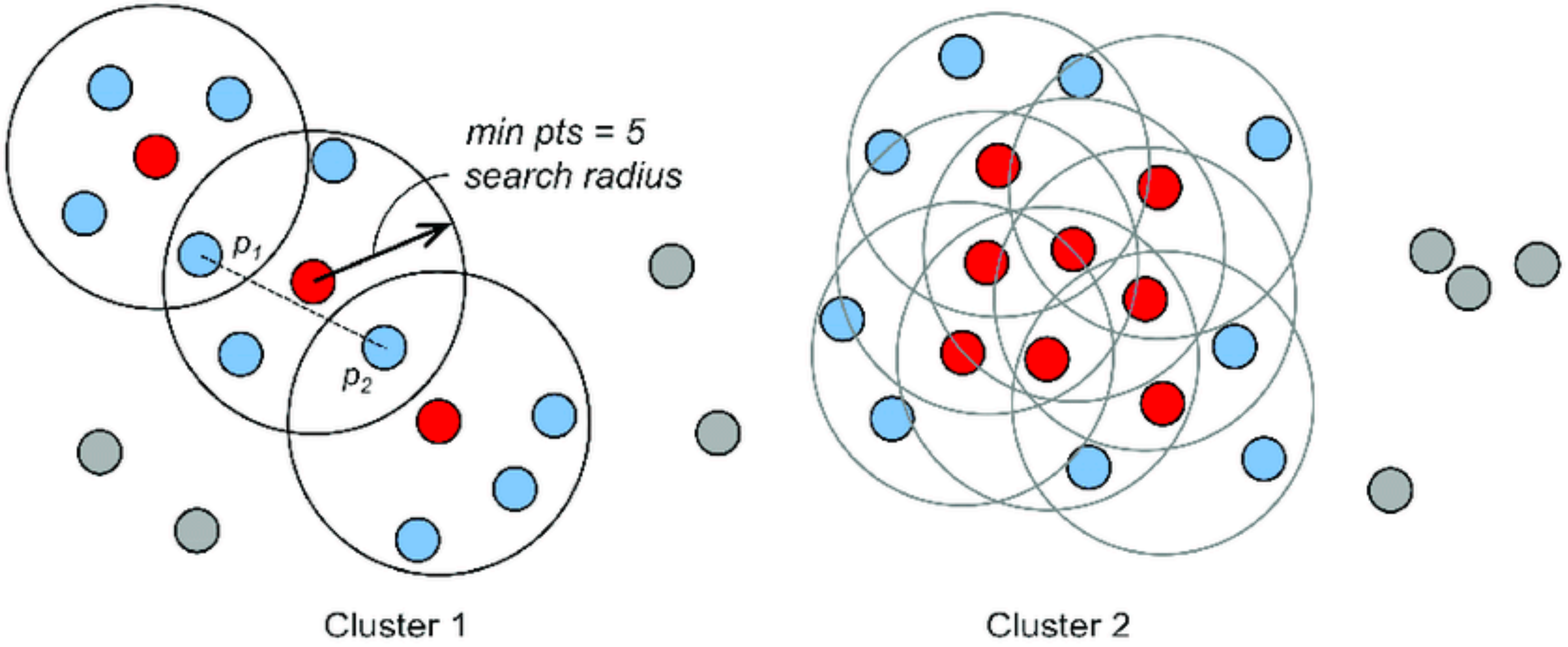
Parallel DBSCAN

Bowen Chen

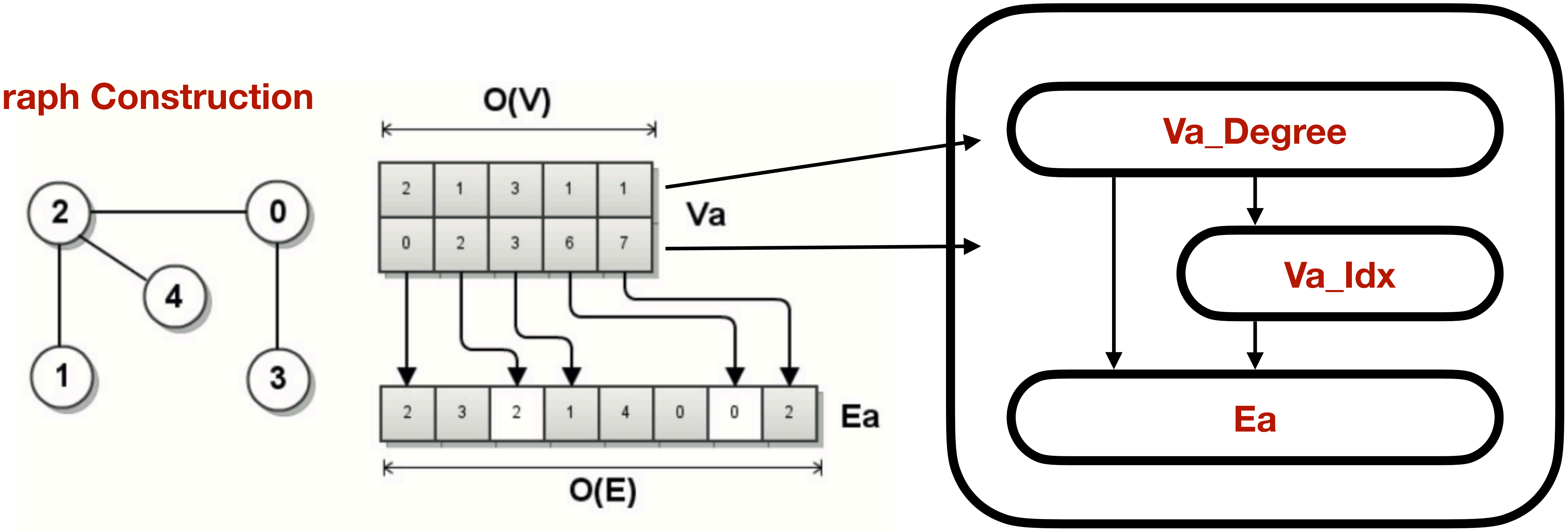
