|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

Студент Сусликов Даниил Владиславович

Группа ИУ7-45Б

Тип практики Производственная

Название предприятия ПАО «Радиофизика»

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

подпись, дата фамилия, и.о.

Руководитель практики **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

подпись, дата фамилия, и.о.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2020 г.

Индивидуальное задание

Создание модели измерителя (РЛС) для решения следующих задач:

* обнаружение сигналов, отраженных от объектов наблюдения (ОН), попавших в рабочую зону РЛC,
* оценка дальности, скорости и углового положения ОН, отраженные сигналы от которых превысили порог,
* формирование единичных замеров.

Оглавление

Введение………………………………………………………………...…..…4

Основная часть…………………………………………………..…………....5

Заключение…………………………………………………………….……..14

Приложение……………………………….……………………..…………..15

Введение

Поставленные задачи для исполнения ранее описанного задания:

1. Определить азимут и угол места нормали, при которых исследуемый объект наблюдения попадёт в главный лепесток (ГЛ) диаграммы направленности (ДН) передающего устройства.
2. Определить азимут и угол места нормали, при которых исследуемый объект наблюдения попадёт в ГЛ ДН принимающего устройства.
3. Определить интервал оцифровки, при котором будет обнаружен отражённый сигнал.
4. Сформировать единичные замеры.
5. Построить визуализацию текущего процесса
6. Выполнить симуляцию цифровой обработки сигнала (ЦОС), построить графики при различных «окнах»

Основная часть

Первая задача - «Определить азимут и угол места нормали, при которых исследуемый объект наблюдения попадёт в ГЛ ДН передающего устройства». Для этого понадобилось воспользоваться методом algorithm, который по данным сообщения передающей РЛС и её координатам точки стояния позволил вычислить азимут и угол места нормали. Для этого изначально потребовалось полученные данные перевести в нужную систему координат — МССК, при помощи алгоритма, описанного ниже.

Пересчёт

ГСК:

φрлс, λрлс, hрлс

ГЦСК:

xрлс, yрлс, zрлс

ГЦСК:

x, y, z

Vx, Vy, Vz

МПСК:

x, y, z

Vx, Vy, Vz

Пересчёт

Пересчёт

МССК:

Di, εi, βi

VD, Vε, Vβ

Δt = (Dон + с\*(tпр- ti)) /

(c - Vон i )

в ГЦСК:

Vотр = Vi +Aотр\* Δt

(для Vx, Vy, Vz )

в ГЦСК:

Aотр = Ai +(Ai+1- Ai) \* Δt

/ (Ti+1 — Ti)

(для Ax, Ay, Az)

Пересчёт

МПСК:

Xотр, Yотр, Zотр

в ГЦСК:

Xотр = Xi +Vотр\* Δt

(для X, Y, Z)

Пересчёт

МCСК:

|εцу — εон| <= ½ \* θε

|βцу — βон| <= ½ \* θβ

МCСК:

D, ε, β

Проверка на попадание

Получив координаты, нужно непосредственно провести проверку на попадание в ГЛ ДН передающего устройства. Это нужно для того, чтобы понять попадает ли объект в область действия посылаемых РЛС-передатчиком импульсов. Для этого использовался метод isInMainLeafSourceAnt.

Координаты и скорости ОН на момент отражения ЗС пересчитываются из ГЦСК в МПСК с началом системы координат, совпадающим с электрическим центром передающей антенны РЛС.

Рассчитывается ширина ГЛ ДН передающей антенны по уровню половинной мощности по формулам

где:

, – размер передающей антенны в азимутальной и угломестной плоскости,

– длина волны,

с = 3\*108 м/с – скорость света в вакууме,

– средняя частота для сигнала ЛЧМ, для сигналов БЧМ и ФКМ.

Направление ГЛ ДН передающей антенны по азимуту и углу места пересчитывается из БСК в МССК:

, (18)

.

