Smallest Enclosing Circle Problem

Raghukul,Vibhor

IIT Kanpur

November 12, 2018

Outline

Outline

Problem

- Given n points in a plane, compute the smallest radius circle enclosing all n points.

Problem

- Given n points in a plane, compute the smallest radius circle enclosing all n points.
- The time complexity of best deterministic algorithm for this problem is $O(n)$ which uses advanced geometry.

Problem

- Given n points in a plane, compute the smallest radius circle enclosing all n points.
- The time complexity of best deterministic algorithm for this problem is $O(n)$ which uses advanced geometry.
- We will present a simple randomised algorithm for this problem with expected $O(n)$ time complexity.

Problem

- Given n points in a plane, compute the smallest radius circle enclosing all n points.
- The time complexity of best deterministic algorithm for this problem is $O(n)$ which uses advanced geometry.
- We will present a simple randomised algorithm for this problem with expected $O(n)$ time complexity.

Ideas

- A circle is completely determined if we are given three or two points lying on its boundary.

Ideas

- A circle is completely determined if we are given three or two points lying on its boundary.
- We can use this idea to get a trivial $O(n^4)$ deterministic algorithm for our problem.

Ideas

- A circle is completely determined if we are given three or two points lying on its boundary.
- We can use this idea to get a trivial $O(n^4)$ deterministic algorithm for our problem.
- This algorithm can be easily modified to give a deterministic algorithm with $O(n^3)$ time complexity.

Ideas

- A circle is completely determined if we are given three or two points lying on its boundary.
- We can use this idea to get a trivial $O(n^4)$ deterministic algorithm for our problem.
- This algorithm can be easily modified to give a deterministic algorithm with $O(n^3)$ time complexity.
- The idea is that given two defining points we can find the third defining point in $O(n)$ time.

Ideas

- A circle is completely determined if we are given three or two points lying on its boundary.
- We can use this idea to get a trivial $O(n^4)$ deterministic algorithm for our problem.
- This algorithm can be easily modified to give a deterministic algorithm with $O(n^3)$ time complexity.
- The idea is that given two defining points we can find the third defining point in $O(n)$ time.
- Check whether these two circles(one constructed taking the 2 defining points as diameteric ends) are valid and return the smallest radius circle enclosing these points.

Outline

$O(n^4)$ algorithm

Input: N points in $2D$ plane
Output: Circle enclosing given N points and having minimum radius

```
P ← list of N points
C ← circle formed by taking P[1], P[2] as diameter.
for i ← 1 to n do
    for j ← i + 1 to n do
        C′ ← circle formed by taking P[i], P[j] as diametric ends.
        if C′.rad < C.rad then
            C = C′
    end
end
for i ← 1 to n do
    for j ← i + 1 to n do
        for k ← j + 1 to n do
            C′ ← Circumcircle of P[i], P[j], P[k]
            if C′.rad < C.rad then
                C = C′
        end
    end
end
return C
```

$O(n^3)$ algorithm

Input: N points in $2D$ plane
Output: Circle enclosing given N points and having minimum radius
$P \leftarrow$ list of N points
$C \leftarrow$ circle formed by taking $P[1], P[2]$ as diameter.
for $i \leftarrow 1$ **to** n **do**
    for $j \leftarrow i + 1$ **to** n **do**
        P_{min} = point subtending least acute angle on line $P[i] \rightarrow P[j]$
        $C_1 \leftarrow$ circle formed by taking $P[i], P[j]$ as diametric ends.
        $C_2 \leftarrow$ circumcircle of $P[i], P[j], P_{min}$.
        $C^{'} \leftarrow$ circle among C_1, C_2 having least radius.
        if $C^{'}.rad < C.rad$ **then**
            $C = C^{'}$
    end
end
return C

Expected $O(n)$ algorithm

Input: N points in $2D$ plane
Output: Circle enclosing given N points and having minimum radius
$P \leftarrow$ list of N points
$P^{'} \leftarrow$ random permutation of P
$C \leftarrow$ circle formed by taking $P^{'}[1], P^{'}[2]$ as diameter.
for $i \leftarrow 3$ **to** n **do**
  if $C.outside(P^{'}[i])$ **then**
    $C \leftarrow$ build_circle$(i-1, P[i])$
end
return C

build_circle

Input: k, P_d
Output: Circle enclosing first k points and P_d being on boundary
$C \leftarrow$ circle formed by taking $P^{'}[1], P_d$ as diameter.
for $i \leftarrow 2$ **to** k **do**
  if $C.outside(P^{'}[i])$ **then**
    $l \leftarrow$ line joining $P^{'}[i], P_d$
    P_{min} = point($[1, i-1]$) subtending least acute angle on l
    $C_1 \leftarrow$ circle formed by taking $P^{'}[i], P_d$ as diametric ends.
    $C_2 \leftarrow$ circumcircle of $P^{'}[i], P_d, P_{min}$.
    $C \leftarrow$ circle among C_1, C_2 having least radius.
end
return C

Lemma 1

If a point P is outside the smallest enclosing circle of set S then it must be one of the defining points of smallest enclosing circle of set $S \cup \{P\}$.

Lemma 1

If a point P is outside the smallest enclosing circle of set S then it must be one of the defining points of smallest enclosing circle of set $S \cup \{P\}$.
Proof : Note that there has to to at least 2 point on the circle. If not then we can compress the circle, so that there are at least 2 points on the circle.