DBSCAN Overview



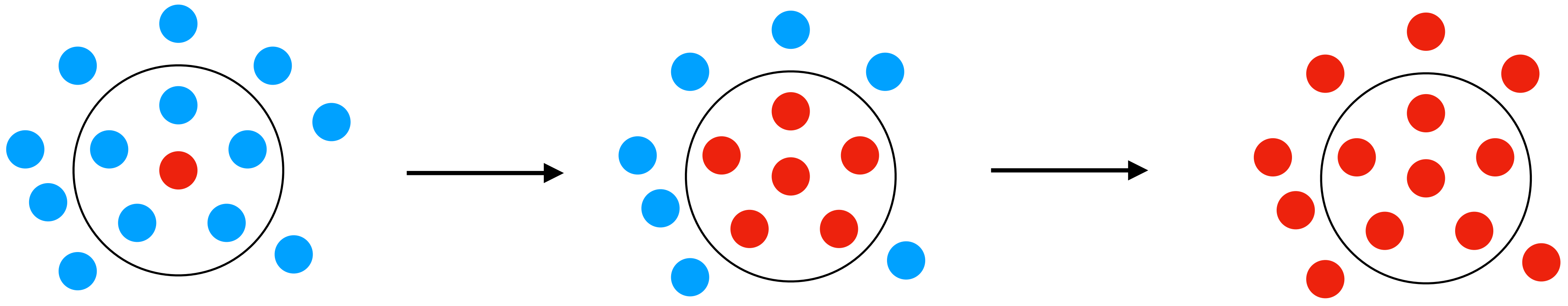
Break down: **1) find core points** **2) growing neighbours**

