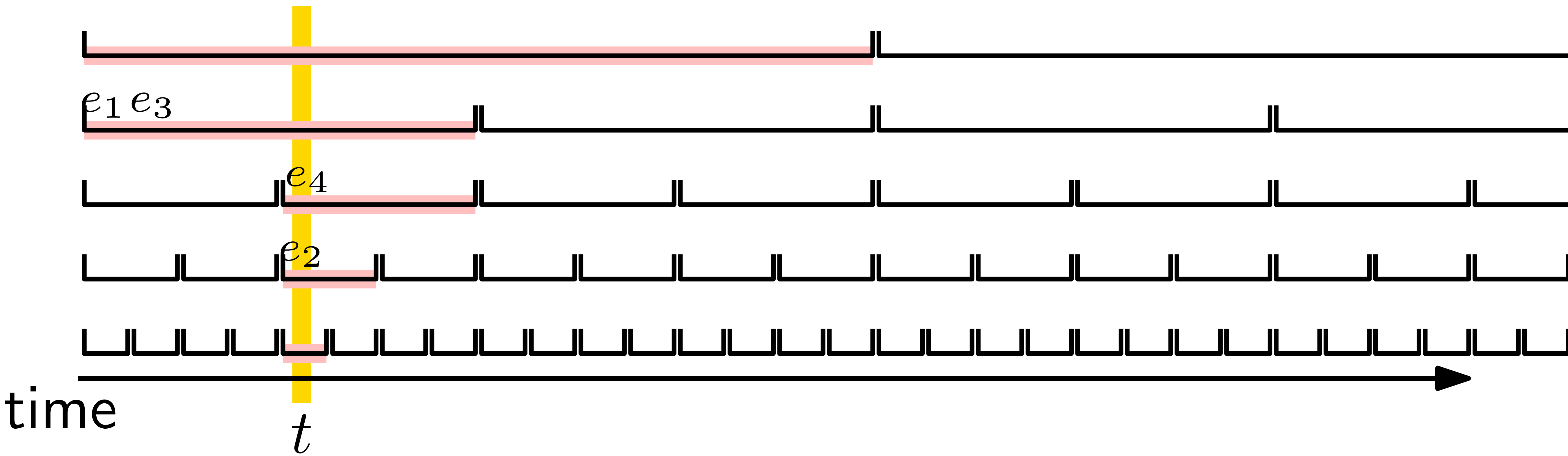
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



? ? ? ?

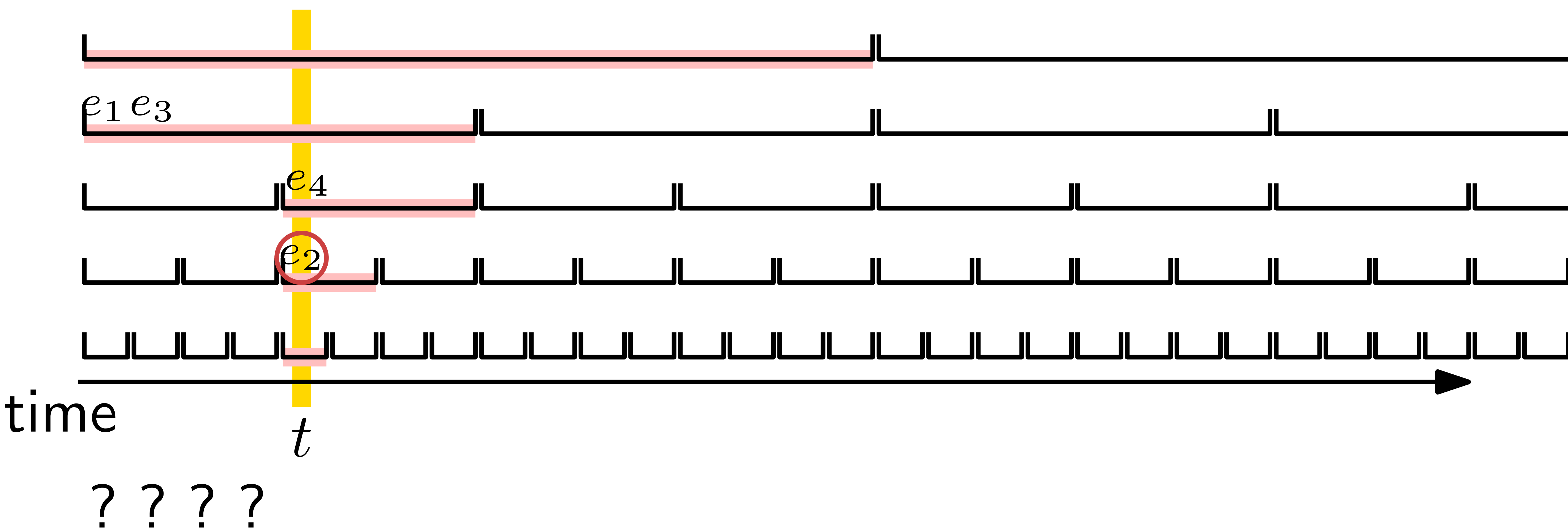
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



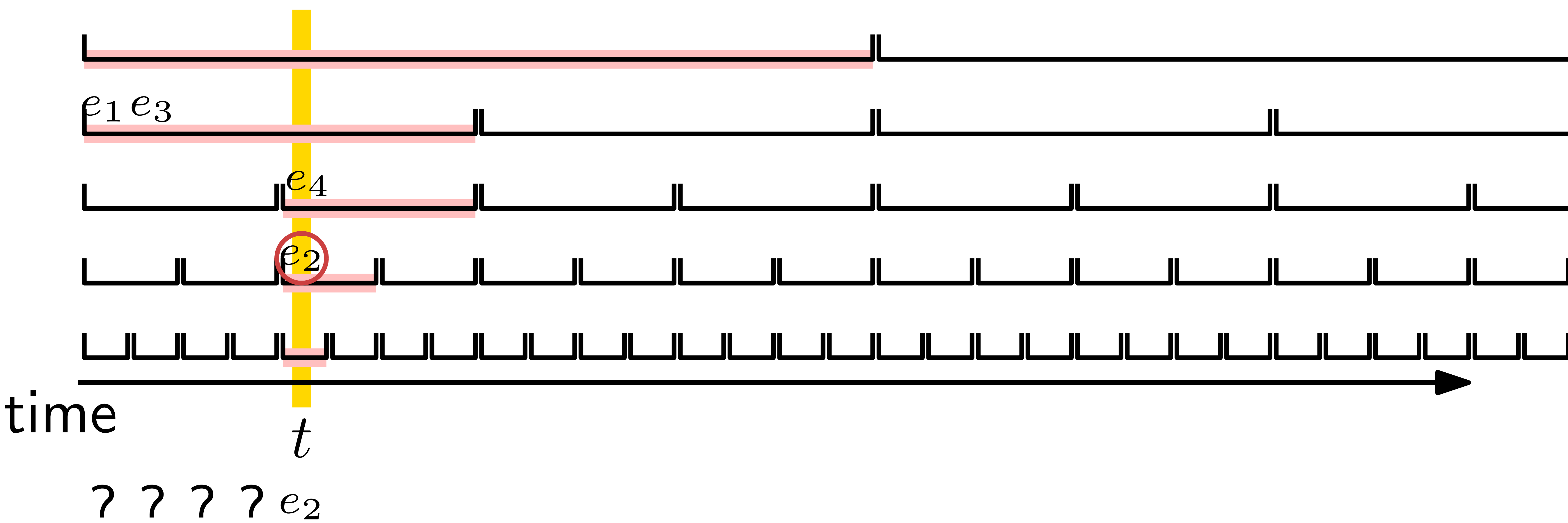
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



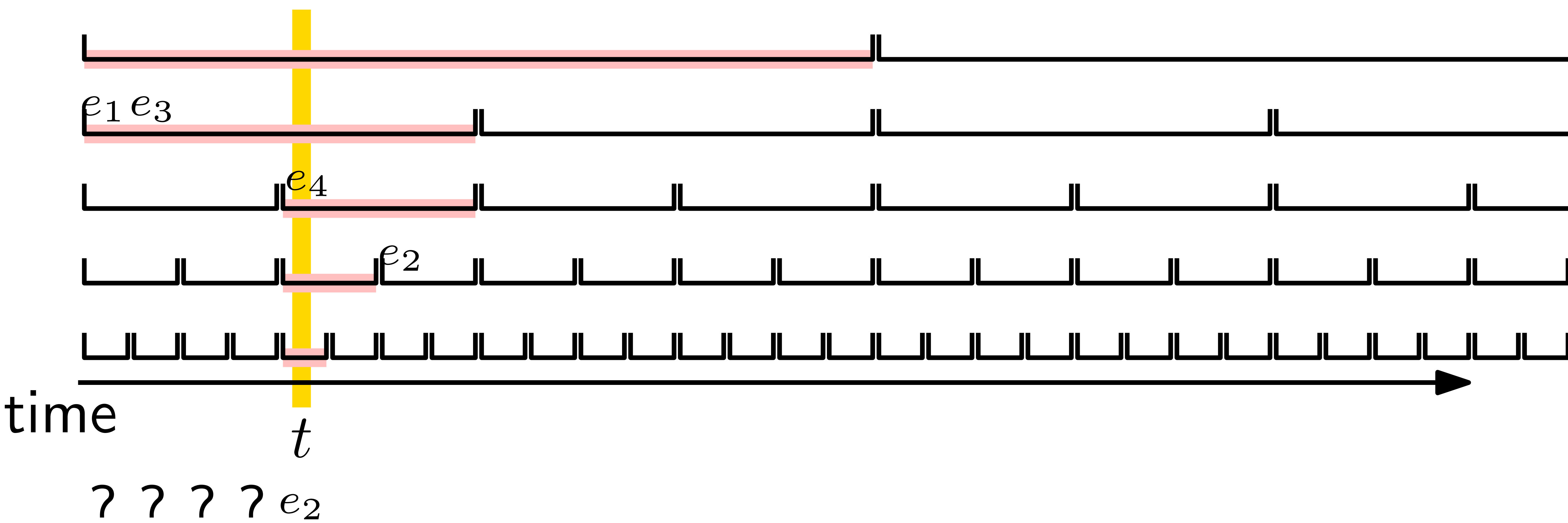
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

