Равномерное распределение точек на сфере

Хашин С.И.
Математический факультет Ивановского государственного университета
khash2@gmail.com

7 февраля 2017 г.

en.wikipedia.org/wiki/Thomson_problem
www.etudes.ru/ru/etudes/tomson

1 Введение

Определение 1.1. Набор точек (конфигурация) на n-мерной сфере будем называть стабильным, если ни одну из них нельзя сдвинуть так, чтобы минимальное расстояние от неё других точек увеличилось. Набор точек будем называть локально стабильным, если ни одну из них нельзя пошевелить так, чтобы минимальное расстояние до других увеличилось. Очевидно, стабильная конфигурация будет локально стабильной.

Определение 1.2. Пусть у некоторой точки из данного набора на n-мерной сфере наименьшее расстояние до других точек равно r. Точки, находящиеся на расстоянии r от данной будем называть соседними для неё. Число r будем называть радиусом окрестности точки.

Замечание 1.3. Отношение «быть соседями» не обязано быть симметричным.

Теорема 1.4. *В локально стабильной конфигурации точек у каждой из них не менее* $n+1$ *соседей.*

Определение 1.5. Радиусом R конфигурации будем называть наименьшее расстояние между её точками.

Определение 1.6. Каркасом локально стабильной конфигурации радиуса R будем называть подмножество, состоящее из точек с радиусом окрестности R.

Определение 1.7. Стабильную конфигурацию будем называть простой, если радиусы окрестностей всех точек одинаковы.

Теорема 1.8. *Каркас локально стабильной конфигурации сам образует локально стабильную конфигурацию, у которой радиус окрестности каждой точки равен* R*, то есть является простой конфигурацией.*

Доказательство. Пусть (a, b) — пара точек, находящаяся на расстоянии R. Все их соседи тоже находятся на расстоянии R и, следовательно, входят в каркас. И так далее. □

Теорема 1.9. *При* $n = 2$ *в простой конфигурации кратность каждой точки или* 3*, или* 4*, или* 5*.*

Доказательство. Уже доказано, что кратность точки не может быть меньше 3. Если кратность точки больше 5, то её соседи находятся слишком близко друг к другу. □

Определение 1.10. Две конфигурации будем называть эквивалентными, если одну можно перевести в другую с помощью ортогонального преобразования (из $O(n + 1)$).

2 Алгоритмы

Алгоритм 1: растяжение кратчайшего ребра.

Находим самое короткое ребро и растягиваем точки на его концах на *eps*. Если за 100 шагов радиус не уменьшился, то уменьшаем *eps*.

Алгоритм 2: стабилизация каждой точки.

На одном шаге алгоритма пробегаем все точки и каждую перемещаем в центр окружности, описанной вокруг её трех ближайших соседей.

Алгоритм 2а: стабилизация каждой точки.

На одном шаге алгоритма пробегаем все точки и каждую шевелим так, чтобы она оказалась на наибольшем возможном расстоянии от всех её соседей.

Теорема 2.1. *Алгоритм (2а) всегда (за бесконечное количество шагов) всегда приводит произвольную конфигурацию* A *к локально стабильной* $S(A)$.

Алгоритм 3: Растяжение каркаса.

На одном шаге алгоритма пробегаем все ребра каркаса и каждое из них последовательно растягиваем на *eps* из центра ребра. После этого делаем стабилизацию всех точек.

3 Результаты

Программа на C++: D:/W/C/sphere

Если количество точек < 5, то ответы очевидны.

?При $N = 5$ две точки располагаются на полюсах и ещё три на экваторе.

При $N = 6$ единственная стабильная конфигурация — октаэдр: две точки на полюсах и ещё четыре на экваторе.

При $N = 7$ имеется одна простая конфигурация:

```
dim= 3, nPoints=7:
Graph:
 0:    1   2   4
 1:    0   3   4   5
 2:    0   3   6
 3:    1   2   5
 4:    0   1   5   6
 5:    1   3   4   6
 6:    2   4   5
distances / 1.256870468479882
 0:        1.000 1.000 1.347 1.000 1.532 1.347
 1:  1.000       1.462 1.000 1.000 1.000 1.532
 2:  1.000 1.462       1.000 1.462 1.462 1.000
 3:  1.347 1.000 1.000       1.532 1.000 1.347
 4:  1.000 1.000 1.462 1.532       1.000 1.000
 5:  1.532 1.000 1.462 1.000 1.000       1.000
 6:  1.347 1.532 1.000 1.347 1.000 1.000
vectors:
-0.590516405489478 -0.594318507911187 -0.545963264333789
 0.415733348177840  0.066593439982578 -0.907045256293273
-0.907499035336535  0.218616770203984  0.358681207549039
-0.072037712498631  0.978581184598533 -0.192845619933914
 0.536338500543042 -0.842653125377819 -0.047715019922826
```

```
 0.921167488564967  0.324795807651856  0.214378500187611
 0.090453169717330 -0.246443399011081  0.964926875556875
 -------------------------------------------------------
```

3d-картинку в Мапл-файле можно увидеть в файле /W/TXT/journal/2016/sphere/res_dim3/007. Геометрически конфигурация выглядит так: равносторонний треугольник в центре, к каждой его стороне присоединено ещё по такому же треугольнику, всего получаем 6 точек. Седьмая точки диаметрально противоположна центру начального треугольника.

