FOUR TYPES OF DATA SKEW AND THEIR EFFECT ON PARALLEL JOIN PERFORMANCE*

Christopher B. Walton

Department of Computer Sciences
The University of Texas at Austin
Austin, Texas 78712-1188

TR-90-12 May 1990

ABSTRACT

Recent work on parallel joins and data skew has concentrated on algorithm design without considering the causes and characteristics of data skew itself. This paper presents a simple analytic model of data skew and identifies four distinct types: tuple population skew, selectivity skew, hash partition skew and join probability skew. To demonstrate the model, a representative algorithm, the GRACE parallel join algorithm, is analyzed. Results of the analysis indicate that skew effects are substantial, and that they vary greatly with the type of skew. Also, skew effects vary substantially with system and data characteristics such as communications speed, cardinality and selectivity.

Keywords: skew, parallelism, joins, performance, database, relative partition model, GRACE algorithm.

---

[*] A version of this report has been submitted to the 1991 International Conference on Data Engineering.

1 Introduction

In recent years, parallel joins have been popular topics for research. Specific algorithms have been proposed in [10, 7, 2, 1, 4, 20]; while more general analyses and comparisons are presented in [16, 3, 11, 9, 12, 17].

The phenomenon of data skew is well-documented[13, 18], and recent reports such as [20, 19, 14] have proposed parallel join algorithms designed to perform well in the presence of skew. However, there has been little effort to characterize data skew in more detail, or to examine what steps of an algorithm are affected by it. This paper identifies four distinct types of skew, and uses a simple analytic model to demonstrate how the effects of data skew differ between skew types. As an illustration of the model and associated methodology, the GRACE[10] algorithm is examined in detail. It is selected because it is simple, efficient, well-known and because a parallel implementation exits[17].

This paper is organized as follows. The first section covers several preliminary topics: section 1.1 describes the model of data skew. Section 1.2 covers goals and assumptions in more detail. Section 1.3 describes the method used to estimate algorithm performance. Section 1.4 explains the format in which results are presented.

Section 2 describes each type of data skew and provides some examples. The GRACE algorithm is described in section 3, while analysis and conclusions are presented in 4.

1.1 The "relative partition" Model of data skew

This section reviews the relative partition model. A more detailed exposition may be found in [18]. Before proceeding, it is useful to define some terminology. A *granule* is taken to be a generic data unit, whose exact interpretation will vary with context. A *partition* is the portion of a relation assigned to a node in multicomputer system.

The relative partition model assumes that one partition has Q times more data than the others, which are all of equal size. The prefix "Q-" denotes the node or granule with the excess data.

The skew parameter Q is defined in terms of the mean partition size. That is, let

$$k_{max} = QK \quad k_i = K\frac{N-Q}{N-1} \tag{1}$$

Where $1 < Q < N$, and K is the mean partition size. k_{max} is the size of the Q-partition, while k_i is the size of any other partition.

While the relative partition model may seem to be an over-simplification, it is actually a reasonable approximation. This is due to the nature of join algorithms: there are multiple phases with barrier synchronization between phases. In many cases there is one data granule per node. Even in cases where each node processes multiple granules between synchronizations, it is a reasonable

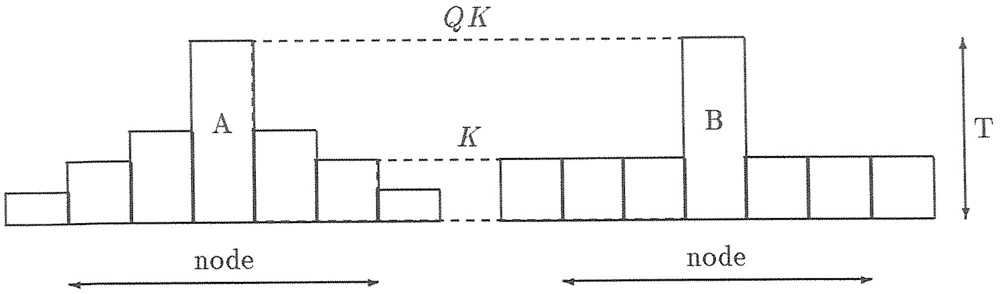

Figure 1: Actual data distribution (A) vs. relative partition model (B)

worst case assumption that the larger granules are collected on a single node. In both cases, the relative partition model is still applicable.

In a computation of this sort, the response time for each phase is determined by the Q-node – the one with the largest workload. Processing time for the other granules has little effect on the response time for each phase, since all nodes must wait for the Q-node to finish.

More formally, the response time T for a phase is $T = f(K, Q)$, where f is a (possibly nonlinear) function of Q, the skew factor, and K, the per-node cardinality. Increasing either variable will increase the response time. The skew factor Q expresses the extent to which processing time for the largest granule exceeds that of the average granule.

The relative partition model exploits this characteristic by focusing on the largest granule and treating all others as of uniform size. Figure 1 shows this simplification. Again, the model is accurate as long as it captures the size of the largest granule.

1.2 Goals and Assumptions

The goal of this report is to use the relative partition model to examine the direct effects of various types of data skew, as compared to a no-skew base case. Several assumptions are made to facilitate such an analysis:

- Join algorithms may be characterized as multiple phase computations with barrier synchronization between phases. This is a prerequisite for use of the relative partition model.
- The Q-node is the same for both relations
- Only one type of skew occurs in each test case. That it, each type of skew will be evaluated in isolation. Future work will consider the more realistic case where several types of skew occur simultaneously.

- There is no overlap between CPU, disk and communications. This is similar to the assumption made in [6]. This assumption simplifies analysis and greatly reduces the number of scenarios that must be examined.
- There is sufficient main memory that even the largest partitions do not overflow main memory. This condition isolates the *direct*, quantitative effects of data skew from indirect, qualitative effects, such as additional input/output required to deal with memory overflow.
- Truncation, rounding, and fragmentation effects are ignored.
- Algorithms do not include bit or semi-join filtering.

1.3 Method of Analysis

The assumption of no overlap between CPU, disk, and communication is justified because it greatly simplifies analysis and does not detract from the goal of this paper. That goal is evaluation of the *relative* increase in response time due to data skew. Thus, the relevant metric is the *ratio* between response times for the skew and non skew cases. The absolute accuracy of response time predictions is a secondary consideration.

Because disk, CPU and communications are assumed not to overlap, the detailed sequence and possible concurrency of operations need not be considered. For the purpose of computing the response time, all operations are commutative, so they can be grouped for computational convenience. For example, all the costs of storing all join buckets to disk can be obtained in a single calculation, rather than calculating the results for each bucket separately.

Section 3 analyzes the GRACE algorithm in detail. The following procedure will be used:

1. decompose the algorithm into steps.
2. compute the response time of each step for the uniform case
3. for each type of data skew
   (a) identify the steps that are affected by skew
   (b) calculate the response time for those steps
   (c) calculate skew multiplier (the ratio between response time for skew and no-skew cases).
   (d) use the uniform case for any steps unaffected by skew.

1.4 Skew Multiplier Tables

The response time for each step can be decomposed into three factors:

$$r_{step} = N_{op} t_{op} M_{skew}$$

where N_{op} is the operation count (tuples, pages, messages, etc.), t_{op} is the operation time, and M_{skew} is a *skew multiplier.*

This decomposition reflects the three major influences on response time: the algorithm and relation sizes (N_{op}); hardware and software performance characteristics (t_{op}); and skew effects (M_{skew}). If there are no skew effects for a given step, then $M_{skew} = 1$. For the uniform case, the response time is the product of N_{op} and t_{op}. The response time for an algorithm is simply the sum of the response times for each of its steps

Operation counts are derived from straightforward (time complexity) analysis of each step. Operation times are a characteristic of the architecture on which an algorithm is implemented. Skew multipliers are obtained by calculating the response time for the skew and non skew cases and taking their ratio. That is:

$$M_{skew} = \frac{r_{skew}}{r_{uni}}$$

where r_{skew} is the response time for the skew case, and r_{uni} is the response time for the uniform case.

This decomposition permits expressions for response time to be presented in *skew multiplier tables* (see, for example, table 1). The leftmost column gives values of N_{op}, the rightmost column gives t_{op},and the middle columns give skew multipliers. For uniform cases, the response time is the product of the right and left columns; for skew cases, a skew multiplier is selected from one of the middle columns.

2 Types of Data Skew

Skew is not a homogeneous phenomenon: a variety of effects can cause non-uniform data distribution. For the purposes of this paper, the effects are grouped into four skew types: tuple population skew, selectivity skew, hash partition skew, and join probability skew. In the following discussion, a partition means the part of a (input or output) relation assigned to a node. If hashing is used, a partition may contain more than one hash bucket.

2.1 Tuple Population Skew (TPS)

The initial distribution of tuples varies between partitions. For example, tuples may be partitioned by clustering attribute, in user specified ranges. Even if the populations are initially the same, differing rates of insertions and deletions may unbalance them over time. A mathematical model of this process is described in [8]. Q times as many tuples are stored on the Q-node as on other nodes.

2.2 Selectivity Skew (SS)

This occurs when the selectivity of local selection and projection predicates varies between nodes. A selection predicate that includes a range selection on the partitioning attribute is an obvious example. Tuples on the Q-node are Q times more likely to be selected during local selection.

2.3 Hash Partition Skew (HPS)

Hash partition skew occurs when there is a mismatch between the distribution of join key values in a relation and the distribution expected by the hash function. Hash partition skew has two different causes:

join key skew occurs when join key values are not uniformly distributed among tuples. For example: join key values follow a normal distribution. Join key skew is a property of data. When it is present, hash bucket sizes will vary even if there is no hash function skew.

hash function skew is a property of the hash function and not the data. It occurs when the number of possible hash key values mapped to each hash bucket varies. That is, the range of the hash function (bucket numbers) is less uniformly distributed than the domain (hash key values) of the hash function. In terms of the relative partition model, the Q-node holds Q times more joinable tuples than other nodes.