И сама проверка:

Если оба условия выполнены, то объект находится в области действия передатчика.

Вторая задача - «Определить азимут и угол места нормали, при которых исследуемый объект наблюдения попадёт в ГЛ ДН принимающего устройства». Аналогично первой задачи нужно провести проверки на попадание в область действия. Это нужно для того, чтобы понять, может ли РЛС-приемник принять сигнал, отражённый от объекта. Метод isInMainLeafReceiveAnt выполняет задачу проверки.

Но так как работа происходит лишь с одной РЛС, это возможно благодаря тому, что РЛС попеременно может быть как и передатчиком, посылающим импульсы, так и приёмник, принимающим отражённый от объекта сигнал, то эта проверка аналогична проверке приёмника, описанной выше.

Третья задача - «Определить интервал оцифровки, при котором будет обнаружен отражённый сигнал». Из-за того, что РЛС является то приёмником, то передатчиком, РЛС принимает сигналы лишь в определенные интервалы. Для того, чтобы сигнал попал в интервал оцифровки, время обнаружения сигнала должно входить в интервал времени его приёма. Описанную выше проверку выполняет метод isInDigitalizationTime.

Выполняется расчёт времени прихода tобн сигналов, отражённых от ОН и попавших в ГЛ ДН приемной антенны , и для дальнейшего рассмотрения выбираются те из них, которые попадают в интервал оцифровки от tпр до tпр+tинт:

где:

,

– время отражения сигнала от ОН,

tпр – время начала оцифровки,

tинт – продолжительность интервала оцифровки,

– скоростное смещение дальности.

Четвёртая задача - «Сформировать единичные замеры». На основе сообщений, полученных от передатчика и приёмника, была составлена структура, содержащая единичные замеры, благодаря методу fillMeasurement. Структура включает в себя

* время отражения
* дальность
* радиальную скорость
* угол места
* азимут
* амплитуду сигнала
* амплитуду шума
* номер объекта в ФЦО
* номер зонда
* номер ДН
* девиацию
* опорную скорость
* длительность импульса
* начальную несущую частоту
* частоту манипуляции ФКМ сигнала
* код вида сигнала
* ЭПР

Пятая задача - «Построить визуализацию текущего процесса». Для визуализации процесса были использованы методы drawEarth, drawTrajectories, drawRlsViewPyramid. Модель Земли была построена методом drawEarth. Траектории движения объектов - методом drawTrajectories, который считывает файлы, содержащие координаты объектов наблюдения в определенное время, и строит их траектории движения. А лучи направления подающихся РЛС импульсов были отображены методом drawRlsViewPyramid.

