

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ им. ПЕТРА ВЕЛИКОГО

ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ И МЕХАНИКИ

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ

ЛАБОРАТОРНАЯ РАБОТА №3  
ПОСТРОЕНИЕ РАЗБИЕНИЯ ОБЛАСТИ  
МЕЖДУ ЗАМКНУТЫМИ КРИВЫМИ

4 КУРС, ГРУППА 3630102/60201

Студент

Д. А. Плаксин

Преподаватель

Баженов А. Н.

САНКТ-ПЕТЕРБУРГ  
2019 г.

# Содержание

1. Список иллюстраций .....	3
2. Постановка задачи .....	4
3. Теория .....	4
3.1. Радиус кривизны .....	4
3.2. Построение разбиения сетки .....	4
4. Реализация .....	5
5. Результаты .....	6
6. Обсуждение .....	7
7. Список литературы .....	7
8. Приложения .....	7

## 1 Список иллюстраций

1	Радиус кривизны трёх точек.....	4
2	Радиус кривизны .....	6
3	Радиус кривизны после медианного сглаживания.....	6
4	Разбиение сетки для сепаратрисы.....	7

## 2 Постановка задачи

- Считать данные из g-файла.
- Вычислить магнитную ось.
- Построить кривую сепаратрисы, отметить на графике магнитную ось.
- Вычислить радиусы кривизны в каждой точке сепаратрисы.
- Применить к полученным результатам медианно сглаживание.
- Построить разбиение области сепаратрисы на сетку.

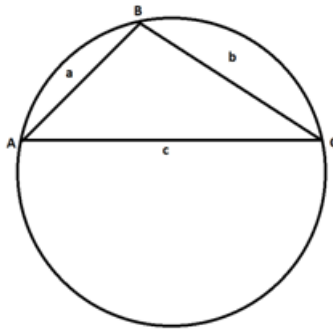
## 3 Теория

Сепаратрисы – граница плазмы, последняя замкнутая поверхность магнитного потока.

Магнитная ось – экстремум магнитного потока.

### 3.1 Радиус кривизны

Рис. 1: Радиус кривизны трёх точек



Возьмём три точки:  $A, B, C$  (рис. 1) и вычислим радиус кривизны в одной из них, например, в точке  $C$ :

По теореме синусов:  $\frac{|a|}{\sin(C)} = 2R$

Из скалярного произведения вычислим косинус угла:  $\cos(C) = \frac{(b,c)}{|b||c|}$

Тогда  $\sin(C) = \frac{|a|}{2 \sin(C)}$

### 3.2 Построение разбиения сетки

- Делим область на 2 сектора (по экватору)
- В каждом секторе находим точку с максимальным радиусом кривизны
- Найдены 4 особые точки (2 на экваторе, 2 в точках с максимальным радиусом кривизны)

- Внутри каждого сектора проводим ещё  $N$  дополнительных отрезков. Итого получено  $4N$  отрезка
- Находим у каждого из отрезков середину и соединяем их между собой
- В итоге получили разбиение пространства (сетку)

## 4 Реализация

Все задания были выполнены на языке программирования *Matlab* в среде разработки *MATLABR2014b* [\[1\]](#)

Данные из фала считаны функцией "gfile\_extractor\_1t" [\[2\]](#)

Радиус кривизны вычислялся по 3 точкам (как радиус окружности, описанной вокруг треугольника)

$R(i)$  вычисляется по трём точкам:  $A = p(i - 1)$ ,  $B = p(i)$ ,  $C = p(i + 1)$ , где  $p$  – точки сепаратрисы

