**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ   
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

|  |  |  |
| --- | --- | --- |
| СОГЛАСОВАНО  Руководитель проекта, старший преподаватель факультета компьютерных наук департамента «Программная инженерия»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. В. Пантюхин  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2020 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Шилов  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2020 г. | |
| |  |  | | --- | --- | | Подп. и дата |  | | Инв. № дубл. |  | | Взам. Инв. № |  | | Подп. и дата |  | | Инв. № подл. | **RU.17701729.04.13-01 ПЗ 01-1** | | **Программа симуляции и визуализации работы радиолокатора Пояснительная записка**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.04.13-01 ПЗ 01-1-ЛУ**  **Исполнитель**  Студент группы БПИ 199  \_\_\_\_\_\_\_\_\_\_\_/Н. Д. Зубарева/  «\_\_\_»\_\_\_\_\_\_\_\_\_\_2020 г.    **Москва 2020** | | |  | |

**УТВЕРЖДЕНО**  
**RU.17701729.04.13-01 ПЗ 01-1-ЛУ**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Подп. и дата |  | | Инв. № дубл. |  | | Взам. Инв. № |  | | Подп. и дата |  | | Инв. № подл. | **RU.17701729.04.13-01 ПЗ 01-1**  \_\_\_\_\_\_\_\_ | | **Программа симуляции и визуализации работы радиолокатора**  **Пояснительная записка**  **RU.17701729.04.13-01 ПЗ 01-1**  **Листов 34**  Москва 2020  СОДЕРЖАНИЕ  [1. ВВЕДЕНИЕ 3](#_Toc41100639)  [1.1 Наименование программы 3](#_Toc41100640)  [1.2 Документы, на основании которых ведется разработка 3](#_Toc41100641)  [2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ 4](#_Toc41100642)  [2.1 Назначение программы 4](#_Toc41100643)  [2.1.1 Функциональное назначение 4](#_Toc41100644)  [2.1.2 Эксплуатационное назначение 4](#_Toc41100645)  [3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ 5](#_Toc41100646)  [3.1 Постановка задачи на разработку программы 5](#_Toc41100647)  [3.2 Описание алгоритма и функционирования программы 5](#_Toc41100648)  [3.2.1 Описание алгоритма программы 5](#_Toc41100649)  [3.2.2 Обоснование выбора алгоритма решения задачи 9](#_Toc41100650)  [3.3 Описание и обоснование выбора метода организации входных и выходных данных 10](#_Toc41100651)  [3.3.1 Описание метода организации входных и выходных данных 10](#_Toc41100652)  [3.3.2 Обоснование выбора метода организации входных и выходных данных 10](#_Toc41100653)  [3.4 Описание и обоснование выбора организации хранения данных 10](#_Toc41100654)  [3.5 Описание и обоснование выбора состава технических и программных средств 10](#_Toc41100655)  [3.5.1 Требования к составу и параметрам технических средств 10](#_Toc41100656)  [3.5.2 Требования к информативной и программной совместимости 10](#_Toc41100657)  [4. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 12](#_Toc41100658)  [4.1 Предполагаемая потребность 12](#_Toc41100659)  [4.2 Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами 12](#_Toc41100660)  [СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 13](#_Toc41100661)  [ПРИЛОЖЕНИЕ 1 14](#_Toc41100662)  [ПРИЛОЖЕНИЕ 2 15](#_Toc41100663)  [ПРИЛОЖЕНИЕ 3 16](#_Toc41100664)  [ПРИЛОЖЕНИЕ 4 17](#_Toc41100665)  [ПРИЛОЖЕНИЕ 5 41](#_Toc41100666) |  |

1. **ВВЕДЕНИЕ**
   1. **Наименование программы**

**Наименование программы на русском языке:** «Программа симуляции и визуализации работы радиолокатора»

**Наименование программы на английском языке:** «Software for Radar Simulation and Visualization»

* 1. **Документы, на основании которых ведется разработка**

Приказ декана факультета компьютерных наук И.В. Аржанцева "Об утверждении тем, руководителей курсовых работ студентов образовательной программы «Программная инженерия» факультета компьютерных наук" № 2.3-02/1112-04 от 11.12.2019

1. **НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ**
   1. **Назначение программы**
      1. Функциональное назначение

Программа симулирует и визуализирует работу радиолокатора по принятию и обработке генерируемых сигналов на фоне генерируемых помех.

* + 1. Эксплуатационное назначение

Программа наглядно демонстрирует работу радиолокационного устройства и помогает лучше понять процесс его работы, может использоваться как для изучения непосредственно методов обнаружения целей в радиолокаторах, так и для симуляции условий в задачах связанных с работой таких радиолокаторов.

1. **ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ**
   1. **Постановка задачи на разработку программы**

Написать программу, которая будет принимать на вход параметры симуляции и на их основе проводить симуляцию и визуализацию работы радиолокатора по обработке информации и поиску целей, выдавая результат в виде изображений и текстовых файлов.

* 1. **Описание алгоритма и функционирования программы**
     1. Описание алгоритма программы
        1. Описание общей схемы работы системы

Глобально работа программы разделена на несколько блоков:

* задание параметров симуляции;
* расчет гипотез о сигнале;
* генерация матрицы наблюдений;
* расчет обратной корреляционной матрицы;
* расчет статистики наблюдений.

Каждая часть работы после задания параметров сопровождается визуализацией получаемых данных. Кроме того, программа гибкая в смысле возможности пропуска этапа при наличии готовых данных и данных, полученных на предыдущих этапах (рис. 1).

