

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное автономное образовательное
Учреждение высшего образования
«Казанский (Приволжский) Федеральный Университет»

ИНСТИТУТ ФИЗИКИ

КАФЕДРА РАДИОЭЛЕКТРОНИКИ

Направление: 03.03.03 - Радиофизика

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

(Бакалаврская работа)

ОЦЕНИВАНИЕ ПАРАМЕТРОВ РАКУРСНОГО РАССЕЙЯНИЯ
РАДИОСИГНАЛОВ НАВИГАЦИОННЫХ СПУТНИКОВ

Работа завершена:

“ ___ ” _____ 2017 г. _____ (Галиуллин А.А.)

Работа допущена к защите:

Научный руководитель:

ассистент кафедры радиоэлектроники

“ ___ ” _____ 2017г. _____ (Терёшин С.Н.)

Заведующий кафедрой:

д.ф.-м.н., профессор

“ ___ ” _____ 2017 г. _____ (Овчинников М. Н.)

Казань - 2017

Оглавление

Введение	3
Глава 1. Нелинейные явления в ионосфере	5
1.1. Нелинейные явления	5
1.2. Воздействие радиоволн на нижнюю ионосферу	7
1.3. Модификация ионосферной плазмы, гигантское ракурсное рассеяние	8
1.4. Многократный гиромагнитный резонанс, сверхмелкомасштабные вытянутые неоднородности	10
Глава 2. ГЛОНАСС	13
2.1. Состав системы ГЛОНАСС.....	13
2.2. Навигационное сообщение	15
Глава 3. Описание созданной программы и её результатов	16
Заключение	24
Список литературы	25
Приложение А	27
Приложение Б	37
Приложение В	45

Введение

Ионосфера — это плазменный слой в верхней атмосфере, на высотах от 60 до 1000 км. Основным источником ионизации является ультрафиолетовое излучение Солнца. Процессы, протекающие в ионосфере, тесно связаны с солнечной активностью, с процессами в магнитосфере и вариациями магнитного поля Земли, с движениями верхней атмосферы. Следствием является сильная изменчивость свойств ионосферы во времени - в течение суток, в зависимости от времени года, с одиннадцатилетним циклом солнечной активности, а также в зависимости от высоты и географической широты. Структура и свойства ионосферы сильно меняются с высотой. Ионосферу принято делить на три области, называемые слоями D, E и F.

Максимум концентрации плазмы достигается в F-слое, на высоте около 300 км, он составляет приблизительно 10^6 электронов в 1 см^3 , спадая в ночное время до $3 \cdot 10^5 \text{ см}^3$. Ниже F-слоя, в E- и D-слоях, концентрация спадает, уменьшаясь до 10^3 см^3 или даже 10^2 см^3 . Иногда концентрация E-слоя существенно увеличивается, тогда его называют спорадическим. В области, лежащей выше 1000 км, ионосфера плавно переходит в магнитосферу.

Ионосфера существенно влияет на процессы распространения радиоволн. От F-слоя отражаются короткие радиоволны. Благодаря большой высоте слоя они распространяются на большие расстояния — до 2-3 тысяч километров. В E- и D-слоях распространяются длинные и средние волны. Вследствие наличия магнитного поля Земли ионосферная плазма анизотропна, что приводит к возникновению двух компонент радиоволн — обыкновенной и необыкновенной. Это явление аналогично двойному лучепреломлению электромагнитных волн в анизотропных кристаллах [5].

Необходимо дальнейшее изучение механизмов генерации естественных ионосферных неоднородностей, нелинейных процессов в ионосферной плазме и в плазме вообще, а также исследование эффектов искусственного воздействия на ионосферу. Данная выпускная работа выполнялась для получения вспомогательного программного инструмента в планировании экспериментов.

Целью данной работы является разработка программы оценивания параметров радиосигналов навигационных спутников, рассеянных на анизотропных ионосферных СММ неоднородностях.

Для достижения цели работы потребовалось решение задач:

разработать и реализовать алгоритм определения угла ракурсного рассеяния для заданных нагревного стенда и приёмного пункта, и высоте области неоднородностей с учётом возможных её размеров;

разработать и реализовать алгоритм определения времени прохождения спутниками участка орбиты, на котором радиосигнал спутника, рассеянный на СММ неоднородностях, будет доступен в приемном пункте с учётом различных частей области неоднородностей.

Глава 1. Нелинейные явления в ионосфере

1.1. Нелинейные явления

Проводить исследование ионосферной плазмы методом воздействия радиоволн, возможно, потому что в плазме с малой концентрацией весьма существенное локальное возмущение достигается сравнительно слабоинтенсивным воздействием. Такие исследования впервые были начаты в 60-х годах в СССР.

На Земле действуют много вещательных и других радиостанций, мощность которых ничуть не меньше мощности установок воздействия на ионосферу, а нередко и превосходит ее. Радиоизлучение станций воздействия специально сфокусировано вблизи вертикального направления. Это дает возможность радиоволнам достичь области резонансов в окрестности максимума концентрации электронов в F-слое. Эффективность воздействия является следствием совпадения частоты возбуждающей волны с частотами собственных колебаний плазмы. В результате в области резонанса происходит сильное возбуждение собственных колебаний электронной плазмы, развивается плазменная турбулентность, что и служит причиной нагрева электронной плазмы, структуризации плазмы, генерации искусственного радиоизлучения, ускорения электронов и ряда других физических явлений.

Особенность плазмы, находящейся в магнитном поле, состоит в том, что, в отличие от обычных жидкостей и газов (в которых существуют только звуковые волны), в плазме существует большое количество разнообразных волн, например:

- плазменные волны — продольные колебания нагретой электронной компоненты плазмы;

- ионно-звуковые волны — совместные продольные колебания электронов и ионов;
- верхнегибридные плазменные волны — электронные колебания поперек магнитного поля;
- бернштейновские моды, связанные с многократным гиромагнитным резонансом,

и другие волны. Этим обусловлено чрезвычайно большое разнообразие нелинейных явлений в ионосферной плазме, возникающих при резонансном воздействии мощных радиоволн.

Для исследования вызванных радиоволнами возмущений плазмы был разработан широкий набор диагностической техники, включающей в себя разнообразные системы радиозондирования, комплексы многочастотных доплеровских измерений, радары некогерентного рассеяния (в том числе один из высокочувствительных в мире радар в Аресибо), специальные камеры и телескопы для оптических измерений, магнетометры и другие научно-измерительные приборы. Проводились также прямые наблюдения со специально запущенных ракет и искусственных спутников.

Первым нелинейным явлением, открытым в 1930г. была кроссмодуляция. При включении люксембургской станции ее передачи стали слышны на других средневолновых станциях, частота которых никак не была связана с этой мощной радиостанцией. Природа кроссмодуляции заключается в следующем. Под действием мощной радиоволны изменяется температура электронов в ионосфере. Речь идет о небольших изменениях температуры, порядка $\Delta T/T \sim 5\%$. И все же такое изменение температуры влияет на частоту соударений электронов, что сказывается на поглощении радиоволн, проходящих в возмущенной области. Для передачи звуковых сигналов амплитуда радиоволны модулируется звуковой частотой $F \sim 1$ кГц. Поскольку радиоволна мощной станции модулирована по амплитуде частотой F , то возникнут колебания возмущений электронной

температуры ΔT с той же частотой F . Следовательно, радиоволны других станций, распространяющиеся в возмущенной области ионосферы, благодаря переменным изменениям электронной температуры ΔT , частоты соударений и поглощения приобретают модуляцию мощной станции.

1.2. Воздействие радиоволн на нижнюю ионосферу

В 1937 г. Бейли опубликовал работу, в которой предложил использовать резонанс между вращением электронов вокруг магнитного поля и радиоволной, имеющей частоту 1,45 МГц. Предполагалось таким способом довести энергию электронов до большой величины и тем самым возбудить значительное свечение ионосферы [1].

По предположению Бейли для осуществления эффекта можно использовать станции с относительно небольшой мощностью 500 кВт, и коэффициентом усиления антенны $G = 100$, так чтобы эффективная мощность излучения достигала величины 50 МВт. Согласно расчетам Бейли такой мощности излучения достаточно, для того чтобы осуществить предсказанный эффект в предположении, что волна свободно достигает высот 130-140 км.

Впервые воздействие мощных радиоволн на ионосферу было осуществлено в СССР. Работа основывалась на идее Бейли. Под Москвой была создана мощная импульсная радиостанция с частотой 1,45 МГц, длительность импульса 5 мс. Антенная система направляла пучок радиоволн вертикально вверх. Именно в этом случае воздействие электромагнитного поля волн на ионосферную плазму является наиболее сильным. В 1961 г. станция начала работать. Эффективная мощность равнялась 100 МВт, т.е. вдвое превышала ту, которую полагал достаточной. Однако ожидаемого эффекта свечения неба наблюдать не удалось. Тогда инженеры этой, заметив, что для достижения максимального эффекта

воздействия можно уменьшить длительность импульса, сократили ее до 0,5 мс. Эффективная мощность излучения в импульсе возросла при этом в 10 раз, достигнув величины в 1 ГВт. Но и в этом случае никакого значительного свечения увидеть не удалось.