Для крайних точек сепаратрисы учитывается её замкнутость

## 5 Результаты

Рис. 2: Радиус кривизны

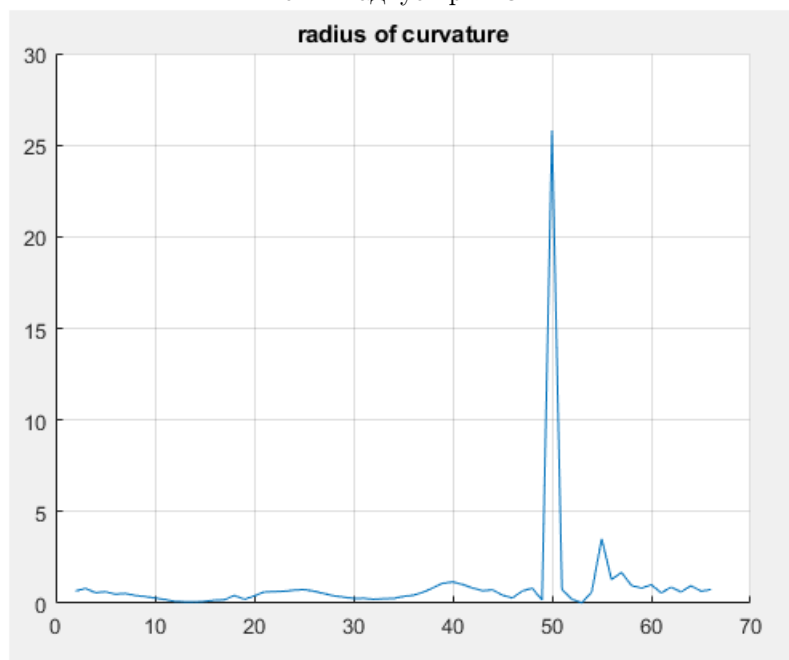


Рис. 3: Радиус кривизны после медианного сглаживания

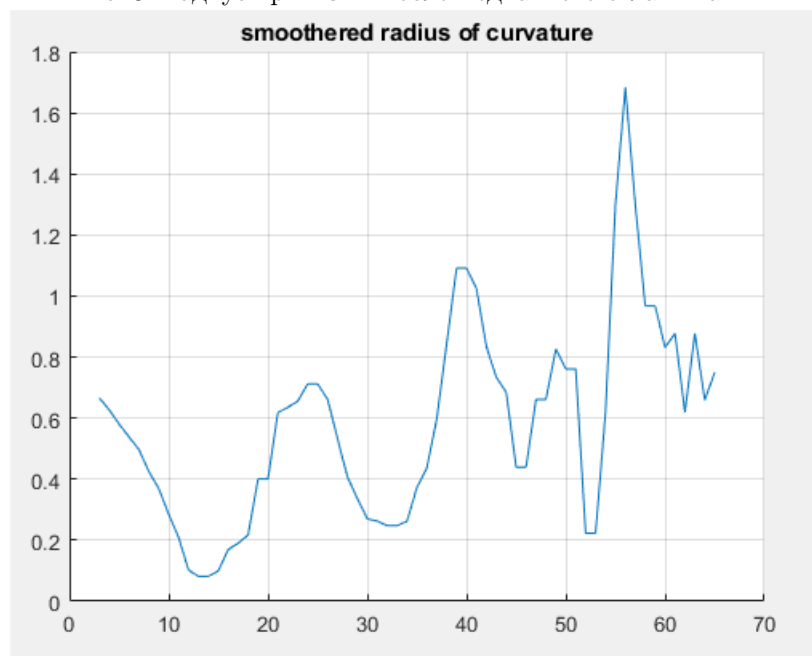
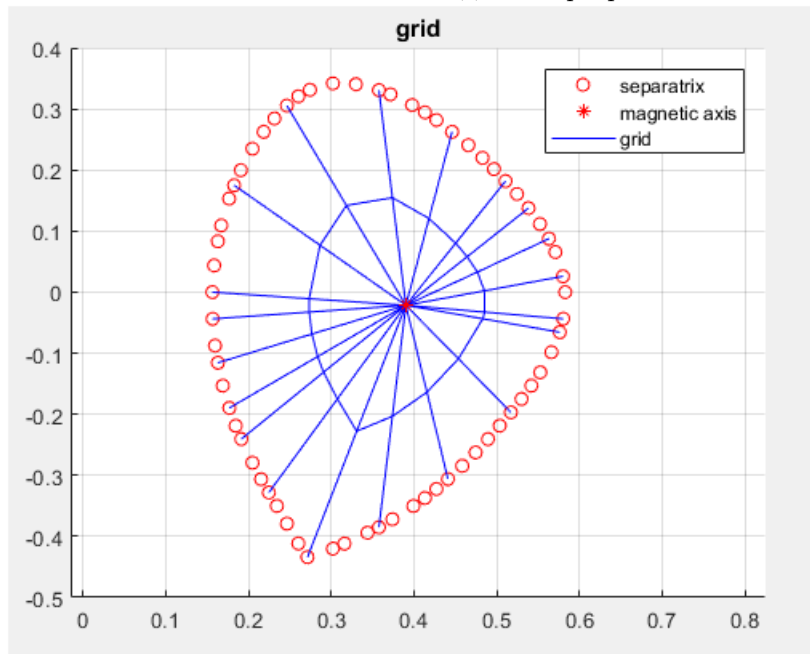


Рис. 4: Разбиение сетки для сепаратрисы



## 6 Обсуждение

После медианного сглаживания «пики» и «впадины» радиуса кривизны стали более отчетливые.