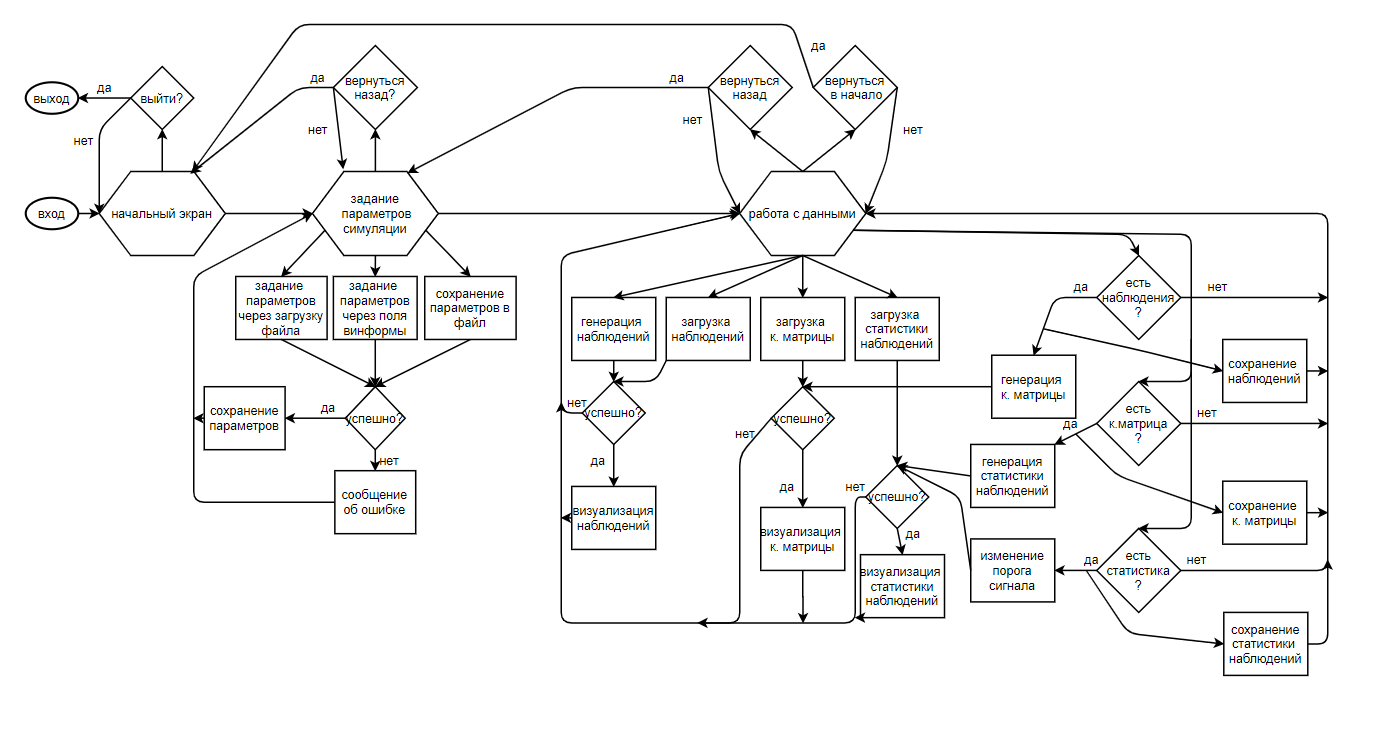


Рисунок 1 – Блок-схема программы.

* + - 1. Описание алгоритма задания параметров симуляции

Параметры симуляции (приложение 2) задаются посредством интерфейса WindowsForms, при этом не используются какие-либо заслуживающие пояснения алгоритмы, подробную информацию о входных данных можно найти в пункте 3.3 данного документа.

* + - 1. Описание алгоритма расчета гипотез о сигнале

Для симуляции работы радиолокатора необходимо сформировать предположения о возможных сигналах по возможным вариациям времени прихода и вариациям угловых координат. Для этого рассчитываются комплексные матрицы U (гипотезы по углам) и S (гипотезы по времени), действительные и мнимые компоненты которых сохраняются соответственно в файлы Ux.txt, Uy.txt, Sx.txt и Sy.txt. Для вычисления U и S используются формулы (1), (2), (3) из [1].

Матрица задается элементами:

(1)

где

M – количество блоков, на которые разбивается вся антенная решетка, равен Mu\*Mv (параметры симуляции),

m = 1,…,M,

p, q – индексы блока антенной решетки при нумерации блоков по строкам (рис. 2), p = 1,…,, q = 1,…,, m = (p-1)\*+ q,

Рисунок 2 – Разбиение антенной решетки на блоки на примере Mu = Mv = 8.

, - предполагаемое положение сигнала по одному и другому углам, изменяются от -1/12 до 1/12 радиана с шагом 1/48 радиана,

, - ширина главного лепестка диаграммы направленности модуля по соответствующим углам, соотносятся с параметрами симуляции Um и Vm,

(2)

- диаграмма направленности модуля,

*–* математическая константа.

Матрица определяется элементами

*,* (3)

где

- целый момент прихода сигнала,

- длительность сигнала.

Реализация этих формул осуществляется в функции CreateFiles.m, там вычисляются матрицы U и S и сохраняются в файлы.

* + - 1. Описание алгоритма генерации матрицы наблюдений

Настоящий радиолокатор с определенной частотой по времени фиксировал бы своей антенной решеткой сигналы, приходящие от целей и шумов, и по некоторому количеству векторов с каждого блока формировал бы матрицу наблюдений, соответственно для симуляции необходимо генерировать подобные наблюдения программными средствами. Для этого используются функции, написанные на языке MATLAB на основе функций из [1], конкретно CreateFiles.m, вызывающая getMatrNabl.m, которая формирует матрицу Y размером M на N (параметр симуляции, количество векторов наблюдений, снятых с каждого блока антенной решетки, по которым формируется статистика).

Матрица Y состоит из комплексных чисел, вычисленных с помощью функции getGauss.m, которая генерирует матрицу требуемых размеров, состоящую из случайных комплексных чисел с нормальным распределением модулей мнимых и вещественных компонент от 0 до 9 методами генерации случайных чисел MATLAB. Затем несколько матриц, полученных таким образом, складываются с подобной матрицей, у которой увеличены значения на месте предполагаемой цели (параметры симуляции u, v). В результате получается матрица требуемого размера, в которой присутствуют сигналы от шумов и предполагаемых целей, что имитирует наблюдения, которые бы получал настоящий радиолокатор, и требуется в качестве входных параметров для симуляции его работы.

* + - 1. Описание алгоритма расчета обратной корреляционной матрицы

После генерации наблюдений и предположений о сигнале происходит расчет корреляционной матрицы. Ее оценка формируется для сигналов и шумов в каждой области пространства (обозревается пространство 60о на 60о градусов, а эффективная разрешающая способность одного модуля антенной решетки составляет примерно 10о на 10о градусов, поэтому пространство разумно разделять на 6\*6 = 36 областей). При этом в каждой области берется количество блоков, равное параметру симуляции pq, то есть в результате матрица R содержит M\*M\*pq элементов.

Вычисление матрицы производится с помощью класса RadarOperations.cs библиотеки RadarLib по реккурентной формуле из [1]:

, (4)

где

,

,

y – элемент матрицы наблюдений Y, сгенерированной выше,

,

- единичная матрица подходящей размерности,

- коэффициент регуляризации,

H – знак эрмитового сопряжения (комплексное сопряжение и транспонирование).

В итоге получается матрица требуемого размера, заполненная комплексными числами, которая разделяется и сохраняется в два файла для действительных и мнимых компонент соответственно.

