

# Аннотация

В данной работе проводится анализ алгоритмов адаптивной компенсации нелинейных паразитных искажений, возникающих в приемо-передающих устройствах телекоммуникаций. Нелинейные искажения образуются в передатчиках базовых станций, а также мобильных устройств при прохождении сигнала через нелинейные цепи передающего тракта, усилитель мощности и дуплексер.

Вследствие компактности расположения компонент в RF-чипсете мобильного устройства, паразитная помеха распространяется из передатчика в приёмную часть уменьшая чувствительность приёмника.

Нелинейные искажения значительно влияют на качество радиочастотных сигналов, создавая помехи в канале связи. Такие нелинейные искажения приводят к увеличению битовой ошибки (Bit Error Rate - BER) на приемнике, генерируют внутриполосные и внеполосные помеховые сигналы и ухудшают качество передачи сигналов соседних полос. Для предотвращения искажений в современных базовых станциях и сотовых устройствах широко используются методы цифровой предискажения DPD, основанные на методе адаптивной компенсации нелинейных искажений.

В работе рассматриваются задачи адаптивной нелинейной компенсации искажений в задаче цифрового предискажения сигнала передатчика, а адаптивной компенсации паразитных помех в приёмном тракте мобильного терминала системы связи стандарта 4G/5G. Приводится математическое описание модели нелинейной помехи, а также используемых в исследовании алгоритмов.

Проводится исследование методов адаптивной компенсации нелинейных искажений, а также различных моделей нелинейных искажений радиосигналов.

Исследование алгоритмов адаптации, а также моделей нелинейных искажений, проводится на основе сигналов, сформированных на физической платформе. Сигнал помехи формируется относительно различных случаев выходной мощности усилителя.

Адаптивный корректор строится на основе классических нелинейных моделей, качественно описывающей процессы формирования нелинейных искажений, а также на основе нейросетевого подхода, позволяющего увеличить обобщающую способность адаптивной нелинейной модели. Особое внимание уделяется вопросам ускорения сходимости алгоритмов адаптации параметров нелинейной модели, а также методам сокращения ресурсов при аппаратной реализации модуля адаптивной коррекции.

# Содержание

<b>Аннотация</b>	i
<b>Введение</b>	v
<b>I.. Методы адаптивной компенсации нелинейных искажений сигнала в приемо-передающих устройствах систем связи</b>	1
1. Постановка задачи адаптивной компенсации нелинейных искажений в устройствах связи . . . . .	1
1.1. Проблема возникновения паразитных помех в приёмном тракте приемо-передающего устройства . . . . .	1
1.2. Схема компенсации паразитных нелинейных помех в приёмном тракте мобильного терминала . . . . .	3
1.3. Проблема возникновения нелинейных искажений сигнала передающего тракта . . . . .	4
1.4. Схема компенсации нелинейных искажений в передатчике приемо-передающего устройства . . . . .	5
1.5. Критерий компенсации нелинейных искажений в устройствах связи . . . . .	6
2. Методы первого порядка для адаптации моделей нелинейных искажений	7
2.1. Метод градиентного спуска . . . . .	8
2.2. Модификации метода градиентного спуска . . . . .	10
3. Методы второго порядка для адаптации моделей нелинейных искажений	11
3.1. Метод Ньютона . . . . .	11
3.2. Модификации метода Ньютона . . . . .	13
3.3. Смешанный метод Ньютона . . . . .	14
3.4. Метод LS . . . . .	16
4. Методы адаптации моделей нелинейных искажений не требующие явного вычисления матрицы Гессе . . . . .	17
4.1. Метод сопряженных градиентов . . . . .	17
4.2. Метод DCD . . . . .	19
4.3. Квазиньютоновские методы . . . . .	23
4.4. Метод Barzilai-Borwein . . . . .	25
4.5. Метод DFP . . . . .	25
4.6. Метод BFGS . . . . .	26

5.	Математические модели линейных и нелинейных искажений в системах связи . . . . .	27
5.1.	Линейная модель искажений сигнала в системах связи . . . . .	27
5.2.	Полиномиальная нелинейная модель искажений в системах связи . . . . .	29
5.3.	Модель нелинейных искажений в системах связи на основе сплайновых полиномов . . . . .	32
5.4.	Модель Вольтерра нелинейных искажений в системах связи . . . . .	34
5.5.	Модель Гаммерштейна нелинейных искажений в системах связи . . . . .	34
5.6.	Модель Винера нелинейных искажений в системах связи . . . . .	36
5.7.	Модель Винера-Гаммерштейна нелинейных искажений в системах связи . . . . .	40
5.8.	Применение нейросетевых структур для аппроксимации нелинейных искажений приемо-передающего тракта . . . . .	42
5.9.	Применение канонического тензорного разложения для аппроксимации многомерных структур . . . . .	43