G-DBSCAN

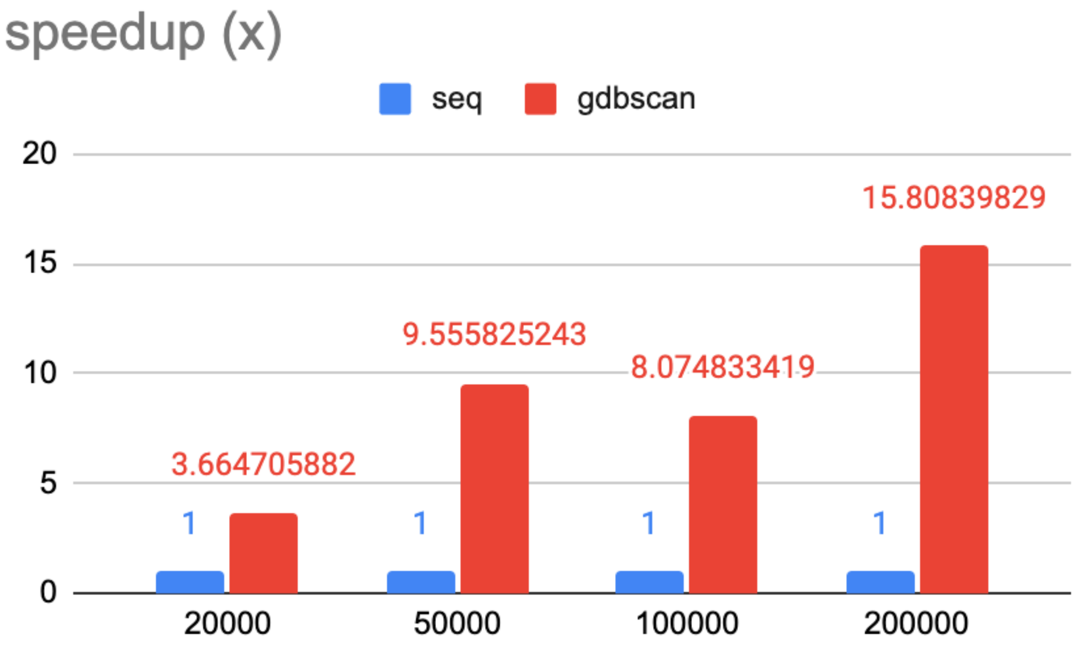
Graph Construction

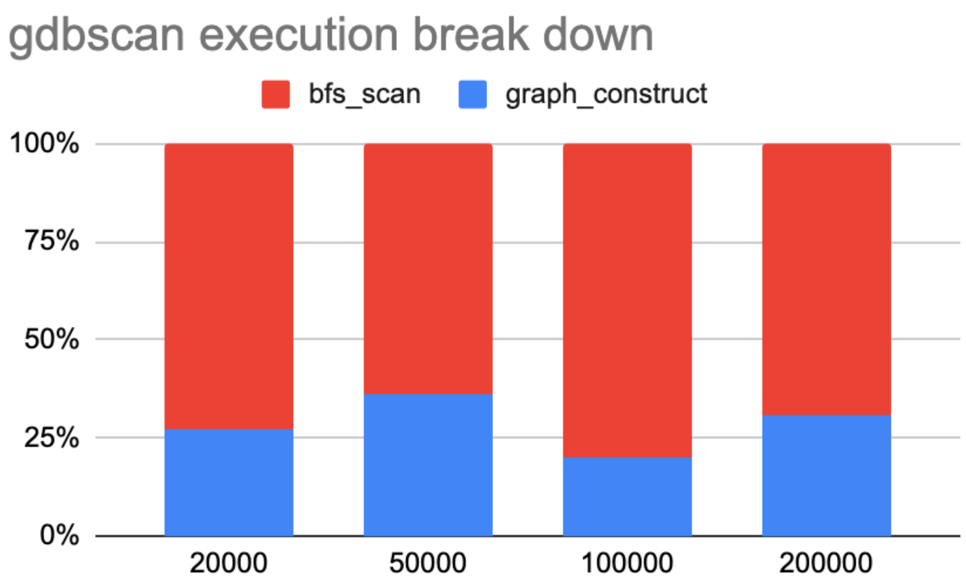


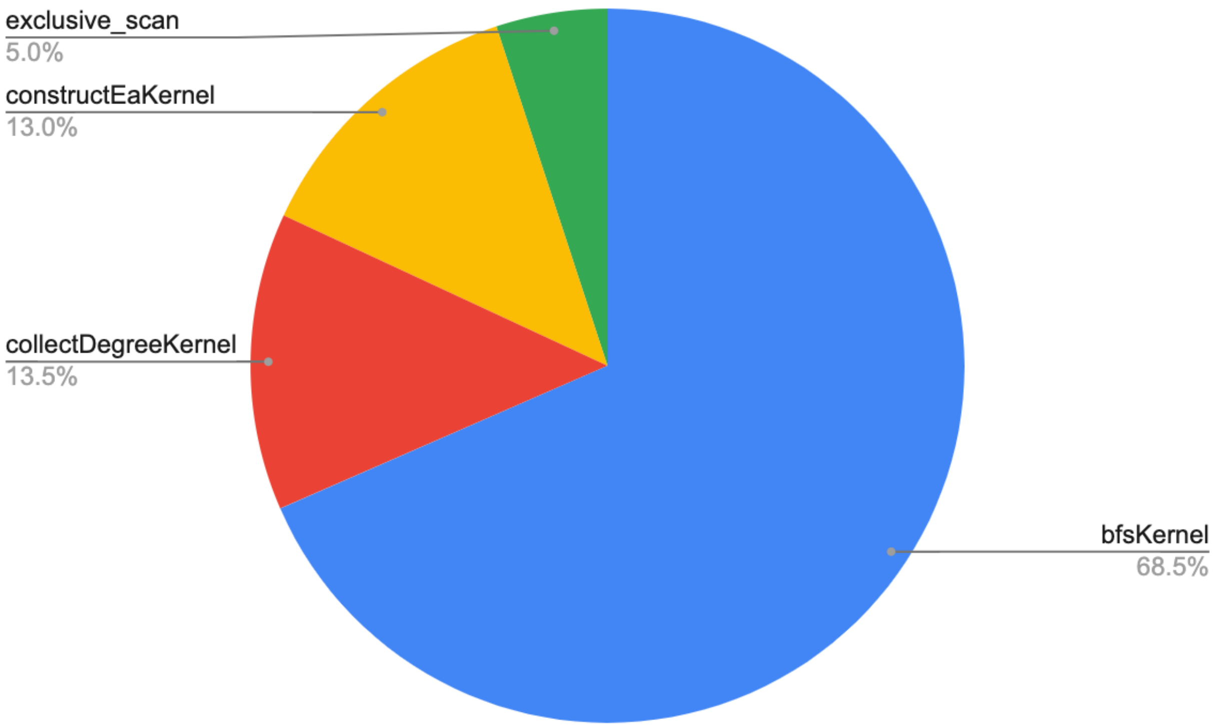