Стоит отметить, что также проводится нахождение корреляционной матрицы функцией Yinvert.m в рамках вызова функции CreateFiles.m для сравнения полученных результатов.

* + - 1. Описание алгоритма расчета статистики наблюдений

Основываясь на предположениях о положении и времени прихода сигнала и на рассчитанной обратной корреляционной матрице, строится статистика наблюдений L. Она зависит от углов и времени прихода сигнала, и ее значение в конкретной точке определяет, есть ли в этой точке пространства сигнал или нет: если значение больше заданного параметра симуляции h – порогового значения сигнала, то делается вывод о наличии сигнала в точке.

В точке пространства значение статистики вычисляется по формуле из [1]:

, (5)

где

вычислены выше.

Матрица L в программе вычисляется функцией CreateL.m и сохраняется в текстовый файл Lfile.txt.

* + - 1. Описание алгоритма визуализации обратной корреляционной матрицы

Визуализация корреляционной матрицы проводится с помощью функции VisR.m, которая вызывается из функции CreatePicture.m и в которую передаются массив с рассчитанной корреляционной матрицей и другие параметры, необходимые для установления размерности. Далее делаются срезы матрицы по времени через фиксацию индексов в измерении матрицы, и на каждом срезе значения матрицы выводятся в окно визуализации, за счет чего создается анимация. Визуализация мнимых и действительных компонент проходит в двух разных окнах одно после другого. Функция также настраивает внешний вид графика, в том числе палитру и ее легенду. Визуализация проводится как после расчета матрицы, так и после добавления существующих файлов.

* + - 1. Описание алгоритма визуализации сигналов

Алгоритм визуализации сигналов общий и применяется на этапе визуализации пробных сигналов по углам и по времени, при визуализации статистики наблюдений и при финальном отображении статистики наблюдений с измененным параметром порога значения сигнала.

За него отвечает функция VisFiles.m, которая принимает на вход параметры для установления размерности матрицы, матрицу и параметр порога значения сигнала. Происходит нахождение минимального и максимального значений сигналов, проверка того, что порог сигнала находится в допустимых пределах, далее формируется массив, состоящий только из элементов изначального массива, значения которых превосходят порог, происходит перерасчет их положения в сферические координаты по формулам из [2]:

, (6)

, (7)

, (8)

где , - это углы, соответствующие каждому сигналу в массиве,

– стандартная функция MATLAB для вычисления единичной матрицы требуемого размера.

Затем настраиваются подписи, легенда графика и цветовая схема, и происходит отрисовка точек в соответствии с их сигналами, точка с максимальным сигналом помечается значком звездочки. Визуализация происходит при вычислении и добавлении существующих файлов.

* + 1. Обоснование выбора алгоритма решения задачи

Алгоритм, описанный в пункте 3.2.1.2 был выбран, так как WindowsForms – распространенный и достаточно универсальный формат для интерфейсов в том числе вычислительных программ, он совмещает удобство и понятность для пользователя и сравнительную легкость для разработчика, так как не требует сложной верстки, как, например, WPF.

Алгоритм из пункта 3.2.1.3 был выбран в силу его относительно несложной программной реализации, кроме того, он был подробно описан в [1], в то время как аналогичные методы не обладают какими-либо особыми свойствами, которые давали бы им преимущество.

Алгоритм, описанный в 3.2.1.4, был реализован как наиболее вычислительно простой алгоритм, без каких-либо усложнений выполняющий требуемые операции и создающий данные с приемлемыми статистическими характеристиками.

Алгоритм, описанный в пункте 3.2.1.5 был выбран после сравнения его с другим методом, также описанным в [1], а именно реализующим параллельные вычисления матрицы на графической карте с помощью расширений CUDA для языка C, позволяющих задавать потоки вычислений, напрямую взаимодействуя с графическим процессором. Хотя последняя реализация и является более быстрой и оптимизированной, выбор был сделан в пользу алгоритма, не привязанного к вычислениям на видеокарте, так как это заставило бы предъявлять более жесткие требования к программному оборудованию и сильно снизило бы универсальность данной программы.

Алгоритм из пункта 3.2.1.6 данного документа был взят, так как он удобен и хорошо подходит для решения данной задачи, хотя вероятнее всего существуют алгоритмы расчета статистики, более точные и математически сложные, но конкретно в этой работе выбранная реализация дает хорошие результаты, соответствующие предъявляемым к ним требованиям, так что мне показалось разумным остановиться на ней.

Алгоритм, описанный в пункте 3.2.1.7, был выбран в связи с тем, что он осуществляет визуализацию именно в том формате, который более всего подходит конкретно для этой задачи, а именно в виде анимации срезов массива.

Алгоритм визуализации, описанный в пункте 3.2.1.8, является адаптированным для данной задачи алгоритмом из [1] и представляет собой довольно распространенную визуализацию объектов в сферических координатах, настроенную под отображение специфичной для данной задачи информации.

Стоит отметить, что большая часть вычислений и визуализация выполняются с использованием программных средств MATLAB, хотя есть аналоги, в том числе вычисления и визуализация на языке C#. Однако осуществлять математические вычисления на неспецифичных для этого языках, как C#, зачастую неудобно в связи с меньшей гибкостью и приспособленностью для этого языка, а среди прочих сред для математических вычислений MATLAB выделяется немаловажной в данном случае функцией создания на основе функций библиотек, которые возможно прикреплять к проектам на .Net, что и было сделано в данной работе. При этом прикрепление функций в виде библиотеки имеет ощутимые преимущества над альтернативами, как, например, запуск из программы отдельных исполняемых файлов, так как классы библиотеки полноценно включаются в проект на C# в целом являются более распространенными элементами разработки.

* 1. **Описание и обоснование выбора метода организации входных и выходных данных**
     1. Описание метода организации входных и выходных данных

Входные данные представляют собой параметры симуляции, задаваемые через интерфейс приложения либо через текстовый файл. Ввод посредством интерфейса осуществляется с помощью текстовых полей, соответствующих каждому параметру и проверяющих корректность вводимых данных.

Выходные данные, согласно требованиям технического задания, организованны в виде текстовых файлов, которые пользователь может сохранить в удобное для себя место с помощью диалогового окна, и в графическом виде.

* + 1. Обоснование выбора метода организации входных и выходных данных

Описанный выше формат входных данных был выбран в соответствии с требованиями, предъявленными к входным данным, также формат текстовых файлов удобен и понятен пользователю, написание файла необходимого вида не требует особых знаний или умений, что делает текстовые файлы предпочтительнее, чем, например, xml-документы. Ввод через интерфейс также был реализован на основе наиболее часто используемых и доступных для понимания способов вводить данные в WindowsForms, а именно через текстовые поля.