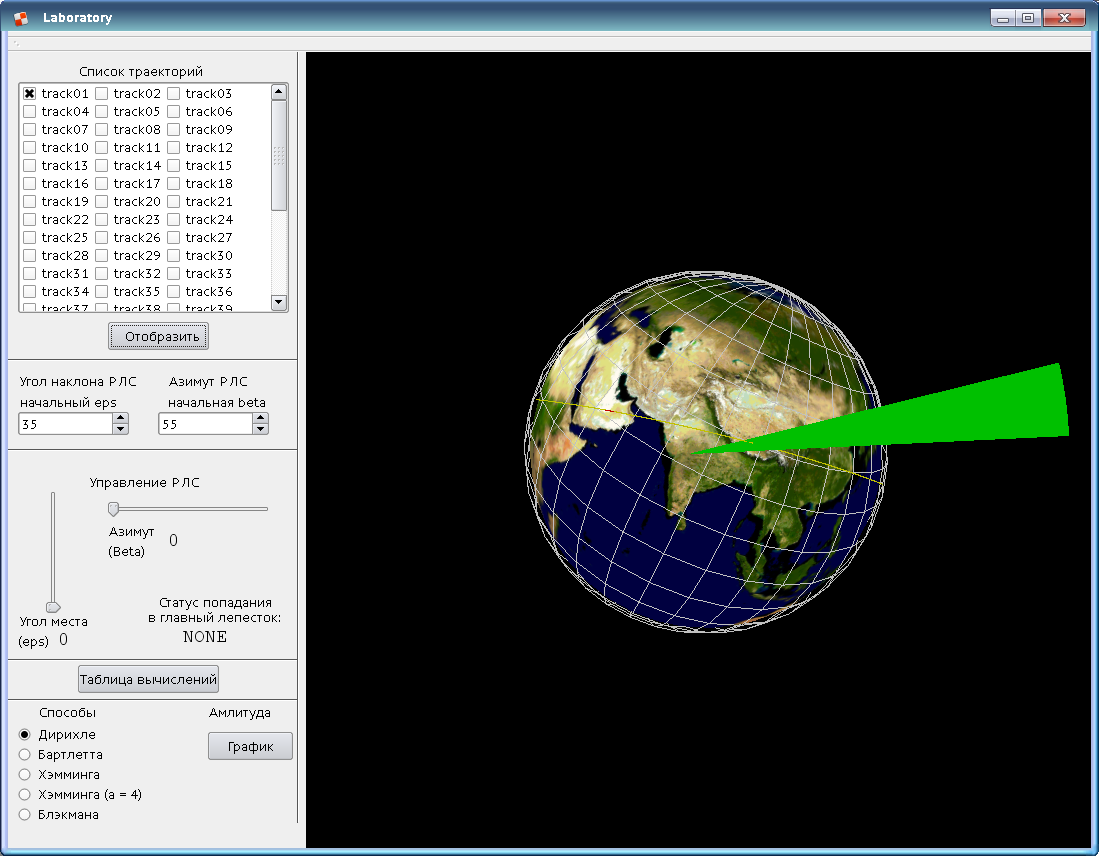


Рис. 1. Визуализация модели

Зелёным отображена область посылаемых сигналов РЛС

Желтым — траектории объектов наблюдения

Красным — вычисленная траектория объекта наблюдения

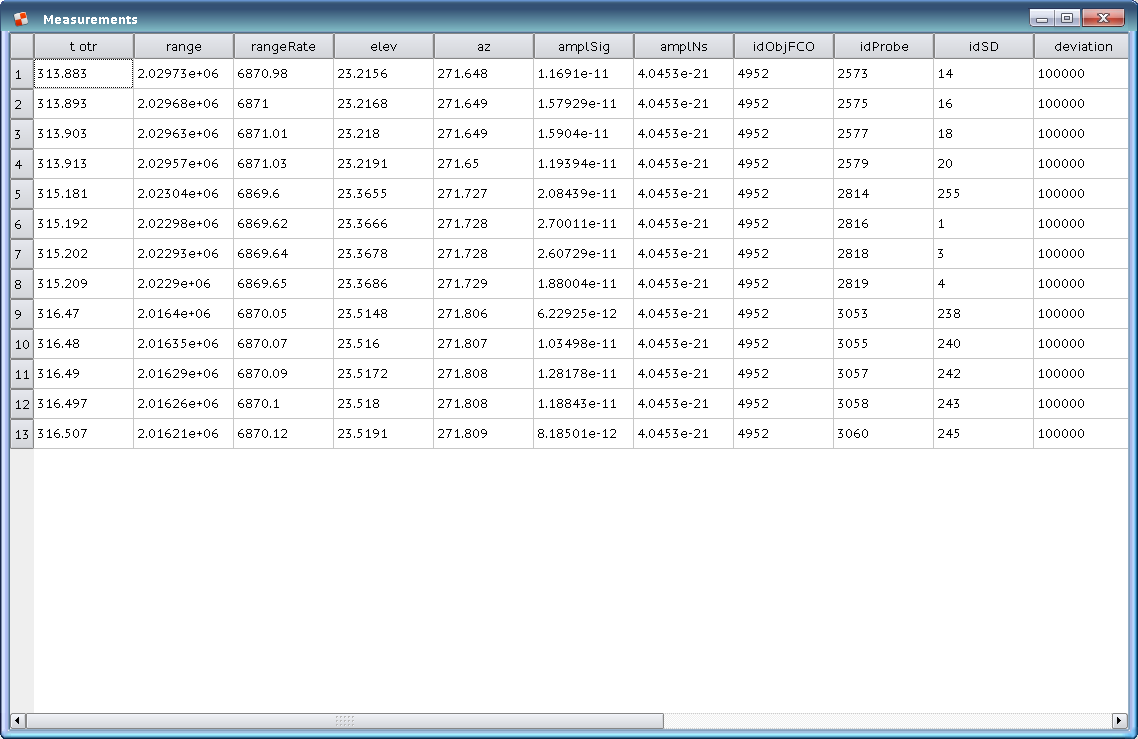
Так же был настроен вывод данных единичный замеров в отдельную таблицу.