2.4 Join Probability Skew (JPS)

Join Probability Skew occurs when size of the join product on each partition differs, although the size of the join inputs on all partitions are uniform. It is a property of a *pair* of relations. As such, it is not manifested until the relations are joined. For this reason, the skew factor Q_{RS} has two subscripts. The size of the join output on the Q-node is Q_{RS} larger than that on other nodes.

2.5 Examples of Data Skew

Some examples will help clarify the nature of HPS and JPS. Let x be a key value; for these examples x has a domain of $\{1 \cdots 8\}$. Let $\mathcal{H}(x)$ be a hash function. Tuples are represented by ordered pairs, consisting of a key value and a letter representing the rest of the tuple.

1. **Hash Function Skew**

   Let $\mathcal{H}(x) = (x \textbf{ div } 6)$

   Let $R = \{(1, A), (2, B), (3, C), (4, D), (5, E), (6, F), (7, G), (8, H)\}$

   The partitions become

   $R_0 = \{1, 2, 3, 4, 5\}$ and $R_1 = \{6, 7, 8\}$

   In this case, hash partition skew occurs because the hash function does not evenly divide the space of possible hash keys. More specifically, Q_R is $5/4$.

2. **Join Key Skew**

   Let

   $R = \{(1, A), (2, B), (3, C), (4, D), (4, E)\}$ and

   $S = \{(1, V), (2, W), (3, X), (4, Y), (4, Z)\}$

   The hash function is $\mathcal{H}(x) = (x - 1) \textbf{ div } 2$

   Then, the partitions are:

   $R_0 = \{(1, A), (2, B)\}$ $R_1 = \{(3, C), (4, D), (4, E)\}$

   $S_0 = \{(1, V), (2, W)\}$ $S_1 = \{(3, X), (4, Y), (4, Z)\}$

   The join output for each partition is

   $R_0 \bowtie S_0 = \{(1, A, V), (2, B, W)\}$

   $R_1 \bowtie S_1 = \{(3, C, X), (4, D, Y), (4, D, Z), (4, E, Y), (4, E, Z)\}$

   Here we see that the hash function evenly divides the space of hash keys, but that the number of tuples in each bucket is still different, since some hash keys occur more frequently than others. Both Q_S and Q_R have a value of $8/5$ in this example

3. **Join Probability Skew**

   Let the hash function be the same as the previous example

   Let $R = \{(1, A), (3, C), (5, E), (6, F)\}$ and

   $S = \{(1, V), (2, W), (5, Y), (6, Z)\}$

   The partitions are:

   $R_0 = \{(1, A), (3, C)\}$ $R_1 = \{(5, E), (6, F)\}$ and

   $S_0 = \{(1, V), (2, W)\}$ $S_1 = \{(5, Y), (6, Z)\}$

   Then, the join output for each partition is

   $R_0 \bowtie S_0 = \{(1, A, V)\}$

$R_1 \bowtie S_1 = \{(5, E, Y), (6, F, Z)\}$

In this case, the number of tuples is the same in both partitions, and each join key occurs with equal frequency. That is, hash partition skew is not present. But the join outputs are of different size because of the way tuples in the two partitions interact. In this example Q_{RS} is 4/3.

2.6 Normalization

In order to accurately compare the effects of uniform and non-uniform *distribution* of data, the total number of tuples in the system at each stage of the algorithm must be the same in both skewed and uniform cases. Therefore, in performing an analytic comparison, the selectivity of local selection and projection predicates (hereafter called local selectivity) must be adjusted so that local selection eliminates the same number of tuples in both cases. Similarly, the number of join output tuples must be consistent between cases.

local selectivity normalization Let α^{net} be the net selectivity (for either relation), while α^{eff} is the effective selectivity at nodes besides the Q-node. The number of tuples at each node is K. (e.g: $K_S \equiv ||S||/N$). The total number of tuples remaining after filtering is

$$N K_S \alpha^{net} \tag{2}$$

For the uniform case, and:

$$Q_S K_S \alpha^{eff} + (N-1) K_S \alpha^{eff} \tag{3}$$

For the skew case. Setting equations 2 and 3 equal and solving for α^{eff} yields the following result:

$$\alpha^{eff} = \frac{N}{Q+N-1} \alpha^{net} \tag{4}$$

Note that in the no-skew case, $Q = 1$, and the above expression reduces to $\alpha^{eff} = \alpha^{net}$.

join selectivity normalization ρ^{eff} expresses the join "selectivity" of two typical granules (those not on the Q-node). The relationship between ρ^{net} and ρ^{eff} can be calculated from the definition of ρ^{net}:

$$\rho^{net} \equiv \frac{||S \bowtie R||}{||S||\ ||R||} \tag{5}$$

Let q denote the skew factor: $Q_S Q_R$ for hash partition skew, and Q_{RS} for join probability skew.

The size of the join on the Q-node is given by:

$$B\frac{K_R}{B}\frac{K_S}{B}q\rho^{eff}_{BQ} \tag{6}$$

Note that $K_S \equiv \|S\|/N$ and $K_R \equiv \|R\|/N$. ρ^{eff}_{BQ} is the effective join selectivity for the most general case, where data skew is present and there are multiple bucket (fragments) on the Q-node.

On the other nodes, the join product size is:

$$B\frac{K_S}{B}\frac{K_R}{B}\rho^{eff}_{BQ} \tag{7}$$

Applying the results in equations 6 and 7, as well as the definition of K_R and K_S, to the expression for ρ^{net} in equation 5 yields:

$$\rho^{net} = \frac{\frac{K_S K_R}{B}q\rho^{eff}_{BQ} + (N-1)\frac{K_S K_R}{B}\rho^{eff}_{BQ}}{N K_S N K_R} \tag{8}$$

Which may be rearranged to:

$$\rho^{net} = \frac{N+q-1}{BN^2}\rho^{eff}_{BQ} \tag{9}$$

Equation 9 implies that:

$$\rho^{eff}_{BQ} = \frac{BN^2}{N+q-1}\rho^{net} \tag{10}$$

From the above expression, it is apparent that ρ^{eff} depends not only on ρ^{net}, but on N, the number of nodes; B,the number of buckets; and the skew factor(s).

Two simplifications of equation 10 are of special interest: the no-skew case, where $q = 1$, and the single bucket case, where $B = 1$. Thus, there are four versions of the effective selectivity.

ρ^{eff}_{BQ}	multiple buckets and skew	$B > 1$	$q > 1$
ρ^{eff}_{1Q}	single bucket and skew	$B = 1$	$q > 1$
ρ^{eff}_{BU}	multiple buckets, no skew	$B > 1$	$q = 1$
ρ^{eff}_{1U}	single bucket, no skew	$B = 1$	$q = 1$

3 The GRACE Parallel Join Algorithm

3.1 Overview

The GRACE algorithm was proposed by Kitsuregawa[10] in 1983. Another useful description may be found in [6]. Schneider and DeWitt[17] report the implementation of a parallel version of the algorithm on the GAMMA database machine [5, 7]. The performance estimates in this report are based on the GAMMA implementation of the GRACE algorithm.

The GRACE algorithm has three distinct phases.

1. Form buckets for smaller relation (S-relation)
    (a) Retrieve S tuples from disk
    (b) Filter tuples
    (c) hash tuples on the join key
    (d) send tuples to join bucket sites
    (e) receive tuples at bucket sites
    (f) store tuples for buckets 0 to $B - 1$.
2. Repeat the above procedure for the larger (R) relation.
3. Join each bucket according to the following procedure:
    (a) retrieve S-tuples
    (b) build a hash table
    (c) retrieve R-tuples
    (d) probe hash table with R tuples
    (e) create join output tuples
    (f) store join results on disk

The remainder of this section examines each step of the GRACE algorithm in detail. Table 3 summarizes the notation used here.

3.2 Uniform Case

For the uniform case, the following assumptions apply:

- each node stores an equal number of tuples.
- join key values are distributed uniformly throughout the relations R and S.
- All hash buckets will be of the same size.

step 1a: retrieve tuples The time to fetch tuples from disk is the product of the number of tuples and their length, divided by the size of a disk page:

$$\frac{KL}{D} t_{disk} \tag{11}$$

step 1b: filter tuples This is the number of tuples multiplied by the cost to filter a single tuple.

$$K t_{filter} \tag{12}$$

step 1c: hash to join bucket sites The number of tuples remaining after the filtering stage is $K\alpha^{net}$, and t_{hash} is the cost of hashing one tuple.

$$K\alpha^{net} t_{hash} \tag{13}$$

step 1d: send tuples The time required to move tuples to the sites of their join buckets is:

$$\frac{N-1}{N} \frac{K\alpha^{net} L}{m} t_{send} \tag{14}$$

Where m is the message size. The $(N-1)/N$ term reflects the fact that $1/N$ of the tuples are already stored at the proper node.

step 1e: receive tuples Since the message traffic is symmetrical, the same expression holds for the number of tuples received at each node.

$$\frac{N-1}{N} \frac{K\alpha^{net} L}{m} t_{recv} \tag{15}$$

step 1f: save tuples The GRACE algorithm writes all buckets to disk.

$$\frac{K\alpha^{net} L}{D} t_{disk} \tag{16}$$

Note that above calculations describe both phase 1 (hash partition of S-relation) and phase 2 (hash partition of R-relation).

The join phase begins by reading S-tuples and building a hash table with them. For all buckets, this requires:

step 3a: retrieve S-tuples

$$\frac{K_S \alpha_S^{net} L_S}{D} t_{disk} \tag{17}$$

step 3b: hash S-tuples

$$K_S \alpha_S^{net} t_{hash} \tag{18}$$

step 3c: retrieve R-tuples Next, R tuples are read and used to probe the hash tables constructed in step 3b.

$$\frac{K_R \alpha_R^{net} L_R}{D} t_{disk} \tag{19}$$

step 3d: probe R-tuples

$$K_R \alpha_R^{net} t_{probe} \tag{20}$$

Finally, matching tuples are joined and written to disk.

step 3e: join buckets

$$K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BU}^{eff} t_{join} \tag{21}$$

step 3f: save join tuples

$$K_R K_S \alpha_S^{net} \alpha_R^{net} \rho_{BU}^{eff} \frac{L_{join}}{D} t_{disk} \tag{22}$$

Where $L_{join} \equiv L_S + L_R - L_{key}$.

