**Санкт-Петербургский политехнический университет Петра Великого**

Институт прикладной математики и механики

Кафедра «Прикладная математика»

# 

# 

**Отчет по дисциплине «Вычислительные комплексы» по лабораторной работе №3  
«Построение разбиения области   
между замкнутыми кривыми»**

##### 

##### 

##### 

##### 

##### 

##### 

##### 

##### 

##### 

##### Выполнил студент группы 3630102/60201 Чепулис М.А.

Преподаватель: Баженов А.Н.

Санкт-Петербург

2019

Оглавление

[Постановка задачи 3](#_Toc23347137)

[Теория 3](#_Toc23347138)

[Радиус кривизны 3](#_Toc23347139)

[Построение разбиения сетки 3](#_Toc23347140)

[Реализация 3](#_Toc23347141)

[Результаты 4](#_Toc23347142)

[Обсуждение 6](#_Toc23347143)

[Литература 7](#_Toc23347144)

[Приложение 8](#_Toc23347145)

[Код программы на Matlab 8](#_Toc23347146)

## 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 (паутинку)

## Теория

Сепаратрисы - граница плазмы, последняя замкнутая поверхность магнитного потока

Магнитная ось – экстремум магнитного потока

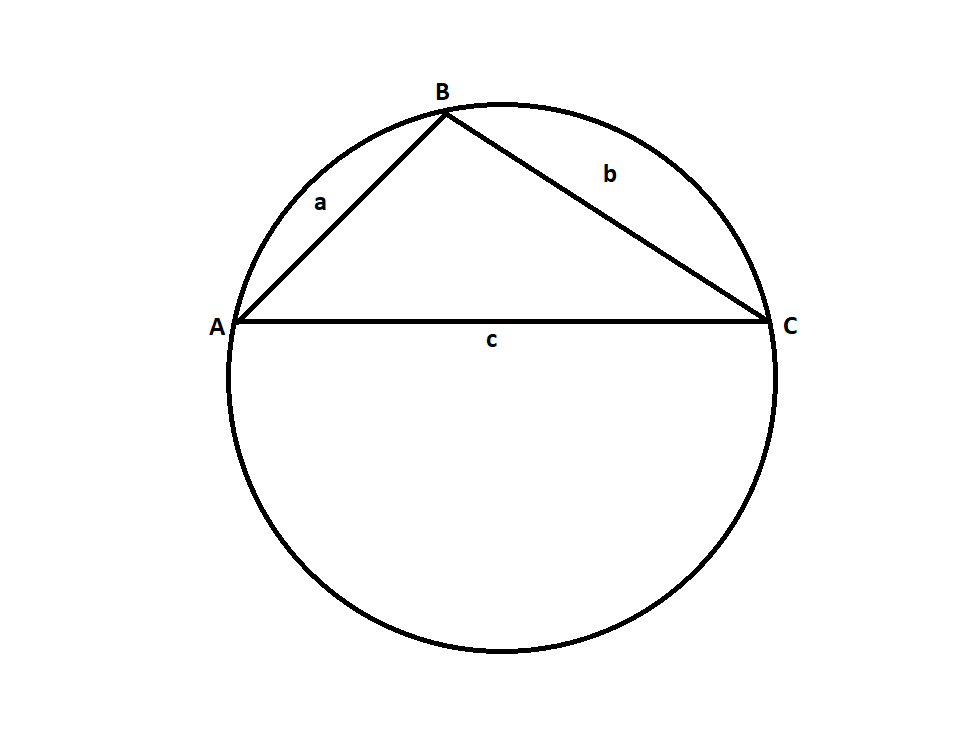
### Радиус кривизны

По тереме синусов:

Из скалярного произведения вычислим косинус угла:

Тогда

Следовательно



### 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)

## Реализация

Все задания были выполнены на языке программирования Matlab в среде разработки MATLAB R2017b [[1]](#а16)

Данные из фала считаны функцией «gfile\_extractor\_1t» [[2]](#ref2)

