Parallel metric trees for multi-billion bodies simulations

Raul H. C. Lopes[1], Peter R Hobson[1]

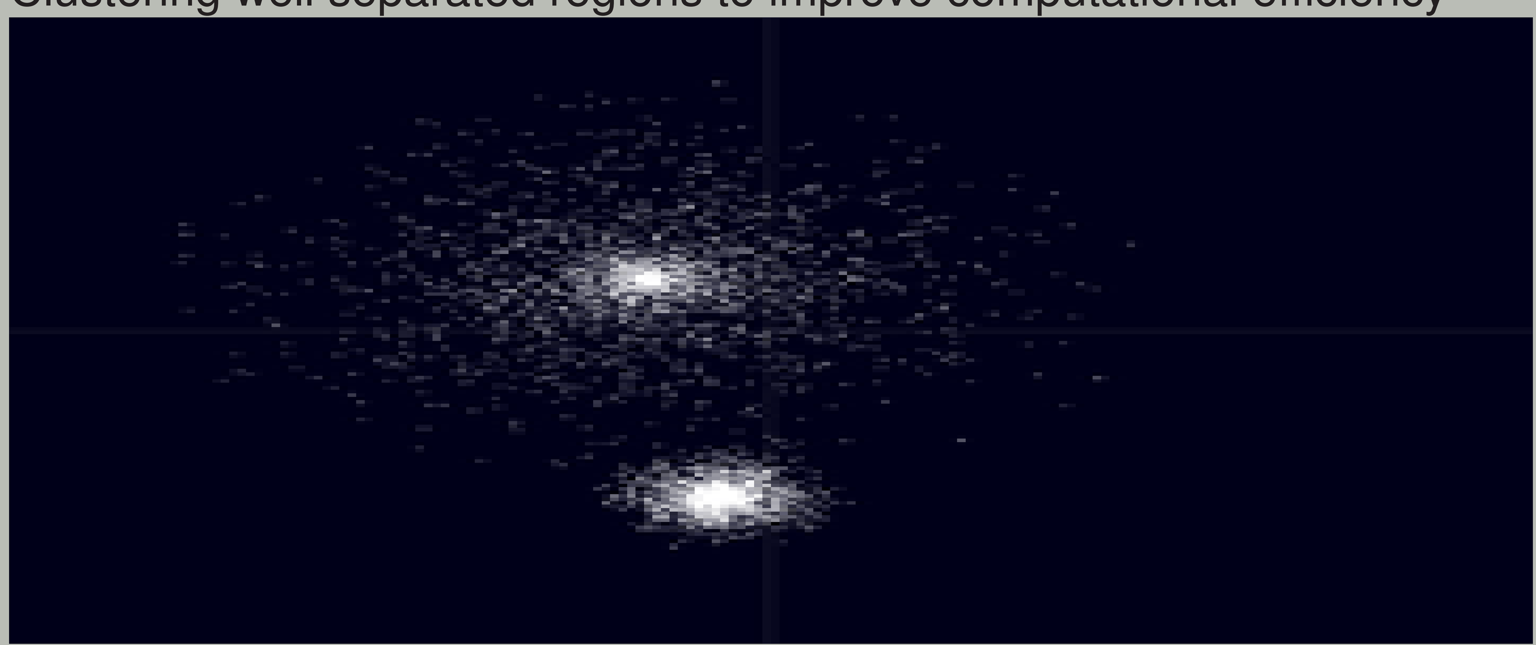
[1]Brunel University London, UK

A research question

- Theoretical foundations and tool
  - Computational Geometry
    Vantage point trees
    Metric trees
  - Parallel algorithms
    Contraction algorithms
    Accumulation trees
  - Parallel architectures
    Multi-core
    GPGPU and extended device computing
    Distributed memory archittecures
- **The research question**
  How effective can the algorithms and data structures of theorectical parallel computing be when applied to practical problems in high dimensional domains.