3.3 Tuple Population Skew

Tuple population skew occurs when a disproportionate number of tuples are stored on one node (the Q-node). However, join keys values are uniformly distributed and there is no hash function skew, so all hash buckets are of uniform size.

Because there are QK tuples stored on the Q-node, the response time for retrieval, filtering, and hashing, and transmitting data (steps 1a to 1d) are all Q times greater than the uniform case. That is, the skew multiplier for these steps is Q.

step 1e: receive tuples Since the hashing and redistribution step disperses extra tuples from the Q-node to other nodes, the other nodes receive more tuples than the Q-node.

The number of tuples sent by the Q-node to each of the other $N-1$ nodes is:

$$\frac{1}{N-1} \frac{N-1}{N} QK\alpha^{net} = \frac{QK\alpha^{net}}{N} \tag{23}$$

The number of tuples sent by any other node to the Q-node (or any other node) is:

$$\frac{1}{N-1}\frac{N-1}{N}\frac{N-Q}{N-1}K\alpha^{net} = \frac{(N-Q)}{N(N-1)}K\alpha^{net} \tag{24}$$

Thus, the total number of tuples received by a non-Q node is:

$$\frac{QK\alpha^{net}}{N} + (N-2)\frac{N-Q}{N(N-1)}K\alpha^{net} = \frac{N^2-2N+Q}{N(N-1)}K\alpha^{net} \tag{25}$$

The skew multiplier is obtained by taking the ratio of the number of tuples in the skew case (equation 25) and the uniform case (equation 15):

$$M_{skew} = \frac{\frac{N^2-2N+Q}{N(N-1)}}{\frac{N-1}{N}} = 1 + \frac{Q-1}{(N-1)^2} \tag{26}$$

3.4 Selectivity Skew

Selectivity Skew is similar to tuple population skew, but it is caused by differences between nodes in the selectivity of local selection and projection. The time required to retrieve and filter tuples is identical to the uniform case.

step 1c: hash tuples The filtering step will yield $QK\alpha^{eff}$ tuples on the Q-node and $K\alpha^{eff}$ tuples on the others. Thus, the response time for the hash step is:

$$QK\alpha^{eff}t_{hash} \tag{27}$$

The skew multiplier is:

$$M_{skew} = \frac{QK\alpha^{eff}t_{hash}}{K\alpha^{net}t_{hash}} = \frac{NQ}{Q+N-1} \tag{28}$$

step 1d: send S-tuples Since the Q-node has more tuples than the others, it determines the response time for the transmission step:

$$\frac{N-1}{N}\frac{QK\alpha^{eff}L}{m}t_{send} \tag{29}$$

The skew multiplier is the ratio of equations 29 and 14:

$$\frac{Q\frac{N-1}{N}\frac{K\alpha^{eff}L}{m}t_{send}}{\frac{N-1}{N}\frac{K\alpha^{net}L}{m}t_{send}} = Q\frac{\alpha^{eff}}{\alpha^{net}} = \frac{QN}{Q+N-1} \tag{30}$$

step 1e: receive tuples Note that the Q-nodes needs more time to send data, but the other nodes need more time to receive it. The number of tuples sent from the Q-node to each other node is:

$$\frac{1}{N-1}\frac{N-1}{N}\frac{QK\alpha^{eff}L}{m}t_{send} = \frac{Q}{N}K\alpha^{eff} \tag{31}$$

While the tuples sent by any non-Q node to each of the remaining nodes is given by:

$$\frac{1}{N-1}\frac{N-1}{N}K\alpha^{eff} = \frac{1}{N}K\alpha^{eff} \tag{32}$$

Thus, the number of tuples received at a non-Q node is

$$\frac{QK\alpha^{eff}}{N} + (N-2)\frac{1}{N}K\alpha^{eff} = \frac{Q+N-2}{N}K\alpha^{eff} \tag{33}$$

The skew multiplier is found by dividing the number of tuples in equation 33 by the number of tuples in the uniform case (from equation 15)

$$M_{skew} = \frac{\frac{Q+N-2}{N}K\alpha^{eff}}{\frac{N-1}{N}K\alpha^{net}} = \frac{N}{N-1}\frac{Q+N-2}{Q+N-1} \tag{34}$$

After the tuple redistribution stage, tuples are uniformly distributed across all nodes. Thus, the time required to store tuples after hashing and redistribution is identical to the uniform case.

3.5 Hash Partition Skew

The time to retrieve, filter and hash S-tuples is the same as the uniform case. In contrast to tuple population skew and selectivity skew, Hash Partition Skew concentrates tuples at the Q-node. The response time for the transmission step is the same as the uniform case: the same number of tuples are sent, but a greater portion of them are routed to the Q-node. The skew multiplier for the receive step is Q, since there are Q times as many tuples sent to the Q-node.

As for the join phase, since there are Q_S times as many S-tuples on the Q-node, the skew multiplier for steps 3a and 3b (retrieve and hash) is Q_S. Similarly, the skew multiplier for steps 3c and 3d (retrieve and probe) is Q_R.

step 3e: join buckets

$$Q_S Q_R K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff} t_{join} \tag{35}$$

step 3f: save join tuples

$$Q_S Q_R K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff} \frac{L_S + L_R - L_{key}}{D} t_{disk} \tag{36}$$

The number of tuples in the last two steps is $Q_S Q_R K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff}$, compared to $K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BU}^{eff}$ tuples for the uniform case. Thus, the skew multiplier is:

$$M_{skew} = \frac{Q_S Q_R K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff}}{K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BU}^{eff}} = Q_S Q_R \frac{\rho_{BQ}^{eff}}{\rho_{BU}^{eff}} = \frac{N Q_S Q_R}{N + Q_S Q_R - 1} \tag{37}$$

3.6 Join Probability Skew

Since join probability skew only affects the size of the join product, not the join inputs, costs for all but the last two steps are the same as the uniform case.

step 3e: building join tuples

$$Q_{RS} K_R K_S \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff} t_{join} \tag{38}$$

step 3f: saving join tuples

$$Q_{RS} K_R K_S \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff} \frac{L_S + L_R - L_{key}}{D} t_{disk} \tag{39}$$

For both step 3e and step 3f the skew multiplier is:

$$M_{skew} = \frac{Q_{RS} K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BQ}^{eff}}{K_S K_R \alpha_S^{net} \alpha_R^{net} \rho_{BU}^{eff}} = Q_{RS} \frac{\rho_{BQ}^{eff}}{\rho_{BU}^{eff}} = \frac{N Q_{RS}}{N + Q_{RS} - 1} \tag{40}$$

3.7 Summary

Skew multipliers for the hash partition phase are summarized in table 1. Skew multipliers for the join phase are given in table 2.

4 Analysis and Conclusions

4.1 Selection of Parameter Values

Calculations were performed to evaluate how system size, relation size, local selectivity, join selectivity, and message processing time modulated the effects of data skew. Specifically, the following cases were considered:

step number	operation count	skew multiplier				operation time
		TPS	SS	JPS	HPS	
1a	$\frac{KL}{D}$	Q	1	1	1	t_{disk}
1b	K	Q	1	1	1	t_{filter}
1c	$K\alpha^{net}$	Q	$\frac{NQ}{Q+N-1}$	1	1	t_{hash}
1d	$\frac{N-1}{N}\frac{K\alpha^{net}L}{m}$	Q	$\frac{NQ}{Q+N-1}$	1	1	t_{send}
1e	$\frac{N-1}{N}\frac{K\alpha^{net}L}{m}$	$1+\frac{Q-1}{(N-1)^2}$	$\frac{N}{N-1}\frac{Q+N-2}{Q+N-1}$	1	Q	t_{recv}
1f	$\frac{K\alpha^{net}L}{D}$	1	1	1	Q	t_{disk}

Table 1: Response Times and Skew Multipliers for Hash Partition Phase

step number	operation count	skew multiplier				operation time
		TPS	SS	JPS	HPS	
$3a$	$\frac{K_S\alpha_S^{net}L_S}{D}$	1	1	1	Q_S	t_{disk}
$3c$	$\frac{K_R\alpha_R^{net}L_R}{D}$	1	1	1	Q_R	t_{disk}
$3b$	$K_S\alpha_S^{net}$	1	1	1	Q_S	t_{hash}
$3d$	$K_R\alpha_R^{net}$	1	1	1	Q_R	t_{probe}
$3e$	$K_SK_R\alpha_S^{net}\alpha_R^{net}N\rho^{net}$	1	1	$\frac{NQ_{RS}}{N+Q_{RS}-1}$	$\frac{NQ_SQ_R}{N+Q_SQ_R-1}$	t_{join}
$3f$	$K_SK_R\alpha_S^{net}\alpha_R^{net}N\rho^{net}\frac{L_{join}}{D}$	1	1	$\frac{NQ_{RS}}{N+Q_{RS}-1}$	$\frac{NQ_SQ_R}{N+Q_SQ_R-1}$	t_{disk}

Table 2: Response Times for Join Phase of GRACE Algorithm

- **system size:** Small (8 nodes), medium-sized (64 nodes), and large (1024 nodes) systems were considered.
- **relation size:** Three relation sizes were examined: 10^5, 10^6, 10^7.
- **local selectivity:** Two values of α^{net} were considered: 1.0, meaning no tuples were eliminated by local selection, and 0.1, corresponding to elimination of 90% of tuples.
- **join selectivity:** two values were used: 10^{-7} and 10^{-8}.
- **message time:** The message time includes the operating system overhead required for a node to process a message. The message time values assume a maximum message size of 256 bytes; if substantially longer messages were used, T_{msg} would have to be adjusted accordingly.

