ON THE CORRECTNESS OF SOME BISECTION-LIKE PARALLEL EIGENVALUE ALGORITHMS IN FLOATING POINT ARITHMETIC*

JAMES W. DEMMEL [†], INDERJIT DHILLON[‡], AND HUAN REN[§]

Abstract. Bisection is a parallelizable method for finding the eigenvalues of real symmetric tridiagonal matrices, or more generally symmetric acyclic matrices.

Ideally, one would like an implementation that was simultaneously parallel, load balanced, devoid of communication, capable of running on networks of heterogenous workstations, and of course correct. But this is surprisingly difficult to achieve.

The reason is that bisection requires a function Count(x) which counts the number of eigenvalues less than x. In exact arithmetic Count(x) is a monotonic increasing function of x, and the logic of the algorithm depends on this. However, monotonicity can fail, and incorrect eigenvalues may be computed, because of roundoff or as a result of using networks of heterogeneous parallel processors. We illustrate this problem, which even arises in some serial algorithms like the EISPACK routine `bisect`, and show several ways to fix it. One of these ways has been incorporated into the ScaLAPACK library.

Key words. symmetric eigenvalue problem, parallel algorithms, monotonicity, correctness, floating point.

AMS subject classifications. 65F15, 65Y05.