High dimensional data processing examples

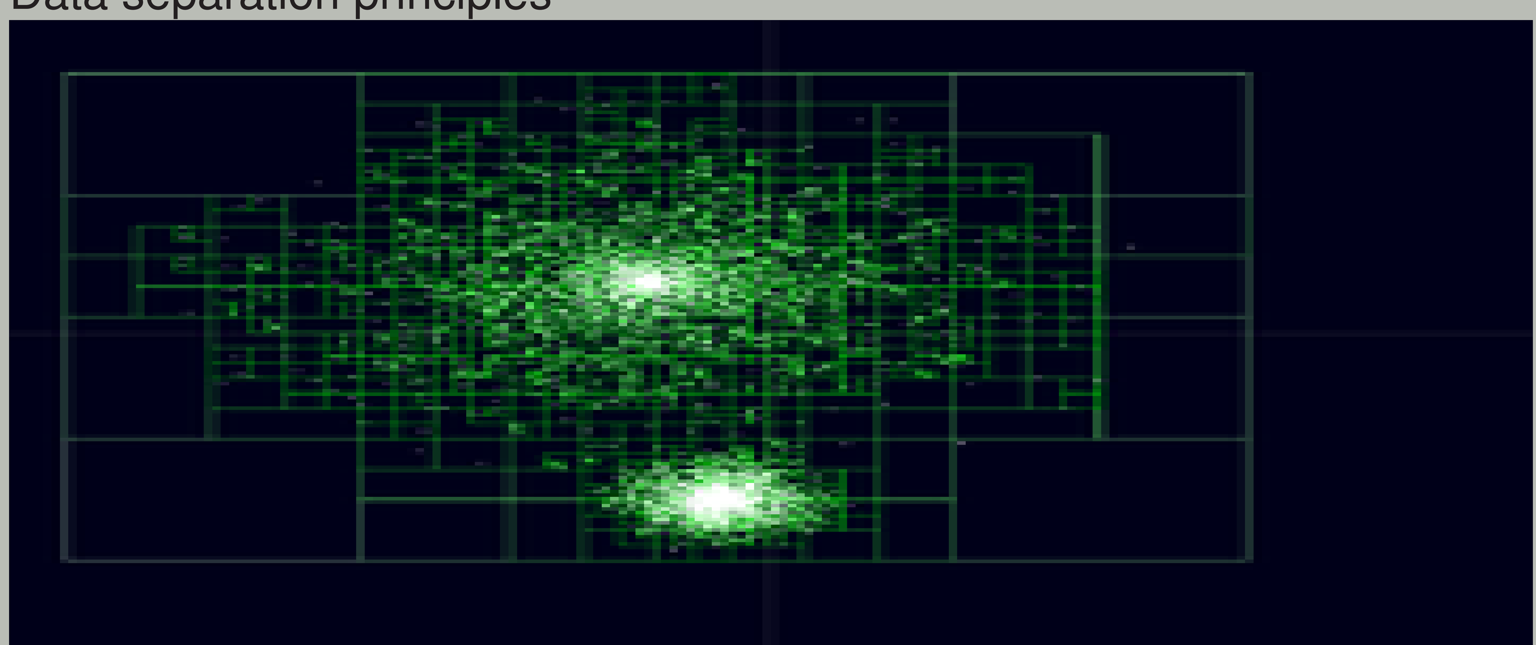
- Clustering well separated regions to improve computational efficiency



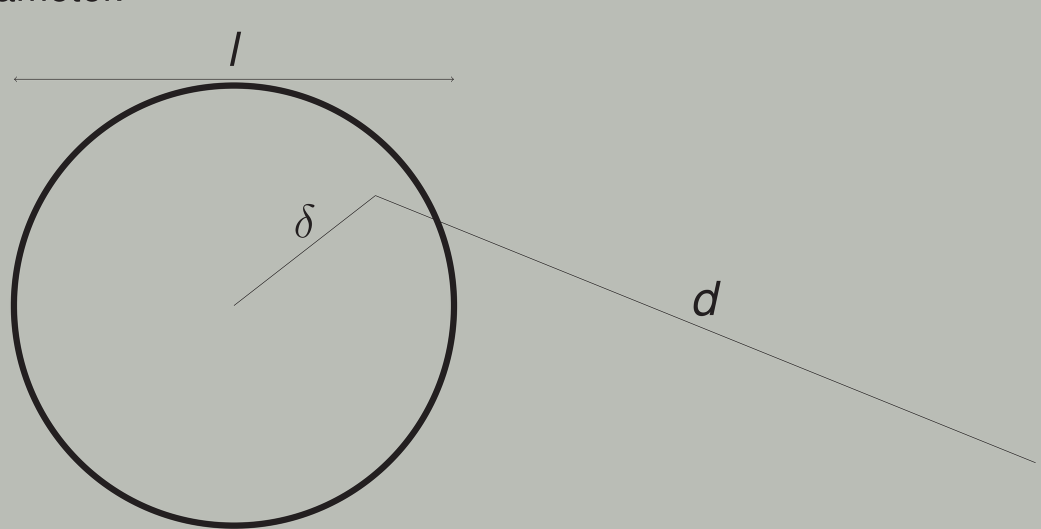
- Clustering well separated blocks of points to improve line detection.
  Possible figure here showing OpenCV HT failure
- Higher order visualization and sequence matching in DNA or data mining in time series.
  Sergey Brin (before Google): *A ball of radius 2 in 20 dimension Euclidean space is a million times larger in volume than a ball of radius*
  Metric trees can reduce partition the space.