BFS SCAN



G-DBSCAN



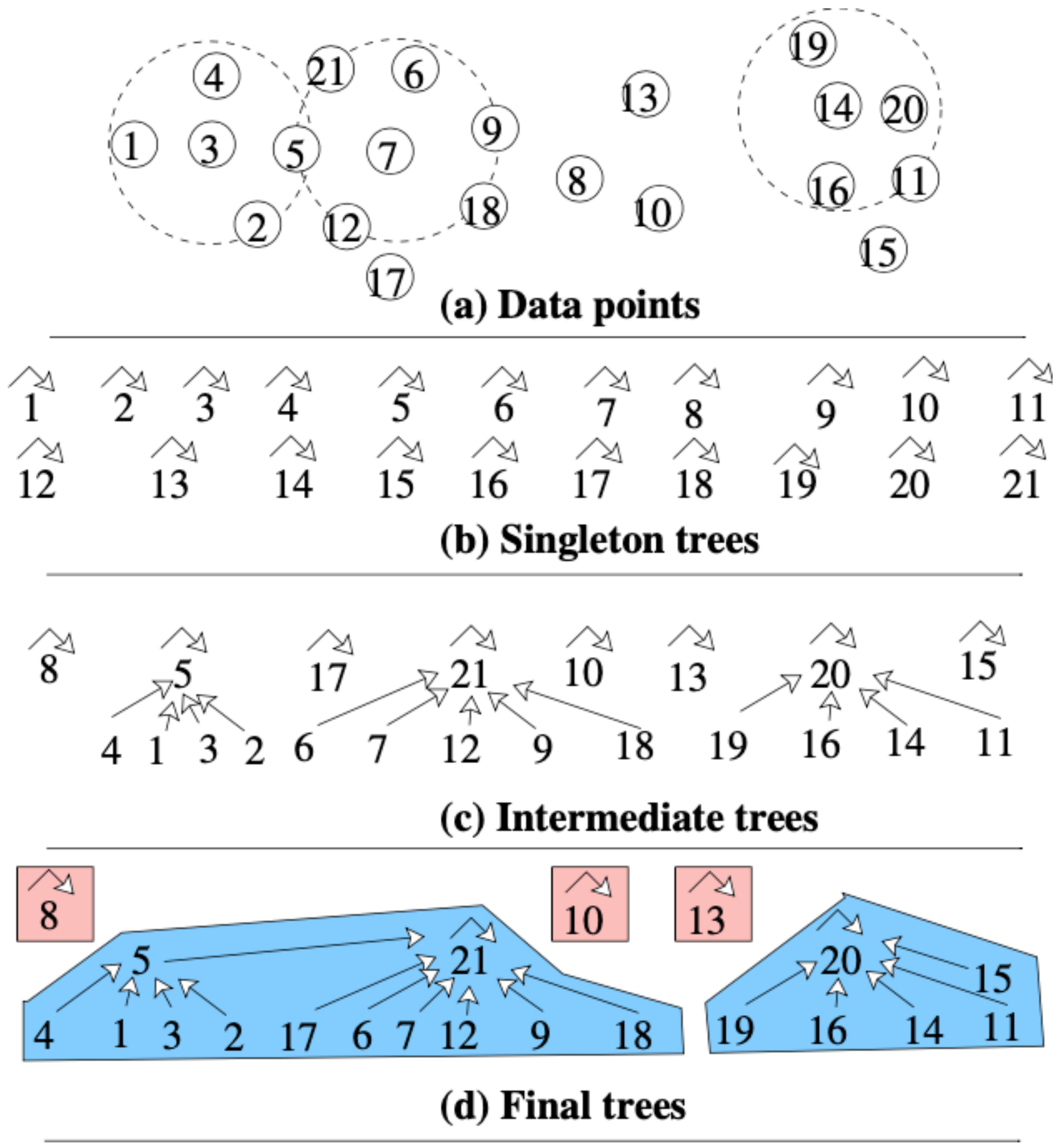




BFS SCAN is the bottleneck!!!

PDSDBSCAN (Disjoint-Set)