Рис. 2. Таблица единичных замеров

Шестая задача - «Выполнить симуляцию ЦОС, построить графики при различных «окнах»». Для выполнения симуляции потребовались данные сообщения приёмника и сам алгоритм симуляции ЦОС, реализованный в методе cosSimulation. Перед работой алгоритма пользователю был представлен выбор способов вычисления амплитуды сигнала - «окна».

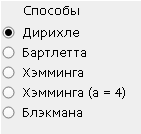


Рис. 3. Меню выбора способа вычисления амплитуды

В ходе алгоритма симуляции ЦОС производится вычисление амплитуд сигнала в определенный момент времени по способу, заранее выбранному пользователем. Полученные значения сохраняются и визуализируются в виде графика, используя метод makePlot.

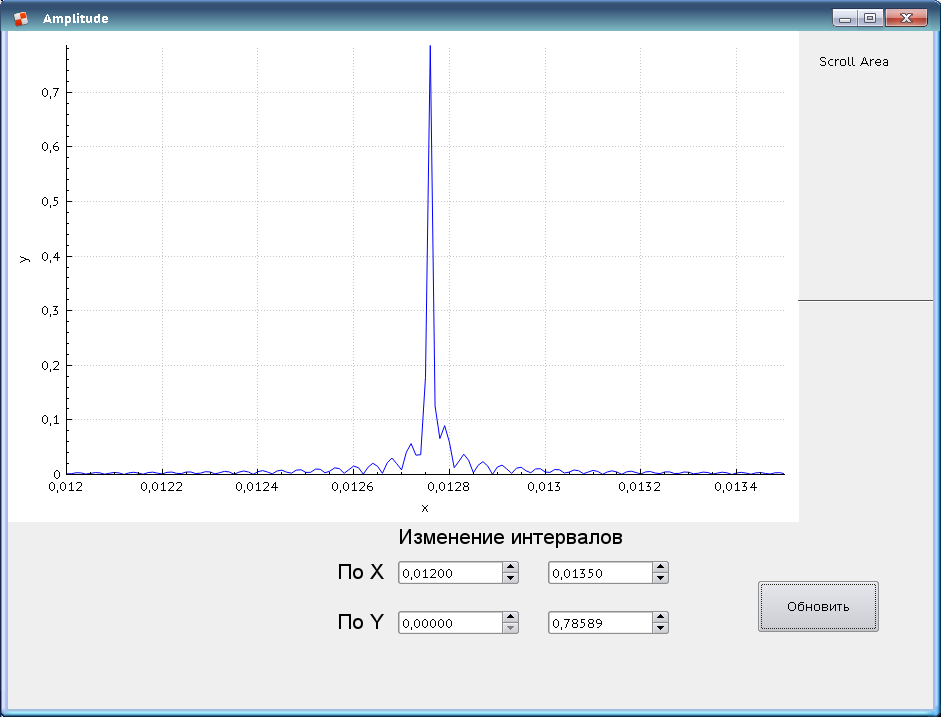


Рис. 4. График зависимости амплитуд отсчёта от временного положения

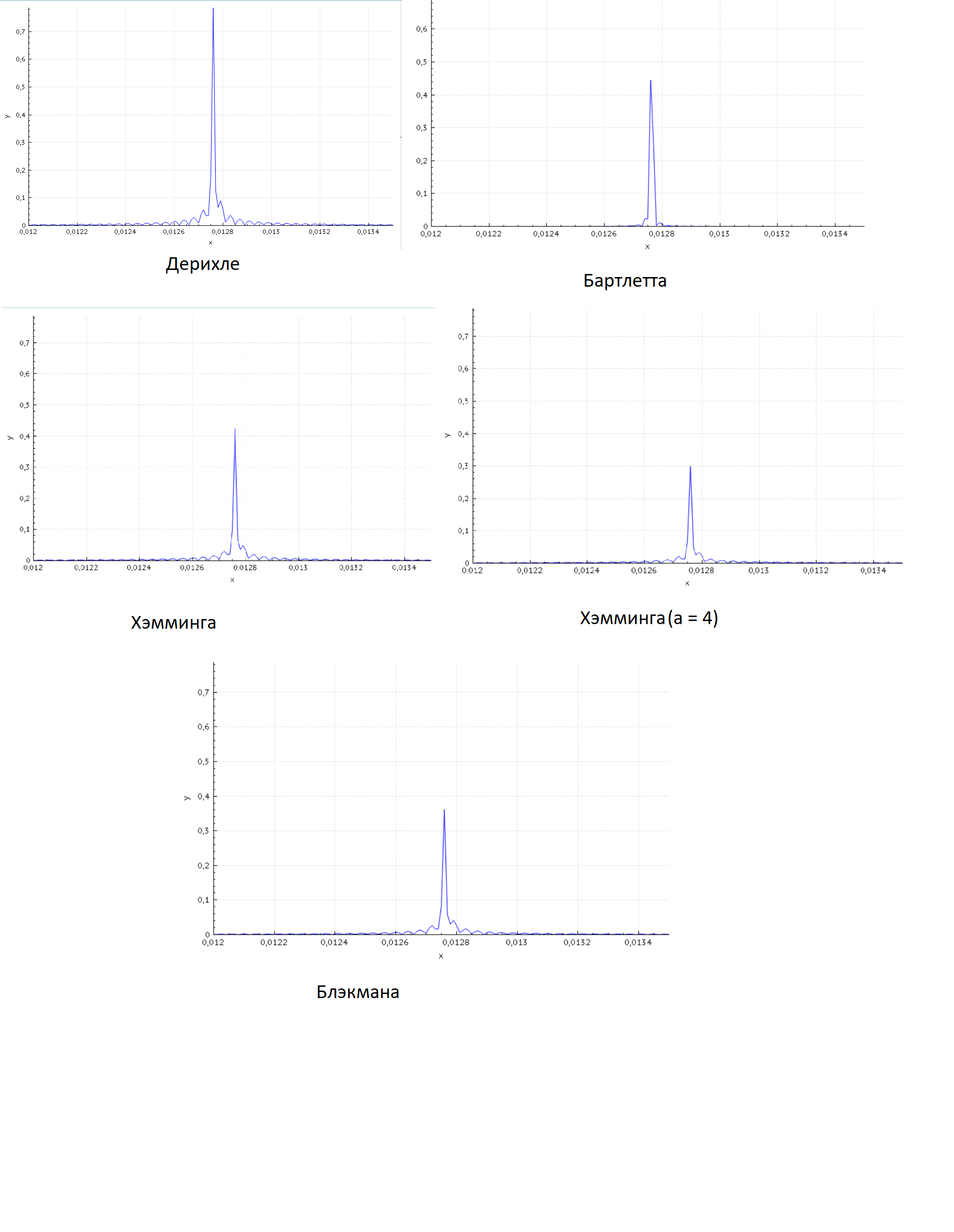


Рис. 5. Сравнение графиков зависимости амплитуд отсчёта от временного положения при использовании разных «окон»

Графики отличаются величиной амплитуды, боковых лепестков и шириной.

