CHAPTER - 6

COMPARATIVE STUDY

Introduction

Concurrent access of data structures can at a high level be designed using two different approaches, locking and lock free. Each of these has its own problems and solutions. Locking is a very controlled way of approaching the problem; however it is also quite pessimistic requiring the locking of all resources, even if it is unlikely that another thread will attempt to access the data simultaneously. As multi processor computer system are becoming increasingly popular, there is great pressure to develop standard data structures which are safe to use in multiprocessor environment and ideally perform better in these environments than their classical counterparts. The main goal of this chapter is to introduce the comparison between classical modified skip list data structure with its comparative data structures like skip list. The comparison is also done according to concurrent access of these data structures.

Comparison between Modified Skip List & Skip List Data Structure

If we do comparison between modified skip list with its correlated data structure like skip list. It is analyzed by { Sartaj [ ] }, in my thesis work I am representing these issues in my own way, there is implementation issues in skip list data structure which makes modified skip list better one. These issues are discussed as follows
The skip list data structure is represented as hierarchy of node with different levels as shown in above fig 6.1 the node with key value 1 has node size three, node with key value 2 has node size 1, and likewise node with key value 5 has node size 4. Because of that variable size of node structure is supported by skip list data structure. The memory representation of this kind of structure used a fixed node size whether all fields of structure are required or not. This Kind of storage requires much more space than Modified skip list data structure. The modified skip list data structure takes less storage space if they implemented by those languages which support dynamic construction of variable size arrays. If we look at algorithm of MSL comparative to skip list data structure, the number of comparison done in the operation on MSL data structure are much more than skip list, still modified skip list takes less execution time than skip list, due to complex structure of skip list.

**Comparison between Modified Skip List & Skip List Data Structure with concurrent Access**

- One of the major comparisons between concurrent access of Modified skip list /skip list is the complexity of algorithm. The algorithmic steps for skip list are more complex, due to its varying hierarchy structure.
• Lock based Concurrent Insertion operation require a sequence of locks for updating the adjacent pointers, where as in case of modified skip list only adjacent node which is effected due to insertion is only lock, so there is extra overhead of lock while constructing them with locking technique, same thing occurs while doing deletion of node.

• The lock–free concurrent priority queue designed with minor modification in modified skip list called threaded modified skip list with worst case insertion complexity of $O(\log n)$ and delete_min is $O(1)$, i.e. delete_min operation is succeed in one attempt where as in a concurrent priority queue based on skip list has a delete_min complexity of $O(\log n)$.