<b>II..Исследование эффективности методов адаптивной компенсации нелинейных искажений в устройствах связи</b>	<b>44</b>	
1.	Структура тестовой платформы . . . . .	44
2.	Формирование тестовых наборов данных . . . . .	45
2.1.	Формирование обучающей выборки для задачи компенсации нелинейной помехи в приёмнике полнодуплексной системы связи . . . . .	45
2.2.	Формирование обучающей выборки для задачи компенсации второй гармоники в приёмнике системы связи . . . . .	46
2.3.	Формирование обучающей выборки для задачи цифровых предискажений в передатчике двухканальной системы связи . . . . .	48
2.4.	Формирование обучающей выборки для задачи цифровых предискажений в передатчике одноканальной системы связи в условиях динамически меняющегося режима работы усилителя мощности . . . . .	50
3.	Исследование алгоритмов и адаптивных моделей компенсации паразитной помехи на приёмнике устройств связи . . . . .	52
3.1.	Компенсация паразитной помехи на приёмнике полнодуплексной системы связи методом адаптации модели Гаммерштейна . . . . .	52
3.2.	Компенсация паразитной помехи 2-ого порядка на основе адаптации классических и нейросетевых структур . . . . .	56
4.	Исследование алгоритмов и адаптивных моделей компенсации нелинейных помех в передатчике устройств связи . . . . .	61
4.1.	Компенсация нелинейных искажений в передатчике устройств связи в условиях динамического изменения выходной мощности нелинейного усилителя . . . . .	61

4.2. Компенсация нелинейных искажений в передатчике двухканальной системы связи . . . . .	61
<b>III.Практическая реализация нелинейных корректоров в системах связи</b>	<b>62</b>
1. Исследование компенсации нелинейных искажений методами не требующими явного обращения матрицы Гессе . . . . .	62
2. Сокращение ресурсов при реализации моделей нелинейных искажений засчет канонического тензорного разложения . . . . .	71
3. Сокращение ресурсов при реализации моделей нелинейных искажений путем оптимизации гиперпараметров модели . . . . .	71
3.1. Метод Orthogonal Least Squares . . . . .	71
3.2. Метод Policy Gradient . . . . .	71
4. Методы экономии ресурсов при реализации нейросетевых моделей компенсации нелинейных искажений . . . . .	71
4.1. Методы экономии ресурсов при реализации цифровых фильтров . . . . .	71
4.2. Сокращение ресурсов при реализации свёрточных сетей для задачи компенсации нелинейных искажений . . . . .	76
<b>Заключение</b>	<b>77</b>
<b>Список литературы</b>	<b>78</b>

# Введение

По мере совершенствования телекоммуникационных технологий возникают задачи повышения скорости передачи информации. Так например разработка систем стандарта 5G предполагает решение задач связанных с увеличением скорости передачи информации по сравнению с системами стандарта 4G. Увеличение скорости передачи требует повышения порядков цифровой модуляции. А это в свою очередь требует улучшения характеристик чувствительности приёмника.

Кроме того, важнейшим направлением развития телекоммуникационных технологий является унификация производства аппаратуры. Так основная тенденция это упрощение аналоговых трактов аппаратуры с компенсацией возникающих искажений адаптивными цифровыми корректорами [1].

Современным направлением решения перечисленных задач является использование адаптивных систем нелинейной коррекции, которые позволяют существенно повысить коэффициент полезного действия передающих трактов системы при сохранении характеристик линейности системы в соответствии с принятыми международными стандартами связи [2]. С точки зрения пользователя мобильного терминала это означает увеличение времени автономной работы телефона.

В данной работе рассматриваются алгоритмы подавления помех в приёмном тракте мобильного устройства, возникающих в передающем тракте путём прохождения через нелинейные элементы, такие как дуплексеры и усилители мощности [3]. Ввиду интегрального исполнения аналоговых компонент, их характеристики имеют существенно нелинейный вид. Кроме того, ввиду компактного расположения компонент в RF-чипсете мобильного терминала, нелинейные компоненты, возникающие в передающем тракте, попадают в приемный тракт мобильного телефона, проходя по различным путям распространения, уменьшая чувствительность приёмника, а соответственно и пропускную способность нисходящего канала связи.

Адаптивная компенсация паразитных нелинейных помех позволяет улучшить пропускную способность нисходящего канала связи, повысить КПД усилителя передатчика, а также ослабить требования, предъявляемые к аналоговым дуплексерам.

В работе приводятся результаты симуляции работы модуля адаптивной коррекции для сигналов помехи различной ширины полосы, а также для различных каналов распространения помехи.