Key Idea: postpone union



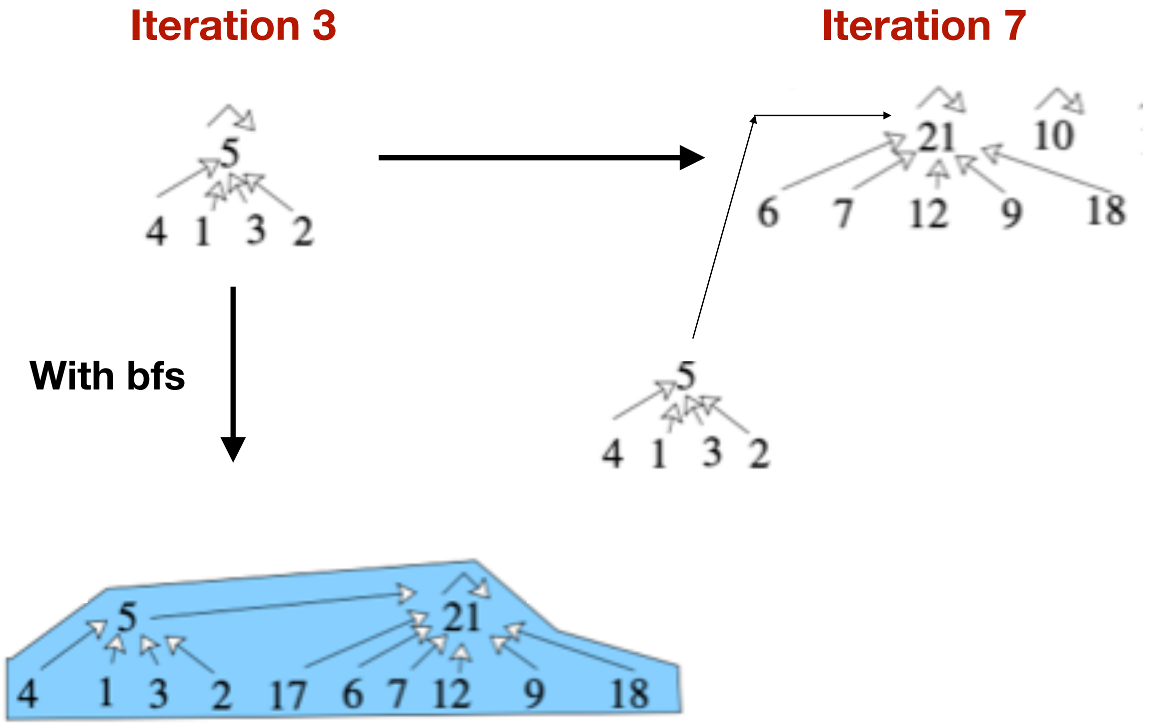
(a) Data points

(b) Singleton trees

(c) Intermediate trees

(d) Final trees

Iterate over points

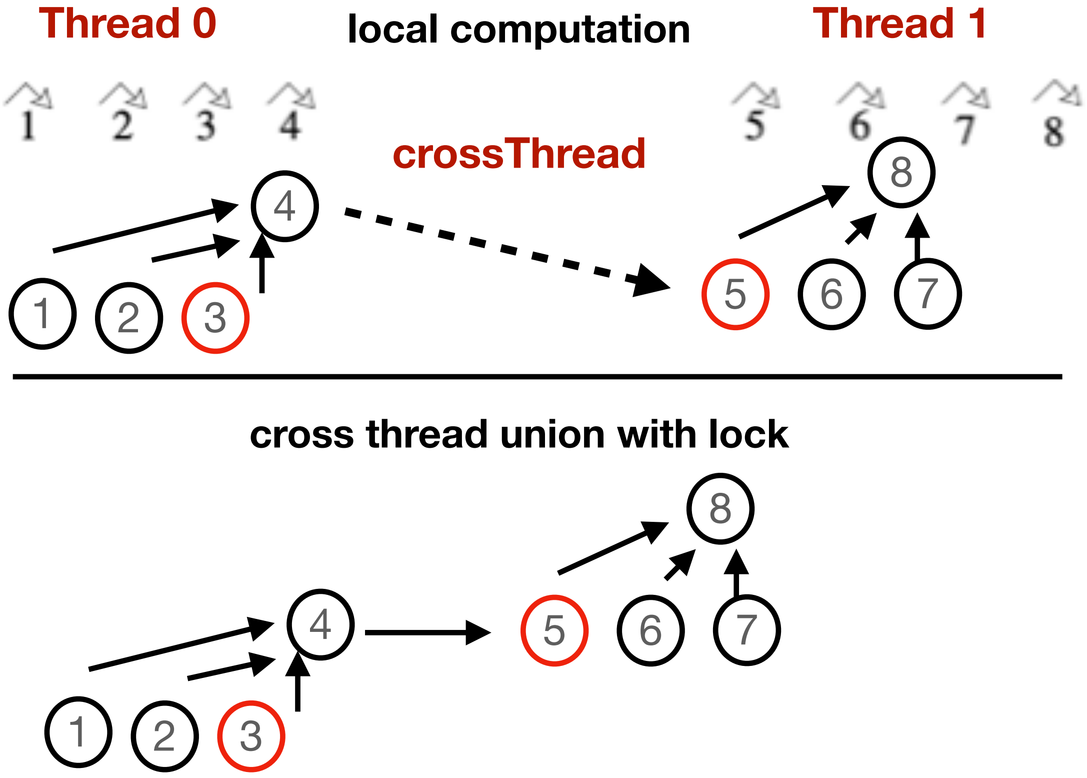


Iteration 3

Iteration 7