Выходные данные организованы в тестовом виде в силу требований технического задания, а также в силу специфики данных, так как представлять их в каком-либо другом формате было бы неудобно. Для отрисовки изображений были выбраны средства MATLAB, так как это позволяет совместить математические вычисления и их легкую визуализацию с относительной простотой работы. Графики MATLAB содержат широкий набор функций и хорошо реализуют отображение информации подобной тематики, в то же время они несложные и взаимодействие с ними интуитивно понятно.

* 1. **Описание и обоснование выбора организации хранения данных**

Внутри программы данные хранятся в виде текстовых файлов в папке с исполняемым файлом, это наиболее удобный способ их хранения, так как такой формат можно с легкостью читать и проверять на корректность как в приложении на C#, так и внутри MATLAB-функций.

* 1. **Описание и обоснование выбора состава технических и программных средств**

Программные и технические средства выбраны на основании минимальных требований, необходимых для работы программы.

* + 1. Требования к составу и параметрам технических средств
* Процессор архитектуры Intel или AMD x86-64 с частотой не менее 1 ГГц;
* Монитор с разрешением 800х600 точек и более;
* Не менее 4 Гб ОЗУ;
* Не менее 5 Гб на жёстком диске;
* Клавиатура, мышь.
  + 1. Требования к информативной и программной совместимости
* Windows XP или более поздняя версия операционной системы (32-разрядные или 64-разрядные);
* Установленный .NET Framework версии 4.5 и выше;
* Доступ к сети интернет или установленный MATLAB Runtime;
* Программа должна быть написана на языке программирования C#. Допускается использование других языков и библиотек для реализации вычислений.

1. **ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ**
   1. **Предполагаемая потребность**

Данная программа предназначена для обучения работе методов обнаружения целей на фоне помех в автоматизированных антенных решетках, будет использована в рамках дисциплин «НИС «Нейросетевые технологии»», «Применение нейросетевых технологий». Может быть использована другими образовательными программами, а также любым пользователем для самообразования.

* 1. **Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами**

Полные аналоги данной программы в открытом доступе отсутствуют. Ближайшим аналогом является набор скриптов MATLAB [1] используемых на дисциплине «Применение нейросетевых технологий». В сравнении с ними программа:

* Обладает наглядным пользовательским интерфейсом;
* Не требует установки системы MATLAB;
* Позволяет задавать большее число параметров целей и помех.

Другие характеристики программы:

* Является бесплатной;
* Может свободно распространяться в информационном пространстве;
* Не имеет срока годности;
* Не требует наличия специфичного технического оборудования или жестких требований к оператору.

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Черемисин О.П., Пантюхин Д.В., Подложнюк В.Д. Оценка технической эффективности применения современных графических ускорителей в задаче обнаружения целей на фоне помех в автоматизированных фазированных антенных решетках // Информатизация и связь. 2009, № 1, С. 151–153.
2. [S. Swapna Kumar](https://www.google.com/search?sxsrf=ALeKk02ouaRuI0qqC1Bh8QM8zKMwvbzA_Q:1590176724380&q=matlab:+easy+way+of+learning+s.+swapna+kumar&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEw2LMipyq0qUoJyTY2zs8sMjLVkspOt9JPy87P1y4syS0pS8-LL84uyrRJLSzLyixax6uQmluQkJlkppCYWVyqUJ1Yq5Kcp5KQmFuVl5qUrFOspFJcnFuQlKmSX5iYW7WBlBACHqDVkcAAAAA&sa=X&ved=2ahUKEwiEnqX2ncjpAhXnlIsKHU_bABIQmxMoATAYegQIBhAD&sxsrf=ALeKk02ouaRuI0qqC1Bh8QM8zKMwvbzA_Q:1590176724380), [S. V. B. Lenina](https://www.google.com/search?sxsrf=ALeKk02ouaRuI0qqC1Bh8QM8zKMwvbzA_Q:1590176724380&q=matlab:+easy+way+of+learning+s.+v.+b.+lenina&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEw2LMipyq0qUoJyiw1Ms-ItsrVkspOt9JPy87P1y4syS0pS8-LL84uyrRJLSzLyixax6uQmluQkJlkppCYWVyqUJ1Yq5Kcp5KQmFuVl5qUrFOsplOkpJOkBRYD8xB2sjAC6PYrBcAAAAA&sa=X&ved=2ahUKEwiEnqX2ncjpAhXnlIsKHU_bABIQmxMoAjAYegQIBhAE&sxsrf=ALeKk02ouaRuI0qqC1Bh8QM8zKMwvbzA_Q:1590176724380). MATLAB: Easy Way of Learning, 2016, 452.

ПРИЛОЖЕНИЕ 1

**ТЕРМИНОЛОГИЯ**

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| **Корреляционная матрица** | Это набор данных и соответствующие им два файла: корреляционная матрица принятого сигнала (с шумами и помехами), разбивается на два файла действительных и комплексных составляющих. |
| **Логгер** | Это текстовое поле, в котором записываются сообщения о начале, окончании и результате выполнения каких-либо процессов. |
| **Наблюдения** | Это набор данных, либо шесть файлов, в которых они содержатся: сформированные предположения о сигнале по углам, предположения о сигнале по времени и сгенерированная на основе параметров симуляции матрица наблюдений (каждый блок информации разбивается на два файла действительных и комплексных составляющих). |
| **Параметры симуляции** | Это набор переменных, используемых в качестве входных данных для симуляции работы радиолокатора. |
| **Пороговое значение сигнала** | Значение сигнала в области, превышение значения статистики над которым значением h означает наличие сигнала в предполагаемой точке, начинающегося в предполагаемое время, а также соответствующий параметр симуляции. |
| **Статистика наблюдений** | Это набор данных и соответствующий им файл: статистика всех предположений о наличии сигнала в области, фактически являющаяся трехмерной функцией, зависящей от двух углов прихода сигнала и времени задержки, и файл, в который эта статистика записана. |

