COL380 REPORT

cs1190404

January 2022

1 Design Choice

In this design we use the simple procedure that a given in the assignment document that first divide data into p buckets of n/p+1 size up to n%p and after that we just follow the following procedure:
1. Divide A into $A_0, A_1, \ldots, A_{p-1}$, p buckets of size n/p $\pm$ 1 each as follows. Each A_i contains contiguous elements of A.
2. From each bucket A_i, select first p elements as pseudo-splitters. Let $R = [r_0, r_1, \ldots, r_{p*p-1}]$ be the sorted list of p 2 pseudo-splitters. This sorting may use ParallelSort or SequentialSort.
3. Select p-1 equally spaced splitters from R as follows. Let $S = [s_0, s_1, \ldots, s_{p-2}]$ be the selected splitters such that $s_j = R[(j+1) * p]$ for j in 0 to p-2.
4. (Using tasks) Split A into p partitions $B_0, B_1, \ldots, B_{p-1}$ such that for any element a in partition $B_i, s_{i-1} < a \leq s_i$. Assume $s_{-1} = -\infty ands_p = \infty$.
5. Let n_i denote the number of elements in partition Bi. Sort each partition B_i in a separate task which uses SequentialSort$(B_i, n_i) if n_i < Threshold$, and ParallelSort$(B_i, n_i, p)$ otherwise. SequentialSort is sequential sorting of your choice implemented in a task.
6. Return concatenation of sorted partitions B_i

The overall orde if oarallelism is O(num of threads).