With bfs

Shared Memory Parallel with omp



Thread 0

local computation

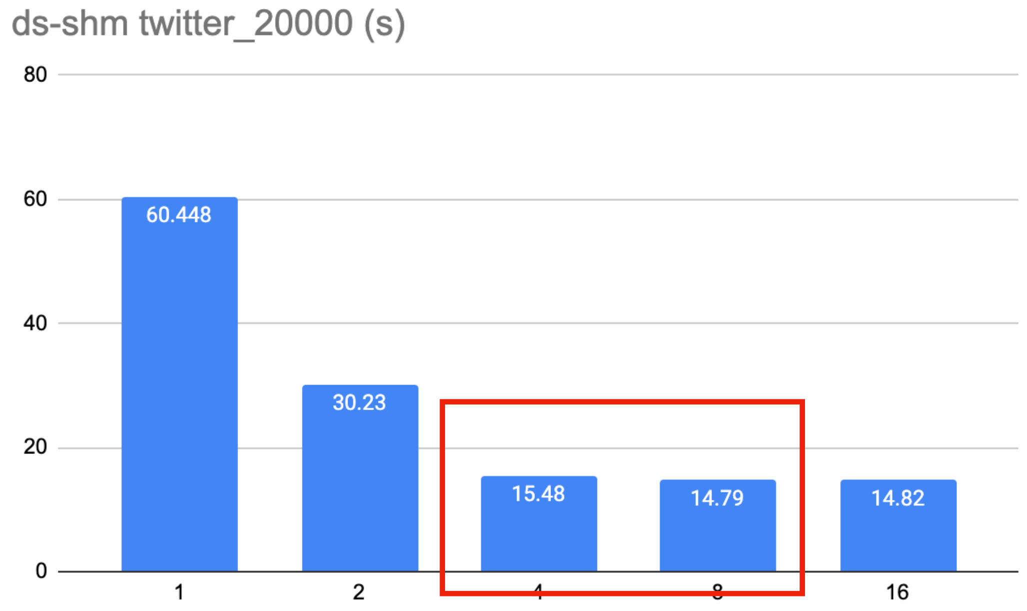
Thread 1

crossThread

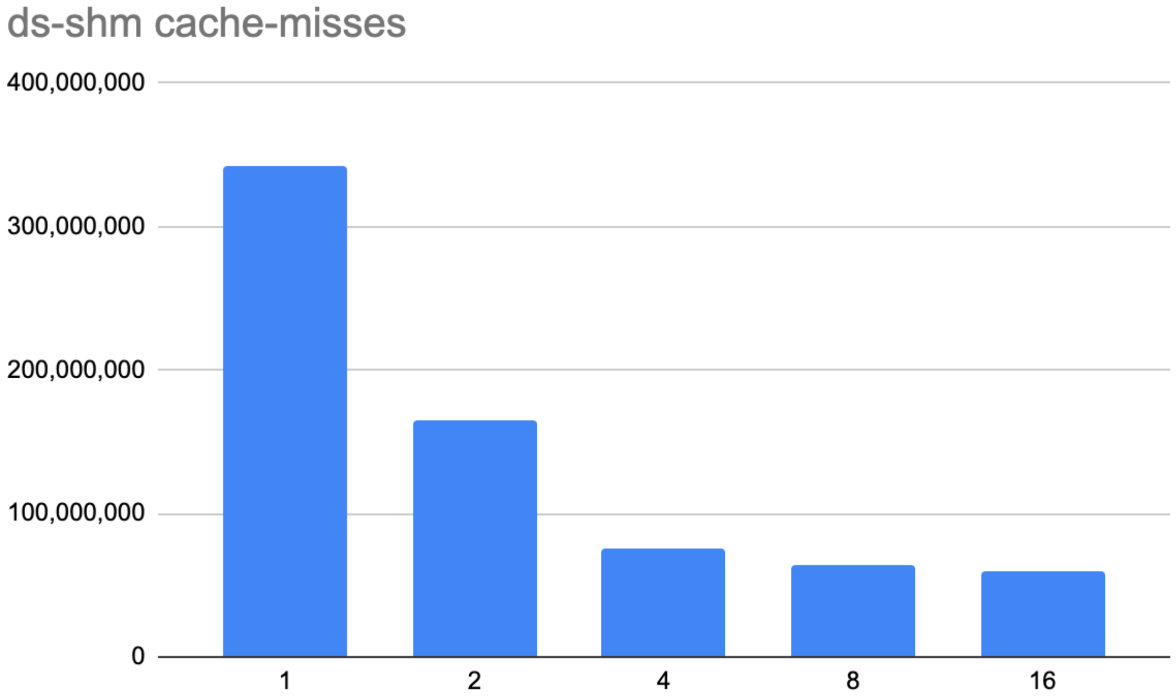
cross thread union with lock

PDSDBSCAN (Disjoint-Set)

