Motivation

We may need to alter the historical sequence of transactions.

- **Dynamization.** Some static algorithms require a dynamic data structure to process the input. These algorithms can be made dynamic by using a retroactive data structure instead.
- **Bad Data.** Data is either missing, was entered incorrectly, or comes from a tainted source (e.g. a sensor has malfunctioned).
- **Efficient Adaptation.** We may want to undo an error made early on without redoing all the work since then.

**Goal:** Maintain a large amount of data which changes over time such that we can efficiently update and query the data. Support updates and queries on current and past versions of the data structure.

Definition

A data structure is **fully retroactive** if it supports queries and updates to current and past versions.

**Persistent** data structures also maintain past versions, but only retroactive data structures propagate updates forward in the timeline.

**General Approach**

- Insert
- Delete
- **Value**
- **Time**
- **Query and/or Update at any point in time.**

**Segment tree over time**

**Normal Query**

**Retroactive Query**

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.

**Approximate Range Searching**

- **Input:** set of points in \( \mathbb{R}^d \).
- Updates can insert or delete points at any time.
- Query: return points in approximate range at time \( t \).

**Figure:** Points within \( Q^- \) must be returned. Points outside \( Q^+ \) are never returned. Points between \( Q^- \) and \( Q^+ \) may or may not be returned.

**Query:** Recursively search stabbing squares

For each Inner Square:

- Reduce to 1-d query using Z-order
- Perform 1-d query in augmented segment tree

**Figure:** Z-order curve projects quadtree down to one dimension.