ПРИЛОЖЕНИЕ 2

**ПАРАМЕТРЫ СИМУЛЯЦИИ**

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Um | Ширина главного лепестка диаграммы направленности модуля по одному углу. |
| Vm | Ширина главного лепестка диаграммы направленности модуля по другому углу. |
| du | Сдвиг до центра просматриваемого сектора по одному углу. |
| dv | Сдвиг до центра просматриваемого сектора по другому углу. |
| u | Предполагаемое положение цели по одному углу. |
| v | Предполагаемое положения цели по другому углу. |
| a | Мощности приходящего сигнала. |
| Tc | Начала прихода сигнала. |
| up1 | Положение помех по одному углу. |
| up2 | Положение помех по одному углу. |
| vp1 | Положение помех по другому углу. |
| vp2 | Положение помех по другому углу. |
| Ap1 | Мощность помех. |
| Ap2 | Мощность помех. |
| gamma | Параметр гамма. |
| \_mu | Коэффициент регуляризации корреляционной матрицы. |
| H | Пороговоге значения мощности сигнала. |
| N | Количества векторов наблюдений, по которым формируется статистика. |
| Mu | Количество модулей, на которые разбивается антенная решетка по длине. |
| Mv | Количество модулей, на которые разбивается антенная решетка по ширине. |
| pq | Количество блоков в 36 областях, по которым составляется корреляционная матрица. |

ПРИЛОЖЕНИЕ 3

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ**

**Классы приложения Coursework**

*Таблица 1 - Классы приложения Coursework*

|  |  |
| --- | --- |
| **Класс** | **Назначение** |
| Program | Точка входа приложения, начало работы винформ. |
| Homescreen | Начальная форма, отображает информацию о программе и об авторе. |
| Input | Форма для задания параметров симуляции, входных данных. |
| Operations | Форма для осуществления различных этапов симуляции, для работы с данными и их визуализации. |

**Классы библиотеки RadarLib**

*Таблица 2 - Классы библиотеки RadarLib*

|  |  |
| --- | --- |
| **Название** | **Назначение** |
| RadarOperations | Класс для методов, реализующих нахождение обратной корреляционной матрицы. |

**Классы подключенной библиотеки MatlabFuncs**

*Таблица 3 - Классы библиотеки MatlabFuncs*

|  |  |
| --- | --- |
| **Название** | **Назначение** |
| **Funcs** | Автосгенерированный на основе матлаб-функций класс, содержащий функции, производящие генерацию данных и расчет статистики, основная вычислительная часть программы. |

ПРИЛОЖЕНИЕ 4

**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ МЕТОДОВ, ПОЛЕЙ И СВОЙСТВ**

**Описание членов класса Program**

*Таблица 4 – Описание членов класса Program*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Методы** | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | *Аргументы* | *Назначение* |
| Main | public | void | - | Главная точка входа приложения, открывает первую форму |

**Описание членов класса Homescreen**

*Таблица 5 – Описание членов класса Homescreen*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Методы** | | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | *Аргументы* | | *Назначение* |
| Homescreen | public | void | - | | Конструктор формы, в котором инициализируются компоненты формы и задается из внешний вид. |
| bMain\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку перехода к главному экрану, на котором отображается название приложения и информация об авторе. |
| bInfo\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку перехода к информации, делает видимым текст, поясняющий работу данного приложения, и невидимым остальной текст. |
| bProceed\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку перехода к работе, скрывает эту форму и открывает форму для задания параметров симуляции на основе этой формы. |
| bExit\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку выхода, закрывает данную форму и приложение. |
| GetAllControlsOfType <T> | private | List<Control> | Control | | Метод, позволяющий получить все элементы определенного типа на форме для их более удобной настройки. |
| **Поля** | | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | | *Назначение* | |
| funcs | private | MatlabFuncs.Funcs | | Экземпляр класса функций библиотеки с функциями матлаб, создается в самом начале работы, чтобы размер формы не перерасчитывался в дальнейшем из-за форматирования. | |
| backColor | private | Color | | Поля для цветов, которые определяют внешний вид формы. | |
| panelColor | private | Color | | Поля для цветов, которые определяют внешний вид формы. | |
| hoverColor | private | Color | | Поля для цветов, которые определяют внешний вид формы. | |
| textColor | private | Color | | Поля для цветов, которые определяют внешний вид формы. | |

