

Abstract

Given a set of prioritized balls with fixed centers in R^d whose radii grow linearly over time, we want to compute the *elimination order* of these balls assuming that when two balls touch, the one with lower priority is 'crushed'. A straightforward algorithm has running time $O(n^2 \log n)$ which we improve to expected $O(\Delta^d n(\log n + \Delta^d))$ where $\Delta = r_{\max}/r_{\min}$ is the ratio between largest and smallest radius amongst the balls. For a natural application of this problem, namely drawing labels on the globe, we have $\Delta = O(1)$. An efficient implementation based on a spherical Delaunay triangulation allows to compute the elimination order for millions of labels on commodity Desktop hardware. Dealing with rounding error induced robustness issues turned out to be one of the major challenges in the implementation.