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

## Результаты

Расcсмаривается набор данных: 34363

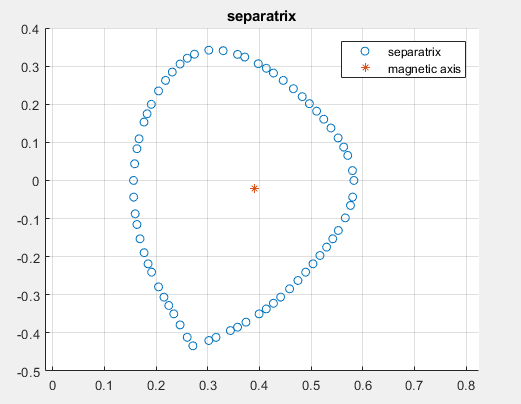


Рисунок график сепаратрисы

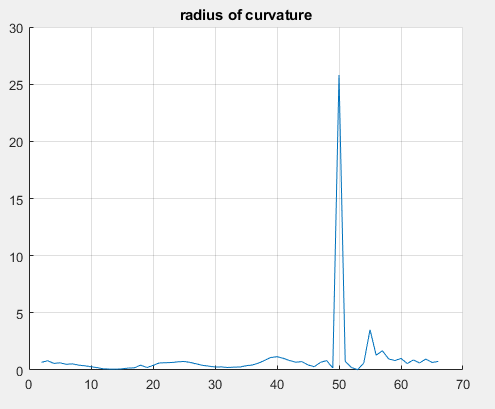


Рисунок радиус кривизны

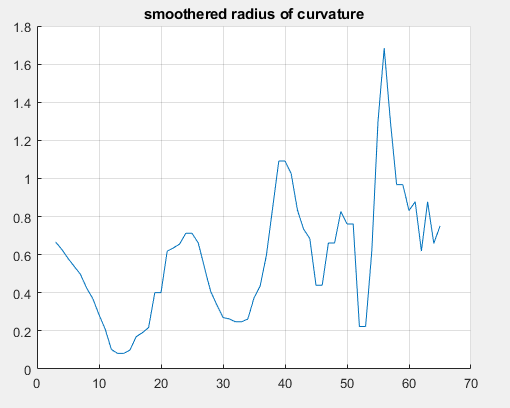


Рисунок радиус кривизны после медианного сглаживания

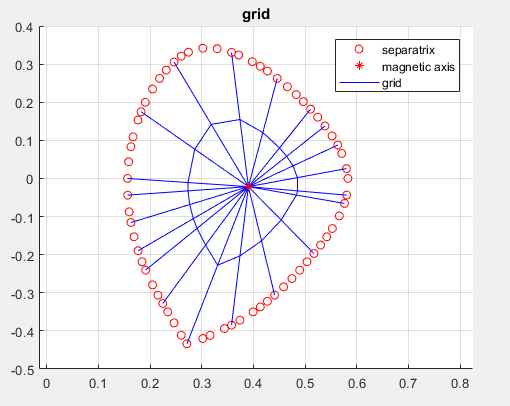


Рисунок разбиение сетки для сепаратрисы

## Обсуждение

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

Было получено разбиение пространства (сетка)

## Литература

1. Документация по Матлаб [Электронный ресурс]   
   Режим доступа: <https://www.mathworks.com/help/> (дата обращения октябрь 2019)
2. Код функции g\_file\_extractor\_1t [электронный ресурс, облачное хранилище]

Режим доступа: <https://cloud.mail.ru/public/5o3T/4G4dD71hL> (дата обращения сентябрь 2019)

1. Пособие к Лабораторным работам [электронный ресурс, облачное хранилище]

Режим доступа: <https://cloud.mail.ru/public/4ra6/5wwqBzMBC/LabPractics.pdf> (дата обращения октябрь 2019)

## Приложение

### Код программы на Matlab

#### Lab\_3\_separatrix

%%

clear all

%%

[flux,RBDRY,ZBDRY,NBDRY,R,Z,time,rdim,zdim] = gfile\_extractor\_1t(34363, 000162, 65);