**Описание членов класса Input**

*Таблица 6 – Описание членов класса Input*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Методы** | | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | *Аргументы* | | *Назначение* |
| Input | public | - | Homescreen, MatlabFuncs.Funcs | | Конструктор формы, определяющий внешний вид формы и вызывающий метод, приводящий форму к дефолтному состоянию. |
| Begin | private | void | - | | Метод, задающий внешний вид формы и значения параметров симуляции их дефолтными значениями, тут же происходит настройка текста надписей и подсказок для пользователя. |
| bGet\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку загрузки параметров из файла. Появляется диалоговое окно для выбора файла, затем вызывается метод получения параметров из файла. Обрабатываются исключения, которые могут возникнуть при чтении файла. |
| GetFromFile | private | void | string[] | | Метод для добавления параметров симуляции из файла. Для считанных строк по одной на каждую переменную в требуемом порядке их следования вызывается метод изменения значения параметра с проверкой корректности. Если данных не хватает, оставшиеся параметры не меняются. |
| ChangeValueI | private | void | ref int, string, TextBox, Label, int, int | | Метод для изменения значения параметра типа int. Вызывается метод проверки корректности задаваемого значения, если данные корректны, меняется поле параметра, надпись в связанном с ним текстовом поле и поле ввода, иначе значение не изменяется, а в поле ввода возвращается предыдущее корректное значение. |
| ChangeValueF | private | void | ref float, string, TextBox, Label, float, float | | Метод для изменения значения параметра типа float. Вызывается метод проверки корректности задаваемого значения, если данные корректны, меняется поле параметра, надпись в связанном с ним текстовом поле и поле ввода, иначе значение не изменяется, а в поле ввода возвращается предыдущее корректное значение. |
| CheckValidityFailedF | public | bool | string, out float, float, float | | Метод проверки корректности вводимых данных для параметра типа float. Проверяется то, что принимаемая на вход строка не пуста, что она парсится в число требуемого типа, что значение этого числа лежит в требуемых границах. Если что-то из этого не выполняется, возвращается булевое значение, говорящее о том, что попытка изменения некорректна. |
| CheckValidityFailedI | public | bool | string, out int, int, int | | Метод проверки корректности вводимых данных для параметра типа int. Проверяется то, что принимаемая на вход строка не пуста, что она парсится в число требуемого типа, что  значение этого числа лежит в требуемых границах.  Если что-то из этого не выполняется,  возвращается булевое значение, говорящее о том, что попытка изменения некорректна. |
| InN\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра N. Вызывается метод изменения значения параметра с проверкой корректности. |
| InMv\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Mv. Вызывается метод изменения значения параметра с проверкой корректности. |
| InMu\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Mu. Вызывается метод изменения значения параметра с проверкой корректности. |
| In\_mu\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра mu. Вызывается метод изменения значения параметра с проверкой корректности. |
| InUm\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Um. Вызывается метод изменения значения параметра с проверкой корректности. |
| InVm\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Vm. Вызывается метод изменения значения параметра с проверкой корректности. |
| Indu\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра du. Вызывается метод изменения значения параметра с проверкой корректности. |
| Indv\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра dv. Вызывается метод изменения значения параметра с проверкой корректности. |
| Inu\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра u. Вызывается метод изменения значения параметра с проверкой корректности. |
| Inv\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра v. Вызывается метод изменения значения параметра с проверкой корректности. |
| Ina\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра a. Вызывается метод изменения значения параметра с проверкой корректности. |
| InTc\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Tc. Вызывается метод изменения значения параметра с проверкой корректности. |
| InH\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра H. Вызывается метод изменения значения параметра с проверкой корректности. |
| Inup1\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра up1. Вызывается метод изменения значения параметра с проверкой корректности. |
| Inup2\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра up2. Вызывается метод изменения значения параметра с проверкой корректности. |
| Invp1\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра vp1. Вызывается метод изменения значения параметра с проверкой корректности. |
| Invp2\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра vp2. Вызывается метод изменения значения параметра с проверкой корректности. |
| InAp1\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Ap1. Вызывается метод изменения значения параметра с проверкой корректности. |
| InAp2\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра Ap2. Вызывается метод изменения значения параметра с проверкой корректности. |
| Ingamma\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра gamma. Вызывается метод изменения значения параметра с проверкой корректности. |
| Inpq\_Leave | private | void | object, EventArgs | | Обработчик покидания текстового поля для задания параметра pq. Вызывается метод изменения значения параметра с проверкой корректности. |
| bSave\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку сохранения параметров. Появляется окно выбора директории, затем файл записывается в выбранное место. Также обрабатываются исключения, которые могут возникнуть при записи. |
| bProceed\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку перехода к  дальнейшей работе с данными, происходит  запись файла входных параметров, создание  и открытие новой формы на основе текущей, сокрытие текущей формы. |
| bExit\_Click | private | void | object, FormClosingEventArgs | | Обработчик кнопки закрытия формы. |
| Input\_FormClosing | private | void | object, FormClosingEventArgs | | Обработчик закрытия формы, который открывает предыдущую форму. |
| GetAllControlsOfType<T> | private | List<Control> | Control | | Метод, позволяющий получить все элементы определенного типа на форме для их более удобной настройки. |
| **Поля** | | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | | *Назначение* | |
| backColor | private | Color | | Поле для цвета фона формы. | |
| panelColor | private | Color | | Поле для цвета панели и кнопок формы. | |
| hoverColor | private | Color | | Поле для цвета элемента формы, над которым находится курсор. | |
| textColor | private | Color | | Поле для цвета текста формы. | |
| prevForm | private | Homescreen | | Поле для ссылки на предыдущую форму, которая откроется в случае закрытия этой. | |
| funcs | private | MatlabFuncs.Funcs | | Поле для ссылки на экземпляр класса функций матлаб. | |
| Um | private | float | | Поле для параметра ширины главного лепестка диаграммы направленности модуля по одному углу. | |
| Vm | private | float | | Поле для параметра ширины главного лепестка диаграммы направленности модуля по другому углу. | |
| du | private | float | | Поле для параметра сдвига до центра просматриваемого сектора по одному углу. | |
| dv | private | float | | Поле для параметра сдвига до центра просматриваемого сектора по другому углу. | |
| u | private | float | | Поле для параметра предполагаемого положения цели по одному углу. | |
| v | private | float | | Поле для параметра предполагаемого положения цели по другому углу. | |
| a | private | float | | Поле для параметра мощности приходящего сигнала. | |
| Tc | private | float | | Поле для параметра начала прихода сигнала. | |
| up1 | private | float | | Поле для параметра положения помех по одному углу. | |
| up2 | private | float | | Поле для параметра положения помех по одному углу. | |
| vp1 | private | float | | Поле для параметра положения помех по другому углу. | |
| vp2 | private | float | | Поле для параметра положения помех по другому углу. | |
| Ap1 | private | float | | Поле для параметра мощности помех. | |
| Ap2 | private | float | | Поле для параметра мощности помех. | |
| gamma | private | float | | Поле для параметра гамма. | |
| \_mu | private | float | | Поле для параметра регуляризации корреляционной матрицы. | |
| H | private | float | | Поле для параметра порогового значения мощности сигнала. | |
| N | private | int | | Поле для параметра количества векторов наблюдений, по которым формируется статистика. | |
| Mu | private | int | | Поле для параметра количества модулей, на которые разбивается антенная решетка по длине. | |
| Mv | private | int | | Поле для параметра количества модулей, на которые разбивается антенная решетка по ширине. | |
| pq | private | int | | Поле для параметра количества блоков в 36 областях, по которым составляется корреляционная матрица. | |