# Глава I

## Методы адаптивной компенсации нелинейных искажений сигнала в приемо-передающих устройствах систем связи

### 1. Постановка задачи адаптивной компенсации нелинейных искажений в устройствах связи

#### 1.1 Проблема возникновения паразитных помех в приёмном тракте приемо-передающего устройства

В данном разделе формулируется задача адаптивной компенсации нелинейных паразитных помех, возникающих на приёмнике приемо-передающего устройства. Приводится схема адаптивной компенсации данного рода помех. А также описывается модель формирования паразитных помех на основе качественного представления физических процессов.

В качестве примера возникновения паразитных помех в приёмном тракте рассмотрим приёмо-передающее устройство, состоящее из двух каналов передатчика и приёмника (2T2R), как это показано на рис. 1. При этом передатчик и приёмник разнесены по частоте.

В процессе прохождения сигнала на несущей частоте  $f_0$  через нелинейные цепи передатчика, генерируются компоненты на кратных частотах  $f_0$ ,  $2f_0$ ,  $3f_0$  и выше. Гармоники порядка выше 3-его, как правило, имеют существенно меньшую мощность по сравнению с гармониками 1-го, 2-го и 3-его порядка.

Среди элементов передающего тракта, представленных на рис. 1, наибольших вклад в формирование нелинейных компонент спектра вносит аналоговый усилитель мощности. В связи с этим будем далее считать, что именно усилитель мощности является источником нелинейности. Согласно стандарту LTE, используемом в системах связи четвёртого поколения 4G, пара передатчик-приёмник может рабо-

тать на таких частотах, при которых первая, вторая или третья гармоника сигнала передатчика попадает в полосу приёмного тракта того же устройства [4]:

- LTE Band 2 (UL 1920–1980 MHz) – LTE Band 2 (DL 1930–1990 MHz)
- LTE Band 8 (UL 880-915 MHz) - LTE Band 3 (DL 1805-1880 MHz)
- LTE Band 17 (UL 704-716 MHz) - LTE Band 4 (DL 2110-2155 MHz)

Кроме того, за счёт малых размеров RF-чипсетов, компоненты внутри чипсете расположены компактно. Вследствие технологических ограничений производства существует сложность обеспечения изоляции отдельных компонент RF-чипсета. В связи с этим появляются различные пути распространения помехи в приёмнике, как это показано на рис. 1.

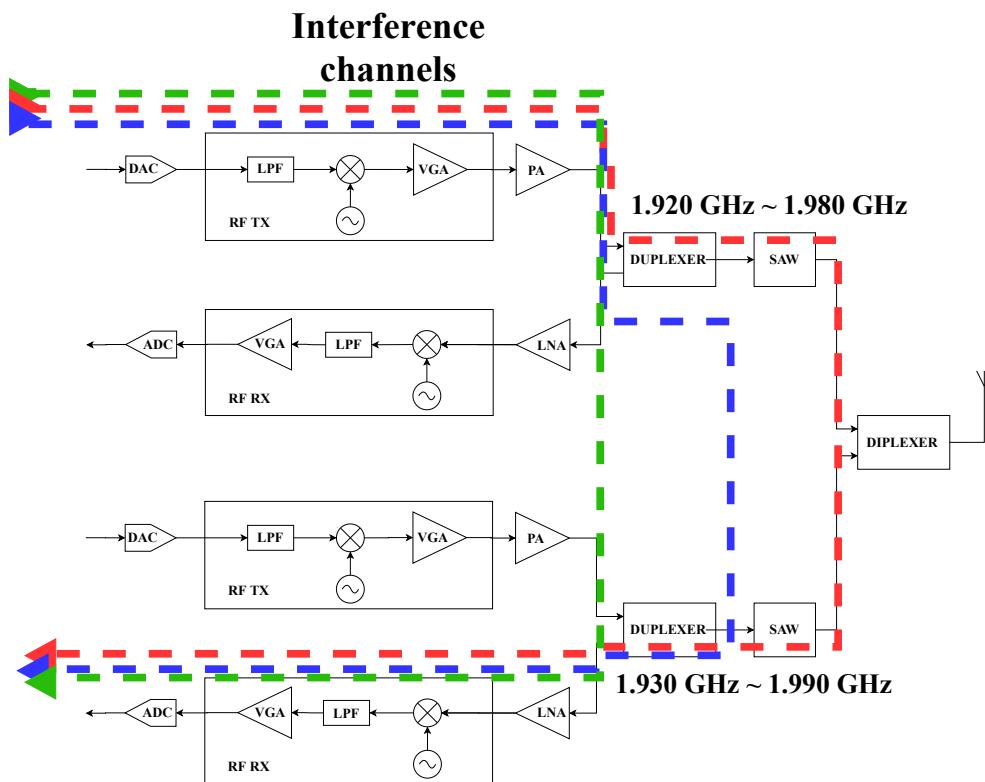


Рис. 1. Каналы распространения 1-ой гармоники нелинейных помех в приёмо-передающем устройстве 2T2R