%%

[arr, ind\_arr] = min(flux);

[flux\_min, min\_j] = min (arr);

min\_i = ind\_arr(min\_j);

figure()

grid on

hold on

plot([RBDRY, RBDRY(1)], [ZBDRY, ZBDRY(1)]);

magnet\_axis = [R(min\_j), Z(min\_i)];

plot(magnet\_axis(1), magnet\_axis(2), "o");

title("separatrix")

legend("separatrix", "magnetic axis")

%%

%вычисляем отдельно для первой и последней точки, учитывая замкнутость кривой

a = [RBDRY(NBDRY),ZBDRY(NBDRY)];

b = [RBDRY(1),ZBDRY(1)];

c = [RBDRY(2),ZBDRY(2)];

r = get\_curv\_radius(a, b, c);

for i=2:NBDRY - 1

a = [RBDRY(i-1),ZBDRY(i-1)];

b = [RBDRY(i),ZBDRY(i)];

c = [RBDRY(i + 1),ZBDRY(i + 1)];

tmp\_r = get\_curv\_radius(a, b, c);

r = [r, tmp\_r];

end

a = [RBDRY(NBDRY-1),ZBDRY(NBDRY-1)];

b = [RBDRY(NBDRY),ZBDRY(NBDRY)];

c = [RBDRY(1),ZBDRY(1)];

tmp\_r = get\_curv\_radius(a, b, c);

r = [r, tmp\_r];

figure()

grid on

hold on

plot(r);

title("radius of curvature")

%%

r\_smoothed = medfilt1(r);

figure()

grid on

hold on

plot(r\_smoothed);

title("smoothered radius of curvature")

%%

% sector = 1:NBDRY;

% sectors = {sector};

% N = 2;

% while length(sectors) < 4 \* N

% result = {};

% for i = 1:length(sectors)

% tmp\_sector = sectors{i};

% if( length(tmp\_sector) <= 4)

% result = {result{:} tmp\_sector};

% else

%

% [s1, s2, s3, s4] = split\_sector(tmp\_sector, r);

% result = {result{:} s1 s2 s3 s4};

% end

% end

% sectors = result

% end

N = 5;

init\_sector = 1:NBDRY;

[s1, s2, s3, s4] = split\_sector\_easy(init\_sector, r\_smoothed);

sectors = {s1, s2, s3, s4};

result = {};

for i = 1:length(sectors)

tmp\_sector = sectors{i};

tmp = split\_linspace(tmp\_sector, N);

result = {result{:} tmp{:}};

end

sectors = result;

segments = [];

middle = magnet\_axis;

R\_segments = [];

Z\_segments = [];

% src\_segments = []

% for i = 1:length(sectors)

% tmp\_sector = sectors{i};

% cur\_ind = tmp\_sector(1);

% tmp\_segment = [R\_segments, mid\_point(1)]

% [Z\_segments, mid\_point(2)]

%

% cur\_point = [RBDRY(cur\_ind),ZBDRY(cur\_ind)];

% plot([cur\_point(1), middle(1)], [cur\_point(2), middle(2)], 'b')

%

% end

for i = 1:length(sectors)

tmp\_sector = sectors{i};

cur\_ind = tmp\_sector(1);

cur\_point = [RBDRY(cur\_ind),ZBDRY(cur\_ind)];

mid\_point = get\_center(cur\_point, middle);

R\_segments = [R\_segments, mid\_point(1)];