  $T_{msg} = 0.2$ms represents computer systems such as the as the Symult with wormhole routing and specialized communications coprocessors. $T_{msg} = 2.0$ms is appropriate for architectures such as the early Hypercubes, which employ store and forward routing.
- **skew parameter** Preliminary analysis of actual data files reported in [19, 15] indicate Q values that range from 2 to 3. The calculations presented here vary the skew parameter from 1 to 3.

Table 3 lists all parameter values used in this analysis. Operation times are similar to those assumed in [16].

For each combination of the above values, response times were calculated for the uniform case and all four skew types. The next section presents some representative results along with detailed analysis. Complete results are listed in appendix A

4.2 Analysis

A number of conclusions may be reached:

skew effects are linear The cost (increase in response time) of data skew is roughly proportional to the skew parameter Q. Table 4 shows results for a system with 8 nodes and 10^6 tuples, and parameter values of $\alpha^{net} = 1.0$, $\rho^{net} = 10^{-7}$ and $T_{msg} = 2.0ms$. Both the response time (in seconds) and *response ratio* (skew response time divided by uniform response time) are shown. Note that $Q = 1.0$ is the uniform case.

skew effects are significant In all cases, when $Q = 3.0$, at least one type of skew will double response time. Table 5 list some representative cases for $Q = 3.0$, $\rho^{net} = 10^{-7}$, and $T_{msg} = 0.2$ ms.

This result is particularly important since TPS can occur in several ways, while HPS occurs whenever the data distribution does not match that expected by the hash function. That is, the forms of skew must likely to occur also have the greatest impact.

parameter	description	typical values	units
N	system size	8; 64; 1024	nodes
D	disk page size	4096	bytes
M	memory size	1 M	bytes
m	message size	256	bytes
t_{disk}	time to read or write a disk page	20	msec
t_{send}	time to send a message	0.2; 2.0	msec
t_{recv}	time to receive a message	0.2; 2.0	msec
t_{hash}	time to hash a tuple	3	μsec
t_{probe}	time to probe hash table	6	μsec
t_{merge}	time to merge two tuples	20	μsec
t_{join}	time to create join output tuple	40	μsec
t_{scan}	time to scan an input tuple	6	μsec
$\|\|R\|\|$	cardinality of R	$10^5; 10^6; 10^7$	tuples
$\|\|S\|\|$	cardinality of S	$10^5; 10^6; 10^7$	tuples
L_R	length of an R-tuple	208	bytes
L_S	S-tuple length	208	bytes
L_{key}	key length	4	bytes
α^{net}	net local selectivity	0.1; 1.0	
ρ^{net}	net join selectivity	$10^{-7}; 10^{-8}$	
Q	skew factor	1.0 to 3.0	

Table 3: parameter values

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	109	1.00	109	1.00	109	1.00	109	1.00
1.5	128	1.17	112	1.02	147	1.35	113	1.04
2.0	146	1.34	114	1.05	187	1.71	118	1.08
2.5	165	1.51	116	1.07	227	2.08	124	1.14
3.0	184	1.68	119	1.09	269	2.46	131	1.20

Table 4: Effects of Increasing Data Skew 10^6 tuples 64 nodes $\alpha^{net} = 1.0$ $T_{msg} = 2.0$ ms $\rho^{net} = 10^{-7}$

skew effects vary with data characteristics Local selectivity, join selectivity, and cardinality all effect the relative magnitude of the costs for each type of skew. More specifically:

1. In table 5, note that HPS has the greatest response ratio of the four skew types when $\alpha^{net} = 1.0$, but that TPS is greatest when $\alpha^{net} = 0.1$. When $\alpha^{net} = 1.0$, every step of the algorithm is performed for all tuples retrieved from disk. Since HPS affects more steps than TPS, it follows that the skew effects will be greater for HPS. However, when $\alpha^{net} = 0.1$, 90% of the tuples are eliminated by local selection. In this situation, skew effects in TPS affect processing of all tuples, while HPS only affects processing of the 10% of tuples that remain after local selection. (HPS does not affect steps 1a and 1b).
2. In most cases, the influence of join selectivity is minor. Referring to table 6, note that for TPS, SS, and HPS, the response rations for both values of ρ^{net} are within a few percent of each other. In most cases considered here, the number of join output tuples is so small that join processing makes only a minor contribution to the total response time. This is not true when relation cardinalities are 10^7; those cases are discussed in the following paragraph.
3. Effects of join selectivity and skew in join steps (both HPS and JPS affect steps 3e and 3f) are more pronounced at large cardinalities. This follows from the fact that the number of join tuples grows quadratically with cardinality. In table 6, note that response ratios for $\rho^{net} = 10^{-7}$ are much greater than those for $\rho^{net} = 10^{-8}$. These are also the only cases where JPS has a larger response ratio than TPS.

card.	N	uniform		TPS		SS		HPS		JPS	
		response(sec)		ratio		ratio		ratio		ratio	
		$\alpha =$ 1.0	$\alpha =$ 0.1	$\alpha =$ 1.0	$\alpha =$ 0.1	$\alpha =$ 1.0	$\alpha =$ 0.1	$\alpha =$ 1.0	$\alpha =$ 0.1	$\alpha =$ 1.0	$\alpha =$ 0.1
10^5	8	84	32	1.70	2.66	1.06	1.02	2.31	1.35	1.01	1.00
	64	11	4.0	1.70	2.65	1.09	1.02	2.31	1.35	1.02	1.00
	1024	0.6	0.2	1.63	2.59	1.10	1.03	2.39	1.41	1.02	1.00
10^6	8	864	316	1.68	2.65	1.06	1.02	2.36	1.35	1.10	1.00
	64	109	40	1.68	2.65	1.09	1.02	2.46	1.35	1.20	1.01
	1024	6.8	2.5	1.68	2.65	1.09	1.03	2.49	1.36	1.23	1.01
10^7	8	10924	3183	1.54	2.64	1.05	1.02	2.81	1.37	1.81	1.03
	64	1377	399	1.54	2.64	1.07	1.02	3.61	1.40	2.61	1.06
	1024	86	25	1.54	2.64	1.07	1.03	3.82	1.41	2.83	1.06

Table 5: Effects of local selectivity when $\rho^{net} = 10^{-7}$ and $T_{msg} = 0.2$ms

card	N	uniform		TPS		SS		HPS		JPS	
		response(sec)		ratio		ratio		ratio		ratio	
		$\rho = 10^{-7}$	$\rho = 10^{-8}$	$\rho = 10^{-7}$	$\rho = 10^{-8}$	$\rho = 10^{-7}$	$\rho = 10^{-8}$	$\rho = 10^{-7}$	$\rho = 10^{-8}$	$\rho = 10^{-7}$	$\rho = 10^{-8}$
10^5	8	84	84	1.70	1.70	1.06	.06	2.31	2.30	1.01	1.00
	64	11	11	1.70	1.70	1.09	1.09	2.31	2.30	1.02	1.00
	1024	0.6	0.6	1.63	1.63	1.10	1.10	2.39	2.37	1.02	1.00
10^6	8	864	841	1.68	1.70	1.06	1.06	2.36	2.31	1.10	1.01
	64	109	106	1.68	1.70	1.09	1.09	2.46	2.31	1.20	1.02
	1024	6.8	6.6	1.68	1.70	1.09	1.10	2.49	2.32	1.23	1.02
10^7	8	10924	8638	1.54	1.68	1.05	1.06	2.81	2.36	1.81	1.10
	64	1377	1091	1.54	1.68	1.07	1.09	3.61	2.46	2.61	1.20
	1024	86	68	1.54	1.68	1.07	1.09	3.82	2.49	2.83	1.23

Table 6: Effect of join selectivity when $\alpha^{net} = 1.0$ and $T_{msg} = 0.2$ ms

communications characteristics also affect skew costs There is also an interaction between communications speed (message passing time) and data skew. The nature of this interaction varies with skew type:

JPS From table 7 it can be seen that the response ratio is always larger with faster messages. Since JPS does not affect message passing steps, the absolute increase in overall response time due to JPS does not vary with T_{msg}. However, the uniform response time – the denominator in response ratio – does increase with T_{msg}. Hence, slow communications decrease the response ratio by increasing its numerator (uniform response time)

SS From table 7, it is apparent that response ratios are greater for large values of T_{msg} (slower communications). Increasing T_{msg} increases the relative importance of steps 1d and 1e, which are affected by selectivity skew.

TPS TPS affects message passing steps. When $\alpha^{net} = 1.0$, a larger value of T_{msg} increases the proportion of the response time consumed by message passing. Hence increasing T_{msg} increases the prominence of skew effects and the response ratio increases. Thus, response ratios are larger for $T_{msg} = 2.0$.

However, when $\alpha^{net} = 0.1$, 90% of tuples are eliminated by local selection, so that the retrieval and local selection steps dominate the response time. The total response time is smaller, so that response ratios are much greater than when $\alpha^{net} = 1.0$. Skew effects in the message passing steps are too small to have an appreciable effect. The $T_{msg} = 0.2$ case has a slightly larger response ratio because the uniform response time is faster.

HPS HPS primarily affects the join phase. Response ratios are larger when $\alpha^{net} = 1.0$ because all tuples are processed by steps affected by HPS. Note that HPS affects the receive step (step 1e), but not the transmit step (step 1d). Thus, increasing T_{msg} increases the uniform response time (which includes two message steps) more than skew effects, so response ratio is slightly less for slow messages.

When $\alpha^{net} = 0.1$, the differences due to T_{msg} are smaller. A larger value of T_{msg} increases skew effects but has little impact on the uniform response time. Thus, the response ratio is larger for $T_{msg} = 2.0$ (compared to $T_{msg} = 0.2$).

			uniform	TPS		SS	HPS		JPS
card	α^{net}	T_{msg}	resp	resp	ratio	ratio	resp	ratio	ratio
10^5	1.0	0.2	11	18	1.70	1.09	24.32	2.30	1.00
		2.0	20	36	1.84	1.49	42.31	2.16	1.00
	0.1	0.2	4.0	10	2.65	1.02	5.32	1.35	1.00
		2.0	4.8	12	2.53	1.20	7.11	1.47	1.00
10^6	1.0	0.2	106	181	1.70	1.09	245.67	2.31	1.02
		2.0	196	361	1.84	1.49	425.62	2.17	1.01
	0.1	0.2	40	105	2.65	1.02	53.31	1.35	1.00
		2.0	49	123	2.53	1.20	71.30	1.47	1.00
10^7	1.0	0.2	1091	1837	1.68	1.09	2685.42	2.46	1.20
		2.0	1991	3637	1.83	1.48	4484.98	2.25	1.11
	0.1	0.2	396	1050	2.65	1.02	535.53	1.35	1.01
		2.0	486	1230	2.53	1.20	715.48	1.47	1.00

Table 7: Effects of message time with 64 nodes and $\rho^{net} = 10^{-8}$ (response times in seconds)

4.3 Conclusions

Several general conclusions can be drawn from this work:

- A simple analytic model provides considerable insight into the effects of data skew.
- Algorithms that were designed for uniform data distribution suffer considerable performance degradation from modest amounts of data skew.
- Skew is not a homogeneous phenomenon, and the effects of each type of skew differ greatly.
- Effects of skew vary with data and system characteristics.

References

[1] Chaitanya K. Baru and Ophir Frieder. Database operations in a cube-connected multicomputer system. *IEEE Transactions on Computers*, C-38(6):920–927, June 1989.

[2] Chaitanya K. Baru, Ophir Frieder, Dilip Dandlur, and Mark Segal. Join on a cube: analysis, simulation, and implementation. In M. Ketsuregawa and H. Tanaka, editors, *Database Machines and Knowledge Base Machines*, pages 61–74, Kluwer Academic Publishers, Norwell, MA, October 1987. [Karuizawa, Nagano, Japan].

[3] Haran Boral. Parallelism and data management. In *Third International Conference on Data and Knowledge Bases*, June 1988. [Jerusalem, Israel].

[4] Alfred G. Dale, F. Furman Haddix, Roy M. Jenevein, and Christopher B. Walton. *Scalability of Parallel Joins on High Performance Multicomputers.* Technical Report TR-89-17, University of Texas at Austin Department of Computer Sciences, Austin, TX, USA, June 1989.

[5] David J. DeWitt et al. Gamma - a high performance backend database machine. In *Twelth Very Large Data Base Conference*, 1986. [Kyoto, Japan, August 1986].

[6] David J. DeWitt, R. Katz, F. Olken, D. Shapiro, Michael Stonebraker, and D. Wood. Implementation techniques for main memory database systems. In *SIGMOD Conference*, June 1984. [Boston, MA, USA].

[7] Robert H. Gerber. *Dataflow Query Processing using Multiprocessor Hash-Partitioned Algorithms.* Technical Report 672, University of Wisconsin Computer Sciences Department, Madison, WI, USA, October 1986.

[8] Peter J. Haas. *Workload Imbalance and Parallel Processing.* Reserach Report RJ 6936, IBM Almaden Research Center, San Jose, CA, USA, July 1989.

[9] R.-C. Hu and R. Muntz. *Removing Skew Effect in Join Operations on Parallel Processors.* Technical Report CSD-890027, University of California at Los Angeles Computer Science Department, Los Angeles, CA, June 1989.

[10] M. Kitsuregawa, M. Nakano, and T. Moto-Oka. Application of hash to database machine and its architecture. *New Generation Computing*, 1(1), 1983.

[11] M. Seetha Lakshimi and Philip S. Yu. Effect of skew on join performances in parallel architecture. In *Symposium on Databases in Parallel and Distributed Systems*, pages 107–120, August 1988. [Austin, TX, USA].

[12] M. Seetha Lakshimi and Philip S. Yu. Limiting factors of join performance on parallel processors. In *Conference on Data Engineering*, pages 488–496, February 1989. [Los Angeles, CA, USA].

[13] C. A. Lynch. Selectivity estimation and query optimization in large databases with highly skewed distributions of column values. In *14th International Conference on Very Large Databases*, 1988.

[14] Edward Omiecinski and Eileen Tien Liu. *The Adaptive-Hash Join Algorithm for a Hypercube Multicomputer.* Technical Report GIT-ICS-89/48, School of Information and Computer Science Georgia Institute of Technology, Atlanta, GA, December 1989.

[15] Matt L. Pinsonneaut and Christopher B. Walton. *Observations of Data Skew in Two Databases.* Technical Report (in prepartion), Department of Computer Sciences, University of Texas at Austin, May 1990.

[16] James P. Richardson, Hongjun Lu, and Krishna Mikkilineni. Design and evaluation of parallel pipelined join algorithms. In *ACM SIGMOD Conference*, pages 399–409, May 1987. [San Francisco, CA, USA,].

[17] Donovan A. Schneider and David J. DeWitt. *A Performance Evaluation of Four Parallel Join Algorithms in a Shared-Nothing Multiprocessor Environment.* Technical Report 836, University of Wisconsin Computer Sciences Department, Madison, WI, USA, April 1989.

[18] Christopher B. Walton. *Investigating Skew and Scalability in Parallel Joins.* Technical Report TR-89-39, Department of Computer Sciences, University of Texas at Austin, December 1989.

[19] Joel L. Wolf, Daniel M. Dias, and Philip S. Yu. *An Effective Algorithm for Parallelizing Hash Joins in the Presence of Data Skew.* Research Report RC 15510, IBM T.J. Watson Research Center, Yorktown Heights, NY, USA, February 1990.

[20] Joel L. Wolf, Daniel M. Dias, and Philip S. Yu. *An Effective Algorithm for Parallelizing Sort Merge Joins in the Presence of Data Skew.* Research Report RC 15138, IBM T.J. Watson Research Center, Yorktown Heights, NY, USA, November 1989.

A Complete Results of Calculations

The following tables list results for all combinations of the parameter values discussed in section 4.1.

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	84.1	1.00	84.1	1.00	84.1	1.00	84.1	1.00
1.5	98.8	1.18	85.6	1.02	111.5	1.33	84.3	1.00
2.0	113.5	1.35	86.9	1.03	139.0	1.65	84.6	1.01
2.5	128.2	1.53	88.2	1.05	166.4	1.98	84.8	1.01
3.0	142.9	1.70	89.3	1.06	193.9	2.31	85.0	1.01

Table 8: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	83.8	1.00	83.8	1.00	83.8	1.00	83.8	1.00
1.5	98.6	1.18	85.4	1.02	111.1	1.33	83.9	1.00
2.0	113.3	1.35	86.7	1.03	138.3	1.65	83.9	1.00
2.5	128.0	1.53	87.9	1.05	165.6	1.98	83.9	1.00
3.0	142.7	1.70	89.0	1.06	192.8	2.30	83.9	1.00

Table 9: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	148.1	1.00	148.1	1.00	148.1	1.00	148.1	1.00
1.5	179.1	1.21	163.0	1.10	191.5	1.29	148.3	1.00
2.0	210.1	1.42	176.3	1.19	235.0	1.59	148.5	1.00
2.5	241.2	1.63	188.2	1.27	278.4	1.88	148.8	1.00
3.0	272.2	1.84	198.9	1.34	321.8	2.17	148.9	1.01

Table 10: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	147.8	1.00	147.8	1.00	147.8	1.00	147.8	1.00
1.5	178.9	1.21	162.8	1.10	191.1	1.29	147.8	1.00
2.0	209.9	1.42	176.1	1.19	234.3	1.59	147.9	1.00
2.5	240.9	1.63	188.0	1.27	277.6	1.88	147.9	1.00
3.0	272.0	1.84	198.7	1.34	320.8	2.17	147.9	1.00

Table 11: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	31.5	1.00	31.5	1.00	31.5	1.00	31.5	1.00
1.5	44.6	1.41	31.7	1.00	34.3	1.09	31.5	1.00
2.0	57.7	1.83	31.8	1.01	37.0	1.17	31.6	1.00
2.5	70.7	2.24	32.0	1.01	39.7	1.26	31.6	1.00
3.0	83.8	2.66	32.1	1.02	42.4	1.35	31.6	1.00

Table 12: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	31.5	1.00	31.5	1.00	31.5	1.00	31.5	1.00
1.5	44.6	1.41	31.7	1.00	34.3	1.09	31.5	1.00
2.0	57.6	1.83	31.8	1.01	37.0	1.17	31.5	1.00
2.5	70.7	2.24	32.0	1.01	39.7	1.26	31.5	1.00
3.0	83.8	2.66	32.1	1.02	42.4	1.35	31.5	1.00

Table 13: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	37.9	1.00	37.9	1.00	37.9	1.00	37.9	1.00
1.5	52.6	1.39	39.4	1.04	42.3	1.11	37.9	1.00
2.0	67.3	1.77	40.8	1.07	46.6	1.23	37.9	1.00
2.5	82.0	2.16	42.0	1.11	50.9	1.34	38.0	1.00
3.0	96.7	2.55	43.0	1.13	55.2	1.46	38.0	1.00

Table 14: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	37.9	1.00	37.9	1.00	37.9	1.00	37.9	1.00
1.5	52.6	1.39	39.4	1.04	42.3	1.11	37.9	1.00
2.0	67.3	1.77	40.8	1.07	46.6	1.23	37.9	1.00
2.5	82.0	2.16	42.0	1.11	50.9	1.34	37.9	1.00
3.0	96.7	2.55	43.0	1.13	55.2	1.46	37.9	1.00

Table 15: Effects of Q value for 10^5 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	10.6	1.00	10.6	1.00	10.6	1.00	10.6	1.00
1.5	12.5	1.18	10.9	1.02	14.1	1.33	10.6	1.00
2.0	14.3	1.35	11.1	1.05	17.6	1.66	10.7	1.01
2.5	16.2	1.53	11.3	1.07	21.0	1.98	10.8	1.01
3.0	18.0	1.70	11.6	1.09	24.5	2.31	10.8	1.02

Table 16: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	10.6	1.00	10.6	1.00	10.6	1.00	10.6	1.00
1.5	12.4	1.18	10.8	1.02	14.0	1.32	10.6	1.00
2.0	14.3	1.35	11.1	1.05	17.4	1.65	10.6	1.00
2.5	16.2	1.53	11.3	1.07	20.9	1.97	10.6	1.00
3.0	18.0	1.70	11.6	1.09	24.3	2.30	10.6	1.00

Table 17: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	19.6	1.00	19.6	1.00	19.6	1.00	19.6	1.00
1.5	23.7	1.21	22.0	1.12	25.3	1.29	19.6	1.00
2.0	27.8	1.42	24.5	1.25	31.0	1.58	19.7	1.00
2.5	31.9	1.63	26.8	1.37	36.8	1.88	19.8	1.01
3.0	36.0	1.84	29.2	1.49	42.5	2.17	19.8	1.01

Table 18: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	19.6	1.00	19.6	1.00	19.6	1.00	19.6	1.00
1.5	23.7	1.21	22.0	1.12	25.3	1.29	19.6	1.00
2.0	27.8	1.42	24.4	1.25	30.9	1.58	19.6	1.00
2.5	31.9	1.63	26.8	1.37	36.6	1.87	19.6	1.00
3.0	36.0	1.84	29.1	1.49	42.3	2.16	19.6	1.00

Table 19: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3.9	1.00	3.9	1.00	3.9	1.00	3.9	1.00
1.5	5.6	1.41	4.0	1.01	4.3	1.09	3.9	1.00
2.0	7.2	1.83	4.0	1.01	4.6	1.17	3.9	1.00
2.5	8.8	2.24	4.0	1.02	5.0	1.26	3.9	1.00
3.0	10.5	2.65	4.0	1.02	5.3	1.35	3.9	1.00

Table 20: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3.9	1.00	3.9	1.00	3.9	1.00	3.9	1.00
1.5	5.6	1.41	4.0	1.01	4.3	1.09	3.9	1.00
2.0	7.2	1.83	4.0	1.01	4.6	1.17	3.9	1.00
2.5	8.8	2.24	4.0	1.02	5.0	1.26	3.9	1.00
3.0	10.5	2.65	4.0	1.02	5.3	1.35	3.9	1.00

Table 21: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	4.8	1.00	4.8	1.00	4.8	1.00	4.8	1.00
1.5	6.7	1.38	5.1	1.05	5.4	1.12	4.8	1.00
2.0	8.6	1.77	5.3	1.10	6.0	1.23	4.8	1.00
2.5	10.4	2.15	5.6	1.15	6.5	1.35	4.8	1.00
3.0	12.3	2.53	5.8	1.20	7.1	1.47	4.8	1.00

Table 22: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	4.8	1.00	4.8	1.00	4.8	1.00	4.8	1.00
1.5	6.7	1.38	5.1	1.05	5.4	1.12	4.8	1.00
2.0	8.6	1.77	5.3	1.10	6.0	1.23	4.8	1.00
2.5	10.4	2.15	5.6	1.15	6.5	1.35	4.8	1.00
3.0	12.3	2.53	5.8	1.20	7.1	1.47	4.8	1.00

Table 23: Effects of Q value for 10^5 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	0.6	1.00	0.6	1.00	0.6	1.00	0.6	1.00
1.5	0.7	1.16	0.6	1.03	0.8	1.35	0.6	1.00
2.0	0.8	1.31	0.7	1.05	1.1	1.69	0.6	1.01
2.5	0.9	1.47	0.7	1.08	1.3	2.04	0.6	1.02
3.0	1.0	1.63	0.7	1.10	1.5	2.39	0.6	1.02

Table 24: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 1.000\ T_{msg} = 0.20\ \rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	0.6	1.00	0.6	1.00	0.6	1.00	0.6	1.00
1.5	0.7	1.16	0.6	1.03	0.8	1.34	0.6	1.00
2.0	0.8	1.31	0.7	1.05	1.0	1.69	0.6	1.00
2.5	0.9	1.47	0.7	1.08	1.3	2.03	0.6	1.00
3.0	1.0	1.63	0.7	1.10	1.5	2.37	0.6	1.00

Table 25: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 1.000\ T_{msg} = 0.20\ \rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1.2	1.00	1.2	1.00	1.2	1.00	1.2	1.00
1.5	1.4	1.20	1.3	1.13	1.5	1.30	1.2	1.00
2.0	1.7	1.40	1.5	1.26	1.9	1.60	1.2	1.00
2.5	1.9	1.60	1.7	1.40	2.3	1.90	1.2	1.01
3.0	2.1	1.80	1.8	1.53	2.6	2.21	1.2	1.01

Table 26: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 1.000\ T_{msg} = 2.00\ \rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1.2	1.00	1.2	1.00	1.2	1.00	1.2	1.00
1.5	1.4	1.20	1.3	1.13	1.5	1.30	1.2	1.00
2.0	1.7	1.40	1.5	1.26	1.9	1.60	1.2	1.00
2.5	1.9	1.60	1.7	1.40	2.3	1.90	1.2	1.00
3.0	2.1	1.81	1.8	1.53	2.6	2.20	1.2	1.00

Table 27: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 1.000\ T_{msg} = 2.00\ \rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	0.2	1.00	0.2	1.00	0.2	1.00	0.2	1.00
1.5	0.3	1.40	0.2	1.01	0.2	1.10	0.2	1.00
2.0	0.4	1.80	0.2	1.02	0.3	1.20	0.2	1.00
2.5	0.5	2.19	0.2	1.02	0.3	1.31	0.2	1.00
3.0	0.5	2.59	0.2	1.03	0.3	1.41	0.2	1.00

Table 28: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	0.2	1.00	0.2	1.00	0.2	1.00	0.2	1.00
1.5	0.3	1.40	0.2	1.01	0.2	1.10	0.2	1.00
2.0	0.4	1.80	0.2	1.02	0.3	1.20	0.2	1.00
2.5	0.5	2.19	0.2	1.02	0.3	1.31	0.2	1.00
3.0	0.5	2.59	0.2	1.03	0.3	1.41	0.2	1.00

Table 29: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	0.3	1.00	0.3	1.00	0.3	1.00	0.3	1.00
1.5	0.4	1.37	0.3	1.06	0.3	1.13	0.3	1.00
2.0	0.5	1.73	0.3	1.12	0.3	1.27	0.3	1.00
2.5	0.6	2.10	0.3	1.18	0.4	1.40	0.3	1.00
3.0	0.7	2.47	0.3	1.24	0.4	1.54	0.3	1.00

Table 30: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	0.3	1.00	0.3	1.00	0.3	1.00	0.3	1.00
1.5	0.4	1.37	0.3	1.06	0.3	1.13	0.3	1.00
2.0	0.5	1.73	0.3	1.12	0.3	1.27	0.3	1.00
2.5	0.6	2.10	0.3	1.18	0.4	1.40	0.3	1.00
3.0	0.7	2.47	0.3	1.24	0.4	1.53	0.3	1.00

Table 31: Effects of Q value for 10^5 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	863.8	1.00	863.8	1.00	863.8	1.00	863.8	1.00
1.5	1011.1	1.17	879.1	1.02	1160.1	1.34	887.8	1.03
2.0	1158.5	1.34	892.6	1.03	1456.8	1.69	912.3	1.06
2.5	1305.8	1.51	904.7	1.05	1751.0	2.03	934.2	1.08
3.0	1453.1	1.68	915.6	1.06	2041.7	2.36	952.7	1.10

Table 32: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	841.0	1.00	841.0	1.00	841.0	1.00	841.0	1.00
1.5	988.3	1.18	856.2	1.02	1115.6	1.33	843.4	1.00
2.0	1135.6	1.35	869.7	1.03	1390.3	1.65	845.8	1.01
2.5	1282.9	1.53	881.9	1.05	1664.7	1.98	848.0	1.01
3.0	1430.3	1.70	892.8	1.06	1938.8	2.31	849.8	1.01

Table 33: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1503.7	1.00	1503.7	1.00	1503.7	1.00	1503.7	1.00
1.5	1814.2	1.21	1653.3	1.10	1959.9	1.30	1527.7	1.02
2.0	2124.8	1.41	1786.4	1.19	2416.5	1.61	1552.1	1.03
2.5	2435.3	1.62	1905.4	1.27	2870.7	1.91	1574.1	1.05
3.0	2745.9	1.83	2012.5	1.34	3321.4	2.21	1592.5	1.06

Table 34: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1480.8	1.00	1480.8	1.00	1480.8	1.00	1480.8	1.00
1.5	1791.3	1.21	1630.5	1.10	1915.4	1.29	1483.2	1.00
2.0	2101.9	1.42	1763.5	1.19	2350.0	1.59	1485.6	1.00
2.5	2412.4	1.63	1882.5	1.27	2784.4	1.88	1487.8	1.00
3.0	2723.0	1.84	1989.7	1.34	3218.5	2.17	1489.7	1.01

Table 35: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	316.0	1.00	316.0	1.00	316.0	1.00	316.0	1.00
1.5	446.6	1.41	317.5	1.00	343.4	1.09	316.2	1.00
2.0	577.3	1.83	318.8	1.01	370.9	1.17	316.4	1.00
2.5	708.0	2.24	320.1	1.01	398.3	1.26	316.7	1.00
3.0	838.6	2.65	321.1	1.02	425.7	1.35	316.9	1.00

Table 36: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	315.7	1.00	315.7	1.00	315.7	1.00	315.7	1.00
1.5	446.4	1.41	317.3	1.00	343.0	1.09	315.8	1.00
2.0	577.1	1.83	318.6	1.01	370.2	1.17	315.8	1.00
2.5	707.7	2.24	319.8	1.01	397.5	1.26	315.8	1.00
3.0	838.4	2.66	320.9	1.02	424.7	1.35	315.8	1.00

Table 37: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	379.9	1.00	379.9	1.00	379.9	1.00	379.9	1.00
1.5	526.9	1.39	394.9	1.04	423.4	1.11	380.2	1.00
2.0	673.9	1.77	408.2	1.07	466.9	1.23	380.4	1.00
2.5	820.9	2.16	420.1	1.11	510.3	1.34	380.7	1.00
3.0	967.9	2.55	430.8	1.13	553.7	1.46	380.8	1.00

Table 38: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	379.7	1.00	379.7	1.00	379.7	1.00	379.7	1.00
1.5	526.7	1.39	394.7	1.04	423.0	1.11	379.7	1.00
2.0	673.7	1.77	408.0	1.07	466.2	1.23	379.8	1.00
2.5	820.7	2.16	419.9	1.11	509.4	1.34	379.8	1.00
3.0	967.7	2.55	430.6	1.13	552.7	1.46	379.8	1.00

Table 39: Effects of Q value for 10^6 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	109.1	1.00	109.1	1.00	109.1	1.00	109.1	1.00
1.5	127.7	1.17	111.6	1.02	147.2	1.35	112.9	1.04
2.0	146.4	1.34	114.0	1.05	186.6	1.71	118.0	1.08
2.5	165.0	1.51	116.4	1.07	227.2	2.08	124.2	1.14
3.0	183.6	1.68	118.8	1.09	268.5	2.46	131.3	1.20

Table 40: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	106.2	1.00	106.2	1.00	106.2	1.00	106.2	1.00
1.5	124.9	1.18	108.7	1.02	140.9	1.33	106.6	1.00
2.0	143.5	1.35	111.2	1.05	175.7	1.65	107.1	1.01
2.5	162.1	1.53	113.6	1.07	210.7	1.98	107.7	1.01
3.0	180.8	1.70	115.9	1.09	245.7	2.31	108.4	1.02

Table 41: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	199.0	1.00	199.0	1.00	199.0	1.00	199.0	1.00
1.5	240.2	1.21	223.5	1.12	259.7	1.30	202.9	1.02
2.0	281.3	1.41	247.6	1.24	321.6	1.62	208.0	1.04
2.5	322.5	1.62	271.3	1.36	384.6	1.93	214.2	1.08
3.0	363.6	1.83	294.7	1.48	448.5	2.25	221.3	1.11

Table 42: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	196.2	1.00	196.2	1.00	196.2	1.00	196.2	1.00
1.5	237.3	1.21	220.7	1.12	253.4	1.29	196.6	1.00
2.0	278.5	1.42	244.7	1.25	310.7	1.58	197.1	1.00
2.5	319.6	1.63	268.5	1.37	368.1	1.88	197.7	1.01
3.0	360.8	1.84	291.8	1.49	425.6	2.17	198.4	1.01

Table 43: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	39.6	1.00	39.6	1.00	39.6	1.00	39.6	1.00
1.5	55.9	1.41	39.8	1.01	43.1	1.09	39.6	1.00
2.0	72.3	1.83	40.1	1.01	46.5	1.18	39.7	1.00
2.5	88.6	2.24	40.3	1.02	50.0	1.26	39.7	1.00
3.0	105.0	2.65	40.6	1.02	53.5	1.35	39.8	1.01

Table 44: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	39.6	1.00	39.6	1.00	39.6	1.00	39.6	1.00
1.5	55.9	1.41	39.8	1.01	43.0	1.09	39.6	1.00
2.0	72.3	1.83	40.1	1.01	46.4	1.17	39.6	1.00
2.5	88.6	2.24	40.3	1.02	49.9	1.26	39.6	1.00
3.0	105.0	2.65	40.5	1.02	53.3	1.35	39.6	1.00

Table 45: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	48.6	1.00	48.6	1.00	48.6	1.00	48.6	1.00
1.5	67.2	1.38	51.0	1.05	54.3	1.12	48.6	1.00
2.0	85.8	1.77	53.4	1.10	60.0	1.24	48.7	1.00
2.5	104.4	2.15	55.8	1.15	65.8	1.35	48.7	1.00
3.0	123.0	2.53	58.2	1.20	71.5	1.47	48.8	1.00

Table 46: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	48.6	1.00	48.6	1.00	48.6	1.00	48.6	1.00
1.5	67.2	1.38	51.0	1.05	54.2	1.12	48.6	1.00
2.0	85.8	1.77	53.4	1.10	59.9	1.23	48.6	1.00
2.5	104.4	2.15	55.8	1.15	65.6	1.35	48.6	1.00
3.0	123.0	2.53	58.1	1.20	71.3	1.47	48.6	1.00

Table 47: Effects of Q value for 10^6 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	6.8	1.00	6.8	1.00	6.8	1.00	6.8	1.00
1.5	8.0	1.17	7.0	1.02	9.2	1.35	7.0	1.04
2.0	9.1	1.34	7.1	1.05	11.7	1.72	7.4	1.09
2.5	10.3	1.51	7.3	1.07	14.3	2.10	7.8	1.15
3.0	11.4	1.68	7.4	1.09	17.0	2.49	8.4	1.23

Table 48: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	6.6	1.00	6.6	1.00	6.6	1.00	6.6	1.00
1.5	7.8	1.17	6.8	1.02	8.8	1.33	6.6	1.00
2.0	8.9	1.35	6.9	1.05	11.0	1.66	6.7	1.01
2.5	10.1	1.52	7.1	1.07	13.2	1.99	6.7	1.02
3.0	11.2	1.70	7.3	1.10	15.4	2.32	6.8	1.02

Table 49: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	12.5	1.00	12.5	1.00	12.5	1.00	12.5	1.00
1.5	15.1	1.21	14.1	1.13	16.3	1.31	12.8	1.02
2.0	17.7	1.41	15.7	1.25	20.2	1.62	13.1	1.05
2.5	20.3	1.62	17.3	1.38	24.3	1.94	13.5	1.08
3.0	22.8	1.83	18.8	1.51	28.4	2.27	14.1	1.13

Table 50: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	12.3	1.00	12.3	1.00	12.3	1.00	12.3	1.00
1.5	14.9	1.21	13.9	1.13	15.9	1.29	12.4	1.00
2.0	17.5	1.42	15.5	1.26	19.5	1.58	12.4	1.00
2.5	20.1	1.63	17.1	1.39	23.1	1.88	12.4	1.01
3.0	22.7	1.84	18.7	1.51	26.8	2.17	12.5	1.01

Table 51: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	2.5	1.00	2.5	1.00	2.5	1.00	2.5	1.00
1.5	3.5	1.41	2.5	1.01	2.7	1.09	2.5	1.00
2.0	4.5	1.82	2.5	1.01	2.9	1.18	2.5	1.00
2.5	5.5	2.24	2.5	1.02	3.1	1.27	2.5	1.00
3.0	6.5	2.65	2.5	1.03	3.3	1.36	2.5	1.01

Table 52: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	2.5	1.00	2.5	1.00	2.5	1.00	2.5	1.00
1.5	3.5	1.41	2.5	1.01	2.7	1.09	2.5	1.00
2.0	4.5	1.82	2.5	1.01	2.9	1.18	2.5	1.00
2.5	5.5	2.24	2.5	1.02	3.1	1.26	2.5	1.00
3.0	6.5	2.65	2.5	1.03	3.3	1.35	2.5	1.00

Table 53: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3.0	1.00	3.0	1.00	3.0	1.00	3.0	1.00
1.5	4.2	1.38	3.2	1.05	3.4	1.12	3.0	1.00
2.0	5.3	1.76	3.3	1.10	3.7	1.24	3.0	1.00
2.5	6.5	2.14	3.5	1.16	4.1	1.36	3.0	1.00
3.0	7.6	2.53	3.7	1.21	4.5	1.48	3.0	1.01

Table 54: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3.0	1.00	3.0	1.00	3.0	1.00	3.0	1.00
1.5	4.2	1.38	3.2	1.05	3.4	1.12	3.0	1.00
2.0	5.3	1.76	3.3	1.10	3.7	1.24	3.0	1.00
2.5	6.5	2.15	3.5	1.16	4.1	1.35	3.0	1.00
3.0	7.6	2.53	3.7	1.21	4.5	1.47	3.0	1.00

Table 55: Effects of Q value for 10^6 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	10924.0	1.00	10924.0	1.00	10924.0	1.00	10924.0	1.00
1.5	12397.4	1.13	11076.4	1.01	16048.8	1.47	13326.4	1.22
2.0	13870.8	1.27	11212.0	1.03	21217.3	1.94	15772.4	1.44
2.5	15344.1	1.40	11333.2	1.04	26135.2	2.39	17967.9	1.64
3.0	16817.5	1.54	11442.3	1.05	30702.5	2.81	19812.8	1.81

Table 56: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	8638.3	1.00	8638.3	1.00	8638.3	1.00	8638.3	1.00
1.5	10111.7	1.17	8790.8	1.02	11601.0	1.34	8878.6	1.03
2.0	11585.1	1.34	8926.3	1.03	14568.0	1.69	9123.2	1.06
2.5	13058.5	1.51	9047.5	1.05	17510.0	2.03	9342.7	1.08
3.0	14531.8	1.68	9156.6	1.06	20416.9	2.36	9527.2	1.10

Table 57: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	17322.4	1.00	17322.4	1.00	17322.4	1.00	17322.4	1.00
1.5	20428.1	1.18	18819.1	1.09	24046.8	1.39	19724.8	1.14
2.0	23533.7	1.36	20149.5	1.16	30814.9	1.78	22170.9	1.28
2.5	26639.3	1.54	21339.8	1.23	37332.5	2.16	24366.4	1.41
3.0	29745.0	1.72	22411.1	1.29	43499.3	2.51	26211.2	1.51

Table 58: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	15036.8	1.00	15036.8	1.00	15036.8	1.00	15036.8	1.00
1.5	18142.4	1.21	16533.4	1.10	19599.0	1.30	15277.0	1.02
2.0	21248.0	1.41	17863.8	1.19	24165.7	1.61	15521.6	1.03
2.5	24353.7	1.62	19054.1	1.27	28707.2	1.91	15741.1	1.05
3.0	27459.3	1.83	20125.4	1.34	33213.8	2.21	15925.6	1.06

Table 59: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3182.7	1.00	3182.7	1.00	3182.7	1.00	3182.7	1.00
1.5	4489.5	1.41	3198.0	1.00	3479.0	1.09	3206.7	1.01
2.0	5796.3	1.82	3211.5	1.01	3775.7	1.19	3231.2	1.02
2.5	7103.1	2.23	3223.6	1.01	4069.9	1.28	3253.2	1.02
3.0	8409.8	2.64	3234.5	1.02	4360.6	1.37	3271.6	1.03

Table 60: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3159.9	1.00	3159.9	1.00	3159.9	1.00	3159.9	1.00
1.5	4466.6	1.41	3175.1	1.00	3434.5	1.09	3162.3	1.00
2.0	5773.4	1.83	3188.7	1.01	3709.2	1.17	3164.7	1.00
2.5	7080.2	2.24	3200.8	1.01	3983.6	1.26	3166.9	1.00
3.0	8387.0	2.65	3211.7	1.02	4257.7	1.35	3168.7	1.00

Table 61: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3822.6	1.00	3822.6	1.00	3822.6	1.00	3822.6	1.00
1.5	5292.6	1.38	3972.2	1.04	4278.8	1.12	3846.6	1.01
2.0	6762.6	1.77	4105.3	1.07	4735.5	1.24	3871.0	1.01
2.5	8232.6	2.15	4224.3	1.11	5189.6	1.36	3893.0	1.02
3.0	9702.6	2.54	4331.4	1.13	5640.3	1.48	3911.4	1.02

Table 62: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	3799.7	1.00	3799.7	1.00	3799.7	1.00	3799.7	1.00
1.5	5269.7	1.39	3949.4	1.04	4234.3	1.11	3802.1	1.00
2.0	6739.7	1.77	4082.4	1.07	4669.0	1.23	3804.6	1.00
2.5	8209.7	2.16	4201.4	1.11	5103.4	1.34	3806.7	1.00
3.0	9679.7	2.55	4308.6	1.13	5537.4	1.46	3808.6	1.00

Table 63: Effects of Q value for 10^7 tuples and 8 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1376.6	1.00	1376.6	1.00	1376.6	1.00	1376.6	1.00
1.5	1563.1	1.14	1401.5	1.02	2102.8	1.53	1759.7	1.28
2.0	1749.6	1.27	1426.0	1.04	2958.3	2.15	2272.1	1.65
2.5	1936.1	1.41	1450.1	1.05	3922.1	2.85	2892.8	2.10
3.0	2122.6	1.54	1473.8	1.07	4971.1	3.61	3598.8	2.61

Table 64: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1090.9	1.00	1090.9	1.00	1090.9	1.00	1090.9	1.00
1.5	1277.4	1.17	1115.8	1.02	1472.3	1.35	1129.2	1.04
2.0	1463.9	1.34	1140.2	1.05	1866.6	1.71	1180.4	1.08
2.5	1650.4	1.51	1164.4	1.07	2271.7	2.08	1242.5	1.14
3.0	1836.8	1.68	1188.1	1.09	2685.4	2.46	1313.1	1.20

Table 65: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	2276.4	1.00	2276.4	1.00	2276.4	1.00	2276.4	1.00
1.5	2687.9	1.18	2521.0	1.11	3227.5	1.42	2659.5	1.17
2.0	3099.4	1.36	2761.9	1.21	4307.9	1.89	3171.9	1.39
2.5	3510.9	1.54	2999.1	1.32	5496.7	2.41	3792.6	1.67
3.0	3922.3	1.72	3232.7	1.42	6770.7	2.97	4498.6	1.98

Table 66: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	1990.7	1.00	1990.7	1.00	1990.7	1.00	1990.7	1.00
1.5	2402.2	1.21	2235.3	1.12	2597.0	1.30	2029.0	1.02
2.0	2813.6	1.41	2476.2	1.24	3216.3	1.62	2080.2	1.04
2.5	3225.1	1.62	2713.4	1.36	3846.4	1.93	2142.3	1.08
3.0	3636.6	1.83	2947.0	1.48	4485.0	2.25	2212.9	1.11

Table 67: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	398.9	1.00	398.9	1.00	398.9	1.00	398.9	1.00
1.5	562.5	1.41	401.4	1.01	437.1	1.10	402.8	1.01
2.0	726.1	1.82	403.9	1.01	476.5	1.19	407.9	1.02
2.5	889.6	2.23	406.3	1.02	517.0	1.30	414.1	1.04
3.0	1053.2	2.64	408.7	1.02	558.4	1.40	421.2	1.06

Table 68: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	396.1	1.00	396.1	1.00	396.1	1.00	396.1	1.00
1.5	559.6	1.41	398.6	1.01	430.8	1.09	396.5	1.00
2.0	723.2	1.83	401.0	1.01	465.6	1.18	397.0	1.00
2.5	886.8	2.24	403.4	1.02	500.5	1.26	397.6	1.00
3.0	1050.4	2.65	405.8	1.02	535.5	1.35	398.3	1.01

Table 69: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	488.9	1.00	488.9	1.00	488.9	1.00	488.9	1.00
1.5	675.0	1.38	513.4	1.05	549.5	1.12	492.7	1.01
2.0	861.1	1.76	537.5	1.10	611.5	1.25	497.9	1.02
2.5	1047.1	2.14	561.2	1.15	674.5	1.38	504.1	1.03
3.0	1233.2	2.52	584.5	1.20	738.3	1.51	511.1	1.05

Table 70: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	486.1	1.00	486.1	1.00	486.1	1.00	486.1	1.00
1.5	672.1	1.38	510.5	1.05	543.2	1.12	486.4	1.00
2.0	858.2	1.77	534.6	1.10	600.6	1.24	486.9	1.00
2.5	1044.3	2.15	558.3	1.15	658.0	1.35	487.6	1.00
3.0	1230.3	2.53	581.7	1.20	715.5	1.47	488.3	1.00

Table 71: Effects of Q value for 10^7 tuples and 64 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	86.1	1.00	86.1	1.00	86.1	1.00	86.1	1.00
1.5	97.8	1.14	87.7	1.02	132.3	1.54	110.8	1.29
2.0	109.4	1.27	89.3	1.04	188.3	2.19	145.4	1.69
2.5	121.1	1.41	90.9	1.06	254.0	2.95	189.6	2.20
3.0	132.7	1.54	92.5	1.07	329.3	3.82	243.4	2.83

Table 72: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	68.2	1.00	68.2	1.00	68.2	1.00	68.2	1.00
1.5	79.9	1.17	69.8	1.02	92.2	1.35	70.7	1.04
2.0	91.6	1.34	71.5	1.05	117.1	1.72	74.2	1.09
2.5	103.2	1.51	73.1	1.07	143.0	2.10	78.6	1.15
3.0	114.9	1.68	74.7	1.09	169.8	2.49	84.0	1.23

Table 73: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	143.2	1.00	143.2	1.00	143.2	1.00	143.2	1.00
1.5	169.1	1.18	159.0	1.11	203.6	1.42	167.9	1.17
2.0	195.0	1.36	174.9	1.22	273.9	1.91	202.4	1.41
2.5	220.9	1.54	190.7	1.33	353.9	2.47	246.7	1.72
3.0	246.9	1.72	206.5	1.44	443.4	3.10	300.5	2.10

Table 74: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	125.3	1.00	125.3	1.00	125.3	1.00	125.3	1.00
1.5	151.2	1.21	141.2	1.13	163.5	1.30	127.8	1.02
2.0	177.2	1.41	157.0	1.25	202.7	1.62	131.2	1.05
2.5	203.1	1.62	172.8	1.38	242.9	1.94	135.7	1.08
3.0	229.0	1.83	188.6	1.51	284.0	2.27	141.0	1.13

Table 75: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 1.000$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	24.9	1.00	24.9	1.00	24.9	1.00	24.9	1.00
1.5	35.1	1.41	25.1	1.01	27.3	1.10	25.2	1.01
2.0	45.3	1.82	25.2	1.01	29.8	1.20	25.5	1.02
2.5	55.5	2.23	25.4	1.02	32.4	1.30	25.9	1.04
3.0	65.7	2.64	25.6	1.03	35.1	1.41	26.5	1.06

Table 76: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	24.7	1.00	24.7	1.00	24.7	1.00	24.7	1.00
1.5	34.9	1.41	24.9	1.01	26.9	1.09	24.8	1.00
2.0	45.1	1.83	25.1	1.01	29.1	1.18	24.8	1.00
2.5	55.4	2.24	25.2	1.02	31.3	1.26	24.8	1.00
3.0	65.6	2.65	25.4	1.03	33.5	1.35	24.9	1.01

Table 77: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 0.20$ $\rho^{net} = 10^{-8}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	30.6	1.00	30.6	1.00	30.6	1.00	30.6	1.00
1.5	42.2	1.38	32.2	1.05	34.4	1.12	30.9	1.01
2.0	53.9	1.76	33.8	1.10	38.4	1.25	31.2	1.02
2.5	65.5	2.14	35.4	1.16	42.4	1.38	31.6	1.03
3.0	77.2	2.52	36.9	1.21	46.5	1.52	32.2	1.05

Table 78: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-7}$

Q	TPS		SS		HPS		JPS	
	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio	resp(sec)	ratio
1.0	30.4	1.00	30.4	1.00	30.4	1.00	30.4	1.00
1.5	42.1	1.38	32.0	1.05	34.0	1.12	30.5	1.00
2.0	53.7	1.76	33.6	1.10	37.6	1.24	30.5	1.00
2.5	65.3	2.15	35.2	1.16	41.3	1.36	30.5	1.00
3.0	77.0	2.53	36.8	1.21	44.9	1.47	30.6	1.01

Table 79: Effects of Q value for 10^7 tuples and 1024 nodes with $\alpha^{net} = 0.100$ $T_{msg} = 2.00$ $\rho^{net} = 10^{-8}$