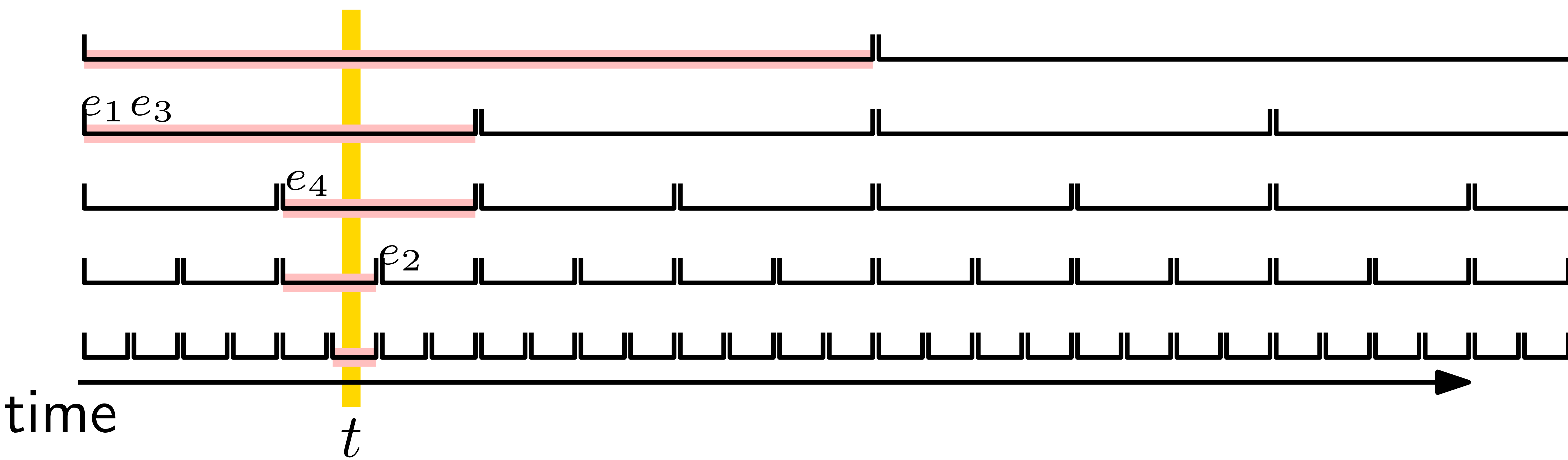
Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



Bucket Strategy for fixed tolerance x

Repeat for all time t
  Query entity e_i from shortest active bucket
  Move e_i to next bucket of length 2^b
    where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



? ? ? ? e_2

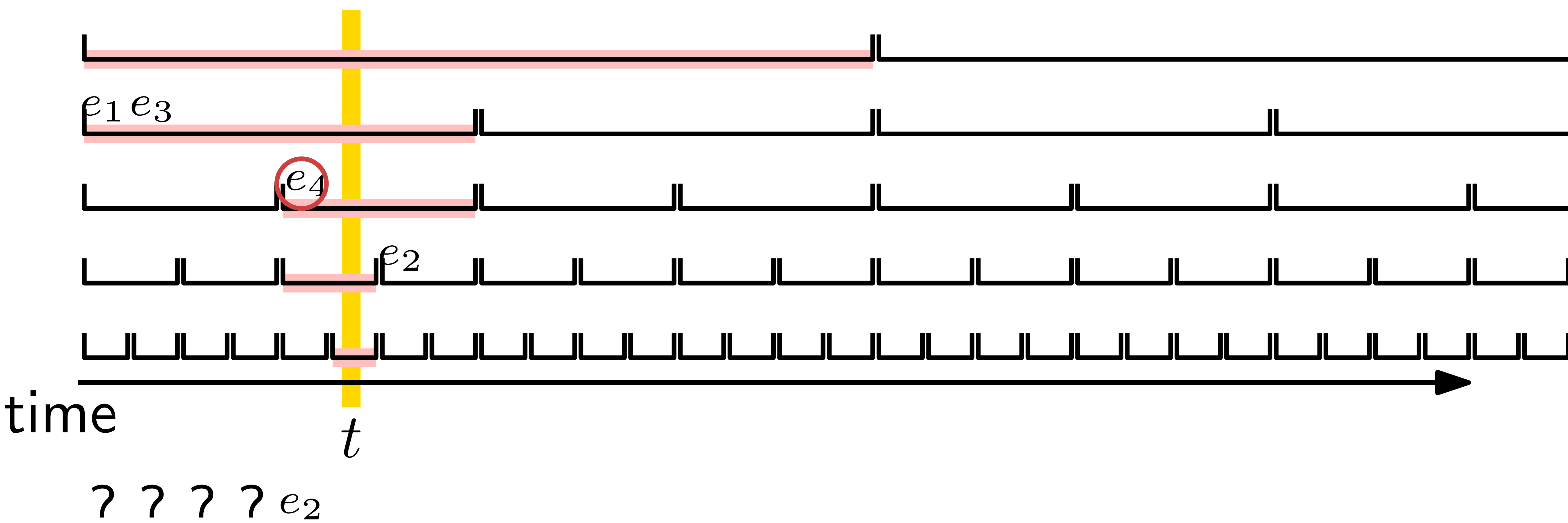
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

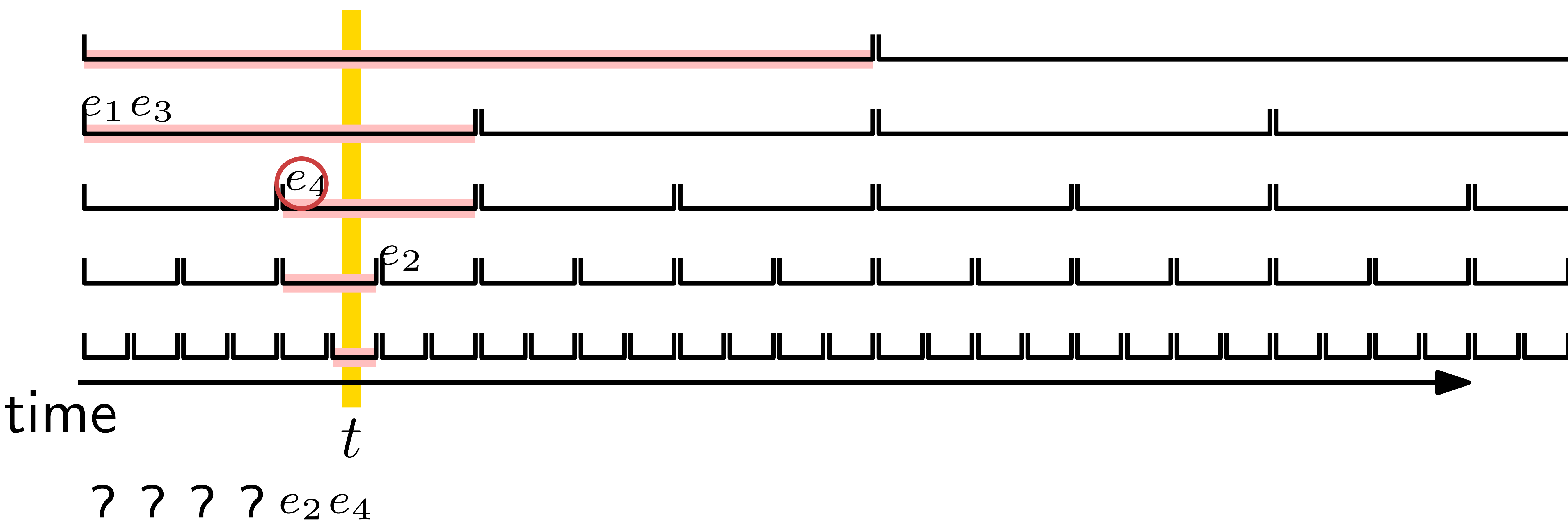
Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



Bucket Strategy for fixed tolerance x

Repeat for all time t
  Query entity e_i from shortest active bucket
  Move e_i to next bucket of length 2^b
    where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



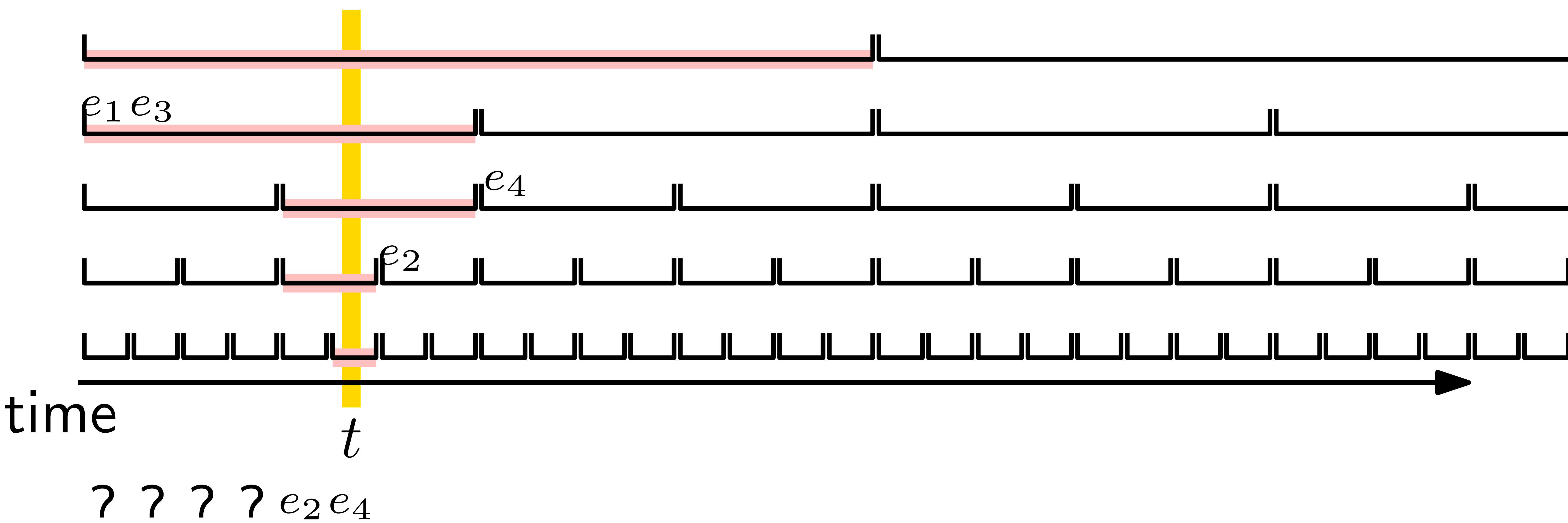
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