Computational Geometry data structures

- Data separation principles



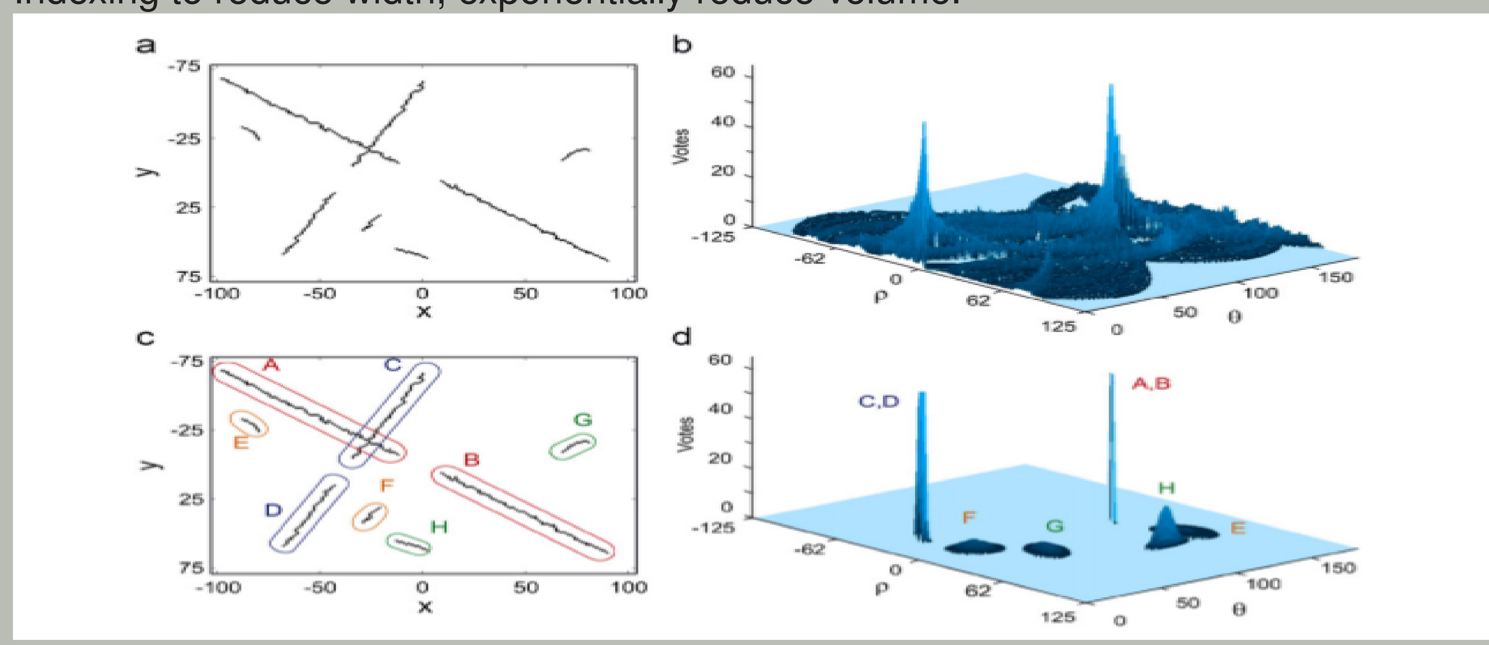
  - Mathematically well founded
  - Implemented using metric trees, or quad-trees.
  - Example: n-body tree codes.
    A particle is well separated from a region when $s/d < \theta$ for
    s, the width of the region
    d, distance between the paricle and the centre of mass of the region
    θ, a given parameter.



