VOLUME 1 NUMBER 4 (July/Aug 2012)
ISSN: 2277 t1891

Fig. 2 illustrates each relation may participate in joins with more than two join attributes.

Step1: While relations R_A, R_B, R_C and R_D still have tuples do following steps.

Step2: / (š µ ‰ o š] ¦ Z]R_A, ŒE, ŒR_C],ÀR_D }) in-input buffer, then move tuples from input buffer to MINER process space.

Step3: Apply join operation for set of matching tuples found using Indexes.

Step4: When Used Memory exceeds Threshold limit then flush those unprocessed tuples that reside in MINER process space for maximum time to disk such that Used Memory must come into below Threshold limit.

Step5: If transmission of R_A, RB, RC and R_D is blocked more than wait threshold then

5a. Bring tuples that have been flushed to disk due to memory constraints to MINER process space.

5b. Apply join operation for set of matching tuples found using Indexes.

5c. If maximum numbers of input tuples are collected in input buffer size is satisfied then repeat steps 2 to 4.

Once all relations have been received in their entirely repeat steps 2 to 4.

There exist multiple join attributes per relation, hence algorithm maintains for each input relation a separate index on each join attribute. In this algorithm hash-table data structure is utilized as indexes. In Step 2 Algorithm is said to be in First Stage, in which tuples will arrive to input buffer. When a new tuple comes, it is • š } Œ v] v Æ] v š Z D] v Œ ‰ Œ } • • • ‰ } ( ] š • Œ o v 3zDá gAn*f*eÞð ( ] D

VOLUME 1 NUMBER 4 (July/Aug 2012)
ISSN: 2277 t1891