**Описание членов класса Operations**

*Таблица 7 – Описание членов класса Operations*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Методы** | | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | *Аргументы* | | *Назначение* |
| Operations | public | - | Input, Homescreen,  Funcs, int, int, int, float, float, float, float, float, float, float, float, float, float, float, float float, float, float, float, float, int | | Конструктор для формы с параметрами - параметрами симуляции. Задает внешний вид формы, значения полей, настраивает подсказки для пользователя. |
| bGetData\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку добавления файлов наблюдений. Вызывает метод, открывающий файлы, проверяющий их корректность и визуализирующий полученные данные. |
| AddData | private | void | - | | Метод для открытия файлов наблюдений с проверкой длины и корректности данных, обновляет информацию в логгере, в случае правильности данных меняет флаг наличия файлов наблюдений, активизирует связанные с этим кнопки, вызывает матлаб функцию визуализации данных. |
| bGetMatrix\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку добавления файлов матрицы. Вызывает метод, открывающий файлы, проверяющий их корректность и визуализирующий полученные данные. |
| AddMatrix | private | void | - | | Метод для открытия файлов матрицы с проверкой длины и корректности данных, обновляет информацию в логгере, в случае правильности данных меняет флаг наличия файлов матрицы, активизирует связанные с этим кнопки, вызывает матлаб функцию визуализации данных. |
| bGetStat\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку добавления файла статистики. Вызывает метод, открывающий файл, проверяющий его корректность и визуализирующий полученные данные. |
| AddStat | private | void | - | | Метод для открытия файлов статистики с проверкой длины и корректности данных, обновляет информацию в логгере, в случае правильности данных меняет флаг наличия файлов статистики, активизирует связанные с этим кнопки, вызывает матлаб функцию визуализации данных, запоминает наибольшее и наименьшее значение сигнала для настройки слайдера. |
| OpenFile | private | bool | string name, bool, int | | Метод для добавления информации из файлов. Открывает диалоговое окно для выбора файла, вызывает метод проверки корректности файла, в случае корректности данных копирует выбранный файл в соответствующий рабочий файл, сохраняет изменения в логгер, обрабатывает исключения, которые могут возникнуть при открытии и чтении файла. |
| CheckData | private | bool | string, bool checkLen, int expectedLen | | Метод для проверки корректности загружаемых файлов. Вызывает метод считывания чисел и проверяет соответствие каждого числа требуемому типу, также сравнивает длину файла с ожидаемой длиной и возвращает результат соответствия файла требованиям. |
| NextNumber | private | string | StreamReader | | Метод для считывания следующего числа из файлового потока. |
| bGenData\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку генерации наблюдений. Вызывает асинхронный метод генерации и визуализации наблюдений. |
| BeginData | private | void | - | | Метод для генерирования и визуализации наблюдений, настраивает вид формы на время осуществления процесса и вызывает метод генерирования и визуализации наблюдений из экземпляра класса функций матлаб, засекает время за которое процесс завершается, обновляет информацию в логгере и флаг наличия наблюдений в случае успеха. |
| bGenMatrix\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку расчета корреляционной матрицы. Вызывает асинхронный метод расчета и визуализации корреляционной матрицы. |
| BeginMatrix | private | void | - | | Метод для расчета и визуализации корреляционной матрицы, настраивает вид формы на время осуществления процесса и вызывает метод расчета и визуализации матрицы из класса библиотеки, засекает время за которое процесс завершается, обновляет информацию в логгере и флаг наличия матрицы в случае успеха. |
| bGenStat\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку расчета статистики. Вызывает асинхронный метод расчета и визуализации статистики. |
| BeginL | private | void | - | | Метод для расчета статистики и визуализации наблюдений, настраивает вид формы на время осуществления процесса и вызывает метод расчета и визуализации статистики наблюдений из экземпляра класса функций матлаб, засекает время за которое процесс завершается, обновляет информацию в логгере и флаг наличия статистики наблюдений в случае успеха, также запоминает минимальное и максимальное значения сигнала из статистики для дальнейшего отображения на слайдере. |
| sliderH\_Scroll | private | void | object, EventArgs | | Обработчик изменения значения на слайдере. Порог сигнала меняется на выставленный на слайдере, либо на минимальное/максимальное осмысленное значение. Соответственно меняется текст на кнопке визуализации. |
| bVisualize\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку визуализации данных. Вызывает асинхронный метод визуализации. |
| BeginVis | private | void | - | | Метод для визуализации данных статистики при изменении порогового значения сигнала, настраивает вид формы на время осуществления процесса и вызывает метод визуализации из экземпляра класса функций матлаб, засекает время за которое процесс завершается, обновляет информацию в логгере. |
| bSaveData\_Click | private | void | object, EventArgs | | Обработчик нажатия на кнопку сохранения файлов наблюдений. В логгер заносится запись о текущем процессе и вызывается метод копирования файла. |
| bSaveMatrix\_Click | Private | void | object, EventArgs | | Обработчик нажатия на кнопку сохранения файлов корреляционной матрицы. В логгер заносится запись о текущем процессе и вызывается метод копирования файла. |
| bSaveStat\_Click | Private | void | object, EventArgs | | Обработчик нажатия на кнопку сохранения файла статистики. В логгер заносится запись о текущем процессе и вызывается метод копирования файла. |
| SaveFile | Private | void | String | | Метод для сохранения файлов в выбранное пользователем место. Заносит состояние задания в логгер, открывает диалоговое окно для сохранения файла и копирует рабочий файл в требуемую директорию. Также обрабатываются возможные исключения. |
| bBackToInput\_Click | Private | Void | object, EventArgs | | Обработчик нажатия на кнопку возвращения к предыдущей форме. Текущая форма закрывается и показывается предыдущая. |
| bBackToHomepage\_Click | Private | Void | object, EventArgs | | Обработчик нажатия на кнопку возвращения на главную страницу. Закрываются предыдущая и текущая формы, открывается форма главной страницы. |
| GetAllControlsOfType<T> | Private | List<Control> | Control container | | Метод, позволяющий получить все элементы определенного типа на форме для их  более удобной настройки. |
| Enable | private | Void | - | | Метод, который переводит кнопки в активное состояние в зависимости от того, могут ли они быть активными (была ли добавлена информация, с которой связана их работа). |
| Disable | private | Void | - | | Метод, переводящий все кнопки в неактивное состояние. |
| **Поля** | | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | | *Назначение* | |
| backColor | private | Color | | Поле для цвета фона формы. | |
| panelColor | private | Color | | Поле для цвета панели и кнопок формы. | |
| hoverColor | private | Color | | Поле для цвета элемента формы, над которым находится курсор. | |
| textColor | private | Color | | Поле для цвета текста формы. | |
| prevForm | private | Input | | Поле для ссылки на предыдущую форму задания параметров симуляции. | |
| homescreen | private | Homescreen | | Поле для ссылки на форму начального экрана. | |
| funcs | private | MatlabFuncs.Funcs | | Поле для ссылки на экземпляр класса функций матлаб. | |
| Um | private | float | | Поле для параметра ширины главного лепестка диаграммы направленности модуля по одному углу. | |
| Vm | private | float | | Поле для параметра ширины главного лепестка диаграммы направленности модуля по другому углу. | |
| du | private | float | | Поле для параметра сдвига до центра просматриваемого сектора по одному углу. | |
| dv | private | float | | Поле для параметра сдвига до центра просматриваемого сектора по другому углу. | |
| u | private | float | | Поле для параметра предполагаемого положения цели по одному углу. | |
| v | private | float | | Поле для параметра предполагаемого положения цели по другому углу. | |
| a | private | float | | Поле для параметра мощности приходящего сигнала. | |
| Tc | private | float | | Поле для параметра начала прихода сигнала. | |
| up1 | private | float | | Поле для параметра положения помех по одному углу. | |
| up2 | private | float | | Поле для параметра положения помех по одному углу. | |
| vp1 | private | float | | Поле для параметра положения помех по другому углу. | |
| vp2 | private | float | | Поле для параметра положения помех по другому углу. | |
| Ap1 | private | float | | Поле для параметра мощности помех. | |
| Ap2 | private | float | | Поле для параметра мощности помех. | |
| gamma | private | float | | Поле для параметра гамма. | |
| \_mu | private | float | | Поле для параметра регуляризации корреляционной матрицы. | |
| H | private | float | | Поле для параметра порогового значения мощности сигнала. | |
| minH | private | float | | Поле для минимального значения сигнала в статистике наблюдений. | |
| maxH | private | float | | Поле для максимального значения сигнала в статистике наблюдений. | |
| N | private | int | | Поле для параметра количества векторов наблюдений, по которым формируется статистика. | |
| Mu | private | int | | Поле для параметра количества модулей, на которые разбивается антенная решетка по длине. | |
| Mv | private | int | | Поле для параметра количества модулей, на которые разбивается антенная решетка по ширине. | |
| pq | private | int | | Поле для параметра количества блоков в 36 областях, по которым составляется корреляционная матрица. | |
| dataE | private | bool | | Поля-флаги для определения стадии работы и мониторинга добавленной информации и опций, открытых в зависимости от этого. | |
| matrixE | private | bool | | Поля-флаги для определения стадии работы и мониторинга добавленной информации и опций, открытых в зависимости от этого. | |
| statE | private | bool | | Поля-флаги для определения стадии работы и мониторинга добавленной информации и опций, открытых в зависимости от этого. | |