Отметим, что в файле 2-300.zip для конфигурации из 7 точек минимальное расстояние оказалось равным 1.175856652.

При $N = 8$ простая конфигурация такова:

```
dim= 3, nPoints=8:
Graph:
 0:    2   3   4   5
 1:    2   5   6   7
 2:    0   1   4   5
 3:    0   4   6   7
 4:    0   2   3   6
 5:    0   1   2   7
 6:    1   3   4   7
 7:    1   3   5   6
distances / 1.215562524130301
 0:        1.554 1.000 1.000 1.000 1.000 1.554 1.414
 1:  1.554       1.000 1.554 1.414 1.000 1.000 1.000
 2:  1.000 1.000       1.554 1.000 1.000 1.414 1.554
 3:  1.000 1.554 1.554       1.000 1.414 1.000 1.000
 4:  1.000 1.414 1.000 1.000       1.554 1.000 1.554
 5:  1.000 1.000 1.000 1.414 1.554       1.554 1.000
 6:  1.554 1.000 1.414 1.000 1.000 1.554       1.000
 7:  1.414 1.000 1.554 1.000 1.554 1.000 1.000
vectors:
-0.124559174623813 -0.946326842415472  0.298245736500954
 0.261926428842505  0.901860830756252  0.343571808828162
 0.747043084770039 -0.171967081673259  0.642147920901444
-0.519767706784792 -0.438648320679231 -0.733095615693457
 0.673250474552469 -0.527829143339171 -0.517812894737773
-0.415409196611112  0.064616633262235  0.907336701603525
 0.188133818623471  0.545998769091702 -0.816389006810171
-0.810617728770794  0.572295154997902 -0.124004650591848
 -------------------------------------------------------
```

3d-картинку в Мапл-файле можно увидеть в файле /W/TXT/journal/2016/sphere/res_dim3/008. Геометрически конфигурация представляет собой два горизонтальных квадрата, один в северном полушарии, один в южном, повернутых друг относительно друга на 45 градусов.

В файле 2-300.zip для конфигурации из 8 точек минимальное расстояние оказалось равным 1.170884645.

При $N = 9$ оптимальная конфигурация определяется однозначно:

```
dim= 3, nPoints=9:
Graph:
 0:    2   6   7   8
```

```
 1:    2   3   5   6
 2:    0   1   3   8
 3:    1   2   4   5
 4:    3   5   7   8
 5:    1   3   4   6
 6:    0   1   5   7
 7:    0   4   6   8
 8:    0   2   4   7
distances / 1.154700538378476
 0:        1.291 1.000 1.633 1.581 1.633 1.000 1.000 1.000
 1:  1.291       1.000 1.000 1.581 1.000 1.000 1.633 1.633
 2:  1.000 1.000       1.000 1.500 1.581 1.500 1.581 1.000
 3:  1.633 1.000 1.000       1.000 1.000 1.581 1.633 1.291
 4:  1.581 1.581 1.500 1.000       1.000 1.500 1.000 1.000
 5:  1.633 1.000 1.581 1.000 1.000       1.000 1.291 1.633
 6:  1.000 1.000 1.500 1.581 1.500 1.000       1.000 1.581
 7:  1.000 1.633 1.581 1.633 1.000 1.291 1.000       1.000
 8:  1.000 1.633 1.000 1.291 1.000 1.633 1.581 1.000
vectors:
 0.440624428616534  0.893008905020655 -0.091570783878726
 0.840614938597269 -0.541259240780844 -0.020123599987448
 0.539315401367989  0.191257596582151  0.820097207407088
-0.081080827704274 -0.765503840632445  0.638302255483847
-0.960929525409981 -0.263812248180803  0.083770787898516
-0.159548345920706 -0.844639137288900 -0.511007879659237
 0.421614124041691  0.072554651598749 -0.903867995306805
-0.559538855902096  0.589629008513341 -0.582455063550269
-0.481071337685105  0.668764305168768  0.566855071592377
 -----------------------------------------------------
```

Отметим, что в этом случае каждая точка имеет ровно 4 соседних, и все они на одном и том же расстоянии 1.154700538378476.

