Computing the smallest enclosing ball of balls

Mike Bostock and Robin Houston

May 15, 2017

Abstract

We consider the problem of computing the smallest ball enclosing a set of balls. There is an elegant randomised algorithm due to Welzl (1991) that computes the smallest ball enclosing a set of points: it was shown by Fischer and Gärtner (2003); Fischer (2005) that a natural extension of Welzl's algorithm to sets of balls does not work in general. We observe that a trivial modification to this algorithm makes it work correctly.

1 Welzl's algorithm

There is an algorithm $\textsc{BallBoundedBy}(T)$ that takes a finite set of points in $\mathbb{R}^d$ and computes the smallest d-ball whose boundary contains all the points in T. This amounts to solving a system of $|T|$ quadratic equations. [Explain how this reduces to one quadratic equation and $|T| - 1$ linear equations, etc.] There may be no such ball, in which case the algorithm returns **null**.

Using $\textsc{BallBoundedBy}$ as a subroutine, the following recursive algorithm due to Welzl (1991) computes the smallest ball containing a finite set of points, with some of them constrained to be on the ball's boundary. Again there may be no such ball, in which case the algorithm will return **null**.

Algorithm 1 Welzl's algorithm for the bounding ball of points

```
{ Compute the smallest ball containing all the points in S ∪ T that has all
the points in T on its boundary. }
function WelzlP(S, T)
    if S = ∅ then
        return BallBoundedBy(T)
    end if
    Choose p ∈ S uniformly at random
    D ← WelzlP(S − {p}, T)
    if D = null or p ∈ D then
        return D
    else
        return WelzlP(S − {p}, T ∪ {p})
    end if
end function
```

We can use this to compute the smallest ball containing all the points in a finite set P simply by calling $\textsc{WelzlP}(P, \emptyset)$.

2 Enclosing ball of balls

Let's write $\mathrm{mb}(S)$ to mean the smallest ball containing all the balls in S, where S is a finite set of balls in $\mathbb{R}^d$. [Show this exists and is unique.] Let's also write $\mathrm{mb}(S;T)$ to mean the smallest ball containing all the balls in $S \cup T$ that has all the balls in T tangent to its boundary. In general $\mathrm{mb}(S;T)$ may not exist, and if it does exist it may not be unique. [Give examples.]

Assume we have an algorithm BallTangentTo(T) that computes $\mathrm{mb}(\emptyset, T)$. The solution procedure is essentially the same as that for BallBoundedBy, with slightly different equations. [Explain further.] As we shall see below, we need only assume that BallTangentTo(T) gives an answer when $\mathrm{mb}(\emptyset, T) = \mathrm{mb}(T)$: in other cases it is allowed to return **null**.

Here is our version of Welzl's algorithm adapted to computing the bounding ball of balls.

Algorithm 2 Compute the bounding ball of balls

```
{ Compute mb(S;T) assuming that mb(S;T) = mb(S ∪ T) }
function WelzlB(S,T)
   if S = ∅ then
      return BallTangentTo(T)
   end if
   Choose B ∈ S uniformly at random
   D ← WelzlB(S − {B}, T)
   if D = null or B ⊈ D then
      return WelzlB(S − {B}, T ∪ {B})
   else
      return D
   end if
end function
```

The only change to the actual algorithm is that a **null** return value from a recursive call is handled in the opposite way: we treat **null** as a ball that contains nothing, rather than as a ball that contains everything. More important is the way we have weakened the description of the algorithm: it is only guaranteed to give an answer when $\mathrm{mb}(S;T) = \mathrm{mb}(S \cup T)$. Of course this makes no difference in the case we principally care about, where the algorithm is initially called with $T = \emptyset$.

3 Proof of correctness

Theorem 3.1. *For any finite sets S, T of d-balls:*

i) *If* WelzlB(S,T) *returns an answer, rather than* **null**, *that answer is equal to* $\mathrm{mb}(S;T)$*;*

ii) *If* $\mathrm{mb}(S;T) = \mathrm{mb}(S \cup T)$ *then* WelzlB(S,T) *returns an answer.*

Proof. We proceed by induction on $|S|$. For the base case: if $S = \emptyset$ then both parts are true by the correctness of BallTangentTo. So assume we have chosen $B \in S$, and that (i) and (ii) are true for WelzlB($S - \{B\}, T$).

Suppose first that $\mathrm{mb}(S - \{B\}; T) = \mathrm{mb}((S - \{B\}) \cup T)$. Then, by the inductive hypothesis, D is equal to $\mathrm{mb}(S - \{B\}; T) = \mathrm{mb}((S - \{B\}) \cup T)$.

If $B \subseteq D$ then D must also be equal to $\mathrm{mb}(S;T)$, as required. If $B \not\subseteq D$ then ...

Next, suppo $\square$

4 Practical considerations

[Just like Welzls original algorithm. We can just shuffle the input balls once at the outset, rather than making a random choice at each step. This is more efficient and still runs in expected linear time. Also the move-to-front heuristic seems to make it faster in practice: explain why this is intuitively plausible, though not supported by rigorous theory. (Also do a more thorough literature search to see if anyone has done any theoretical analysis of the move-to-front heuristic.)]

5 Experimental results

[Compare our algorithm to the MSW algorithm, and report how much faster it is.]

6 Move-to-front

[Can we say anything about whether the move-to-front heuristic has the effect that recursive calls are always tight?]

References

Kaspar Fischer. *Smallest enclosing balls of balls*. PhD thesis, Swiss Federal Institute of Technology, ETH Zürich, 2005.

Kaspar Fischer and Bernd Gärtner. The smallest enclosing ball of balls: Combinatorial structure and algorithms. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, SCG '03, pages 292–301, New York, NY, USA, 2003. ACM. ISBN 1-58113-663-3. doi: 10.1145/777792.777836. URL http://doi.acm.org/10.1145/777792.777836.

Emo Welzl. Smallest enclosing disks (balls and ellipsoids). *New Results and New Trends in Computer Science*, pages 359–370, 1991.