PFLOCK Report

Andres Calderon

University of California, Riverside

February 7, 2020

GeoSpark GSJoin algorithm...

Algorithm 4 *GSJoin* algorithm for range join and distance join query

Data: (repartitioned) SRDD A and (repartitioned) SRDD B
Result: PairRDD in schema <Left object from A, right object from B>
/* **Step1: Zip partitions** */
1 **foreach** *partition pair from SRDD A and B with the same grid ID i* **do**
2 | Merge two partitions to a bigger partition that has two sub-partitions;
3 Return the intermediate SRDD C;
/* **Step2: Run partition-level local join** */
4 **foreach** *partition P in the C* **do**
5 | **foreach** *object O_A in the sub-partition from A* **do**
6 | | **if** *an index exists in the sub-partition from B* **then**
| | | // Filter phase
7 | | | Query the spatial index of this partition using the O_A's MBR;
| | | // Refine phase
8 | | | Check the spatial relation using real shapes of O_A and candidate objects O_Bs;
| | | /* **Step3: Remove duplicates** */
9 | | | Report <O_A, O_B> pair only if the reference point of this pair is in P;
10 | | **else**
11 | | | **foreach** *object O_B in the sub-partition from B* **do**
12 | | | | Check spatial relation between O_A and O_B;
| | | | /* **Step3: Remove duplicates** */
13 | | | | Report <O_A, O_B> pair only if the reference point of this pair is in P;
14 Generate the result PairRDD;