При $N = 10$ имеется две жестких конфигурации

```
dim= 3, nPoints=10:
Graph:
 0:    6   7   8
 1:    2   4   5   9
 2:    1   5   6   7
 3:    4   5   6   8
 4:    1   3   8   9
 5:    1   2   3   6
 6:    0   2   3   5
 7:    0   2   9
 8:    0   3   4   9
 9:    1   4   7   8
distances / 1.091426291403272
 0:        1.785 1.414 1.268 1.671 1.671 1.000 1.000 1.000 1.414
 1:  1.785       1.000 1.576 1.000 1.000 1.605 1.268 1.605 1.000
 2:  1.414 1.000       1.605 1.642 1.000 1.000 1.000 1.818 1.518
 3:  1.268 1.576 1.605       1.000 1.000 1.000 1.785 1.000 1.605
 4:  1.671 1.000 1.642 1.000       1.118 1.642 1.671 1.000 1.000
 5:  1.671 1.000 1.000 1.000 1.118       1.000 1.671 1.642 1.642
```

```
 6:  1.000 1.605 1.000 1.000 1.642 1.000       1.414 1.518 1.818
 7:  1.000 1.268 1.000 1.785 1.671 1.671 1.414       1.414 1.000
 8:  1.000 1.605 1.818 1.000 1.000 1.642 1.518 1.414       1.000
 9:  1.414 1.000 1.518 1.605 1.000 1.642 1.818 1.000 1.000
vectors:
-0.397981897051590 -0.813854721104546 -0.423380328493258
 0.423713759616227  0.503008549627403  0.753292140483105
 0.905094771426086 -0.340289791742127  0.254963354961574
-0.167418462455308  0.478944679712917 -0.861732587409686
-0.352937184498042  0.928509723479127  0.115347462925333
 0.750926922908179  0.596322426018253 -0.283739882073984
 0.529986726919842 -0.355559727721860 -0.769864500617575
-0.022873852545099 -0.798584785124385  0.601447527085767
-0.968979450683996  0.095526162454173 -0.227933271899739
-0.593871406178195  0.110796098434274  0.796894583678215
 -------------------------------------------------------
```

и

```
dim= 3, nPoints=10:
Graph:
 0:    2   3   5   8   9
 1:    4   5   9
 2:    0   3   7   9
 3:    0   2   6   7   8
 4:    1   6   7
 5:    0   1   8   9
 6:    3   4   7   8
 7:    2   3   4   6
 8:    0   3   5   6
 9:    0   1   2   5
distances / 1.051462224237076
 0:        1.618 1.000 1.000 1.902 1.000 1.618 1.618 1.000 1.000
 1:  1.618       1.618 1.902 1.000 1.000 1.618 1.618 1.618 1.000
 2:  1.000 1.618       1.000 1.618 1.618 1.618 1.000 1.618 1.000
 3:  1.000 1.902 1.000       1.618 1.618 1.000 1.000 1.000 1.618
 4:  1.902 1.000 1.618 1.618       1.618 1.000 1.000 1.618 1.618
 5:  1.000 1.000 1.618 1.618 1.618       1.618 1.902 1.000 1.000
 6:  1.618 1.618 1.618 1.000 1.000 1.618       1.000 1.000 1.902
 7:  1.618 1.618 1.000 1.000 1.000 1.902 1.000       1.618 1.618
 8:  1.000 1.618 1.618 1.000 1.618 1.000 1.000 1.618       1.618
 9:  1.000 1.000 1.000 1.618 1.618 1.000 1.902 1.618 1.618
vectors:
 0.357542923874702  0.407412578180216  0.840344006182991
-0.943077567208058  0.228457170200941 -0.241685795223281
 0.678146722754013  0.726837973127375 -0.108736301381280
 0.943077567207698 -0.228457170202000  0.241685795223683
-0.357542923875387 -0.407412578179369 -0.840344006183110
-0.644428790709348  0.099396302630303  0.758176700201419
 0.302986163794970 -0.929462562667924 -0.210472632775816
 0.644428790708790 -0.099396302630217 -0.758176700201904
 0.125680947199892 -0.616237448526609  0.777467495490553
-0.302986163795189  0.929462562668023  0.210472632775064
 -------------------------------------------------------
```

3.1 Жёсткие фигуры

Треугольник является жёсткой фигурой при фиксированной длине ребра.

Жёсткими фигурами будут также объединение нескольких треугольников, склеенных по ребрам.

Ребро в конфигурации будем называть внутренним, если обе грани, в которые оно входит являются треугольниками. Такие ребра можно не рисовать.

Таким образом, ребро соединяющее вершины (a, b) будет внутренним, если существуют вершины c и d такие, что существуют ребра (a, c), (b, c), (a, d) и (b, d).

Если две жёсткие фигуры касаются друг друга более, чем в одной точке, то они вместе тоже образуют жёсткую фигуру.