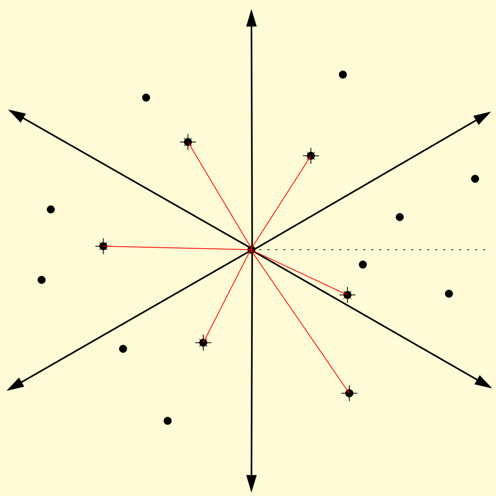
The Closest Pair Problem

Suppose that we want to maintain the closest pair among n moving points in the plane. Our kinetic closest-pair (κ-CP) algorithm is based on a new static CP algorithm.

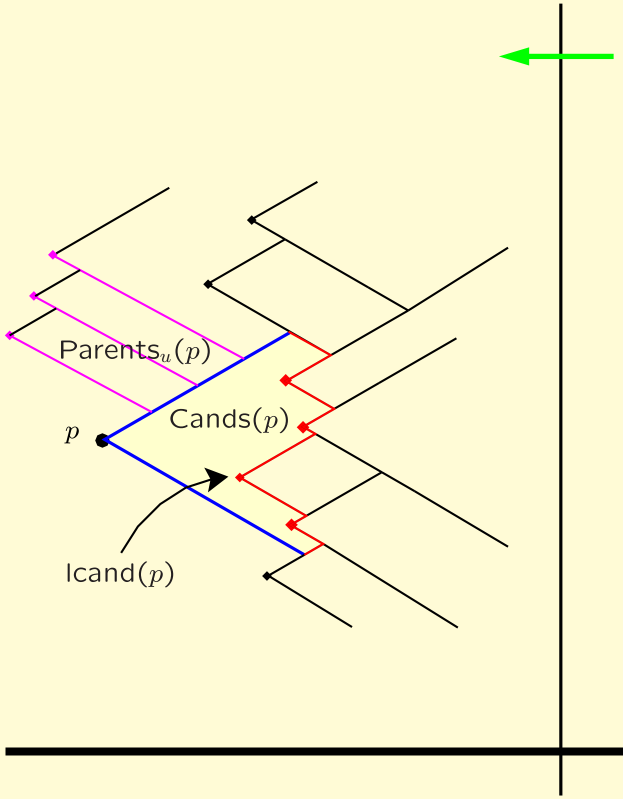


We maintain, for each point, directional nearest neighbors in each of three cones.

The CP must be one of the $3n$ pairs thus defined.

Sweeping to Obtain Directional Neighbors

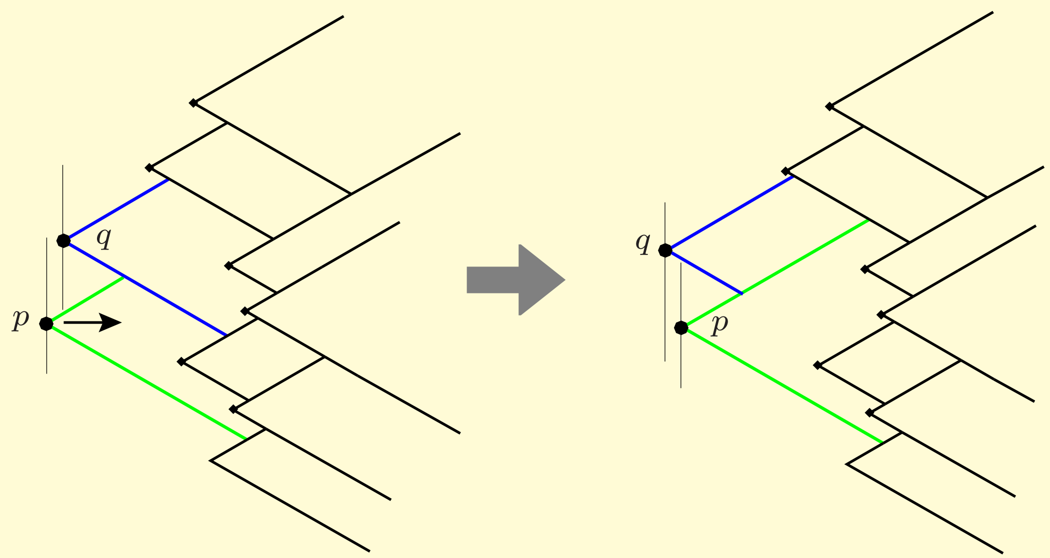
The horizontal sweep computes three data structures for each point p, Cands(p), Parents$_u(p)$ and Parents$_d(p)$.



Each of the three sweeps can be done in $O(n \log n)$ time and will detect all directional neighbors in its direction.

Kinetizing the CP Algorithm

For κ-CP, each of Cands(p), Parents$_u(p)$ and Parents$_d(p)$ is maintained as a balanced tree supporting efficient merges and splits. The only relevant events are changes in the order of the points along the 0°, 60°, and 120° directions.



Completing the CP Maintenance

There are $O(n^2)$ order exchange events to process, each of which can be processed in polylog time.

To determine the CP, we neeed to also maintain a κ-T on the distances of all the points to their directional neighbors. This tournament must handle flight plan updates or discrete changes, whenever these directional neighbors change.

It can be shown that that the cost of processing the tournament events is also roughly quadratic.

Thus κ-CP is efficient, responsive, local, and compact.