Summary
The authors of this paper present a new lookup protocol that outperforms many of the previous lookup protocols. Chord is able to achieve scalability without using hierarchy, and has much less complexity at the cost of not stopping queries after they have passed a threshold. In order for the protocol to perform well there must be a constant updating of the finger tables which hold information about successor to a node. There is an update process which allows for current nodes to detect new nodes through an exchange of messages at a specified time interval. They use a technique called “stabilization” for the operation and to ensure concurrentacy is possible. The load is rebalanced as new nodes inter the Chord. Chord is able to save the memory required by only allowing nodes to know about n of their successors. If a request can’t be satisfied it is simply passed on to the successor that the node thinks is the closest node that would contain the requested key. The protocol runs in both recursive and iterative forms. They explain the results characteristics such as load balancing, path length and node failures in their simulated environment. The number of node failures concurrently seems to correspond to the number of node failures almost exactly. They implement their Chord protocol on top of RON and show that scalability is feasible as projected.

Pros
● They provide some brief examples of applications where Chord would be useful.
● They explain several theorems as to why the Chord system outperforms other lookup systems
● Failures and joins can occur concurrently as long as failures do not disrupt the lookup queries.
● They utilize consistent hashing to help with load balancing
● The path length in general has a good length O(log n) which helps achieve good performance
● Chord is attractive because it is simple and they are able to prove it’s performance through theorems.

Cons
● They were only able to present preliminary experiment results
● They don’t explain how partitioning could be a problem, but they do it as future work
● The authors don’t explain if any negative effects Chord might have on a network other than latency, which made me question how simple the implementation would be for this protocol.