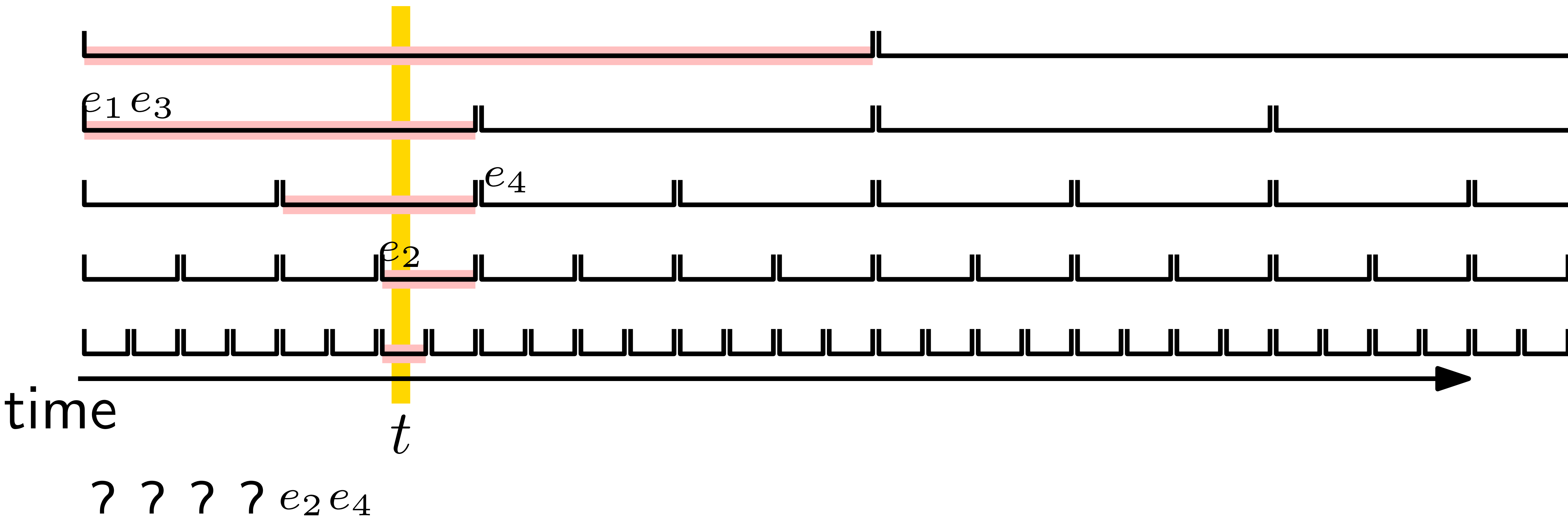
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



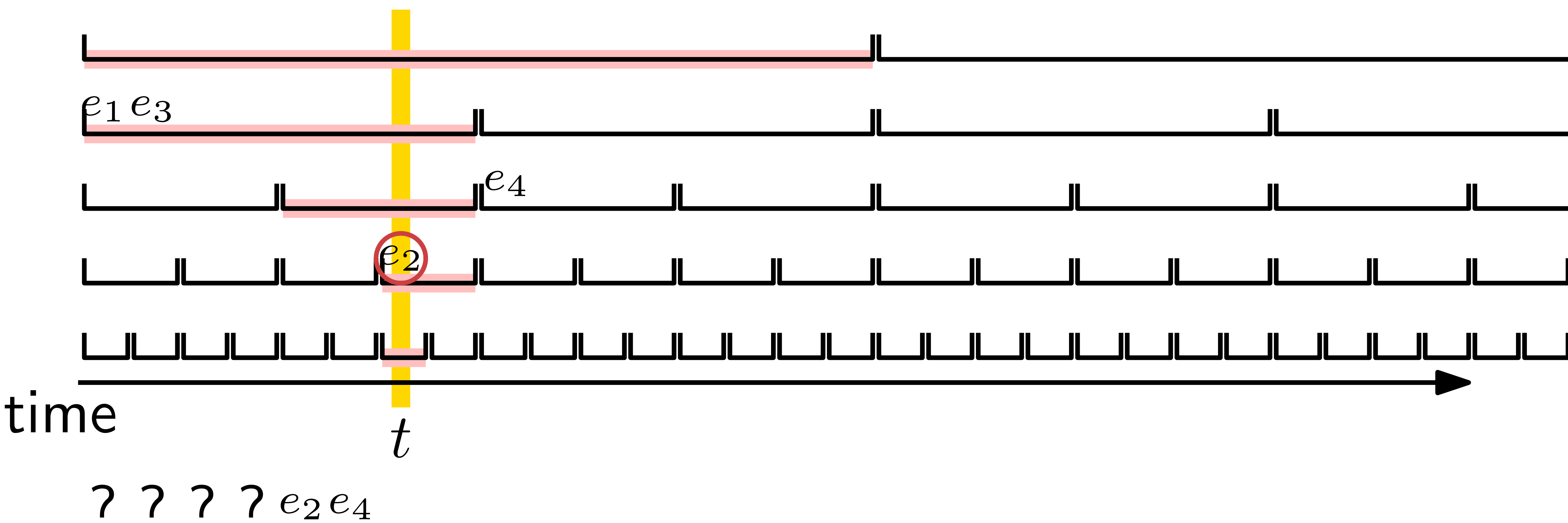
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



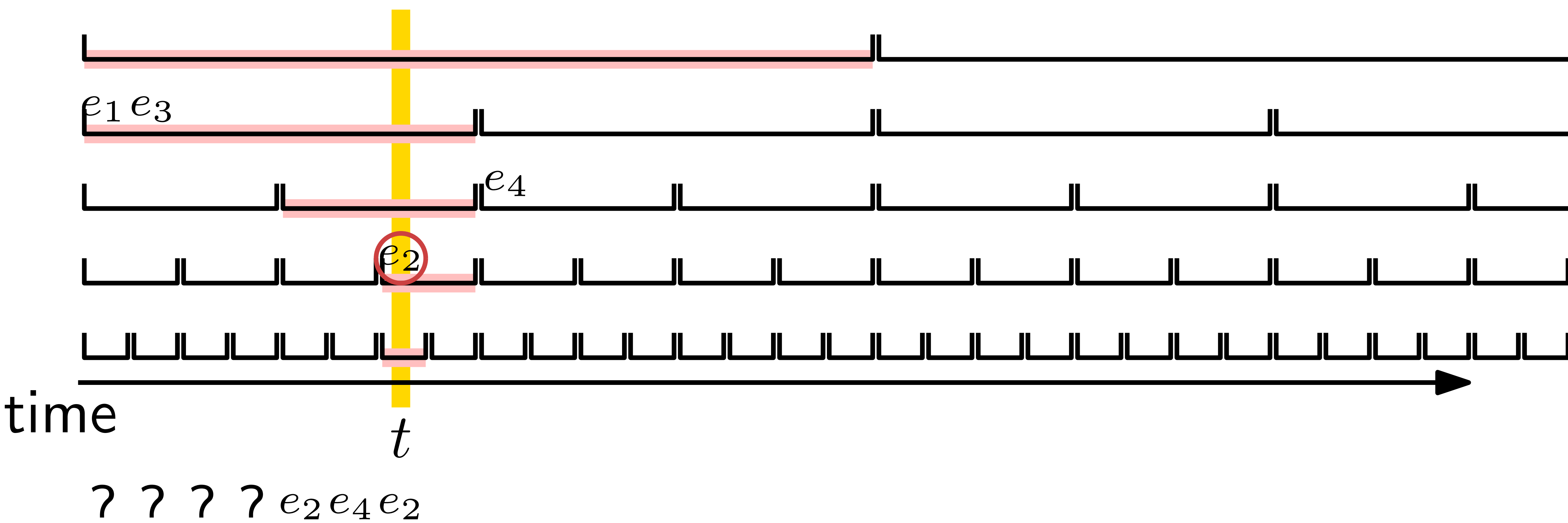
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



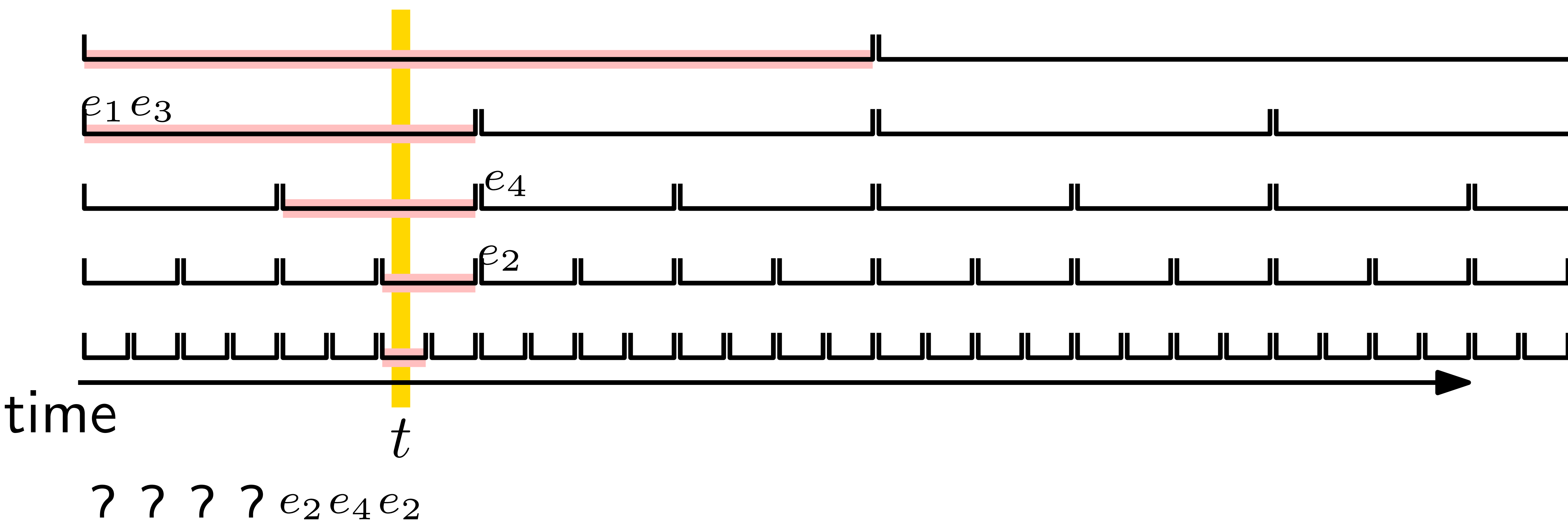
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