**Описание членов класса RadarOperations**

*Таблица 8 – Описание членов класса RadarOperations*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Методы** | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | *Аргументы* | *Назначение* |
| Start | public | TimeSpan | int, int, int, float, string, string, string, string | Вызываемый извне метод класса, который вызывает методы считывания матрицы наблюдений, расчета обратной корреляционной матрицы и записи ее в файл, кроме того осуществляется измерение времени всего процесса обработки информации, и оно возвращается в точку вызова. |
| ReadData | private | void | int, int, int, Complex[], string, string | Метод, с помощью которого через потоки считывается матрица наблюдений, сгенерированная ранее. Матрица, находящаяся в двух файлах (для вещественных и мнимых компонент), объединяется в массив комплексных чисел. |
| MatrixInv | private | void | int, int, int, float, Complex[], Complex[] | Метод, осуществляющий расчет обратной корреляционной матрицы на основе матрицы наблюдений. В том числе происходит ее изначальное заполнение, затем расчет значений по матрице наблюдений и инверсия. |
| ComplexMAD | private | void | Complex, Complex, Complex | Метод, выполняющий сложение первого передаваемого комплексного числа с произведением двух следующих: a += b \* c. |
| WriteData | private | void | int, int, Complex[], string, string | Метод, в котором через потоки в два файла (для вещественных и мнимых компонент) записывается рассчитанная корреляционная матрица. |

**Описание используемых MATLAB-функций**

*Таблица 9 – Описание используемых MATLAB-функций*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Методы** | | | | |
| *Имя* | *Мод. Доступа* | *Тип* | *Аргументы* | *Назначение* |
| CreateFiles | - | - | 19 аргументов | Функция для создания и визуализации файлов наблюдений. |
| CreateL | - | - | 7 аргументов | Функция для создания и визуализации файлов статистики наблюдений. |
| CreatePicture | - | - | 7 аргументов | Функция для разделения и вызова функций визуализации файлов наблюдений, корреляционной матрицы и статистики без их создания. |
| essentCounts | - | - | 3 аргумента | Функция для расчета дополнительных параметров симуляции – массивов проверяемых гипотез. |
| getGauss | - | - | 2 аргумента | Функция для создания матрицы из случайных чисел с нормальным распределением. |
| getMatrNabl | - | - | 15 аргументов | Функция для генерации матрицы наблюдений. |
| getPhase | - | - | 6 аргументов | Функция для вычисления фазы сигнала на основании времени его прихода. |
| sinx | - | - | 1 аргумент | Функция для вычисления поэлементного выражения (Sin(x))/x. |
| VisFiles | - | - | 17 аргументов | Функция для визуализации файлов наблюдений и статистики в формате отображения сигналов. |
| VisR | - | - | 16 аргументов | Функция для визуализации файлов корреляционной матрицы в формате анимации. |
| Yinvert | - | - | 4 аргумента | Функция для обращения матрицы наблюдений, используется для сравнения и проверки результатов нахождения корреляционной матрицы. |

ПРИЛОЖЕНИЕ 5

**ДИАГРАММА КЛАССОВ**

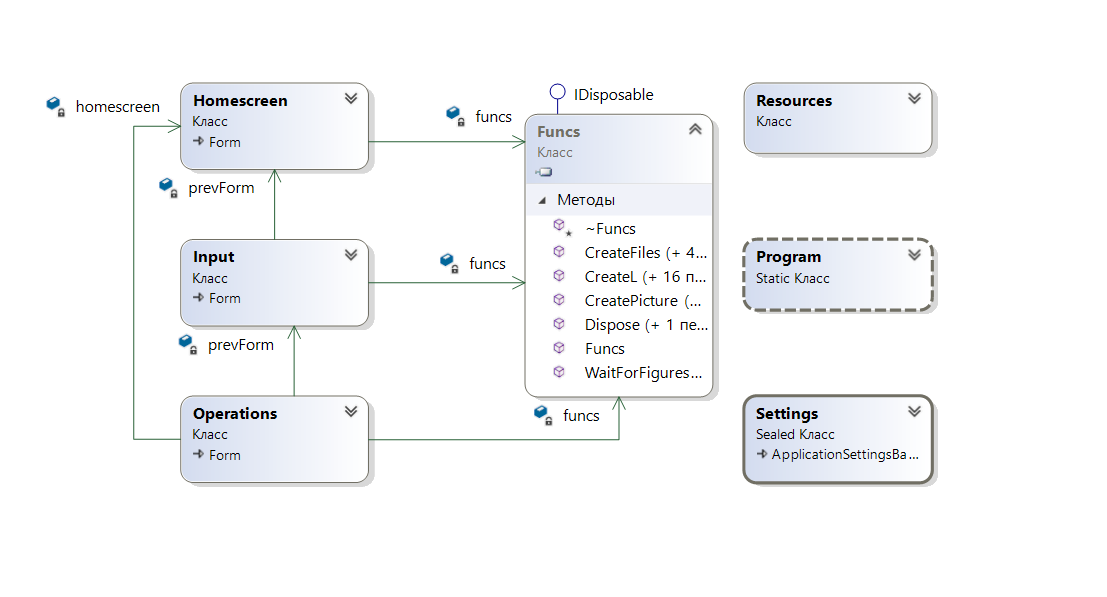


Рисунок 3 – Диаграмма классов приложения Windows Forms.

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подпись | Дата |
| измененных | замененных | Новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |