

Лабораторная работа № 2

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ И МЕТОД СТАТИСТИЧЕСКИХ ИСПЫТАНИЙ

Чем обычно занимаются научные работники? Проводят различные эксперименты, а затем анализируют их результаты. Проводить натурные эксперименты (с изготовлением стэнда, лабораторной установки) довольно дорого, поэтому большинство ученых занимаются имитационным моделированием. Пишется программа, которая моделирует некий процесс, который происходит в реальности. Затем эту модель можно испытывать, чаще всего с помощью метода статистических испытаний. Этот подход мы и попробуем применить к анализу формата передачи данных АРТ.

Я максимально упростил содержание работы. Я выдаю Вам код, который представляет собой имитационную модель передачи пачки синхроимпульсов по каналу АРТ (файл [code_lab2.py](#)). Она моделирует передачу пачки прямоугольных импульсов от спутника на Землю с помощью амплитудной и частотной модуляции (АМ/ЧМ) и выдает на выход ошибку оценки позиции этой пачки на Земле (ошибка в отсчетах, на сколько отсчетов сдвинута оцененная позиция относительно истинной). Вам необходимо использовать метод статистических испытаний, чтобы оценить насколько хорошо решается задачи оценки положения пачки при различных параметрах (в части, от уровня шума).

Метод статистических испытаний необходим, так как Ваша имитационная модель содержит элементы случайности, в частности, шум, который добавляется при распространении сигнала. Т. е. однократный запуск модели не дает нам полного представления о ее свойствах. В методе статистических испытаний мы многократно запускаем нашу имитационную модель при разных реализациях случайных факторов, а затем подсчитываем статистику той величины, которая нас интересует. В нашем случае мы попробуем оценить вероятность правильного детектирования (детектирование будем считать правильным, если ошибка по модулю меньше заданного числа) при заданных параметрах системы и шума.

Задания работы (статистические испытания) можно, в принципе, выполнить и не вникая сильно в суть работы имитационной модели, а обращаясь с ней как с “черным ящиком”. Хотя это может быть интересно, поэтому я добавил далее описание основных моментов, а также запишу видео с дополнительными устными пояснениями.

2.1. Необходимые сведения

2.1.1. Сведения о формате АРТ

Студенты, которые на первом курсе изучали курс “Основы программирования” уже сталкивались с форматом АРТ (automatic picture transmission) в рамках лабораторной работы. Те, кто хочет узнать большей

информации о нем могут обратиться к описанию упомянутой лабораторной работы
(<https://drive.google.com/drive/folders/123ieX5bCAZherv-TtJIRL9LCDJK6IyON>)

Формат АРТ используется для передачи снимков с метеорологических спутников на Землю. Снимки могут использоваться для предсказания погоды. Это довольно старый аналоговый формат. Но три старых американских метеорологических спутника, которые его используют (NOAA-15, 18, 19), еще продолжают работать. В новых метеоспутниках для передачи данных используются более современные цифровые форматы передачи данных. Нам формат АРТ удобен тем, что мы можем принимать реальные сигналы (для этого не нужно сложного оборудования) для учебных целей, а также он хорошо подходит для моделирования (не сильно сложный).

Последовательность основных операций при передаче и приеме спутниковых снимков в формате АРТ показана на рис. 1. Исходным объектом для передачи является цифровое изображение Земли (с уменьшенным разрешением), сформированное радиометром AVHRR на спутнике. Для передачи на Землю оно преобразуется в одномерный аналоговый сигнал согласно формату АРТ.