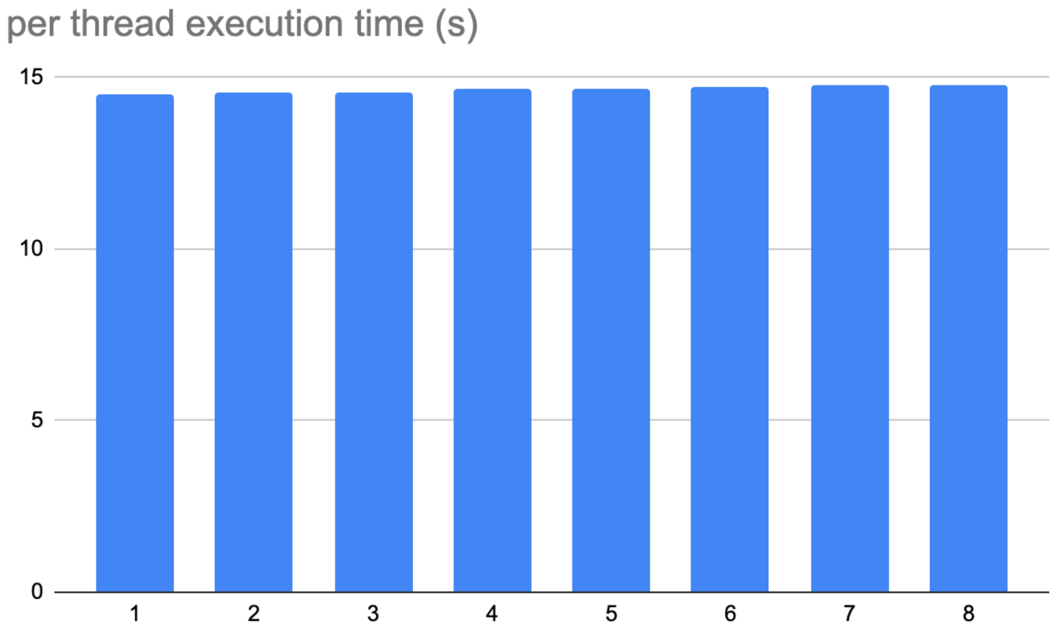
Execution time with different threads



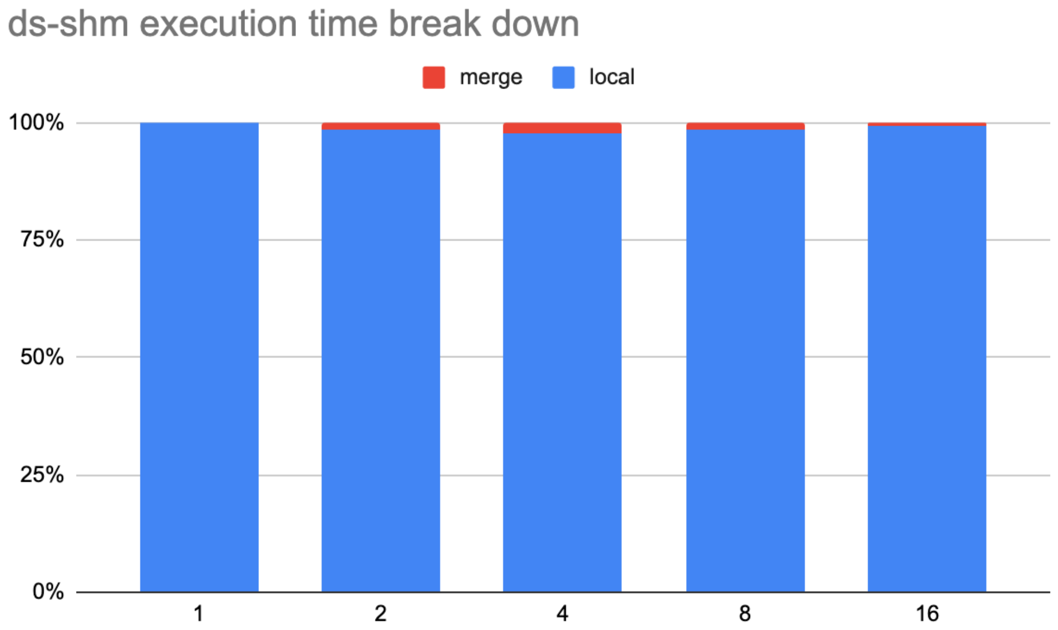
Cache Misses



Workload Distribution



Execution Time Breakdown

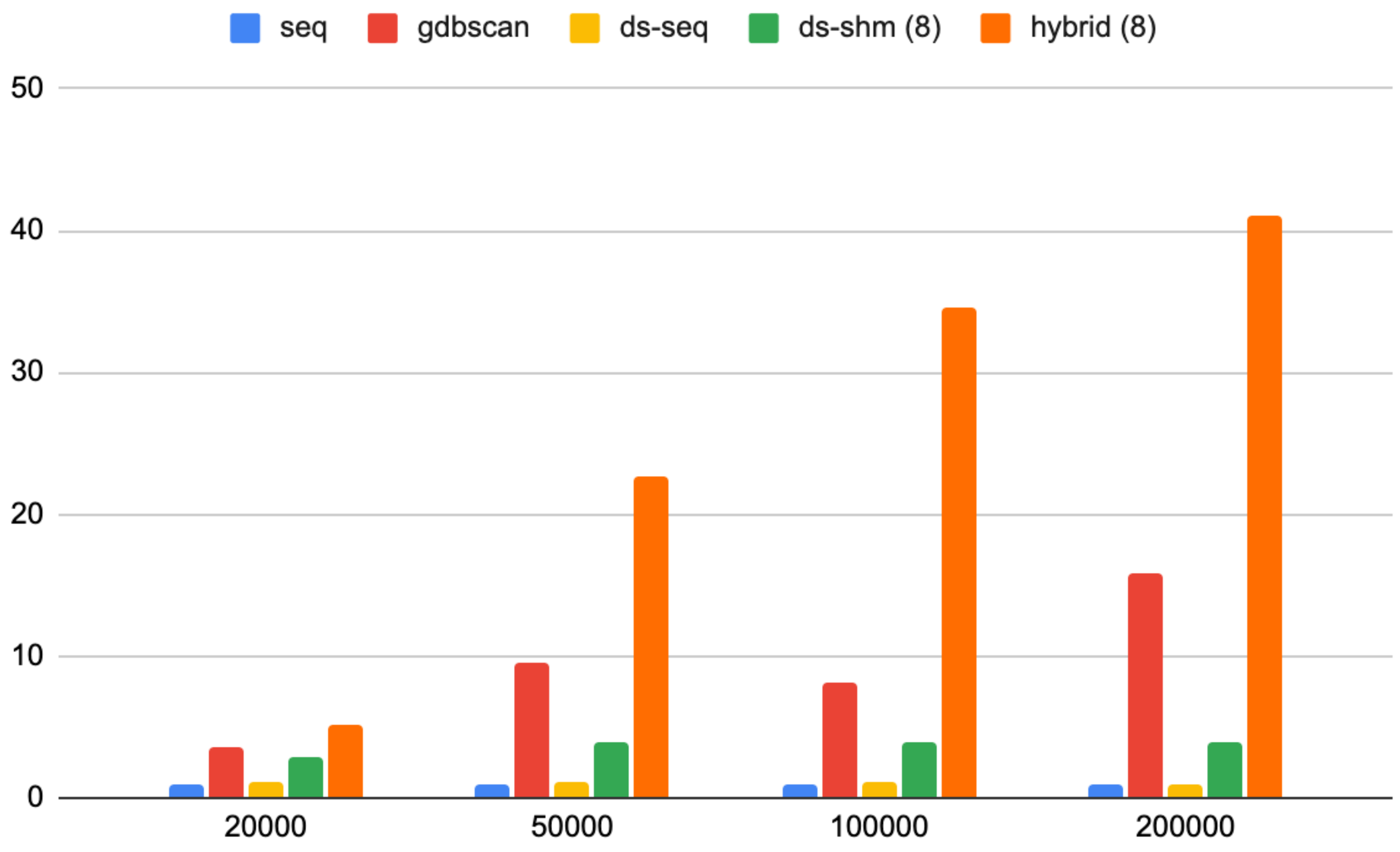


Local Computation is the bottleneck!!!