## 7 Список литературы

- [1] Документация по Матлаб: <https://www.mathworks.com/help/> (дата обращения октябрь 2019)
- [2] Код функции `g_file_extractor_1t`: <https://cloud.mail.ru/public/5o3T/4G4dD71hL> (дата обращения сентябрь 2019)
- [3] Пособие к Лабораторным работам <https://cloud.mail.ru/public/4ra6/5wwqBzMBC/LabPractics.pdf>

## 8 Приложения

Код отчёта: [https://github.com/MisterProper9000/computing-complex/tree/Lab-3-separatrix/Lab\\_3\(separatrix\)/texReport/lab3.tex](https://github.com/MisterProper9000/computing-complex/tree/Lab-3-separatrix/Lab_3(separatrix)/texReport/lab3.tex)

Код лабораторной: [https://github.com/MisterProper9000/computing-complex/tree/Lab-3-separatrix/Lab\\_3\(separatrix\)/texReport/Lab3.m](https://github.com/MisterProper9000/computing-complex/tree/Lab-3-separatrix/Lab_3(separatrix)/texReport/Lab3.m)

```

1 [flux ,RBDY,ZBDY,NBDY,R,Z,time ,rdim ,zdim] = gfile_extractor_1t(34363,
   000162, 65);
2
3 [arr , ind_arr] = min(flux);
4 [flux_min , min_j] = min (arr);
5 min_i = ind_arr(min_j);
6

```

```

7 figure()
8 grid on
9 hold on
10 plot([RBDY, RBDY(1)], [ZBDY, ZBDY(1)]);
11 magnet_axis = [R(min_j), Z(min_i)];
12 plot(magnet_axis(1), magnet_axis(2), "o");
13 title("separatrix")
14 legend("separatrix", "magnetic axis")
15
16
17 a = [RBDY(NBDY), ZBDY(NBDY)];
18 b = [RBDY(1), ZBDY(1)];
19 c = [RBDY(2), ZBDY(2)];
20 r = get_curv_radius(a, b, c);
21 for i=2:NBDY-1
22     a = [RBDY(i-1), ZBDY(i-1)];
23     b = [RBDY(i), ZBDY(i)];
24     c = [RBDY(i+1), ZBDY(i+1)];
25     tmp_r = get_curv_radius(a, b, c);
26     r = [r, tmp_r];
27 end
28
29 a = [RBDY(NBDY-1), ZBDY(NBDY-1)];
30 b = [RBDY(NBDY), ZBDY(NBDY)];
31 c = [RBDY(1), ZBDY(1)];
32 tmp_r = get_curv_radius(a, b, c);
33 r = [r, tmp_r];
34 figure()
35 grid on
36 hold on
37 plot(r);
38 title("radius of curvature")
39
40 %%
41
42 r_smoothed = medfilt1(r);
43 figure()
44 grid on
45 hold on
46 plot(r_smoothed);
47 title("smoothened radius of curvature")
48 %%
49 % sector = 1:NBDY;
50 % sectors = {sector};
51 % N = 2;
52 % while length(sectors) < 4 * N
53 %     result = {};
54 %     for i = 1:length(sectors)
55 %         tmp_sector = sectors{i};
56 %         if( length(tmp_sector) <= 4)
57 %             result = {result{:} tmp_sector};
58 %         else
59 %
60 %             [s1, s2, s3, s4] = split_sector(tmp_sector, r);
61 %             result = {result{:} s1 s2 s3 s4};
62 %         end
63 %     end
64 %     sectors = result
65 % end
66
67 N = 5;
68 init_sector = 1:NBDY;
69 [s1, s2, s3, s4] = split_sector_easy(init_sector, r_smoothed);
70 sectors = {s1, s2, s3, s4};
71 result = {};
72 for i = 1:length(sectors)