Lemma 1

If a point P is outside the smallest enclosing circle of set S then it must be one of the defining points of smallest enclosing circle of set $S \cup \{P\}$.
Proof : Note that there has to to at least 2 point on the circle. If not then we can compress the circle, so that there are at least 2 points on the circle.
Also the boundary points define the circle completely, if not, then again we can compress the circle. Suppose P is not a defining point, by above statement there are some defining point on this new circle. These defining points were present before P was added, so the radius of circle was at least equal to the radius of circle defined by these points. After adding P they are defining point, so the radius of this equal to the radius defined by these point. In other words the circle didn't change, after adding P. But this contradicts our assumption that P is outside the smallest enclosing circle.

Lemma 2

Probability that point P_i lies outside the smallest enclosing circle of points $P_1, ..., P_{i-1}$ is $\leq \frac{3}{i}$.

Lemma 2

Probability that point P_i lies outside the smallest enclosing circle of points $P_1, ..., P_{i-1}$ is $\leq \frac{3}{i}$.

Proof : By Lemma 1, if the smallest enclosing circle is changed on adding the i^{th} point then it must be one of the defining points of smallest enclosing circle of first i points, call this circle C_i.

Lemma 2

Probability that point P_i lies outside the smallest enclosing circle of points $P_1, ..., P_{i-1}$ is $\leq \frac{3}{i}$.

Proof : By Lemma 1, if the smallest enclosing circle is changed on adding the i^{th} point then it must be one of the defining points of smallest enclosing circle of first i points, call this circle C_i.

There can be at most 3 defining points for C_i because C_{i-1} is not same as C_i, so the proability that i^{th} point is defining point for C_i is $\leq \frac{3}{i}$.

Lemma 3

Given two defining points then the third defining point is the point which form the least acute angle with these two points.

Lemma 3

Given two defining points then the third defining point is the point which form the least acute angle with these two points.

Proof : The angle subtended by a chord on the boundary of the circle is least compared to all other points in the same sector and for a point to be the defining point it must lie on the boundary.

Lemma 3

Given two defining points then the third defining point is the point which form the least acute angle with these two points.

Proof : The angle subtended by a chord on the boundary of the circle is least compared to all other points in the same sector and for a point to be the defining point it must lie on the boundary.
So the third defining point(if any) is the point which form the least acute angle with these two points.

Outline

Analysis

- Worst case time complexity of our randomized algorithm is $O(n^3)$.

Analysis

- Worst case time complexity of our randomized algorithm is $O(n^3)$. This occurs when all the points lie outside the existing circle, and while updating circle, removing q does not give a defining point on every step of recursion.

Analysis

- Worst case time complexity of our randomized algorithm is $O(n^3)$. This occurs when all the points lie outside the existing circle, and while updating circle, removing q does not give a defining point on every step of recursion.
- Expected time complexity of our randomized algorithm is $O(n)$.

Analysis

- Worst case time complexity of our randomized algorithm is $O(n^3)$. This occurs when all the points lie outside the existing circle, and while updating circle, removing q does not give a defining point on every step of recursion.
- Expected time complexity of our randomized algorithm is $O(n)$.
  - The expected runtime of $build\_circle()$ function is $O(i)$.

Experimental Results

Comparison between run time of the 3 algorithms discussed.

N	Expected $O(n)$ algorithm	$O(n^3)$ Algorithm	$O(n^4)$ Algorithm
10	0.063	0.186	0.248
20	0.088	1.112	3.203
30	0.101	3.395	14.635
40	0.132	7.888	44.322
50	0.153	15.107	105.807
60	0.190	26.222	219.731
70	0.206	42.041	408.725
80	0.223	62.511	695.651
90	0.229	88.957	1109.614
100	0.261	122.576	1700.718

Table 1: All the values given above are in ms. Tested over 50 randomly generated examples for each N.

Experimental Results

Comparison between $O(n^3)$, $O(n)$ algorithms.

N	Expected $O(n)$ algorithm	$O(n^3)$ Algorithm
100	0.261	121.167
200	0.437	959.231
300	0.631	3163.430
400	0.945	7615.576
500	1.040	14999.119
600	1.318	25769.805
700	1.703	43088.605
800	1.712	66676.151
900	1.876	93234.026
1000	2.188	120870.487

Table 2: All the values given above are in ms. Tested over 20 randomly generated examples for each N.

Experimental Results

Statistics of $O(n)$ algorithm.

N	Average Time	Worst Case Time	Standard deviation
10	0.067	0.098	0.0114
100	0.277	0.737	0.127
1000	2.181	4.328	0.729
10000	19.894	48.375	8.953
100000	215.114	739.233	109.248
1000000	2259.028	4374.806	1121.353

Table 3: All the values given above are in ms. Tested over 50 randomly generated examples for each N.

Experimental Results

Number of time new point lies outside current circle.

N	no. of times oint lies outside (Avg value)
10	3.667
100	8.529
1000	14.843
10000	21.235
100000	28.273
1000000	33.727

Table 4: Tested over 50 randomly generated examples for each N.

Experimental Results

Percentage exceeding Average run time.

Average time exceeded by(%)	10	100	1000	10000	100000
10	23.529	31.373	37.255	29.412	33.333
20	9.804	25.49	29.412	25.49	29.412
50	0.0	9.804	15.686	15.686	19.608
100	0	1.961	3.922	1.961	3.922

Table 5: Tested over 250 randomly generated examples for each N.