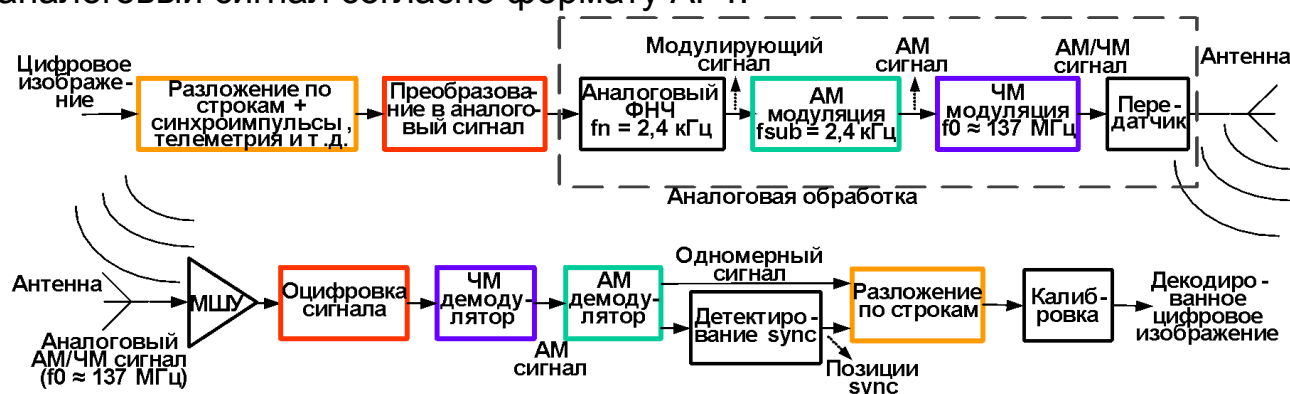


Рис. 1. Формирование и обработка сигнала в формате АРТ

Цифровое изображение (снимок) раскладывается по строкам, каждая строка дополняется вспомогательной информацией. Полученный цифровой сигнал преобразуется в аналоговый, где каждой строке формата АРТ (строке изображения вместе со вспомогательной информацией) соответствует длительность $0,5$ с. Этим сигналом сначала модулируется поднесущее колебание (subcarrier) с частотой $f_{sub} = 2400$ Гц с помощью аналоговой амплитудной модуляции. Перед АМ модуляцией информационный сигнал пропускается через аналоговый фильтр нижних частот (ФНЧ) с частотой среза (natural frequency) $f_n = 2400$ Гц, так как при АМ модуляции ширина боковой полосы не может быть больше, чем несущая частота (в данном случае поднесущая). Затем сформированный

сигнал используется для модулирования по частоте колебания несущей частоты (carrier frequency) $f_0 \approx 137$ МГц. Девияция частоты составляет 17 кГц. Используется круговая правовращательная поляризация передающей антенны, мощность передатчика составляет 37 дБмВт (5 Вт).

При передаче со спутника на наземную станцию приема сигнал ослабляется и подвергается искажениям. Учитывая, что плотность потока энергии обратно пропорциональна расстоянию (когда спутник пролетает прямо над местом приема, оно составляет порядка 800 км), уровень принимаемого сигнала на выходе антенны будет очень слабым. Хороший АРТ приемник должен иметь чувствительность порядка 0,2...0,3 мкВ, отношение сигнал/шум (с/ш) должно быть больше 20 дБ. Перед приемником полезно включить малошумящий усилитель (МШУ). Учитывая, что девияция частоты ЧМ сигнала составляет 17 кГц, а также наличие доплеровского сдвига частоты, когда спутник приближается или отдаляется от приемной станции, полоса приемника должна составлять порядка 40 кГц.

Усиленный сигнал поступает на вход приемника. Приемник может быть выполнен различными способами. Например, можно использовать чисто аналоговый демодулятор, осуществляющий ЧМ и АМ демодуляцию с помощью аналоговых схем, после чего выполнить оцифровку. Однако в последнее время получила распространение технология приема, когда сигнал оцифровывается, а затем обработка осуществляется программным способом (именно этот вариант показан на рис. 1).

После осуществления ЧМ и АМ демодуляции полученный сигнал нужно преобразовать в изображение. Для этого нужно определить начало каждой строки, с этой целью в каждую строку формата АРТ добавлены пакеты синхроимпульсов (далее будем их обозначать словом sync).

2.1.2. Постановка задачи

Алгоритм декодирования изображения в формате АРТ основан на оценке положения пачек синхроимпульсов, после чего происходит разложение сигнала по строкам изображения. Однако кроме результата декодирования интерес представляет анализ показателей качества работы алгоритма: при каких условиях он работоспособен, какое качество при заданных параметрах он может обеспечивать.

Для ответа на эти вопросы нужно исследовать ошибку измерения положения синхроимпульсов, представляющую собой случайную величину, ввиду того что помехи в канале распространения сигнала

являются случайными. Эта ошибка будет зависеть от параметров системы приемопередачи информации и параметров помех в канале связи. Надежность оценки положения пачек синхроимпульсов будем характеризовать с помощью вероятности правильного обнаружения пачки синхроимпульсов. Обнаружение будем считать верным, если модуль ошибки оценки положения не более заданной константы tol (т. е. оценка сдвинута относительно истинной позиции не более, чем на tol отсчетов).

Решить подобную задачу аналитически крайне трудно. Здесь на помощь приходит метод статистических испытаний. Сначала нам нужно создать имитационную модель, которая будет описывать передачу данных в формате APT со спутника NOAA (я Вам ее выдаю). А Вам нужно написать код для статистических испытаний.

Структура пачки синхроимпульсов (sync) показана на рис. 2.

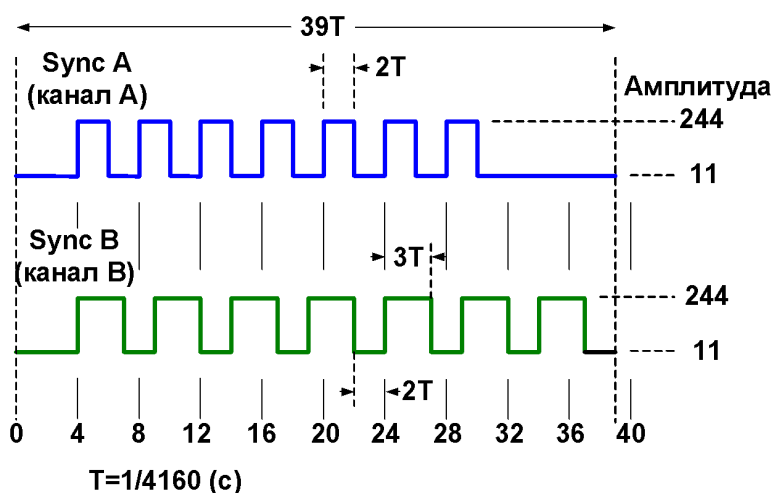


Рис. 2. Пачки синхроимпульсов
(в каналах A и B) в формате APT передачи спутниковых снимков

2.1.3. Описание имитационной модели

Схема на рис. 1 иллюстрирует, какие процессы происходят при передаче сигнала в формате APT со спутника на Землю.

Рассмотрим реализацию данной модели с помощью численного моделирования. На рис. 3 показан пример такой (упрощенной) реализации. Исходным объектом является фрагмент строки формата APT, а результатом моделирования – ошибка оценки положения пачки синхроимпульсов. Для упрощения задачи формируемый фрагмент строки будет содержать только смоделированную пачку синхроимпульсов, остальные отсчеты – нулевые. То есть влиянием характера передаваемых данных на качество оценки положения пачек синхроимпульсов мы пренебрегаем.

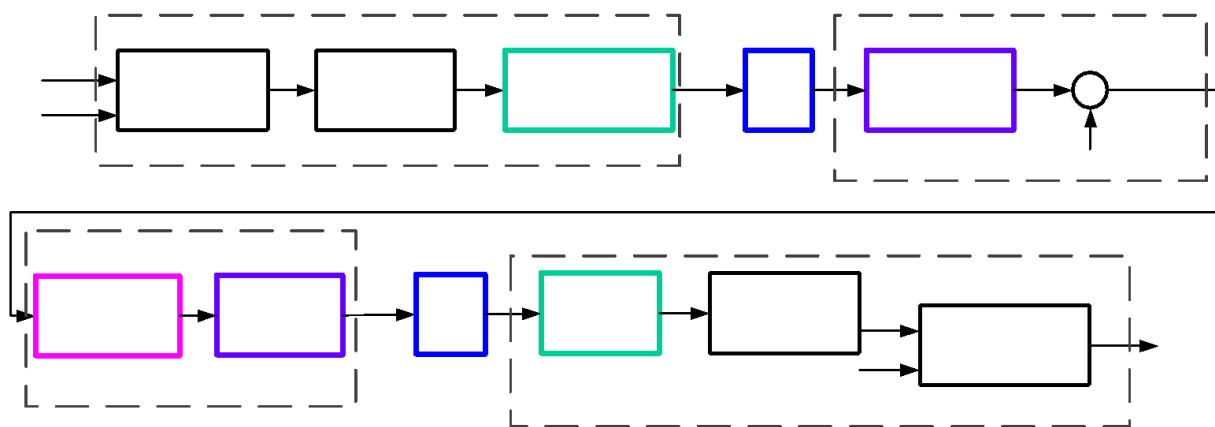


Рис. 3. Пример реализации имитационной модели канала АРТ для численного моделирования

Вместо аналоговой обработки (обработки непрерывных сигналов) при моделировании на ЭВМ мы должны использовать цифровое моделирование. В данной реализации предлагается проводить формирование и обработку АМ сигналов с частотой дискретизации 11025 Гц. Непосредственное моделирование ЧМ модулированных сигналов с несущей частотой 137 МГц потребовало бы использования частоты дискретизации более 274 МГц, что крайне проблематично для обычных персональных компьютеров.

Поэтому при написании и тестировании реализации модели, во-первых, рекомендуется использовать несущую частоту 70 кГц (а не 137 МГц). Использовать разную частоту дискретизации в разных частях модели. Когда мы имеем дело с АМ сигналом, то достаточно использовать частоту дискретизации 11025 Гц. Но когда мы переходим к АМ/ЧМ сигналу, нам необходимо увеличить частоты дискретизации в $N = 25$ раз (если несущая частота 70 кГц, так как максимальная частота спектра увеличивается, вспоминайте теорему Котельникова).

После того как программа статистических испытаний, использующая имитационную модель, будет полностью написана и проверена, можно будет провести моделирования с другими несущими частотами (частота дискретизации должна быть выбрана соответственно), чтобы оценить, как это будет влиять на результат.

В реальности выбор несущей частоты определяется такими факторами, как:

- возможность реализации передающей и приемной антенных систем;
- условия распространения и помехи на выбранной частоте;
- ширина спектра передаваемого сигнала.

Из курса физики известно, что эффективное излучение электромагнитной энергии можно осуществить лишь при условии, что геометрические размеры излучающей системы соизмеримы с длиной

волны. Например, эффективно излучить сигнал с частотой 1 МГц (длина волны 300 м) со спутника не представляется возможным.

При увеличении частоты несущей также появляется возможность передавать сигналы с более широким спектром, так как условия передачи и обработки сигналов в радиоаппаратуре определяются в основном не абсолютной шириной спектра сигнала, а относительной, то есть деленной на центральную частоту.

Повышение, а затем понижение частоты дискретизации при имитационном моделировании позволяет уменьшить количество операций на этапах амплитудной модуляции и демодуляции, а также снизить порядок цифровых фильтров на этих этапах.

Также при цифровом моделировании вместо аналоговых фильтров следует использовать цифровые фильтры. Расчет цифровых фильтров – это предмет курса цифровой обработки сигналов, который Вам еще предстоит. Поэтому пока можете не сильно вникать в принцип их работы.

Особенности реализации отдельных элементов будут более детально рассмотрены в порядке выполнения работы.

2.1.4. Частотная модуляция

Вкратце поясним смысл частотной модуляции. Математическую модель простого гармонического колебания с постоянной частотой f_0 (назовем такое колебание “несущей”) можно задать выражением

$$A_0 \cos(2\pi f_0 t + \varphi_0),$$

где A_0 - амплитуда;

φ_0 - начальная фаза (выражение $\psi(t) = 2\pi f_0 t + \varphi_0$ называют полной фазой).

Но такое колебание не несет в себе никакой информации. Мы можем рассмотреть более сложную модель

$$s_{ch}(t) = A_0 \cos(\psi(t)), \quad (1)$$

где полная фаза $\psi(t)$ - некоторая функция времени. В этом случае вводят понятие мгновенной частоты, которая определяется как производная от полной фазы $\psi(t)$:

$$f(t) = \frac{\partial \psi(t)}{\partial t}.$$

При частотной модуляции мы используем модель (1), кодируя полезную информацию в закон изменения частоты $f(t)$. Соответственно, можем выразить полную фазу через мгновенную частоту (это следует из предыдущего выражения):

$$\psi(t) = \int_0^t 2\pi f(t_1) dt_1 + \varphi_0 \quad (2)$$

где φ_0 – начальная фаза сигнала (в момент времени $t = 0$).

Раз $f(t)$ – это наш информационный сигнал, то выполнив преобразование (2) и используя модель (1) мы получим частотно модулированный сигнал.

2.2. Порядок выполнения работы

По имитационной модели я запишу отдельное видео, которое поясняет, какую роль играет каждый фрагмент ее кода. Код модели для лучшего понимания разбит на “пункты”: “р 1”, “р 2”, “р 3” и т.д.

1. Запустите несколько раз всю программу (файл [code_lab2.py](#)). Если все сделано правильно, то с большой долей вероятности ошибка при каждом испытании должна быть равна нулю.

Попробуйте изменить значение `snr` (в пункте 7 задавалось равным 10) на, например, 0.7 и запустите программу несколько раз. Разные запуски скорее всего приведут к разным результатам, т. е. однократный запуск не позволяет сделать вывод о качестве работы системы.

2. На основе выданного кода создайте функцию [apt_error_model](#), которая при вызове вычисляет модуль ошибки оценки положения пачки синхроимпульсов. Формат вызова

`error = apt_error_model(Fs, n_resample, f0, snr, thresh)`,

где `Fs` – исходная частота дискретизации;

`n_resample` – коэффициент передискретизации (мы использовали значение 25);

`f0` – несущая частота, Гц (для частотной модуляции);

`snr` – отношение с/ш (см. пункт 7);

`thresh` – порог (для нормированной корреляции) детектирования пачек синхроимпульсов (см. пункт 13).

Полученную функцию [apt_error_model](#) сохраните в отдельном модуле [simulation_model.py](#). Протестируйте созданную функцию (импортируя ее при запуске кода основной программы). Нужно убедиться, что функция работает при различных исходных данных, при этом для высоких отношений с/ш модуль ошибки почти всегда равен нулю, а при уменьшении параметра `snr` – модуль ошибки возрастает.

3. Создайте функцию [detect_probability](#) для вычисления вероятности правильного обнаружения пачки синхроимпульсов методом

статистических испытаний. Эта функция должна `n_test` запускать функцию `apt_error_model` и в итоге рассчитать массив ошибок `err_list`. Также функция должна оценивать вероятность правильного совмещения P . Для этого нужно подсчитать количество правильных обнаружений и поделить на общее число экспериментов (`n_test`). Обнаружение считаем правильным, если модуль ошибки меньше или равен параметру `tol`.

Например, если
`err_list = [0, 11, 104, 0, 1]`
`tol = 2`
то вероятность P равна $3/5$.

Формат вызова такой:

`P, err_list = detect_probability(snr, thresh, tol, n_test),`

где `snr` – отношение с/ш;

`thresh` – порог (для нормированной корреляции) детектирования пачек синхроимпульсов;

`tol` – максимально допустимая ошибка, при которой считаем обнаружение пачки правильным;

`n_test` – количество испытаний;

P – это рассчитанная вероятность правильного обнаружения;

`err_list` – массив ошибок, имеющий `n_test` элементов.

Протестируйте функцию `detect_probability` (убедитесь, что функция работает при различных исходных данных и что для полученного массива ошибок `error_list` вероятность P рассчитывается правильно).