```



```

73     tmp_sector = sectors{i};
74     tmp = split_linspace(tmp_sector, N);
75     result = {result{:} tmp{:}};
76 end
77 sectors = result;
78
79 segments = [];
80 middle = magnet_axis;
81 R_segments = [];
82 Z_segments = [];
83
84 % src_segments = []
85 % for i = 1:length(sectors)
86 %     tmp_sector = sectors{i};
87 %     cur_ind = tmp_sector(1);
88 %     tmp_segment = [R_segments, mid_point(1)]
89 %     [Z_segments, mid_point(2)]
90 %
91 %     cur_point = [RBDY(cur_ind), ZBDY(cur_ind)];
92 %     plot([cur_point(1), middle(1)], [cur_point(2), middle(2)], 'b')
93 %
94 % end
95 for i = 1:length(sectors)
96     tmp_sector = sectors{i};
97     cur_ind = tmp_sector(1);
98     cur_point = [RBDY(cur_ind), ZBDY(cur_ind)];
99     mid_point = get_center(cur_point, middle);
100     R_segments = [R_segments, mid_point(1)];
101     Z_segments = [Z_segments, mid_point(2)];
102 end
103
104
105 figure()
106 grid on
107 hold on
108 plot([RBDY, RBDY(1)], [ZBDY, ZBDY(1)], "r");
109 plot(magnet_axis(1), magnet_axis(2), "or");
110 title("grid")
111
112
113 for i = 1:length(sectors)
114     tmp_sector = sectors{i};
115     cur_ind = tmp_sector(1);
116     cur_point = [RBDY(cur_ind), ZBDY(cur_ind)];
117     plot([cur_point(1), middle(1)], [cur_point(2), middle(2)], 'b')
118 end
119
120 for i = 1:length(R_segments)-1
121     plot([R_segments(i), R_segments(i + 1)], [Z_segments(i), Z_segments(i + 1)], 'b')
122 end
123 plot([R_segments(length(R_segments)), R_segments(1)], [Z_segments(length(R_segments)), Z_segments(1)], 'b')
124 plot(magnet_axis(1), magnet_axis(2), "or");
125 legend("separatrix", "magnetic axis", "grid")
126
127
128
129 function r = get_curv_radius(A, B, C)
130     a = B - A;
131     b = C - B;
132     c = C - A;
133     cos = sum(c.*b)/(norm(c)*norm(b));
134     sin = sqrt(1 - cos^2);
135     r = norm(a) / (2 * sin);
136 end

```

```

137
138
139 function [s1, s2, s3, s4] = split_sector(sector, r)
140     separator_ind = fix(length(sector) / 2);
141
142     first_sector = sector(1:separator_ind);
143     second_sector = sector(separator_ind + 1:length(sector));
144     [min_val , first_min_ind] = min(r(first_sector));
145     [max_val , first_max_ind] = max(r(first_sector));
146
147     [min_val , second_min_ind] = min(r(second_sector));
148     [max_val , second_max_ind] = max(r(second_sector));
149     second_min_ind = second_min_ind + length(first_sector);
150     second_max_ind = second_max_ind + length(first_sector);
151
152     sep_1 = min(first_min_ind, first_max_ind);
153     sep_2 = max(first_min_ind, first_max_ind);
154     sep_3 = min(second_min_ind, second_max_ind);
155     sep_4 = max(second_min_ind, second_max_ind);
156     if(sep_4 == length(sector))
157         s1 = sector(1 : sep_1);
158     else
159         s1 = sector([sep_4 + 1: length(sector), 1 : sep_1]);
160     end
161     s2 = sector(sep_1 + 1 : sep_2);
162     s3 = sector(sep_2 + 1 : sep_3);
163     s4 = sector(sep_3 + 1: sep_4);
164 end
165
166
167 function [s1, s2, s3, s4] = split_sector_easy(sector, r)
168     separator_ind = fix(length(sector) / 2);
169
170     first_sector = sector(1:separator_ind);
171     second_sector = sector(separator_ind + 1:length(sector));
172     [max_val , first_max_ind] = max(r(first_sector));
173     [max_val , second_max_ind] = max(r(second_sector));
174
175     second_max_ind = second_max_ind + length(first_sector);
176
177     sep_1 = first_max_ind;
178     sep_2 = length(first_sector);
179     sep_3 = second_max_ind;
180     sep_4 = length(sector);
181
182     s1 = sector(1 : sep_1);
183     s2 = sector(sep_1 + 1 : sep_2);
184     s3 = sector(sep_2 + 1 : sep_3);
185     s4 = sector(sep_3 + 1: sep_4);
186 end
187
188
189 function [result] = split_linspace(sector, N)
190     result = {};
191     cur_step = ceil(length(sector) / N);
192     cur_left = 1;
193     cur_right = cur_step;
194     reduce = ceil(length(sector));
195     for i = 1:N
196         if(reduce == 0)
197             return
198         end
199         tmp = sector(cur_left:cur_right);
200         result = {result{:} tmp};
201         reduce = reduce - length(tmp);
202         cur_left = cur_right + 1;

```

```

203         cur_step = ceil(reduce / (N - i));
204         cur_right = cur_right + cur_step;
205     end
206 end
207
208
209 function [result] = split_linspace(A, B)
210     result = A + (B - A) / 2;
211 end

```