Z\_segments = [Z\_segments, mid\_point(2)];

end

figure()

grid on

hold on

plot([RBDRY, RBDRY(1)], [ZBDRY, ZBDRY(1)], "r");

plot(magnet\_axis(1), magnet\_axis(2), "or");

title("grid")

for i = 1:length(sectors)

tmp\_sector = sectors{i};

cur\_ind = tmp\_sector(1);

cur\_point = [RBDRY(cur\_ind),ZBDRY(cur\_ind)];

plot([cur\_point(1), middle(1)], [cur\_point(2), middle(2)], 'b')

end

for i = 1:length(R\_segments)-1

plot([R\_segments(i), R\_segments(i + 1)], [Z\_segments(i), Z\_segments(i + 1)], 'b')

end

plot([R\_segments(length(R\_segments)), R\_segments(1)], [Z\_segments(length(R\_segments)), Z\_segments(1)], 'b')

plot(magnet\_axis(1), magnet\_axis(2), "or");

legend("separatrix", "magnetic axis", "grid")

%%

#### Get\_curv\_radius

function r = get\_curv\_radius(A, B, C)

a = B - A;

b = C - B;

c = C - A;

cos = sum(c.\*b)/(norm(c)\*norm(b));

sin = sqrt(1 - cos^2);

r = norm(a) / (2 \*sin);

end

#### split\_sector

function [s1, s2, s3, s4] = split\_sector(sector, r)

separator\_ind = fix(length(sector) / 2); % делим нацело на 2 части

first\_sector = sector(1:separator\_ind);

second\_sector = sector(separator\_ind + 1:length(sector));

[min\_val , first\_min\_ind] = min(r(first\_sector));

[max\_val , first\_max\_ind] = max(r(first\_sector));

[min\_val , second\_min\_ind] = min(r(second\_sector));

[max\_val , second\_max\_ind] = max(r(second\_sector));

second\_min\_ind = second\_min\_ind + length(first\_sector); % т.к. индекс получаем относительно массива r(second\_sector) (36 элементов), а хотим иметь абсолютный

second\_max\_ind = second\_max\_ind + length(first\_sector);

sep\_1 = min(first\_min\_ind, first\_max\_ind);

sep\_2 = max(first\_min\_ind, first\_max\_ind);

sep\_3 = min(second\_min\_ind, second\_max\_ind);

sep\_4 = max(second\_min\_ind, second\_max\_ind);

if(sep\_4 == length(sector))

s1 = sector(1 : sep\_1);

else

s1 = sector([sep\_4 + 1: length(sector), 1 : sep\_1]);

end

s2 = sector(sep\_1 + 1 : sep\_2);

s3 = sector(sep\_2 + 1 : sep\_3);

s4 = sector(sep\_3 + 1: sep\_4);

end

#### split\_sector\_easy

function [s1, s2, s3, s4] = split\_sector\_easy(sector, r)

separator\_ind = fix(length(sector) / 2); % делим нацело на 2 части

first\_sector = sector(1:separator\_ind);

second\_sector = sector(separator\_ind + 1:length(sector));

[max\_val , first\_max\_ind] = max(r(first\_sector));

[max\_val , second\_max\_ind] = max(r(second\_sector));

second\_max\_ind = second\_max\_ind + length(first\_sector);% т.к. индекс получаем относительно массива r(second\_sector) (36 элементов), а хотим иметь абсолютный

sep\_1 = first\_max\_ind;

sep\_2 = length(first\_sector);

sep\_3 = second\_max\_ind;

sep\_4 = length(sector);

s1 = sector(1 : sep\_1);

s2 = sector(sep\_1 + 1 : sep\_2);

s3 = sector(sep\_2 + 1 : sep\_3);

s4 = sector(sep\_3 + 1: sep\_4);

end

#### split\_linspace

function [result] = split\_linspace(sector, N)

result = {};

cur\_step = ceil(length(sector) / N);

cur\_left = 1;

cur\_right = cur\_step;

reduse = ceil(length(sector));

for i = 1:N

if(reduse == 0)

return

end

tmp = sector(cur\_left:cur\_right);

result = {result{:} tmp};

reduse = reduse - length(tmp);

cur\_left = cur\_right + 1;

cur\_step = ceil(reduse / (N - i));

cur\_right = cur\_right + cur\_step;

end

end

#### get\_center

function [result] = split\_linspace(A, B)

result = A + (B - A) / 2;

end