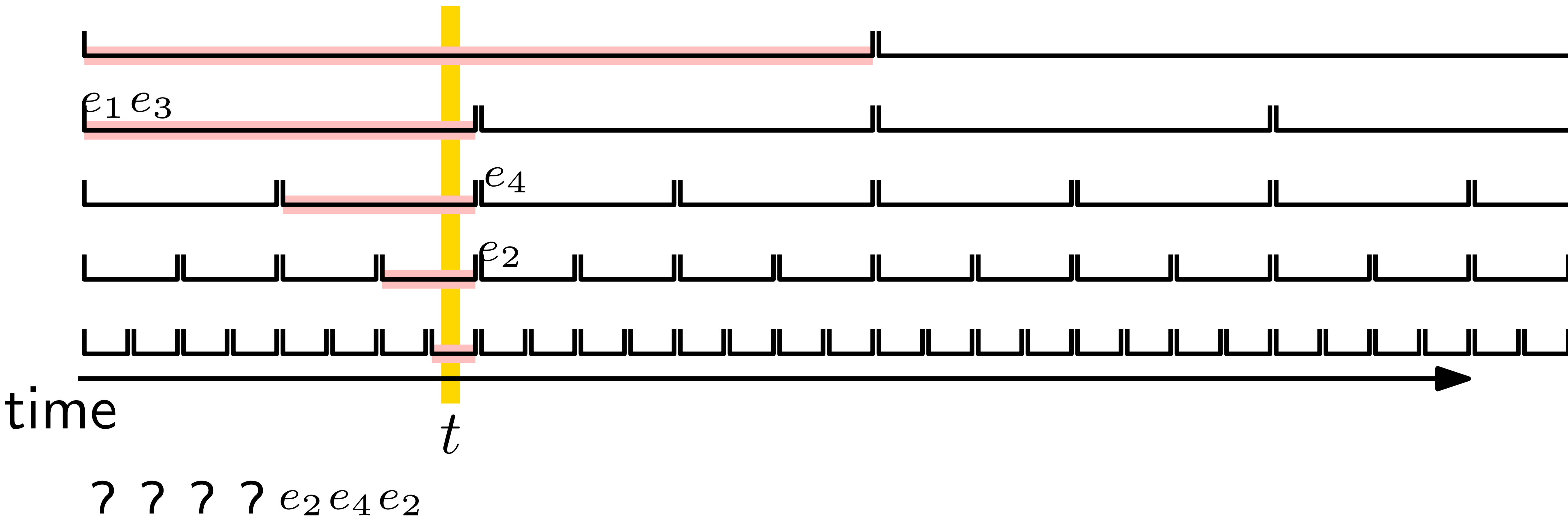
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

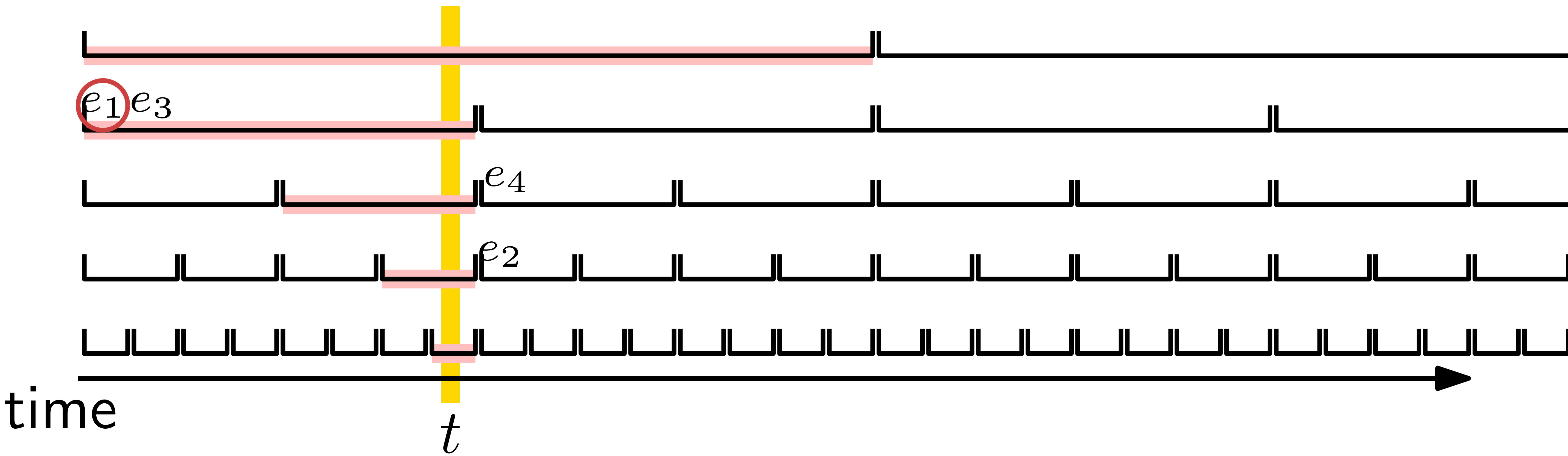
Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



Bucket Strategy for fixed tolerance x

Repeat for all time t
  Query entity e_i from shortest active bucket
  Move e_i to next bucket of length 2^b
    where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



? ? ? ? e_2 e_4 e_2

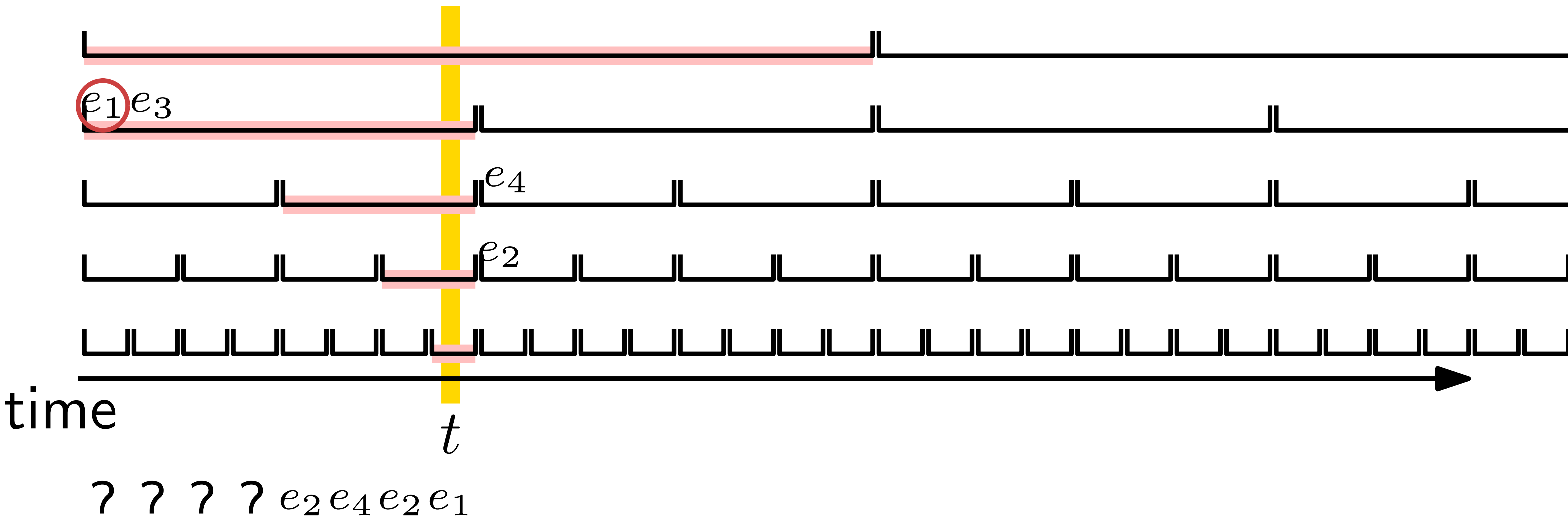
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