Причина этого состояла в том, что столь мощная волна испытывает так называемое сильное самовоздействие. Иначе говоря, в нижних слоях ионосферы, на высотах 60-80 км, благодаря очень большой частоте соударений при сильном нагреве электронов под действием поля волны происходит эффективное самопоглощение волны. В результате в верхние слои (110-140 км) значительная мощность не проходит. Нелинейное поглощение, наблюдавшееся в этих экспериментах, достигало 30 дБ (измерения были проведены впоследствии), т.е. мощность волны из-за самопоглощения понижалась почти в 1000 раз.

1.3. Модификация ионосферной плазмы, гигантское ракурсное рассеяние

В 1960 году впервые было указано на возможность возмущения не только нижней ионосферы, но и F-слоя. В конце 60-х годов, в ходе исследований на станции в Платтевилле (США, Колорадо) наблюдалось много новых нелинейных явлений, вызвавших значительный интерес. Было обнаружено явление, которое получило название гигантское ракурсное рассеяние. При включении мощной станции можно было слышать на больших расстояниях очень малоинтенсивные станции. Например, разговоры таксистов, имеющих передатчик мощностью несколько ватт на частоте около 50 МГц, были слышны от Нью-Йорка до Сан-Франциско. Обычно такие станции слышны только в пределах

видимости, т.е. на расстояниях в несколько десятков километров, а здесь наблюдалось распространение радиоволн на несколько тысяч километров. Возникло впечатление, что с включением мощной станции появился новый способ радиосвязи на дальние расстояния. Этот вопрос представлял значительный практический интерес [1].

Было понято, что в направлениях вдоль магнитного поля над Платтевилем возникает большая группа сильно вытянутых неоднородностей плазмы. Их можно представлять как набор зеркал, вытянутых строго вдоль магнитного поля, но случайно ориентированных в направлении, ортогональном к полю. В результате, падающая на ионосферу волна рассеивается в направлениях, ортогональных магнитному полю. Это и есть ракурсное рассеяние. Таким образом, появился новый вид распространения радиоволн, который был детально экспериментально исследован. Сечение ракурсного рассеяния, действительно, оказалось огромным, указывающим на то, что возникающие возмущения ионосферы чрезвычайно велики.

Была продемонстрирована возможность использования ракурсного рассеяния для телевизионных передач на большие расстояния. Как выяснилось, для практического использования оно большого значения не приобрело.

Исследования в Платтевилле показали, что новые явления возникают только под действием мощной обыкновенной радиоволны (О-волны), распространяющейся в вертикальном или близком к нему направлении. Такие явления получили название модификации ионосферы под действием мощных радиоволн. Кроме гигантского ракурсного рассеяния наблюдались и другие явления, такие как: аномальное поглощение искусственное радиоизлучение ионосферы, сильное искусственное оптическое свечение ионосферы и др.

Модификация ионосферы мощными радиоволнами происходит в основном в F-слое. Иногда явления аналогичного типа наблюдаются в слое E-спорадическом.

1.4. Многократный гиромагнитный резонанс, сверхмелкомасштабные вытянутые неоднородности

При определенных частотах возмущающей волны в ионосфере можно осуществлять сразу два резонансных эффекта. Первый — это обычный резонанс. Он возникает в области ВГ-резонанса на определенной высоте в ионосфере, зависящей от частоты и высотного профиля концентрации электронов. Второй резонанс — это резонанс с частотой, кратной частоте гиромагнитного вращения электронов. Таким образом, двойной резонанс возникает в тех областях ионосферы, где выполняется условие $\omega \approx n\omega_H \approx \omega_{UH}(z)$. Условие кратного гирорезонанса в F-слое ионосферы может выполняться при значениях $n = 2, 3, 4, 5, 6$ и даже больших, в зависимости от величины максимальной концентрации электронов F-слоя.

Вблизи двойного резонанса происходит очень сильное изменение всех явлений, возникающих в области ВГ-резонанса при воздействии мощной волны на ионосферу. А именно радикально изменяется количество сильно вытянутых неоднородностей (вплоть до их полного исчезновения). Соответственно изменяется гигантское ракурсное рассеяние, аномальное и широкополосное поглощение радиоволн, а также полностью трансформируется ИРИ. Кроме того, возникает система новых сверхмелкомасштабных сильно вытянутых неоднородностей. Вместе с ними появляется сильное ракурсное рассеяние радиоволн ультравысокой частоты (УВЧ), вплоть до 1 - 2 ГГц.

Природа эффекта состоит в том, что в двойном резонансе появляются волны нового типа, так называемые бернштейновские моды. Физически эти волны связаны с гиромагнитным вращением электронов вокруг направления магнитного поля. Оказывается, что и в случаях кратного гирорезонанса возможен фазовый синхронизм колебаний электрического поля волны с гиромагнит-

ным вращением. Для этого необходимо только, чтобы в плазме возникали пространственные неоднородности, т.е. волны, длины которых находятся во вполне определенном соотношении с радиусом гироманнитного вращения электрона. Эти волны и называют бернштейновскими модами. Поскольку в условиях ионосферы радиус имеет величину порядка 1-2 см, то и длина волны бернштейновских мод поперек магнитного поля мала: 10—40 см. Вследствие фазового синхронизма энергия возмущающей волны может эффективно перекачиваться в бернштейновские моды.

Сильно вытянутые ионосферные неоднородности имеют большой масштаб, 7 — 10 м, а масштаб бернштейновской волны существенно меньше, 10—40 см. Следовательно, распространение бернштейновской волны в неоднородностях носит адиабатический характер. Необходимо рассмотреть четырехволновый процесс. Теория этого процесса разрабатывалась рядом авторов, но в окончательной форме была построена Д. Хуангом и С. Куо. В четырехволновом процессе взаимодействуют верхнегибридная волна (ω_1, k_1) , бернштейновская мода (ω_2, k_2) , мощная радиоволна накачки $(\omega, k_0 = 0)$ и низкочастотная электростатическая волна (ω_s, k) . Основные соотношения частотного и фазового синхронизма, имеют вид $\omega_1 + \omega_s = \omega = \omega_2 - \omega_s$, $k_1 + k = 0 = k_2 - k$. Видно, что частота бернштейновской моды всегда лежит выше частоты мощной радиоволны накачки [1].

В процессе четырехволнового взаимодействия, генерирующего бернштейновскую моду, важную роль играет ВГ-волна. А ВГ-волна не является свободной — она захвачена вытянутой неоднородностью, иначе говоря, это стоячая волна со стационарной фазой, которая образуется двумя волнами, бегущими в противоположных направлениях. Но, поскольку четырехволновый распад происходит линейно, каждая из этих волн, бегущих вправо и влево, генерирует бернштейновскую моду. Таким образом, бернштейновская мода тоже образует стоячую волну внутри неоднородности. Стоячая бернштейновская волна так же, как и ВГ-волна, имеет большую амплитуду. В результате возни-

кают неоднородности плазмы, которые имеют продольный масштаб, т.е. вдоль магнитного поля, — порядка 1 км. Это и есть вытянутые сверхмелкомасштабные (СММ) неоднородности. Их поперечный размер составляет лишь несколько десятков сантиметров. Линейная трансформация (рассеяние) бернштейновских волн на СММ-неоднородностях объясняет генерацию широкого сдвинутого вверх максимума искусственного радиоизлучения.

Появление сверхмелкомасштабных вытянутых неоднородностей вблизи многократного гирорезонанса, может служить причиной и других эффектов. Главный из них — сильное ракурсное рассеяние радиоволн ультравысокой частоты. Значительное рассеяние могут испытывать радиоволны очень высокой частоты — до 1-3 ГГц. Такое рассеяние значительно слабее гигантского ракурсного рассеяния сверхвысокочастотных (СВЧ) радиоволн на стационарных сильно вытянутых ионосферных неоднородностях. Однако частотный диапазон эффекта на порядок выше.

СММ-неоднородности возникают внутри сильно вытянутых неоднородностей. Последние появляются вследствие нагрева электронов резонансно возбужденными ВГ-волнами. Поэтому температура электронов внутри неоднородностей в 2-3 раза превышает среднюю. Градиенты температуры приводят к появлению квазистационарного электрического поля. Мелкомасштабные неоднородности дрейфуют в этом поле. Скорость дрейфа может достигать 30-100 м/с. Эта скорость должна определять доплеровскую ширину частотного спектра рассеянных СММ-неоднородностями радиоволн.

Глава 2. ГЛОНАСС

2.1. Состав системы ГЛОНАСС

Глобальная навигационная спутниковая система (ГЛОНАСС) — российская спутниковая система навигации. Система ГЛОНАСС в расширенной конфигурации включает в себя следующие составные части:

- Космический комплекс системы ГЛОНАСС, состоящий из орбитальной группировки, средств выведения, наземного комплекса управления;
- Функциональные дополнения, включая широкозонную систему функционального дополнения ГНСС – систему дифференциальных коррекций и мониторинга, а также региональные и локальные системы мониторинга и дифференциальной навигации;
- Система высокоточной апостериорной эфемеридно-временной информации;
- Средства фундаментального обеспечения ГЛОНАСС – системы оперативного определения параметров вращения и ориентации Земли, системы формирования государственной шкалы всемирного скоординированного времени, геодезической основы РФ;
- Навигационная аппаратура потребителей для гражданского и специального применения [7].

Основой системы являются 24 спутника, движущихся над поверхностью Земли в трёх орбитальных плоскостях с наклоном орбитальных плоскостей $64,8^\circ$ и высотой орбит 19400 км. Принцип измерения аналогичен системе навигации NAVSTAR GPS. Основное отличие от системы GPS в том, что спутники ГЛОНАСС в своём орбитальном движении не имеют резонанса (синхронности) с вращением Земли, что обеспечивает им большую стабильность. Таким обра-

зом, группировка КА ГЛОНАСС не требует дополнительных корректировок в течение всего срока активного существования.

Передаваемые каждым НКА системы ГЛОНАСС в составе оперативной информации эфемериды описывают положение в связанной с Землей геоцентрической системе координат ПЗ-90.02, определяемой следующим образом:

Начало координат расположено в центре масс Земли; ось Z направлена на Условный полюс Земли, как определено в рекомендации Международной службы вращения Земли (IERS); ось X направлена в точку пересечения плоскости экватора и начального (нулевого) меридиана, установленного Международным бюро времени (BIH); ось Y дополняет геоцентрическую прямоугольную систему координат до правой. В этой системе координат положение точки в пространстве определяется значениями координат X, Y, Z .

Геодезические координаты точки в системе координат ПЗ-90.02 относятся к эллипсоиду, значения большой полуоси и полярного сжатия. Геодезическая широта B точки M определяется как угол между нормалью к поверхности эллипсоида и плоскостью экватора. Геодезическая долгота L точки M определяется как угол между плоскостью нулевого меридиана и плоскостью меридиана, проходящего через точку M . Положительное направление счета долгот - от нулевого меридиана к востоку от 0 до 360 градусов. Геодезическая высота H определяется как расстояние по нормали от поверхности эллипсоида до точки M [6].

2.2. Навигационное сообщение

По своему содержанию навигационное сообщение подразделяется на оперативную и неоперативную информацию [6].

Оперативная информация относится к тому НКА, с борта которого передается данный навигационный радиосигнал и содержит:

- оцифровку меток времени НКА;
- сдвиг шкалы времени НКА относительно шкалы времени системы ГЛОНАСС;

Неоперативная информация содержит альманах системы, включающий в себя:

- данные о состоянии всех НКА системы (альманах состояния);
- сдвиг шкалы времени каждого НКА относительно шкалы времени системы ГЛОНАСС (альманах фаз);
- параметры орбит всех НКА системы (альманах орбит);
- сдвиг шкалы времени системы ГЛОНАСС относительно UTC(SU) и другие параметры.

Глава 3. Описание созданной программы и её результатов

Для разработки программы требовалось учесть сведения о системе спутников ГЛОНАСС, особенности расположения и антенных систем нагревных стендов, зависимость направления силовых линий магнитного поля от географических координат и высотой над поверхностью Земли.

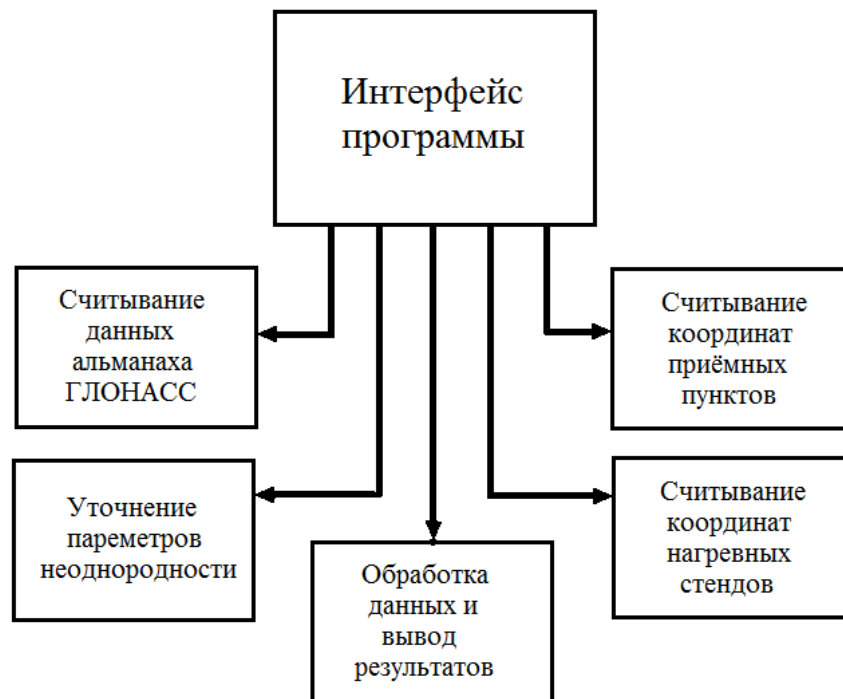


Рис. 1. Блок-схема программы

Алгоритм работы программы:

Сначала происходит последовательное считывание данных каждого спутника из альманаха ГЛОНАСС. Далее подпрограмма обработки вычисляет такие параметры, такие как угол и координаты ракурсного рассеяния, время «видимости» спутника и другие параметры. После обработки, полученные значения выводятся в виде таблиц.

Теперь рассмотрим подробнее подпрограмму расчёта. Листинг приведён в Приложении Б.

В начале все угловые величины переходят из радианной меры в градусную.

По географическим координатам пункта нагрева и высоте неоднородности вычисляются её координаты в геоцентрической системе координат.

$$\begin{aligned} X_i &= (R + h) \cos(\beta) \cos(\lambda), \\ Y_i &= (R + h) \cos(\beta) \sin(\lambda), \\ Z_i &= (R + h) \sin(\beta) \end{aligned} \quad (1)$$

где R – экваториальный радиус Земли, h – высота неоднородности, λ – долгота пункта, β – долгота пункта.

В случае выбора режима расчёта нескольких точек неоднородности, вычисляются компоненты смещения из центра неоднородности и координаты этих точек.

Аналогично, используя формулы (1) вычисляются координаты приёмного пункта.

Компоненты магнитного поля в местной системе координат вычисляются по формулам (2):

$$\begin{aligned} X_{oM} &= \cos(inc) * \cos(\pi - dec), \\ Y_{oM} &= \cos(inc) * \sin(\pi - dec), \\ Z_{oM} &= \sin(\pi - inc) \end{aligned} \quad (2)$$

где, inc – магнитное наклонение, dec – магнитное склонение.

Наклонение и склонение магнитного поля Земли были вычислены при помощи World Magnetic Model (WMM2015) software – программы, вычисляющей характеристики магнитного поля соответственно WMM 2015.

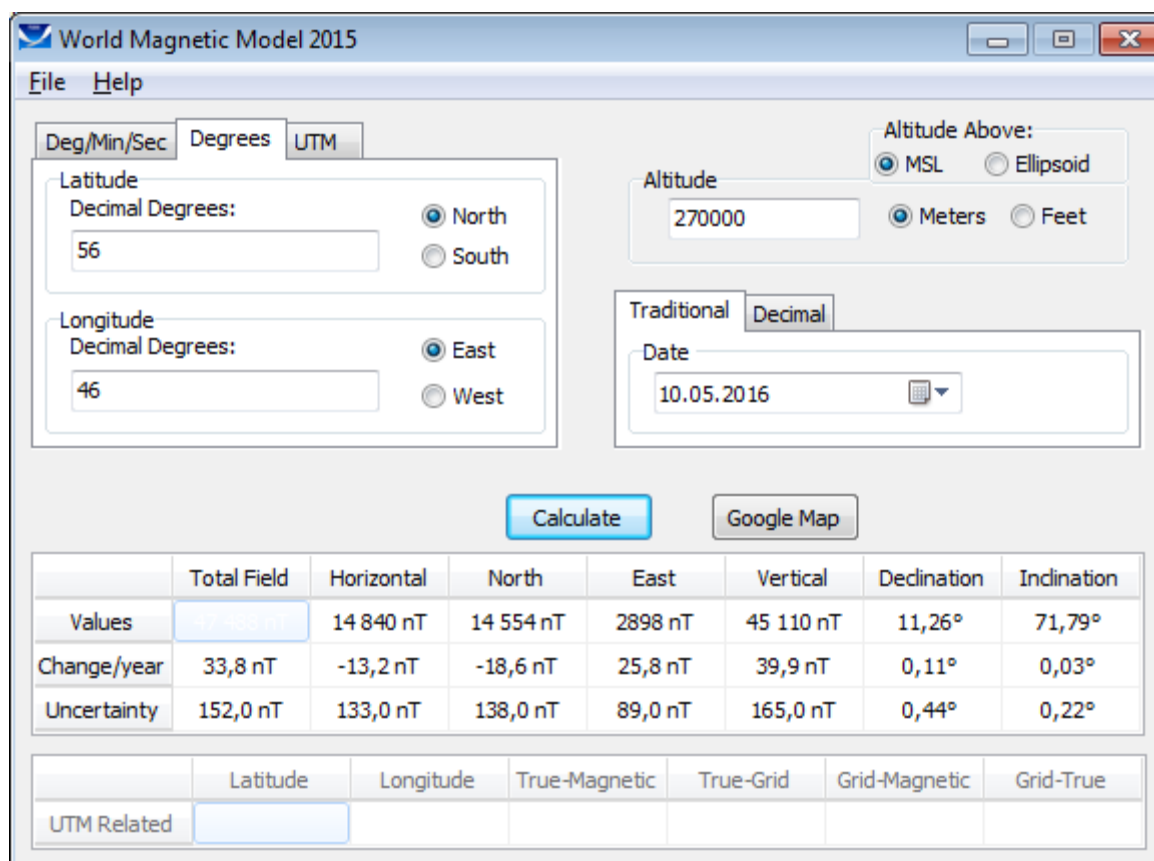


Рис. 2. Интерфейс программы World Magnetic Model (WMM2015) software

Для перехода из местной системы координат в геоцентрическую используется матрица поворота:

$$M(\theta) = \begin{pmatrix} \cos \theta & \mp \sin \theta \\ \pm \sin \theta & \cos \theta \end{pmatrix} \quad (3)$$

где θ – угол поворота.

Поворот выполняется путём умножения матрицы поворота на вектор-столбец, описывающий вращаемую точку. Координаты в результате поворота точки имеют вид:

$$\begin{aligned} x' &= x \cos \theta \mp y \sin \theta, \\ y' &= \pm x \sin \theta + y \cos \theta \end{aligned} \quad (4)$$

Далее происходит вычисление компонент вектора рассеивания и угла между векторами магнитного поля и рассеивания.

Рассчитаем точность расчёта времени ракурсного рассеяния. Период вращения спутников – драконический период равен 40544 секунд. Угол поворота равен 360° . В программе реализована поочерёдная проверка ракурсного рассеяния каждые $0,01^\circ$, то есть этим достигается точность в $40544/(36000) \approx 11,26$ секунд.

После вычисления времени возможного совпадения и угла ракурсного рассеяния определяются координаты спутника и компоненты вектора излучения. Компоненты вектора излучения вычисляются аналогично компонентам вектора рассеивания, только координаты приёмного пункта заменяются координатами спутников.

Вычисленные координаты спутника, используются для определения возможности прямой передачи сигнала в пункт приема. Для этого используется дополнительная функция pass. Листинг дополнительных функций приведён в Приложении В.

Далее происходит проверка условия совпадения углов между векторами магнитного поля и рассеивания и между векторами излучения и магнитного поля с учётом погрешностей.

При выполнении условия определяются такие параметры, как время поступления сигнал в пункт приёма, его продолжительность, номер спутника, с которого передаётся сигнал и координаты спутника в начальный и конечный

момент передачи. Часть вычисленных значений при выполнении функции обработки выводятся в текстовый файл, для возможности их проверки.

Для написания программы расчета параметров ракурсного рассеяния радиосигналов навигационных спутников использовалась среда программирования Turbo C++.

На рисунке 3 изображён интерфейс программы.

Выберете файл альманаха ГЛОНАСС

 (Файлы альманаха хранятся в директории ... \Coursework\Almanac)

Рассчитать только центр
 Укажите количество точек неоднородности для расчёта

Сура

Долгота пункта "нагрева"

Широта пункта "нагрева"

Магнитное склонение пункта "нагрева"

Магнитное наклонение пункта "нагрева"

Дата

Высота неоднородности

Полуширина диаграммы направленности

Радиус неоднородности

Изменить :
 УГОЛ
 РАДИУС

АОЭ

Долгота пункта приёма

Широта пункта приёма

Ракурсный угол в центре неоднородности

55 50 21.59 48 48 43.58	АОЭ Астрономи-я обсерва-я Энгельгардта
55 50 55.65 48 48 29.32	Циклон
55 47 29.83 49 7 5.23	ФизФак КГУ
55 51 40.78 48 33 19.12	ЭФ КГУ
56 19 30.67 44 1 16.20	НИРФИ
55 28 38.79 37 18 52.70	ИЗМИРАН
48 27 35.46 44 46 15.96	Светлый Яр
46 13 2.52 48 12 13.46	Евпракино

Рис. 3. Главное окно программы

Пользователю программы доступны следующие возможности:

выбирать режим расчёта – расчёт центра неоднородности, или задать количество точек на неоднородности, для которых будет проводиться расчёт;

изменять параметры неоднородности – её высоту и радиус, а также угол диаграммы направленности антенны «нагрева». Параметры взаимозависимы;

выбирать координаты пунктов «нагрева» и приёма радиосигнала из базы данных, а также вносить другие координаты;

устанавливать параметры магнитного поля над пунктом «нагрева» - магнитное наклонение и склонение.

Результаты вычислений выводятся в виде таблиц:

N спутника	t сигнала			
1	1д 9ч 24м	1д 10ч 52м		
2	0д 21ч 10м	0д 22ч 2м		
3	0д 22ч 6м	0д 23ч 57м	1д 21ч 9м	1д 21ч 56м
4	0д 23ч 35м	1д 1ч 43м	1д 22ч 4м	1д 23ч 54м
5	0д 4ч 8м	0д 5ч 59м	1д 1ч 29м	1д 3ч 28м
6	0д 6ч 10м	0д 8ч 18м	1д 4ч 9м	1д 6ч 0м
7	0д 7ч 50м	0д 9ч 50м	1д 6ч 1м	1д 8ч 8м
8	0д 9ч 34м	0д 10ч 58м	1д 7ч 42м	1д 9ч 44м
9	0д 7ч 38м	0д 9ч 46м	1д 6ч 8м	1д 7ч 57м
10	0д 9ч 33м	0д 11ч 32м	1д 7ч 31м	1д 9ч 38м
11	0д 12ч 6м	0д 13ч 57м	1д 9ч 27м	1д 11ч 27м
12				
13	0д 15ч 51м	0д 17ч 51м	1д 14ч 3м	1д 16ч 10м
14	0д 17ч 42м	0д 19ч 3м	1д 15ч 49м	1д 17ч 50м
15	0д 5ч 12м	0д 5ч 52м	1д 17ч 18м	1д 18ч 51м
16	0д 6ч 7м	0д 7ч 58м	1д 5ч 11м	1д 5ч 57м
17	0д 19ч 58м	0д 21ч 48м	1д 17ч 20м	1д 19ч 21м
18	0д 21ч 53м	1д 0ч 0м	1д 19ч 45м	1д 21ч 34м
19	0д 23ч 42м	1д 1ч 44м	1д 21ч 53м	1д 23ч 59м
20	1д 1ч 30м	1д 2ч 58м	1д 23ч 39м	2д 1ч 42м
21	0д 13ч 11м	0д 13ч 57м	2д 1ч 21м	2д 2ч 52м
22	0д 14ч 5м	0д 15ч 54м	1д 13ч 12м	1д 13ч 50м
23	0д 15ч 36м	0д 17ч 44м	1д 14ч 5м	1д 15ч 55м
24	0д 17ч 24м	0д 19ч 24м	1д 15ч 24м	1д 17ч 32м

Рис. 4. Таблица 1

В таблице 1 отображаются времена ракурсного рассеяния сигналов сгруппированные по номерам спутников.

N	N спутника	t сигнала	Длительность	F спутника (МГц)
1	5	0д 3ч 35м	15м	1602.5625
2	5	0д 3ч 56м	0м	1602.5625
3	5	0д 4ч 2м	10м	1602.5625
4	5	0д 4ч 18м	0м	1602.5625
5	5	0д 4ч 23м	4м	1602.5625
6	5	0д 4ч 35м	3м	1602.5625
7	15	0д 4ч 46м	15м	1602
8	5	0д 4ч 49м	2м	1602.5625
9	15	0д 5ч 7м	10м	1602
10	5	0д 5ч 19м	2м	1602.5625
11	15	0д 5ч 27м	0м	1602
12	5	0д 5ч 32м	3м	1602.5625
13	15	0д 5ч 37м	0м	1602
14	5	0д 5ч 42м	0м	1602.5625
15	5	0д 5ч 46м	3м	1602.5625
16	15	0д 5ч 47м	4м	1602
17	6	0д 5ч 49м	44м	1599.75
18	16	0д 5ч 51м	34м	1601.4375
19	5	0д 5ч 55м	8м	1602.5625
20	15	0д 5ч 56м	0м	1602

Рис. 5. Таблица переупорядоченных данных

В таблице выводятся следующие данные:

номер точки, занимаемой спутником в орбитальной группировке;

время начала периода возможного попадания радиоволны после рассеяния в заданный пункт приёма;

длительность периода возможного попадания радиоволны после рассеяния в заданный пункт приёма;

частота радиосигнала в соответствии с частотным планом системы ГЛОНАСС.

Порядок вывода - по возрастанию времени начала периода возможного попадания радиоволны после рассеяния в заданный пункт приёма.

Заключение

Разработан и реализован алгоритм определения угла ракурсного рассеяния для заданных нагревного стенда и приёмного пункта и высоте области неоднородностей с учётом её возможных размеров;

Разработан и реализован алгоритм определения времени прохождения спутниками участка орбиты, на котором радиосигнал спутника, рассеянный на СММ неоднородностях, будет доступен в приемном пункте с учётом различных частей области неоднородностей.

Создана программа оценивания параметров радиосигналов навигационных спутников, рассеянных на анизотропных ионосферных СММ неоднородностях. Программу будет полезно применять при планировании и проведении возможных в будущем экспериментов исследования СММ вытянутых искусственных неоднородностей ионосферной плазмы.

Список литературы

1. Гуревич А.В. Нелинейные явления в ионосфере /А.В.Гуревич //Успехи физических наук.- 2007.- Т.77, №11.- С.1145-1147.
2. Насыров А.М. Рассеяние радиоволн анизотропными ионосферными неоднородностями /А.М.Насыров.- Казань: Изд-во КГУ, 1991.- 152 с.
3. Измерения вариаций полного электронного содержания вдоль геомагнитной широты во время работы нагревного стенда «СУРА». Экспериментальные результаты за 2013 – 2014 гг. /Д.А.Когогин, И.А.Насыров, С.М.Грач и др. //Труды международной байкальской молодежной научной школы по фундаментальной физике и Конференции молодых ученых «Взаимодействие полей и излучения с веществом».- 2015.- С.130-132
4. Дэвис К. Радиоволны в Ионосфере/К.Дэвис: Пер. с англ.- М.: Мир, 1973.- 504 с.
5. Гинзбург В.Л. Распространение электромагнитных волн в плазме /В.Л.Гинзбург.- М.: Наука, 1967.- 684 с.
6. ГЛОНАСС. Интерфейсный контрольный документ. Навигационный радиосигнал в диапазонах L1, L2 (редакция 5.1) /М., 2008.- 74 с.
7. Глобальная навигационная спутниковая система ГЛОНАСС [Электронный ресурс] //Информационно-аналитический центр координатно-временного и навигационного обеспечения.- Режим доступа: <https://www.glonass-iac.ru/guide/gnss/glonass.php>
8. Бартнев В.А. Выбор периода обращения спутников ГЛОНАСС /В.А.Бартнев, Ш.Ш.Кавтрашвили, В.А.Юксеев //Вестник СибГАУ.- 2005.- №3.- С.105-108.

9. Понятов А. Зачем греют небо. Мифы и правда /А.Понятов //Наука и жизнь.- 2013.- № 8.- С.2-13.
10. Савич У. Программирование на С++. 4-е изд. /У.Савич.- СПб: Питер; Киев: Издательская группа ВНУ, 2004.- 781 с.: ил.
11. Культин Н.Б. Основы программирования в Turbo С++ /Н.Б.Культин.- СПб: БХВ-Петербург, 2007.- 464 с.: ил. + CD-ROM

Приложение А

Листинг модуля главной формы Main.cpp

```
//-----
#include <vcl.h>
#pragma hdrstop

#include "Main.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
Tfm_Main *fm_Main;
//-----

    __fastcall Tfm_Main::Tfm_Main(TComponent* Owner)
    : TForm(Owner)
    {
    }

    double shirPr[64], dolgPr[64] ;

    std::string namePr[64],descPr[64];

    double shirN[64], dolgN[64] ;

    std::string nameN[64], descN[64];

    string bufStr;
    // istringstream strPotok;

    extern int dlitM [];
    extern int day_0, dayA, mA, yearA ;

    extern double lambU [ nSp ], tU [ nSp ], tS [ nSp ] [ 200 ];
    extern int    nnT[nSp], maxnnT, nf[nSp] ;

    extern int nomSpl,  timeNom [ maxNom ], nTspl [ maxNom ],
                dlit [ maxNom ];

    extern float      bF [ maxNom ], IF [ maxNom ], bL [ maxNom ],
                IL [ maxNom ];

    /*int compare (const void * a, const void * b)
    {
        return ( (int*)a - (int*)b );
    } */
```

```
//-----
void __fastcall Tfm_Main::alm_bnClick(TObject *Sender)
{
    ifstream potok;

    char buf[62];

    AnsiString fName;

    int nT, i;
    int day, month, year;
    double tmp_dat, t_dat;
    double vis4, vis400;

    if ( alm_otd ->Execute() )
    {
        fName = alm_otd ->FileName;

        potok.open ( fName.c_str(), ios::in );

        potok >> dayA >> mA >> yearA; // считывание даты получение альманаха

        vis4 = yearA % 4 ;
        vis400 = year % 400 ;

        if ( (vis4 == 4) || (vis400 == 400) )
            dlitM [1] = 29; // високосный год
        else
            dlitM [1] = 28; // невисокосный год

        for ( nT = 0; nT < nSp; nT++)
            tU [ nT ] = -1;

        day_0 = dayA - 2;

        p_Day->Text = IntToStr( day_0 ) + "." +
                    IntToStr ( mA ) + "." +
                    IntToStr( yearA );

        for (i = 1; i < mA; i++)
            day_0 += dlitM[i-1] ; // перевод даты в дни

        potok.getline( buf, 31);

        while ( potok )
        {
            potok >> nT; // считывание номера спутника
```

```

nT--;

potok >> nf[nT]; // считывание номера литерной частоты
potok >> tmp_dat; // считывание признак здоровья

if (tmp_dat == 0) // пропуск данных при признаке здоровья = 0
{
    potok >> tmp_dat >> tmp_dat >> tmp_dat;
    potok >> tmp_dat;
    potok >> tmp_dat >> tmp_dat >> tmp_dat;
    potok >> tmp_dat;
    potok >> tmp_dat >> tmp_dat >> tmp_dat >> tmp_dat >> tmp_dat;
    potok >> tmp_dat >> tmp_dat >> tmp_dat;
    potok.getline( buf, 31);
    continue;
}

potok >> day >> month >> year; // дата

for (i = 1; i < month; i++)
    day = day + dlitM[i-1] ; // перевод даты в дни

day -= day_0;

if ( day < 0 )
    ShowMessage ( " Странно " + IntToStr ( day ) ); // проверка даты

potok >> t_dat; // считывание времени прохождения восходящего узла
t_dat += day * 24 * 3600;
if ( t_dat > tU [ nT ] ) // считывание остальных переменных
{
    tU [ nT ] =t_dat;
    potok >> tmp_dat >> tmp_dat >> tmp_dat;
    potok >> lambU [ nT ];
}
else
    potok >> tmp_dat >> tmp_dat >> tmp_dat >> tmp_dat;

potok >> tmp_dat >> tmp_dat >> tmp_dat >> tmp_dat >> tmp_dat;

potok >> dayA >> mA >> yearA;
potok.getline( buf, 31);

}
potok.close();

}
else

```

```

        ShowMessage (" Необходимо выбрать файл ! " );
    }
    //-----

void __fastcall Tfm_Main::b_obrabClick(TObject *Sender)
{
    double ang;

    TdanRakurs danRakurs;

    danRakurs.flag = !cb_centр ->Checked;

    danRakurs.LNs = StrToFloat ( p_LN ->Text ); // считывание данных заданных в про-
грамме
    danRakurs.BNs = StrToFloat ( p_BN ->Text );
    danRakurs.hNs = StrToFloat ( p_hN ->Text );
    danRakurs.RNs = StrToFloat ( p_RN ->Text );
    danRakurs.DAs = StrToFloat ( p_DA ->Text );
    danRakurs.LRs = StrToFloat ( p_LR ->Text );
    danRakurs.BRs = StrToFloat ( p_BR ->Text );
    danRakurs.incMs = StrToFloat ( p_incM ->Text );
    danRakurs.decMs = StrToFloat ( p_decM ->Text );

    ang = obrab ( danRakurs ); // запуск функции обработки

    p_ugol ->Text = FloatToStr ( ang * Rad2Deg );

    int atS, dtS , htS, mtS, stS;

    fm_win_tab ->tab_rez ->ColCount = maxnnT + 1; // таблица результатов расчёта

    fm_win_tab ->tab_rez ->Cells [0][0] = "N спутника";
    fm_win_tab ->tab_rez ->Cells [1][0] = "t сигнала";
    //->tabSort ->Cells [1][0] = "N спутника";

    for (int i = 0; i < nSp; i++)
    {
        fm_win_tab ->tab_rez ->Cells [ 0 ] [ i+1 ] = IntToStr ( i + 1 );
        for ( int j = 0; j < nnT [ i ] ; j++)
        {
            atS = int (tS[i][j]) ;

            dtS = atS / (3600 * 24) ;
            atS -= dtS* (3600 * 24);
            htS = atS / 3600 ;
        }
    }
}

```

```

        atS -= htS* 3600;
        mtS = atS / 60 ;

        fm_win_tab ->tab_rez ->
            Cells [ j + 1 ] [ i+1 ] = FloatToStr ( dtS ) + "д" + " " + Float-
ToStr ( htS ) + "ч"
            + " " + FloatToStr ( mtS ) + "м" ;
    }

}
fm_win_tab->Show();

//   qsort (timeNom, maxNom, sizeof(timeNom[0]), compare);

int tmin, n, iMin, lastMin= -1;

int NomSort[maxNom];

fm_tabSort ->tabSort ->RowCount = nomSpl;

fm_tabSort ->tabSort ->Cells [0][0] = "N";
fm_tabSort ->tabSort ->Cells [1][0] = "N спутника";
fm_tabSort ->tabSort ->Cells [2][0] = "t сигнала";
fm_tabSort ->tabSort ->Cells [3][0] = "Длительность";
fm_tabSort ->tabSort ->Cells [4][0] = "F спутника (МГц)";

double frec [nSp];
int nT;

for (nT = 0; nT <= nSp; nT++)
    frec [nT] = 1602 + nf[nT]*0.5625;

for ( n = 0; n < nomSpl; n++)
{
    tmin = INT_MAX;
    for ( int i = 0; i < nomSpl; i++)
    {
        if ( (timeNom[ i ] > lastMin) && ( timeNom[i] < tmin ) )
            { tmin = timeNom[ i ]; iMin = i; }
        else
            if ( (timeNom[ i ] == lastMin) && ( i > NomSort[ n - 1 ] ) )
                { tmin = timeNom[ i ]; iMin = i;
                break;
            }
    }
    NomSort [ n ] = iMin; lastMin = tmin;
}

double f;
int tmp;
for (int i = 0; i < nomSpl; i++)

```

```

{

    tmp = nTspl [ NomSort[i] ] ;

    fm_tabSort ->tabSort -> Cells [ 0 ] [ i+1 ] = IntToStr ( i + 1 );
    fm_tabSort ->tabSort -> Cells [ 1 ] [ i+1 ] =
        IntToStr ( nTspl [ NomSort[i] ] + 1 );

        atS = timeNom [ NomSort[i] ];

        dtS = atS / (3600 * 24) ;
        atS -= dtS * (3600 * 24);
        htS = atS / 3600 ;
        atS -= htS * 3600;
        mtS = atS / 60 ;

    fm_tabSort ->tabSort -> Cells [ 2 ] [ i+1 ] =
        FloatToStr ( dtS ) + "д" + " " + FloatToStr ( htS ) + "ч"
        + " " + FloatToStr ( mtS ) + "м" ;

        atS = dlit [ NomSort[i] ] ;
        mtS = atS / 60 ;

    fm_tabSort ->tabSort -> Cells [ 3 ] [ i+1 ] =
        FloatToStr ( mtS ) + "м" ;

    f = frec [ tmp ];

    fm_tabSort ->tabSort -> Cells [ 4 ] [ i+1 ] =
        FloatToStr ( f );
}

    fm_tabSort ->Show();

}

//-----

void __fastcall Tfm_Main::R_DBClick(TObject *Sender)
{
    char cstrBuf [ 64 ];
    ifstream fpotok;
    int i, n;

    if ( alm_otd ->Execute() )
    {
        mem->Lines->Add(" Пункты приёма ");
        //bufStr = alm_otd ->FileName;
        mem->Lines->LoadFromFile( alm_otd ->FileName ) ;
    }
}

```



```

        //fpotok.open ( alm_otd ->FileName.c_str() );
        //fpotok.open ( "E:\\DISTR\\ttt\\PUNKTYP.txt" );
double tmpBuf;

double s_tmp1, s_tmp2, s_tmp3;
double d_tmp1, d_tmp2, d_tmp3;

int nPunkt = mem->Lines->Count;

for ( i = 0; i < nPunkt; i++ )
{
    istrstream ( mem->Lines->Strings[i].c_str() ) >> s_tmp1
        >> s_tmp2 >> s_tmp3 >> d_tmp1 >> d_tmp2 >> d_tmp3
        >> namePr[i] >> descPr[i];

    shirPr[i] = s_tmp1 + (s_tmp2 / 60) + (s_tmp3 / 3600);
    dolgPr[i] = d_tmp1 + (d_tmp2 / 60) + (d_tmp3 / 3600);

    R_cbox->Items->Add(namePr[i].c_str());

}
R_cbox->ItemIndex = 0;
p_LR->Text = dolgPr[0];
p_BR->Text = shirPr[0];
}
else
    ShowMessage ( " Необходимо выбрать файл ! " );
}
//-----

void __fastcall Tfm_Main::R_cboxChange(TObject *Sender)
{
    int i = R_cbox->ItemIndex;

    p_LR->Text = dolgPr[i];
    p_BR->Text = shirPr[i];
}
//-----

void __fastcall Tfm_Main::N_DBClick(TObject *Sender)
{
    char cstrBuf [ 64 ];
    ifstream fpotok;
    int i, n;

    if ( alm_otd ->Execute() )
    {
        mem->Lines->Add(" Пункты нагрева ");
        mem->Lines->LoadFromFile( alm_otd ->FileName );
    }
}

```

```

double tmpBuf;

double s_tmp1, s_tmp2, s_tmp3;
double d_tmp1, d_tmp2, d_tmp3;

int nPunkt = mem->Lines->Count;

for ( i = 0; i < nPunkt; i++ )
{
    istringstream ( mem->Lines->Strings[i].c_str()) >> s_tmp1
        >> s_tmp2 >> s_tmp3 >> d_tmp1 >> d_tmp2 >> d_tmp3
        >> nameN[i] >> descN[i];

    shirN[i] = s_tmp1 + (s_tmp2 / 60) + (s_tmp3 / 3600);
    dolgN[i] = d_tmp1 + (d_tmp2 / 60) + (d_tmp3 / 3600);

    N_cbox->Items->Add(nameN[i].c_str());
}
N_cbox->ItemIndex = 0;
p_LN->Text = dolgN[0];
p_BN->Text = shirN[0];
}
else
    ShowMessage ( " Необходимо выбрать файл ! " );
}
//-----

void __fastcall Tfm_Main::N_cboxChange(TObject *Sender)
{
    int i = N_cbox->ItemIndex;

    p_LN->Text = dolgN[i];
    p_BN->Text = shirN[i];
}
//-----

void __fastcall Tfm_Main::b_wind2Click(TObject *Sender)
{
    fm_win_pic->Show();

    fm_win_pic->Canvas->Ellipse(4, 4, 302, 302) ;
}
//-----

void __fastcall Tfm_Main::p_DACHange(TObject *Sender)
{
    double DA;
    TdanRakurs danRakurs;
}

```

```

danRakurs.DAs = StrToFloat ( p_DA ->Text );

if ( RG ->ItemIndex == 0 )
{
    DA = danRakurs.DAs * Deg2Rad;
    danRakurs.RNs = StrToFloat ( p_hN ->Text ) * tan( DA );
    p_RN ->Text = FloatToStr( danRakurs.RNs );
}
}
//-----

void __fastcall Tfm_Main::p_RNChange(TObject *Sender)
{
    double RN, DA, hN;
    TdanRakurs danRakurs;

    danRakurs.RNs = StrToFloat ( p_RN ->Text );
    danRakurs.hNs = StrToFloat ( p_hN ->Text );

    if ( RG ->ItemIndex == 1 )
    {
        RN = danRakurs.RNs;
        hN = danRakurs.hNs;
        DA = atan (RN / hN);
        danRakurs.DAs = DA * Rad2Deg;
        p_DA ->Text = FloatToStr( danRakurs.DAs );
    }
}
//-----

void __fastcall Tfm_Main::RGClick(TObject *Sender)
{
    if ( RG ->ItemIndex == 1 )
    {
        p_RN ->Enabled = true ; p_DA ->Enabled = false;
    }
    else
    {
        p_RN ->Enabled = false ; p_DA ->Enabled = true;
    }
}
//-----

void __fastcall Tfm_Main::p_hNChange(TObject *Sender)
{
    double RN, DA, hN;
    TdanRakurs danRakurs;

    danRakurs.hNs = StrToFloat ( p_hN ->Text );
    hN = danRakurs.hNs ;

    if ( RG ->ItemIndex == 0 )

```

```

    {
        danRakurs.DAs = StrToFloat ( p_DA ->Text );
        DA = danRakurs.DAs * Deg2Rad;
        danRakurs.RNs = StrToFloat ( p_hN ->Text ) * tan( DA );
        p_RN ->Text = FloatToStr( danRakurs.RNs );
    }

    if ( RG ->ItemIndex == 1 )
    {
        danRakurs.RNs = StrToFloat ( p_RN ->Text );
        RN = danRakurs.RNs;
        DA = atan (RN / hN);
        danRakurs.DAs = DA * Rad2Deg;
        p_DA ->Text = FloatToStr( danRakurs.DAs );
    }
}
//-----

void __fastcall Tfm_Main::cb_centrClick(TObject *Sender)
{
    if (cb_centr ->Checked)
    {
        L_nkT ->Enabled = false;
        p_nkT ->Enabled = false;
    }
    else
    {
        L_nkT ->Enabled = true;
        p_nkT ->Enabled = true;
    }
}
//-----

```

Приложение Б

Листинг модуля основных подпрограмм вычислений vych.cpp

```
//-----
#pragma hdrstop

#include "vych.h"
//-----

#pragma package(smart_init)

using namespace std;

extern int dlitM [];
extern int day_0, dayA, mA, yearA ;

extern double lambU [ nSp ], tU [ nSp ],
             tS [ nSp ] [ 200 ], tSF [ nSp ] [ 200 ], tSL [ nSp ] [ 200 ];
extern int nnT[ nSp ], maxnnT, nnTp[ nSp ], maxnnTp ;

extern int nomSpl,   timeNom [ maxNom ], nTspl [ maxNom ],
             dlit [ maxNom ];

extern float        bF [ maxNom ], IF [ maxNom ], bL [ maxNom ],
             IL [ maxNom ];

double obrab ( TdanRakurs danRakurs )
{
    double LN, BN, hN, LR, BR, incM, decM, rN;

    double RyoZ,
           XI[nKT + 1], YI[nKT + 1], ZI[nKT + 1],
           Xrec, Yrec, Zrec, XoM, YoM, ZoM,
           deltA[nKT + 1], oldDel[nKT + 1], povVokrZ;
    double XM1, XM, YM, ZM, XR, YR, ZR,
           XS[nKT + 1], YS[nKT + 1], ZS[nKT + 1], t, tFirst, tLast, Rox;
    const double magM = 10000;
    double angle[nKT + 1], ang, L, B, alpha, grad10;

    double Xsat, Ysat, Zsat, betF, betL, lamF, lamL,
           betC, lamC, betCg, lamCg;

    double L1[ nSp ], L2[nSp][nKT], Lrez, trez[nSp][nKT], Ldir, tdir[nSp],
           tmax[nSp][nKT + 1], tmin[nSp][nKT + 1];

    double const c = 300000; // км/с
```

```
ofstream potok, potok2, pot3, pot4;
```

```
potok.open ( "tttt.txt", ios::out );
potok2.open ( "tttt2.txt", ios::out );
pot3.open ( "t3.txt", ios::out );
pot4.open ( "t4.txt", ios::out );
```

```
// перевод угловых величин из градусов в радианы
```

```
LN = danRakurs.LNs * Deg2Rad;
BN = danRakurs.BNs * Deg2Rad;
LR = danRakurs.LRs * Deg2Rad;
BR = danRakurs.BRs * Deg2Rad;
incM = danRakurs.incMs * Deg2Rad;
decM = danRakurs.decMs * Deg2Rad;
```

```
hN = danRakurs.hNs;
rN = danRakurs.RNs;
bool flag25 = danRakurs.flag;
```

```
int nT; // номер спутника
nomSpl = 0; // номер сплошной
```

```
// incM = 72 ,decM = 12 ;
// main_form ->s_cbox ->Items ->Clear();
```

```
double rhN = rZ + hN;
```

```
ZI[0] = ( rhN ) * sin(BN);
Ryoz = ( rhN ) * cos(BN); // координаты неоднородности
XI[0] = Ryoz * cos(LN); // в геоцентрической СК
YI[0] = Ryoz * sin(LN);
```

```
int nk;
```

```
if ( flag25 )
```

```
for ( nk = 1 ; nk <= nKT ; nk++ )
{
```

```
double az = nk * 2 * M_PI / nKT;
XoM = rN * cos( az ); // компоненты смещения из центра
YoM = rN * sin( az ); // в местной СК
ZoM = 0;
```

```
XM1 = XoM*cos( M_PI_2 - BN ) + ZoM*sin( M_PI_2 - BN ); //
```

поворот местной системы координат

```
ZM = -XoM*sin( M_PI_2 - BN ) + ZoM*cos( M_PI_2 - BN ); // вокруг
```

оси Y и оси Z

```
XM = XM1*cos( LN ) - YoM*sin( LN ); // компоненты
YM = XM1*sin( LN ) + YoM*cos( LN ); // в геоцентрической СК
```

```
XI[nk] = XI[ 0 ] + XM;
```

```

        YI[nk] = YI[ 0 ] + YM;
        ZI[nk] = ZI[ 0 ] + ZM;
    }

Zrec = rZ * sin(BR);
Ryoz = rZ * cos(BR); // координаты приёмного пункта
Xrec = Ryoz * cos(LR); // в геоцентрической СК
Yrec = Ryoz * sin(LR);

XoM = magM * cos ( incM ) * cos( M_PI - decM ); // компоненты вектора маг-
                                                нитного поля
YoM = magM * cos ( incM ) * sin( M_PI - decM ); // в местной СК
ZoM = magM * sin( - incM );

XM1 = XoM * cos( M_PI_2 - BN ) + ZoM * sin( M_PI_2 - BN ); // поворот
                                                           местной системы координат
ZM = -XoM * sin( M_PI_2 - BN ) + ZoM * cos( M_PI_2 - BN ); // вокруг оси У и
оси Z

XM = XM1 * cos( LN ) - YoM * sin( LN ); // компоненты вектора магнитного
                                           поля
YM = XM1 * sin( LN ) + YoM * cos( LN ); // в геоцентрической СК

if ( cross( XM, YM, ZM, XI[0], YI[0], ZI[0], &betC, &lamC ) )
    { betCg = betC * Rad2Deg; lamCg = lamC * Rad2Deg; }
else
    betCg = 100;

pot3 << " btCg = " << betCg << " lamCg = " << lamCg << "\n\n";

XS[0] = Xrec - XI[0]; // компоненты вектора рассеивания
YS[0] = Yrec - YI[0]; // в геоцентрической СК
ZS[0] = Zrec - ZI[0];

if ( flag25 )
    for ( nk = 1 ; nk < nKT + 1 ; nk++ )
        {
            XS[nk] = Xrec - XI[nk]; // компоненты вектора рассеивания
            YS[nk] = Yrec - YI[nk]; // в геоцентрической СК
            ZS[nk] = Zrec - ZI[nk];

            L1[nk] = sqrt ( XS[nk]*XS[nk] + YS[nk]*YS[nk] + ZS[nk]*ZS[nk] );
        }

double cosA;

cosA = (XS[0]*XM + YS[0]*YM + ZS[0]*ZM)
    /sqrt ((XS[0]*XS[0] + YS[0]*YS[0]
    + ZS[0]*ZS[0])*(XM*XM + YM*YM + ZM*ZM));

```

```

angle[0] = acos( cosA );    //угол между векторами магнитного поля и рассеи-
                             вания
if ( flag25 )
for ( nk = 1 ; nk < nKT + 1 ; nk++ )
{
    cosA = (XS[nk]*XM + YS[nk]*YM + ZS[nk]*ZM)
           /sqrt ((XS[nk]*XS[nk] + YS[nk]*YS[nk] +
                   ZS[nk]*ZS[nk])*(XM*XM + YM*YM + ZM*ZM));

    angle[nk] = acos( cosA ); // компоненты вектора рассеивания
}

int nkV = 1;

if ( flag25 )
    nkV = nKT + 1;

for ( nT = 0; nT < nSp; nT++ )
{
    nnT [ nT ] = 0;
    nnTp [ nT ] = 0;
}

maxnnT = 0;
maxnnTp = 0;

for ( nT = 0; nT < nSp; nT++ )
{
    if (tU[ nT ] < 0)
        continue;

    //main_form ->s_cbox -> Items ->Add(IntToStr ( nT ) );
    for ( nk = 0; nk < nkV; nk++ )
        oldDel[nk] = 0;

    bool signalp = false;

    for ( grad10 = -72000; grad10 <= 72000 ; grad10++ )
    {
        alpha = 0.01 * grad10 * Deg2Rad;    // градусы -> радианы

        t = (grad10*Ts)/36000 ;              // время возможного совпаде-
                                             ния углов

        Rox = Rorb*sin(alpha);
        Xsat = Rorb*cos(alpha); // координаты спутников
        Ysat = Rox *cos(beta);
        Zsat = Rox *sin(beta);
    }
}

```



```

povVokrZ = M_PI * ( lambU[nT] - t / (3600*12) );

XM1 = Xsat*cos( povVokrZ ) - Ysat*sin( povVokrZ ); //
Ysat = Xsat*sin( povVokrZ ) + Ysat*cos( povVokrZ ); //
Xsat = XM1;

if ( pass(Xrec, Yrec, Zrec, Xsat, Ysat, Zsat) )
{
  Ldir = sqrt( (Xsat-Xrec)*(Xsat-Xrec) + (Ysat-Yrec)*
              (Ysat-Yrec) + (Zsat-Zrec)*(Zsat-Zrec) );
  tdir[nT] = Ldir/ c;
}
else
  tdir[nT] = 0;

double aDelta, minDelta = 99999;

t += tU[nT];

for (nk=0; nk < nkV; nk++)
{
  XR = Xsat - XI[nk] ; // компоненты вектора
                        // излучения
  YR = Ysat - YI[nk] ;
  ZR = Zsat - ZI[nk] ;

  L2[nT][nk] = sqrt ( (XR)*(XR) + (YR)*(YR) + (ZR)*(ZR) );

  Lrez = L1[nk] + L2[nT][nk];
  trez[nT][nk] = Lrez / c ;

  ang = acos(
            (XR*(-XM) + YR*(-YM) + ZR*(-ZM))
            /sqrt ((XR*XR + YR*YR + ZR*ZR)*(XM*XM + YM*YM
            + ZM*ZM)) );
  // угол между векторами излучения и магнитного поля

  deltA[nk] = angle[nk] - ang ;
  aDelta = abs(deltA[nk]);
  if ( aDelta < minDelta )
    minDelta = aDelta;

//
  if ( (aDelta < 0.001) || (deltA[nk] * oldDel[nk] < 0) )
  if ( (aDelta == 0.00000) || (deltA[nk] * oldDel[nk] < 0) )
  {
    if ( nk == 0 )
    {
      nnT[nT] ++;
      if ( nnT [ nT ] > maxnnT )
        maxnnT = nnT [ nT ];
    }
  }
}

```

```

        tS[nT] [ nnT[nT] - 1 ] = t;
    }
    tLast = t;
    betL = Rad2Deg * asin ( Zsat / Rorb );
    lamL = Rad2Deg * atan2 ( Ysat, Xsat );

    pot3 << " " << betL;
    pot4 << " " << lamL;

    if ( ! signalp )
    {
        tFirst = t;
        betF = betL;
        lamF = lamL;
        signalp = true;
    }
    potok << "\n nT = " << nT << " " << t << " aD = " <<
aDelta ;
    }

    oldDel[nk] = deltA[nk];
}
if ( signalp && ((minDelta > 2*Deg2Rad) || ((t - tLast)>250)) )
{
    nnTp[nT] ++;
    if ( nnTp [ nT ] > maxnnTp )
        maxnnTp = nnTp [ nT ];

    tSF[nT] [ nnTp[nT] - 1 ] = tFirst;
    tSL[nT] [ nnTp[nT] - 1 ] = tLast;

    //pot3 << " " << betF << " " << betL;
    //pot4 << " " << lamF << " " << lamL;

    potok2 << "\n nT= " << nT
        << " dt= " << tLast - tFirst << " tF= " << tFirst <<
        " tL= " << tLast << " " << t << " aD = " <<
aDelta ;

    potok2 << "\n betF= " << betF << " lamF= " << lamF
        << " betL= " << betL << " lamL= " << lamL;

    timeNom [ nomSpl ] = int ( tFirst );
    nTspl [ nomSpl ] = nT;
    dlit [ nomSpl ] = int ( tLast-tFirst );
    bF [ nomSpl ] = betF;
    lF [ nomSpl ] = lamF;

    bL [ nomSpl ] = betL;
    lL [ nomSpl ] = lamL;

    nomSpl ++;

```

```

        signalp = false;
    }
}
}
potok.close();
potok2.close(); pot3.close(); pot4.close();
return angle[0];
}

```

double rakurs (TdanRakurs danRakurs)

```

// double LN, double BN, double hN, double LR, double BR)
{

```

```

    double LN, BN, hN, LR, BR, incM, decM;

```

```

    double RyoZ,XI,YI,ZI,Xrec,Yrec,Zrec,XoM,YoM,ZoM, delTA, oldDel, povVokrZ;
    double XM1, XM, YM, ZM, XR, YR, ZR, XS, YS, ZS, t, Rox;
    const double magM = 10000;
    double angle, ang, /*incM = 72 ,decM = 12 , */ L,B,alpha;
    double Xsat, Ysat, Zsat;

```

```

// перевод угловых величин из градусов в радианы

```

```

LN = danRakurs.LNs * Deg2Rad;
BN = danRakurs.BNs * Deg2Rad;
LR = danRakurs.LRs * Deg2Rad;
BR = danRakurs.BRs * Deg2Rad;
incM = danRakurs.incMs * Deg2Rad;
decM = danRakurs.decMs * Deg2Rad;

```

```

hN = danRakurs.hNs;

```

```

ZI = ( rZ + hN ) * sin(BN);
RyoZ = ( rZ + hN ) * cos(BN); // координаты неоднородности
XI = RyoZ * cos(LN); // в геоцентрической СК
YI = RyoZ * sin(LN);

```

```

Zrec = rZ * sin(BR);
RyoZ = rZ * cos(BR); // координаты приёмного пункта
Xrec = RyoZ * cos(LR); // в геоцентрической СК
Yrec = RyoZ * sin(LR);

```

```

XoM = magM * cos ( incM ) * cos( M_PI - decM ); // компоненты вектора магнитного
поля

```

```

YoM = magM * cos ( incM ) * sin( M_PI - decM ); // в местной СК
ZoM = magM * sin( - incM );

```

```

XM1 = XoM * cos( M_PI_2 - BN ) + ZoM * sin( M_PI_2 - BN ); // поворот мест-
ной системы координат
ZM = -XoM * sin( M_PI_2 - BN ) + ZoM * cos( M_PI_2 - BN ); // вокруг оси Y и оси Z

```

```

XM = XM1*cos( LN ) - YoM*sin( LN ); // компоненты вектора магнитного поля
YM = XM1*sin( LN ) + YoM*cos( LN ); // в геоцентрической СК

XS = Xrec - XI ; // компоненты вектора рассеивания
YS = Yrec - YI ; // в геоцентрической СК
ZS = Zrec - ZI ;

double cosA;

cosA = (XS*XM + YS*YM + ZS*ZM)/sqrt ((XS*XS + YS*YS + ZS*ZS)*(XM*XM +
YM*YM + ZM*ZM));

angle = acos( cosA ); //угол между векторами магнитного поля и рассеивания

return angle;
}

```

Приложение В

Листинг вспомогательных подпрограмм dop.cpp

```
//-----
#pragma hdrstop

#include "dop.h"
//-----
#pragma package(smart_init)

bool cross
( double px, double py, double pz,
  double x1, double y1, double z1,
  double *rezb, double *rezl )
{
  bool logRez;
  // x^2 + y^2 + z^2 = r^2
  //
  // (x-x1)/px = (y-y1)/py = (z-z1)/pz
  //
  // Выражаем уравнения для координат,
  // подставляем в уравнение сферы,
  // находим координаты пересечения
  // вектора магнитного поля и
  // сферической поверхностью
  // с радиусом орбиты спутника

  double r2 = Rorb * Rorb, rx, ry, rz, irx, iry, irz,
    rx2, ry2, rz2 ;
  double D, a, b, c, p1, p2, b2;
  double x2 = x1*x1, y2 = y1*y1, z2 = z1*z1,
    s1, s2, s3, s4, tmp, tmp2;
  double px2 = px*px, py2 = py*py, pz2 = pz*pz;

  rx = (py2*py*x1 + py*pz2*x1 - px*py2*y1 - px*py*pz*z1 -
  px*sqrt(py2*(pz2*(r2 - x2 - y2) + 2*py*pz*y1*z1 +
  2*px*x1*(py*y1 + pz*z1) + py2*(r2 - x2 - z2) +
  px2*(r2 - y2 - z2))))/
  (py*(px2 + py2 + pz2));

  if ( ( rx - x1 ) * px > 0 )
    rx = (py2*py*x1 + py*pz2*x1 - px*py2*y1 - px*py*pz*z1 +
  px*sqrt(py2*(pz2*(r2 - x2 - y2) + 2*py*pz*y1*z1 +
  2*px*x1*(py*y1 + pz*z1) + py2*(r2 - x2 - z2) +
  px2*(r2 - y2 - z2))))/
```

```

(py*(px2 + py2 + pz2));

rx2 = rx*rx;

ry = -((px*py*x1 - px2*y1 - pz2*y1 + py*pz*z1 +
sqrt(py2*(pz2*(r2 - x2 - y2) + 2*py*pz*y1*z1 +
2*px*x1*(py*y1 + pz*z1) + py2*(r2 - x2 - z2) +
px2*(r2 - y2 - z2)))))/(px2 + py2 + pz2));
if ( ( ry - y1 ) * py > 0 )
    ry = -((px*py*x1 - px2*y1 - pz2*y1 + py*pz*z1 -
sqrt(py2*(pz2*(r2 - x2 - y2) + 2*py*pz*y1*z1 +
2*px*x1*(py*y1 + pz*z1) + py2*(r2 - x2 - z2) +
px2*(r2 - y2 - z2)))))/(px2 + py2 + pz2));

ry2 = ry*ry;

rz = (-((px*py*pz*x1) - py2*pz*y1 + px2*py*z1 + py2*py*z1 +
pz*sqrt(py2*(pz2*(r2 - x2 - y2) + 2*py*pz*y1*z1 +
2*px*x1*(py*y1 + pz*z1) + py2*(r2 - x2 - z2) +
px2*(r2 - y2 - z2)))))/(py*(px2 + py2 + pz2)) ;

if ( ( rz - z1 ) * pz > 0 )
    rz = (-((px*py*pz*x1) - py2*pz*y1 + px2*py*z1 + py2*py*z1 -
pz*sqrt(py2*(pz2*(r2 - x2 - y2) + 2*py*pz*y1*z1 +
2*px*x1*(py*y1 + pz*z1) + py2*(r2 - x2 - z2) +
px2*(r2 - y2 - z2)))))/(py*(px2 + py2 + pz2)) ;

rz2 = rz*rz;

tmp = (rx2+ry2+rz2)/r2 ;

logRez = true;

if (!( (0.999 < tmp) && (tmp < 1.001) ) )
    logRez = false;

*rezl = atan2( ry, rx );// XYZ_TO_LB
*rezb = asin( rz / Rorb);

tmp = (rx-x1)/px; // подстановка в уравнение
tmp2 = (ry-y1)/py;

if ( (!(0.999 < tmp/tmp2) && (tmp/tmp2 < 1.001) ) )
    logRez = false;

tmp2 = (rz-z1)/pz;

if ( (!(0.999 < tmp/tmp2) && (tmp/tmp2 < 1.001) ) )
    logRez = false;

```

```
        return logRez;
    }

bool pass
( double x1, double y1, double z1,
  double x2, double y2, double z2)
{
    // x1 = Xsat , x2 = Xrec ;
    //
    // Проверка пересечения
    // радиосигнала с
    // Земным шаром

    bool logRez;
    double sp;

    logRez = true;

    sp = (x1-x2)*x2 + (y1-y2)*y2 + (z1-z2)*z2 /sqrt( ((x1-x2)*(x1-x2)+
      (y1-y2)*(y1-y2)+(z1-z2)*(z1-z2))*(x2*x2 + y2*y2 + z2*z2) );

    if (sp < 0 )
        logRez = false;

    return logRez;
}
```