Все графики верны, нужно лишь умножать значения амплитуды на коэффициент соответствующий выбранному «окну».

Заключение

По итогу практики была разработана модель для решения следующих задач:

* обнаружение сигналов, отраженных от объектов наблюдения (ОН), попавших в рабочую зону РЛC,
* оценка дальности, скорости и углового положения ОН, отраженные сигналы от которых превысили порог,
* формирование единичных замеров.

Визуализация модели была произведена при помощи библиотек Qt и OpenGL.

Был прописан алгоритм для симуляции цифровой обработки сигналов (ЦОС).

Была произведена оценка дальности до объекта наблюдения и получены его координаты относительно точки стояния РЛС.

Приложение

Структуры:

typedef struct table

{

int ntr\_in\_VKO;

double t;

//long long ntr;

double x;

double y;

double z;

double vx;

double vy;

double vz;

double ax;

double ay;

double az;

double x\_or;

double y\_or;

double z\_or;

double pizl;

int id;

}table;

typedef struct RadiationControl

{

int head;

double tMod;

long probeID;

long servType;

double sinAE;

double sinAB;

long objObsrvNum;

long long timeStartRad;

long probeDuration;

int signalType;

double probeMeanFreq;

double probeDeviation;

double probeFaseFreq;

double pcmPolynomV;

double pcmPolynoH;

bool probeV;

bool probeH;

}RadiationControl;

typedef struct ReceivingControl

{

int head;

double tMod;

long probeID;

long servType;

long recvDiagramNum;

double sinAE;

double sinAB;

long objObsrvNum;

long long timeStartRad;

// ProbeSignalType signalType;

long long probeDuration;

int signalType;

double probeMeanFreq;

double probeDeviation;

double probeFaseFreq;

double pcmPolynomV;

double pcmPolynoH;

double radSpeed;

double quantizFreq;

double timeStartRecv;

double recvDuration;

bool sepDetect;

double falseAlarmP;

double minThreshold;

}ReceivingControl;

typedef struct measurement

{

double time;

double range;

double rangeRate;

double elev;

double az;

double amplSig;

double amplNs;

unsigned int idObjFCO;

unsigned int idProbe;

unsigned int idSD;

double deviation;

double refVel;

double duration;

double initFreq;

double phaseFreq;

int signalType;

double RCS;

}meauserement;

void algorithm(string& radControlFile, string &recControlFile, string& trajectoryFile, double phi, double lyambda, double h);

void isInMainLeafSourceAnt(RadiationControl& rad\_control, Vector3& mssk\_coords);

void isInMainLeafReceiveAnt(ReceivingControl& receiving\_control, Vector3& mssk\_coords);

Методы:

void isInDigitalizationTime(RadiationControl &rad\_control,\

ReceivingControl& receiving\_control,\

Vector3& mssk\_coords, Vector3& v\_otr, Vector3& rls\_coords\_gsk);

void fillMeasurement(table& cur\_situation,\

RadiationControl& rad\_control,ReceivingControl rec\_control,\

Vector3& mssk\_coords, Vector3& v\_r, double amplitude, double noize\_amplitude,\

double rcs, measurement& measure);

void drawEarth();

void drawTrajectories();

void drawRlsViewPyramid(RadiationControl &rad\_control,\

Vector3& mssk\_coords,\

Vector3& rls\_coords\_gcsk,\

Vector3& rls\_coords\_gsk);

void cosSimulation(RadiationControl& rad\_control, ReceivingControl& rec\_control,\

Vector3& mssk\_coords, Vector3& v\_r, ofstream& file);

void makePlot();