PREFACE

Due to the advent of paralel computers and because of the growing need for high speed processing of large amount of data many efficient sequential algorithms in computational geometry have been developed . But in comparison with the number of sequential algorithms the number of parallel algorithms developed in the area is very few.

Intersection and proximity problems are two important classes of computational geometry problems. The study of the intersection problems have started from the very fact that two objects can not occupy the same place at the same time. Proximity problems are studied in the cases where it is required to have access to a suitable neighbourhood of the objects.

The motivation of this thesis is to develop parallel algorithms for some intersection and proximity problems which arise in a wide range of application areas and are solved sequentially by different authors but no parallel algorithms exist for these problems.

ABSTRACT

In this thesis a study has been carried out on the design and analysis of parallel algorithms for certain intersection and proximity problems.

The thesis contains parallel algorithms for the computation of the contour of a set of iso-oriented rectangles and a set of rectilinear polygons and for reassembling rectilinear polygons from edges .These problems belong to the class of intrsection problems. The proximity problems that have been solved in this thesis include stationing guards in a rectilinear art-gallery, finding the internal as well as the external geodesic diameters of a simple polygon. Parallel algorithms for these problems have not been considered before. Most of the parallel algorithms designed in this thesis are on the SM.SIMD CREW model of parallel computer.

In Chapter three a parallel algorithm has been designed for the problem of computing the contour of a set of iso-oriented rectangles which runs in O(log n) time using $O(n^2)$ processors where n is the total number of edges of the input rectangles.

In Chapter four a parallel algorithm has been presented for the contour computation of a set of rectilinear polygons with the same time and processor complexity as in Chapter 3 where n is the total no. of edges of the input polygons.

In Chapter five an $O(\log^2 n)$ time parallel algorithm using $O(n^2)$ processors has been presented for the problem of