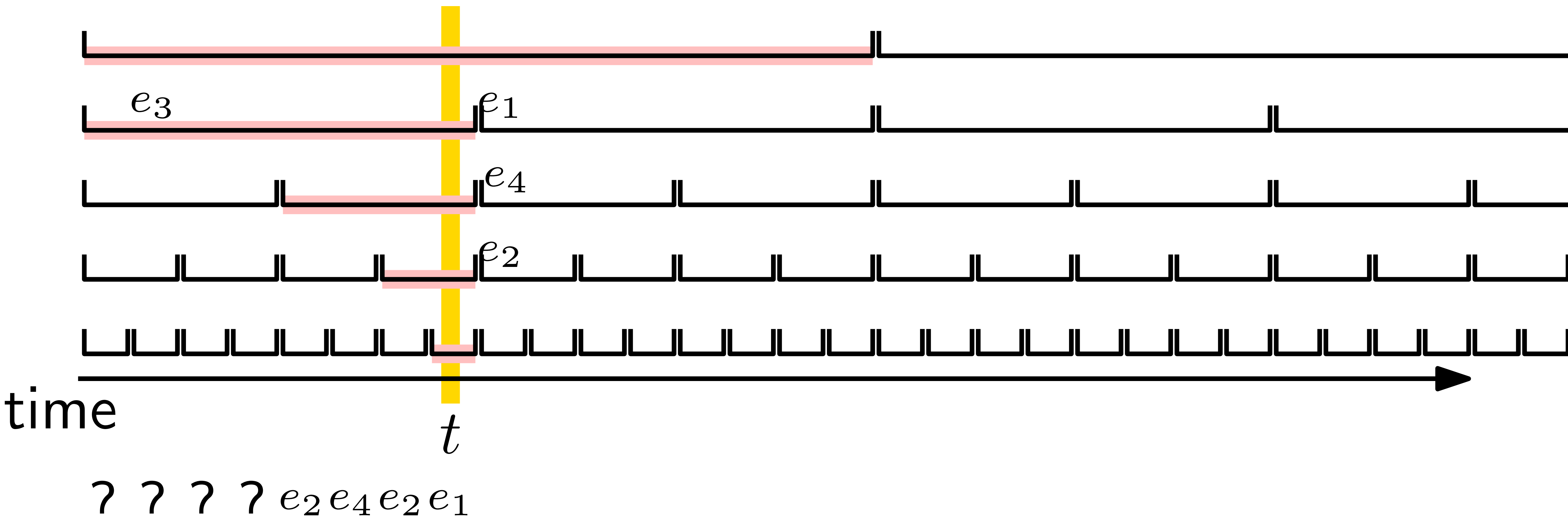
Bucket Strategy for fixed tolerance x

Repeat for all time t

Query entity e_i from shortest active bucket

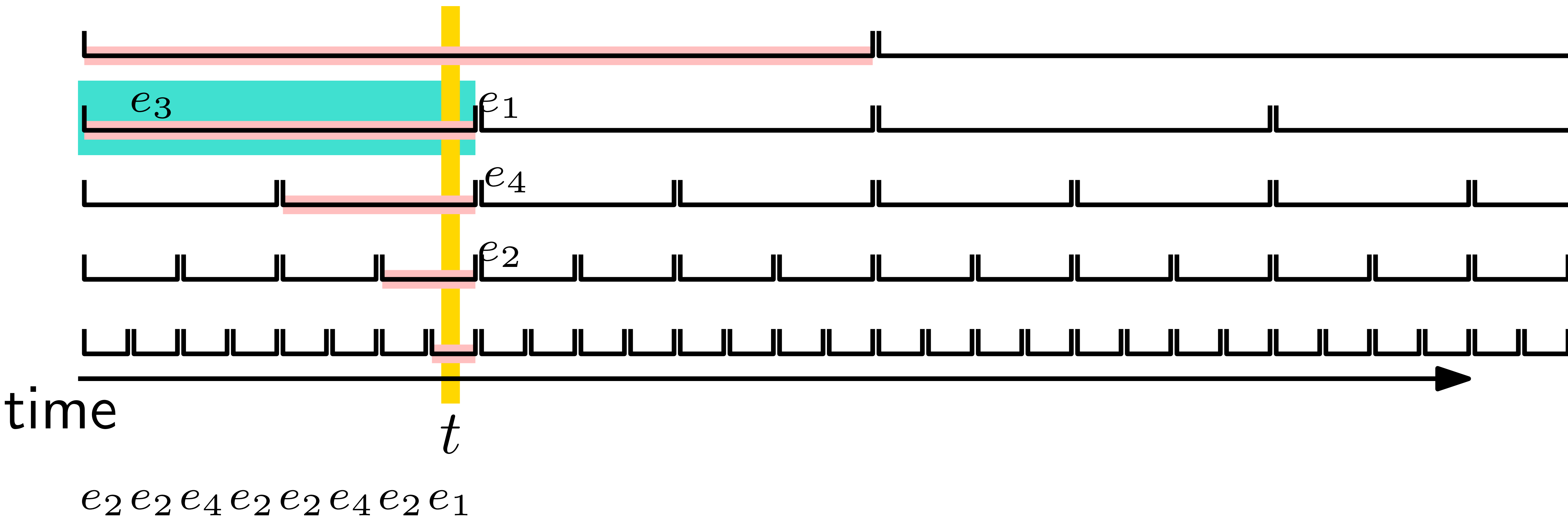
Move e_i to next bucket of length 2^b

where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$



Bucket Strategy for fixed tolerance x

Repeat for all time t
  Query entity e_i from shortest active bucket
  Move e_i to next bucket of length 2^b
    where $b = \lfloor \lg(\widetilde{r}_i(x,t)/Q) \rfloor$
  If any ending bucket is uncleared, fail



Unavoidable interference - moving entities

Sustained x-density implies high ply

Lemma. *For any time interval T with $|T| \geq |E|$, if $\sum_{t \in T} \sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x,t)} \geq c_d |T|$ then any query strategy suffers* **ply** $\Omega(x)$ *at some time in T.*

Unavoidable interference - moving entities

Sustained x-density implies high ply

Lemma. *For any time interval* T *with* $|T| \geq |E|$*, if* $\sum_{t \in T} \sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x,t)} \geq c_d |T|$ *then any query strategy suffers* **ply** $\Omega(x)$ *at some time in* T*.*

Theorem. *If the Bucket Strategy fails then any query strategy suffers* **ply** $\Omega(x)$ *at some time in the uncleared bucket's time interval.*

Unavoidable interference - moving entities

Sustained x-density implies high ply

Lemma. *For any time interval* T *with* $|T| \geq |E|$*, if* $\sum_{t \in T} \sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x,t)} \geq c_d |T|$ *then any query strategy suffers* **ply** $\Omega(x)$ *at some time in* T*.*

Theorem. *If the Bucket Strategy fails then any query strategy suffers* **ply** $\Omega(x)$ *at some time in the uncleared bucket's time interval.*

Theorem. *If the Bucket Strategy does not fail over time interval* T *then it keeps* **degree** *at most* x *during* T*.*

Adaptive Bucket Strategy when congestion may vary

1. If Bucket Strategy fails using tolerance x, we can increase tolerance to $2x$ knowing ply $\Omega(x)$ is unavoidable

Adaptive Bucket Strategy when congestion may vary

1. If Bucket Strategy fails using tolerance x, we can increase tolerance to $2x$ knowing ply $\Omega(x)$ is unavoidable

   but queries made using tolerance x may prevent success using $2x$ (even when $2x$ is possible).

Adaptive Bucket Strategy when congestion may vary

1. If Bucket Strategy fails using tolerance x, we can increase tolerance to $2x$ knowing ply $\Omega(x)$ is unavoidable

   but queries made using tolerance x may prevent success using $2x$ (even when $2x$ is possible).

2. If Bucket Strategy is not failing using tolerance x, when do we try to achieve $x/2$?

Adaptive Bucket Strategy when congestion may vary

1. If Bucket Strategy fails using tolerance x, we can increase tolerance to $2x$ knowing ply $\Omega(x)$ is unavoidable

   but queries made using tolerance x may prevent success using $2x$ (even when $2x$ is possible).

2. If Bucket Strategy is not failing using tolerance x, when do we try to achieve $x/2$?

   perform speculative queries in parallel

Future Work

A simpler optimal strategy

A distributed optimal strategy

Other measures of interference potential

Future Work

A simpler optimal strategy

A distributed optimal strategy

Other measures of interference potential

Thank you

Unavoidable interference

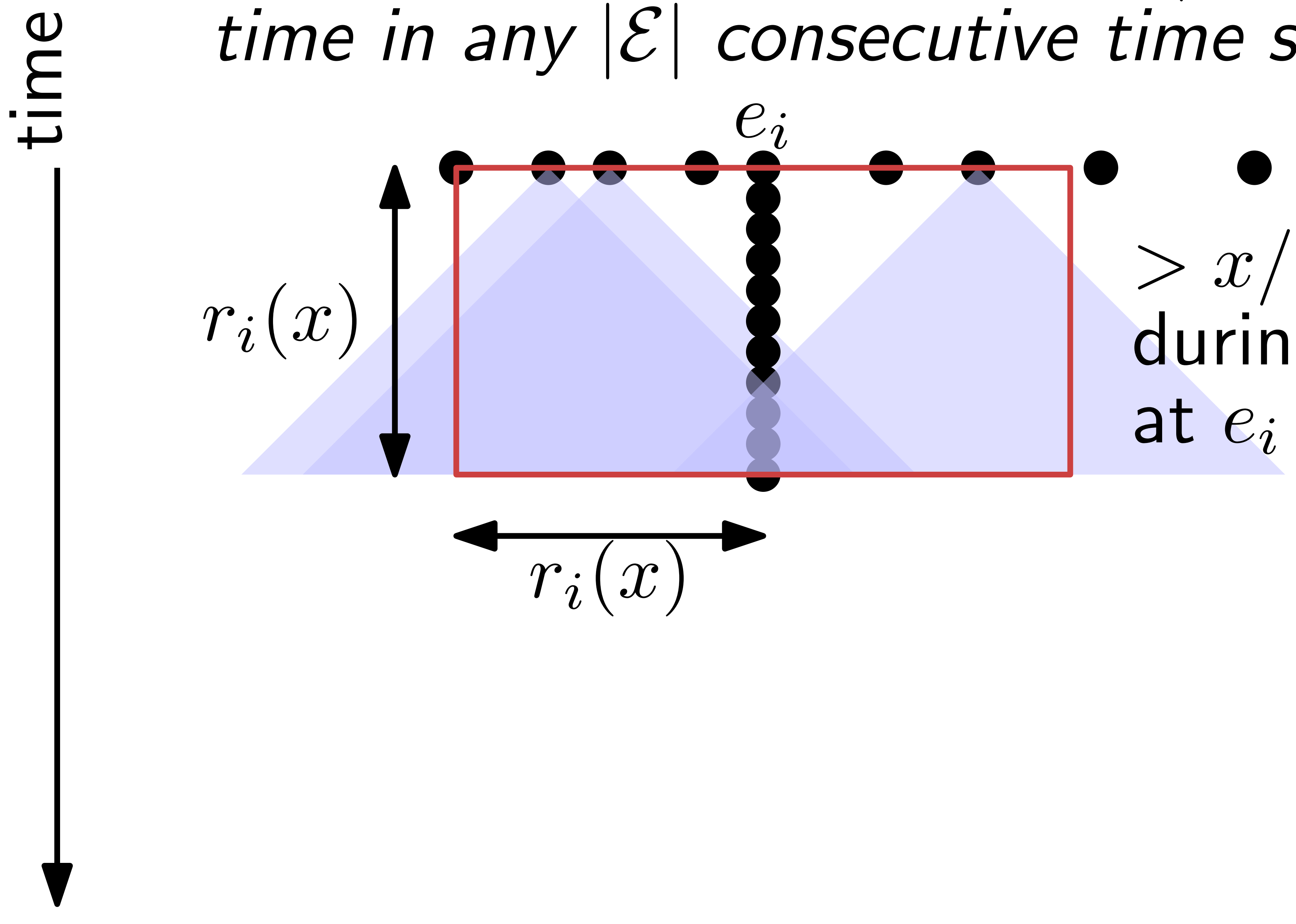
kissing in dim d

Lemma. *If* $\sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} \geq 4\kappa_d$ *then any query strategy suffers* **ply** $\geq x/2$ *at some time in any* $|\mathcal{E}|$ *consecutive time steps.*

Unavoidable interference

kissing in dim d

Lemma. *If* $\sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} \geq 4\kappa_d$ *then any query strategy suffers* **ply** $\geq x/2$ *at some time in any* $|\mathcal{E}|$ *consecutive time steps.*



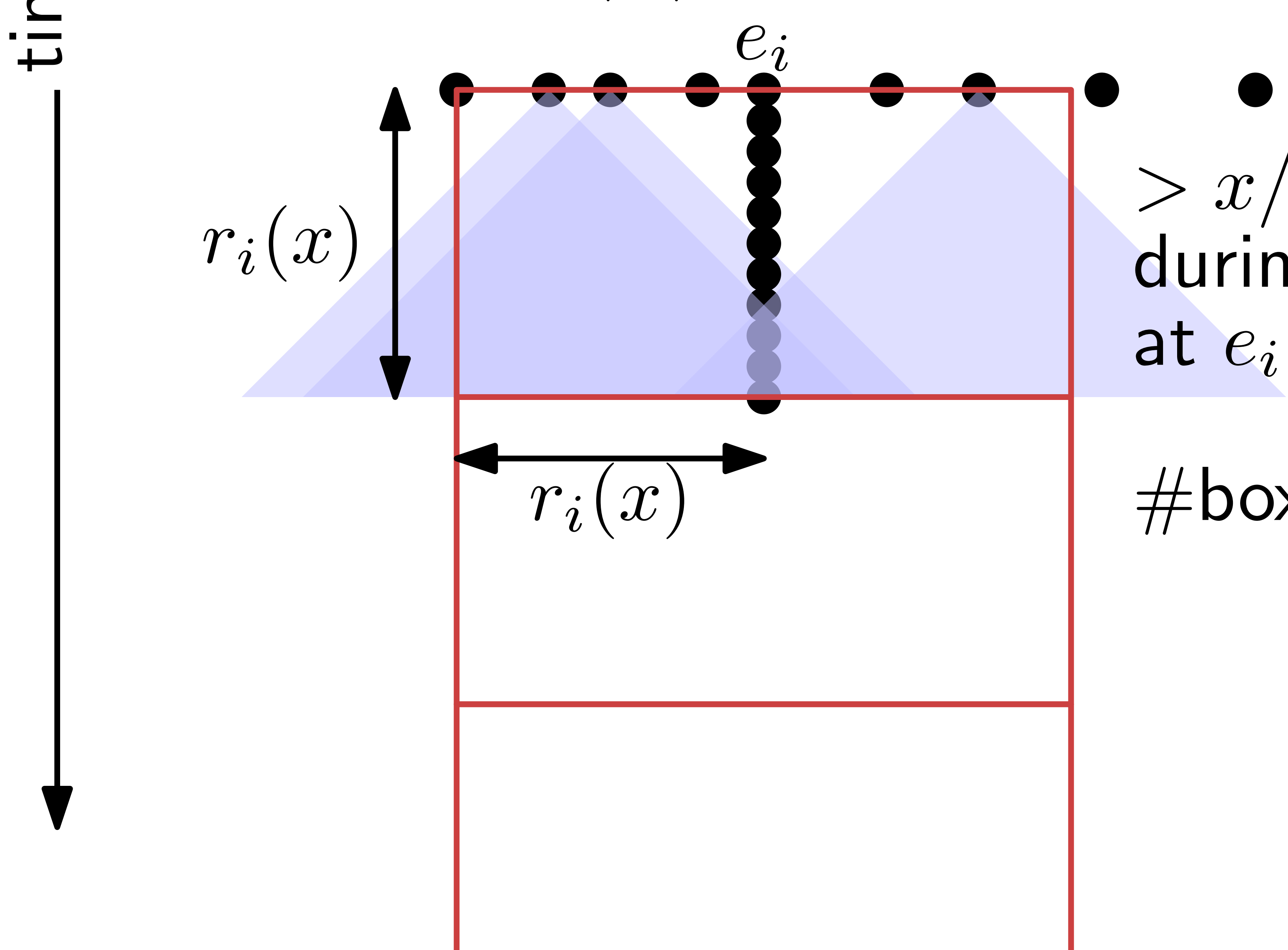
Suppose not...

$> x/2$ nbrs_x queried during box (or ply $\geq x/2$ at e_i at end of box)

Unavoidable interference

kissing in dim d

Lemma. *If* $\sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} \geq 4\kappa_d$ *then any query strategy suffers* **ply** $\geq x/2$ *at some time in any* $|\mathcal{E}|$ *consecutive time steps.*



Suppose not...

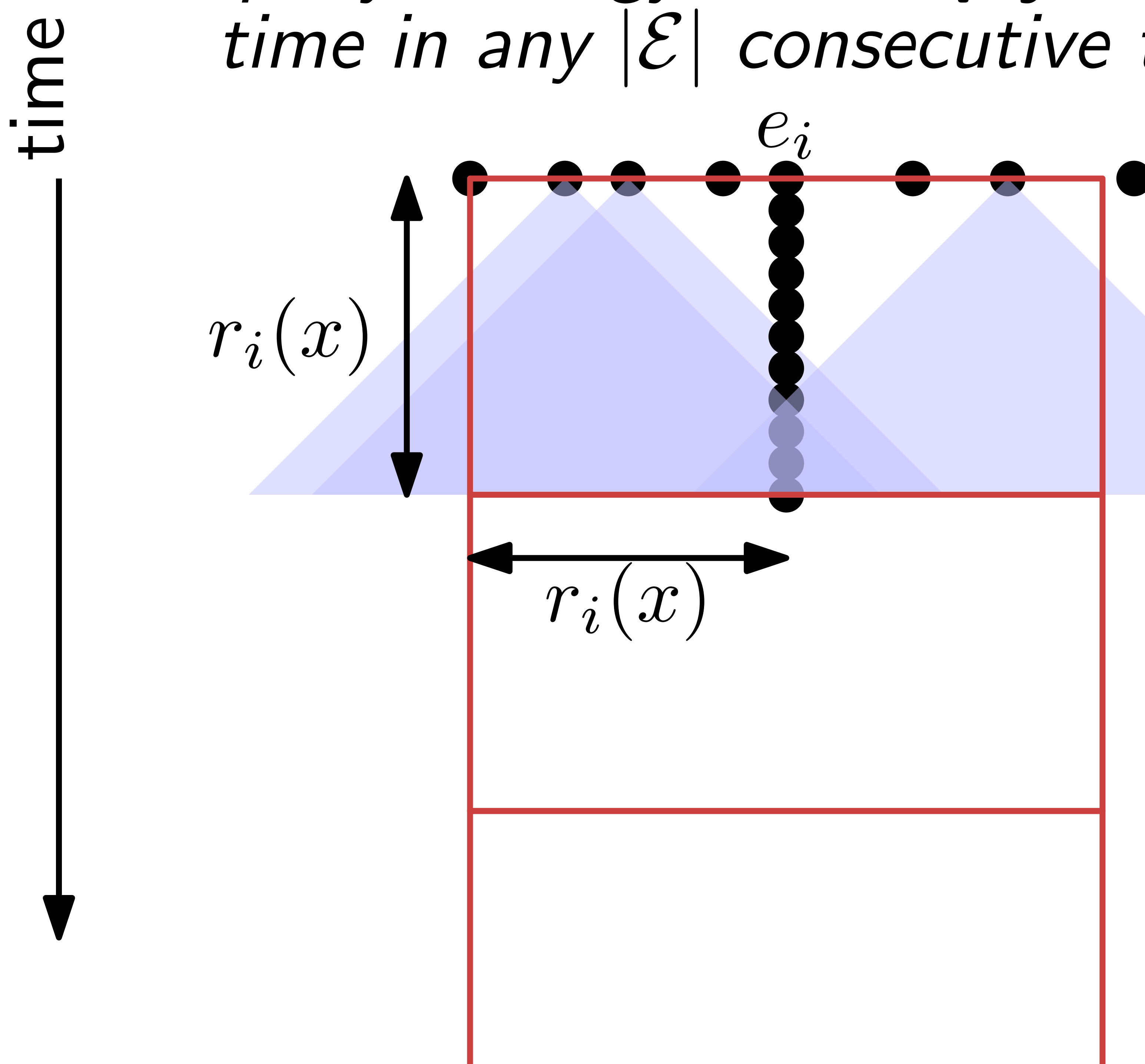
$> x/2$ nbrs$_x$ queried during box (or ply $\geq x/2$ at e_i at end of box)

#boxes $\geq \sum_{e_i \in \mathcal{E}} \left\lfloor \frac{|\mathcal{E}|}{r_i(x)} \right\rfloor$

Unavoidable interference

kissing in dim d

Lemma. *If* $\sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} \geq 4\kappa_d$ *then any query strategy suffers* **ply** $\geq x/2$ *at some time in any* $|\mathcal{E}|$ *consecutive time steps.*



Suppose not...

$> x/2$ nbrs$_x$ queried during box (or ply $\geq x/2$ at e_i at end of box)

#boxes $\geq \sum_{e_i \in \mathcal{E}} \left\lfloor \frac{|\mathcal{E}|}{r_i(x)} \right\rfloor$

Each entity is nbr$_x$ of at most $x\kappa_d$ entities e_i.

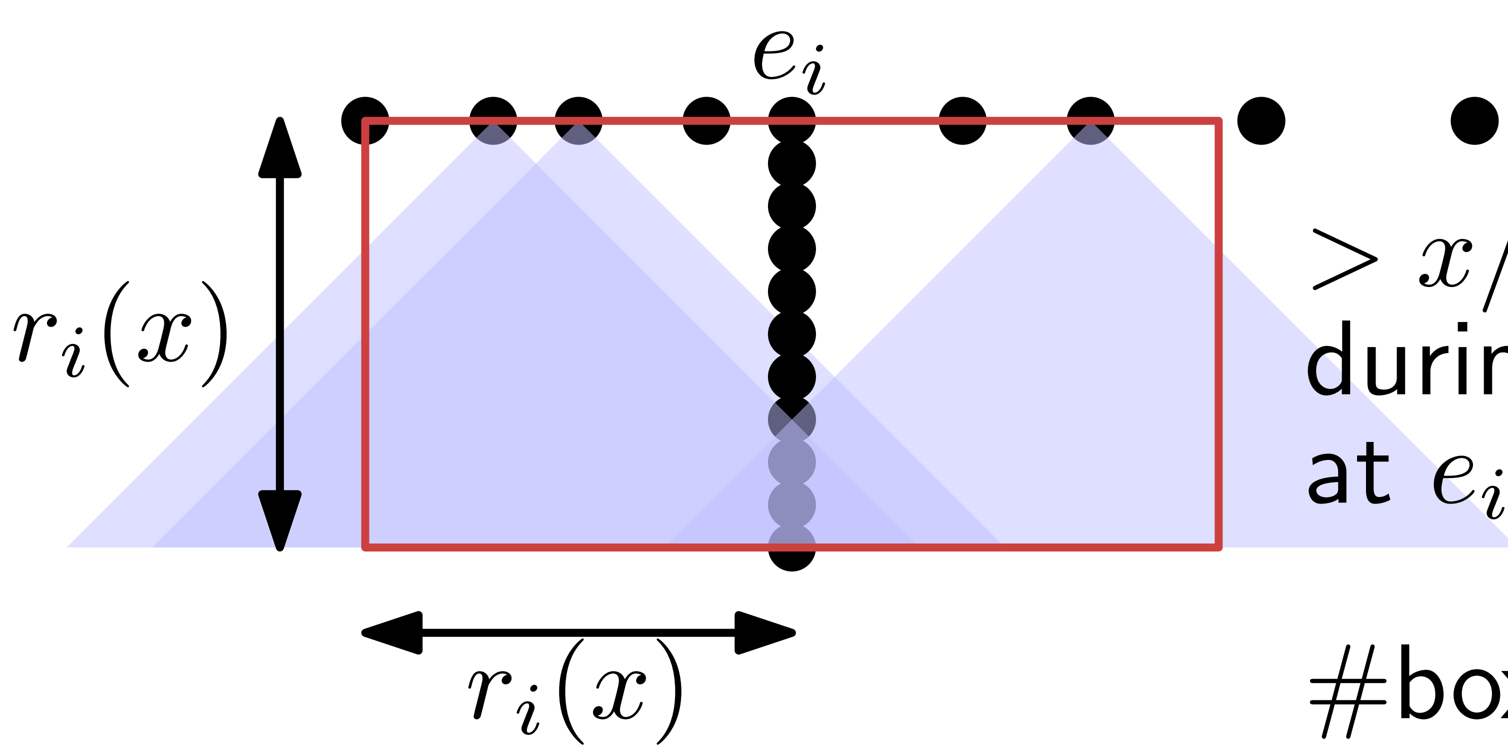
geometric lemma

Unavoidable interference

kissing in dim d

Lemma. *If* $\sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} \geq 4\kappa_d$ *then any query strategy suffers* **ply** $\geq x/2$ *at some time in any* $|\mathcal{E}|$ *consecutive time steps.*

time



Suppose not...

$> x/2$ nbrs$_x$ queried during box (or ply $\geq x/2$ at e_i at end of box)

$\#\text{boxes} \geq \sum_{e_i \in \mathcal{E}} \left\lfloor \frac{|\mathcal{E}|}{r_i(x)} \right\rfloor$

$|\mathcal{E}| = \#\text{queries} \geq \frac{1}{x\kappa_d} \frac{x}{2} \#\text{boxes}$

$> \frac{|\mathcal{E}|}{4\kappa_d} \sum_{e_i \in \mathcal{E}} \frac{1}{r_i(x)} \geq |\mathcal{E}|$

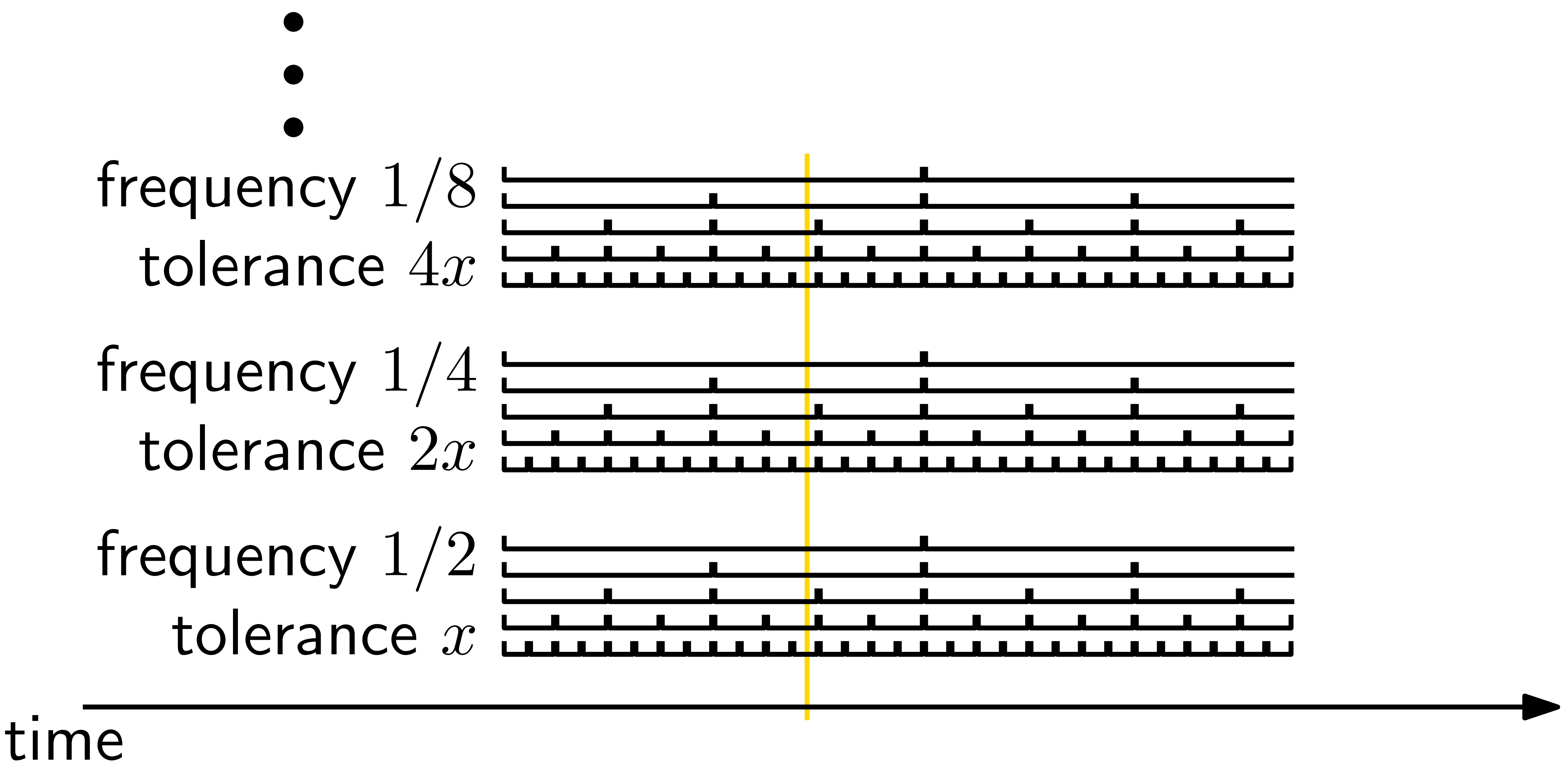
$\Rightarrow\Leftarrow$

Each entity is nbr$_x$ of at most $x\kappa_d$ entities e_i.

geometric lemma

Adapt to larger tolerance

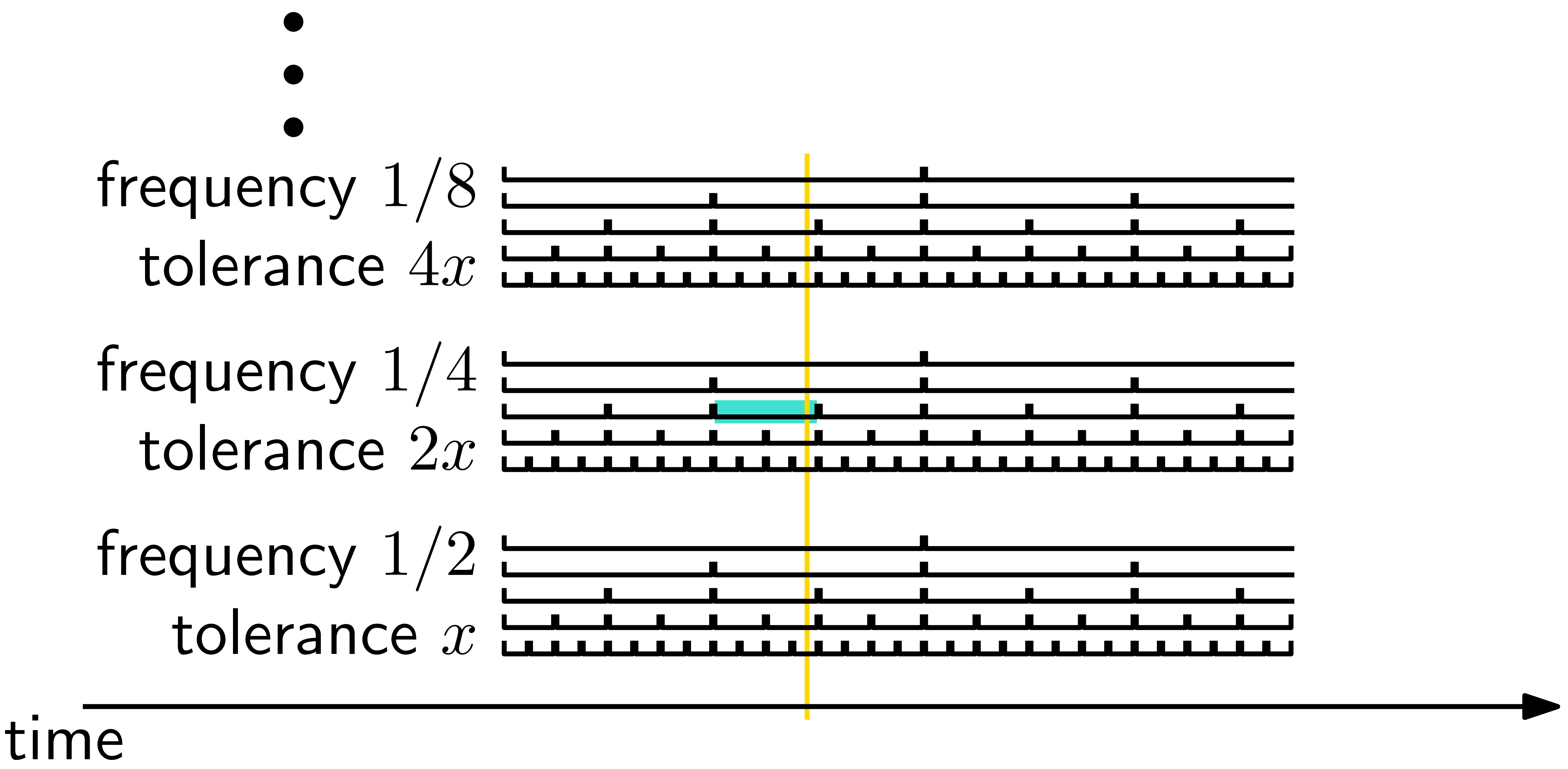
Rough idea: Run multiple Bucket Strategies in parallel



Adapt to larger tolerance

Rough idea: Run multiple Bucket Strategies in parallel

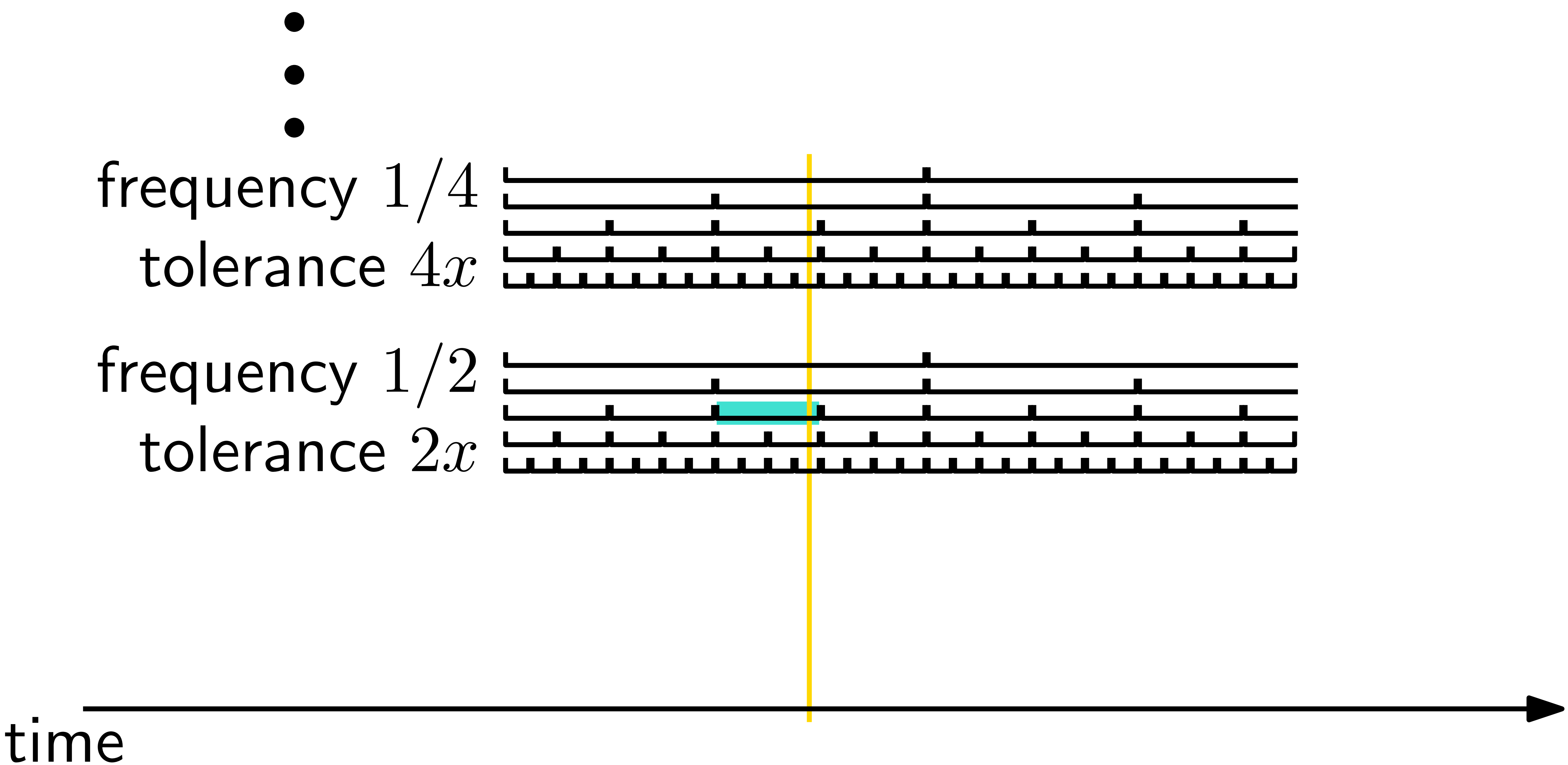
Failure of any implies ply $\Omega(x)$ unavoidable [scaling lem]



Adapt to larger tolerance

Rough idea: Run multiple Bucket Strategies in parallel

- Failure of any implies ply $\Omega(x)$ unavoidable [scaling lem]
- Stop strategy for x and divvy up its frequency

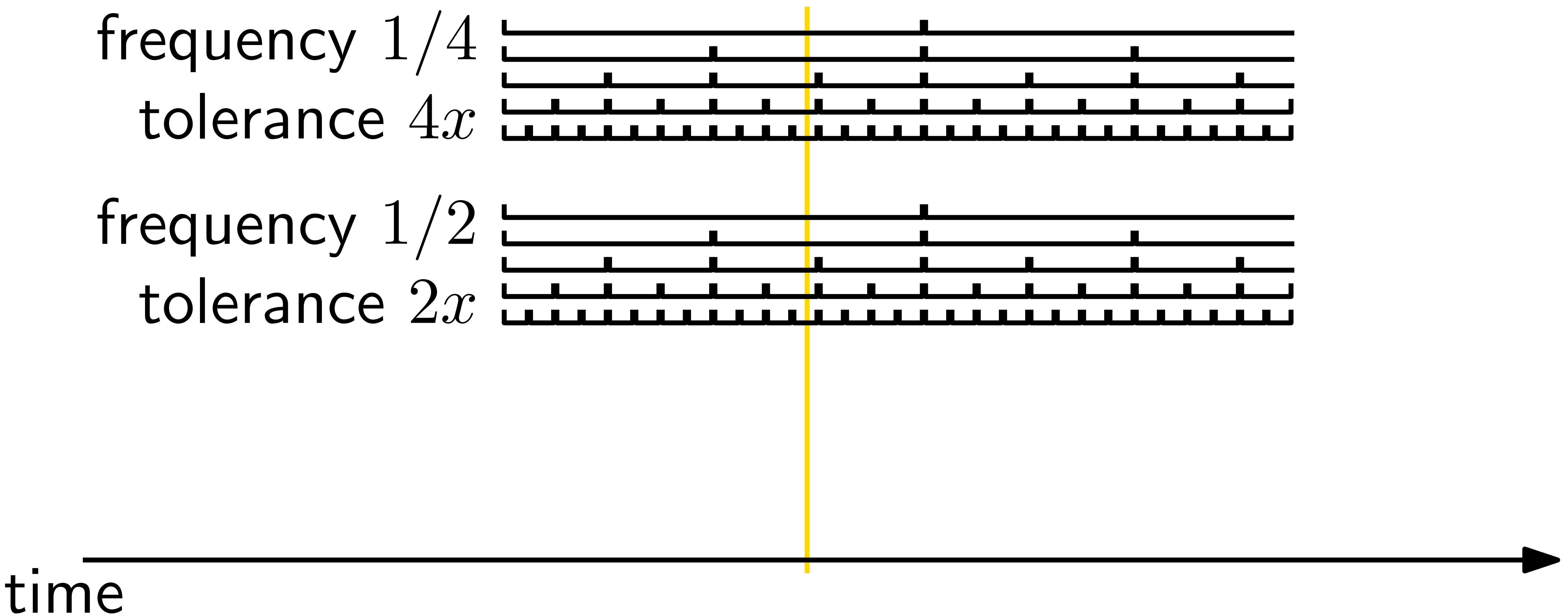


Adapt to larger tolerance

Rough idea: Run multiple Bucket Strategies in parallel

- Failure of any implies ply $\Omega(x)$ unavoidable [scaling lem]
- Stop strategy for x and divvy up its frequency
  - Empty all active buckets to special queues

$\vdots$



Adapt to smaller tolerance

Rough idea: In parallel with everything else,
Round-robin query all entities.

Adapt to smaller tolerance

Rough idea: In parallel with everything else,
Round-robin query all entities.

Let E_0 be those that are **not** $(x/2)$-safe for n steps
If $|E_0| > n/2$ then restart
If $|E_0| = 0$ then add Bucket Strategy $x/2$

region cannot contain more than $x/2$ entities in the next n steps

Adapt to smaller tolerance

Rough idea: In parallel with everything else,
Round-robin query all entities.

Let E_0 be those that are **not** $(x/2)$-safe for n steps
If $|E_0| > n/2$ then restart
If $|E_0| = 0$ then add Bucket Strategy $x/2$

for $k = 1$ to $\lg(n)$
Round-robin query E_{k-1} for $n/2^k$ steps
Let E_k be those that are **not** $(x/2)$-safe for $n/2^k$ steps
If $|E_k| > n/2^k$ then restart
If $|E_k| = 0$ then add Bucket Strategy $x/2$