The paper evaluates all the advantages and disadvantages of the deterministic database systems using different experiments in the Calvin codebase. In doing so, it plans to aid in selecting a database by considering the workload and cluster configuration of the system. The paper evaluates different claims made by different papers and experiments conducted, to fully understand these systems. The paper explains several tradeoffs that occur when we try to achieve the advantages of the deterministic databases systems. Deterministic execution requires the transactions to be processed in such a way that guarantees the same final state, on the same transactional input.

The paper explains the differences between traditional and deterministic systems on the parameters of Transaction Processing, Concurrency, Agreement on Input and Commit Protocols. Few of the advantages and disadvantages of the features of the deterministic systems are as follows:
1). No nondeterministic aborts: This results in simplified commit protocols, but in cases of transaction overload on a node or out-of-memory errors, it cannot abort the transaction.
2). The transactions from the input are placed in a sequence, resulting in easy logging and recovery of the systems in case of the failures, but increases the latency in the system, due to preprocessing of the transaction sequence by an additional layer involving optimistic lock prediction protocol (OLPP), where the transactions need to be aborted and restarted, if these predictions turn out to be incorrect.
3). Since locks are acquired in the order of transactions, this eliminates the possibilities of any deadlock, but lacks concurrency, thereby resulting in low throughput.

The experiments are implemented in both the deterministic and non deterministic database system, to determine different parameters in which deterministic systems outperforms the traditional systems. Few of these parameters are single-node, multi-node distributed and partition. On the evaluation under contention transactions, it is observed that non-deterministic execution suffers bottlenecks for the overhead of two-phase commit protocol and the possibilities of deadlocks. Under multi-node distributed transactions, it is observed that as there are more distributed transactions, there is a drop in throughput. Under TPC benchmark, as the percentage of the distributed transactions increases, the advantages of the deterministic systems is increased. On experimenting on the OLPP, it is noticed that the processing of the distributed transactions is independent of it.

Strong Points:
1). The paper provides a thorough study of the deterministic database systems, as and when all the features of the deterministic systems will fail or be competitively benefit the system.
2). The paper finds an important correlation between the number of distributed transaction sizes and the determinism factor, that they are independent of each other.

3). The paper provides a much better comparison when testing systems under distributed transaction, by using more than 2 nodes, compared to previous paper implementations.
4). The paper highlights and infers the point that increase in distributed transactions percent has an increase in the poor performance in the scalability of the non deterministic systems.
5). The paper has managed to back each of its claims by referring to a previous section or explanation already mentioned in the paper.

Weak Points:
1). Unpredictable constraints on resources can result in a significant decrease in the performance of deterministic systems.
2). As transactions cannot be reordered at ease in a deterministic approach, there is a wait on faster machines for data dependency on the slower machines.
3). Deterministic systems have comparatively more latency than a non deterministic system, due to the fact that there is a processing layer which processes each of the transactions in a deterministic prior to running them actually in the system.

Question:
1). The paper mentions little information about the OLPP protocol. How does it compare to other mechanisms available for the same purpose?
2). What were the improvements or different database applications or systems that were designed, which incorporated the learning of this paper?