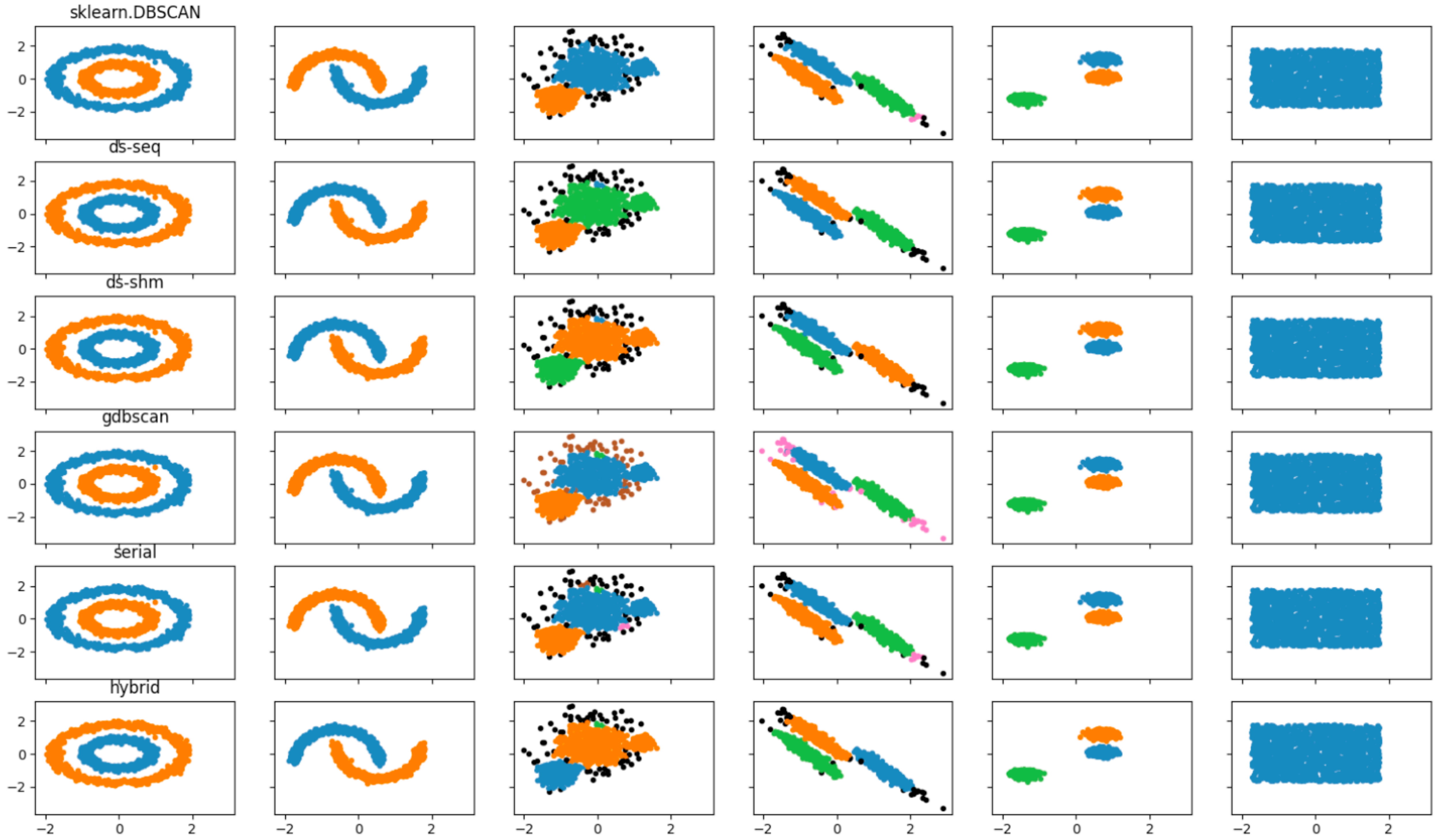
Hybrid

Key Idea: dispatch computation intensive workload to cuda,
Then leverage disjoin set for efficient merge (union)

Graph Construction (cuda) → **Disjoint Set (omp)**



Correctness Validation


sklearn.DBSCAN
ds-seq
ds-shm
gdbscan
serial
hybrid