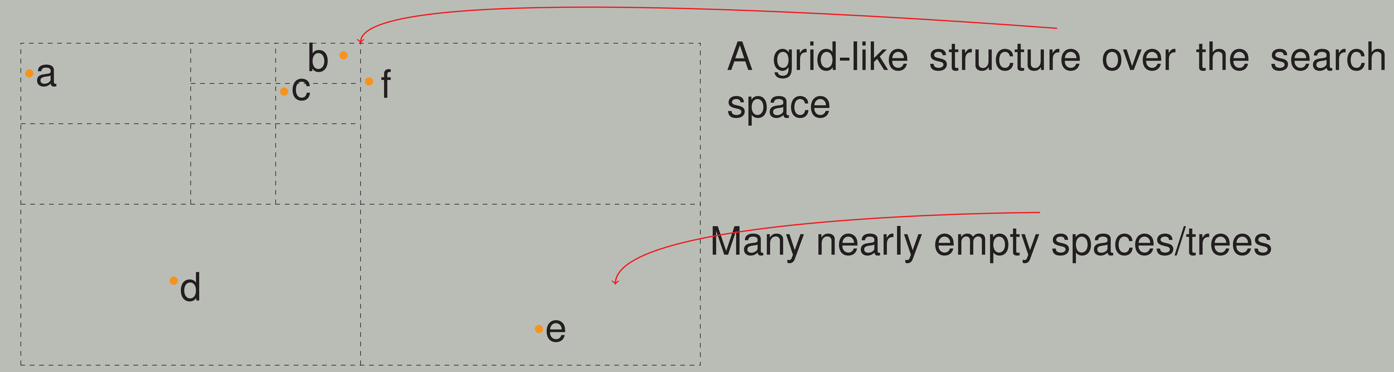
- Problems
  - Parallel algorithm to extract pairs separated regions and particles.
  - Parallel accumulation algorithms. Accumulation can be:
    FMM Taylor expansion;
    Accumulation of force and mass in Barnes-Hut;
    Binning (vote counting) in algorithms for variations of Hough Transform.

Trees for high dimensional data

- Target
  - Indexing to reduce width, exponentially reduce volume.



  - Balancing of construction to optimize sharing of parallel workload.
- Data structures to separate high dimensional data
  - Octrees, and quadtrees tend to be inefficient in space and parallel work balancing;
  - k-d-trees can become unbalanced or have irregularly split regions;



  - Metric and vantage-point tree might offer round regions and binary trees good for upward and downward accumulation.

Metric trees

- Binary search tree.
- Indexed by distance.
- A node defines a sphere of the points living within a distance from its centre.
- Axioms that regulate the construction of a mertic tree
  - Each node is define by a central point C, a radius r , left and right subtrees.
  - Points on the left subtree are within a distance r from C.
  - Points on the right subtree are outside distance r from C.
  - Central point is any point in the indexed set.
  - Radius is the median of distances to that central point.
- Observations:
  - Each node is weight balanced.
  - Each node can be split with linear work, if we assume that a median can be computed with linear work.

Point separation

- Two sets of points P and Q are separated when
$$\frac{max(diam(Q), diam(R))}{d(Q, R)} < \theta$$
- A separated pairs decomposition of a set P is a set of pairs $(A_i, B_i)\}$ where
  - each $A_i, B_i \subset P$
  - A_i and B_i are θ-separated.
- Parallel complexity
  An $O(n \lg n)$ work parallel algorithm with depth $O(\lg n)$ can be defined based the observation that:
  - if sets P and Q are not separated then the separation set of P and Q is given by the separation sets of P_0 and Q and P_1 and Q, where P_0 and P_1 are children of Q.

N-body tree code

- Algorithm: each Physical time step simulated by a sequence of three paralle bulk steps.
  - In parallel: build a metric tree.
    Complexity: Computation with depth $\lg n$.
  - In parallel: build separated sets.
  - In parallel: for each pair (A_i, B_i), apply cell to cell computation to evalute force, centre of mass, position.

Acknowledgement

Raul Lopes and Peter Hobson would like to acknowledge financial support from the UK Science and Technology Facilities Council.

Particle Physics Group, School of Engineering and Design, Brunel Universirty, Uxbridge, UB8 3PH, UK peter.hobson@brunel.ac.uk