Пример спектра сигнала передатчика на несущей частоте  $f_0 = 1.93$  ГГц и помехи, возникающей в приёмнике на несущей частоте передатчика  $f_1 = 1.96$  ГГц изображен на рис. 2. Задача модуля компенсации заключается в том, чтобы понизить уровень этой помехи для обеспечения требуемого уровня чувствительности приёмника.

Таким образом, одной из задач данной работы является разработка и исследование модуля адаптивной нелинейной коррекции, который позволит компенсировать

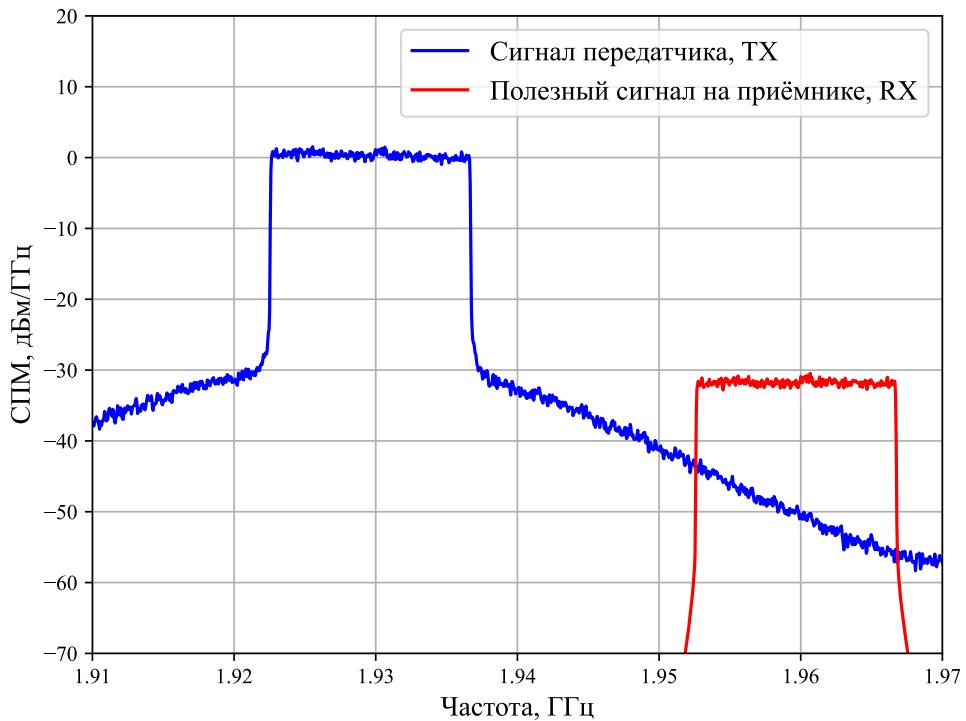


Рис. 2. Спектральная плотность мощности сигнала передатчика на выходе нелинейного УМ на частоте  $f_0 = 1.93$  ГГц и полезного сигнала приёмника на частоте  $f_1 = 1.96$  ГГц

паразитные помехи в приёмнике. Кроме того, такой подход позволит снизить требования к блокам фильтрации аналогового сигнала: дуплексерам и полосовым фильтрам.

## 1.2 Схема компенсации паразитных нелинейных помех в приёмном тракте мобильного терминала

Как отмечалось ранее, будем считать, что основным источником нелинейных помех в приёмном тракте является аналоговый усилитель мощности в передающем тракте мобильного терминала [1].

Предлагается идентифицировать помехи, попадающие в приёмник путем адаптации нелинейной модели по критерию минимума среднего квадрата ошибки [5]. Такая схема компенсации нелинейных помех изображена на рис. 3.

Исходный цифровой сигнал  $x$  на несущей частоте  $f_0$  проходит через блок нелинейной модели  $f(\cdot)$ , в результате чего на её выходе формируется цифровой сигнал  $y = f(x)$ . Сигнал проходит по передающему тракту: через ЦАП, RF-модуль, аналоговый усилитель мощности, затем вследствие наличие различных путей распространения, паразитные гармоники исходного сигнала на частотах  $f_0, 2f_0, 3f_0$  попадают на приёмную часть.

Проходя по приёмному тракту помеха приобретает вид цифрового сигнала, обозначенного как  $d$ . Помимо помехи в приёмную часть терминала поступает полезный

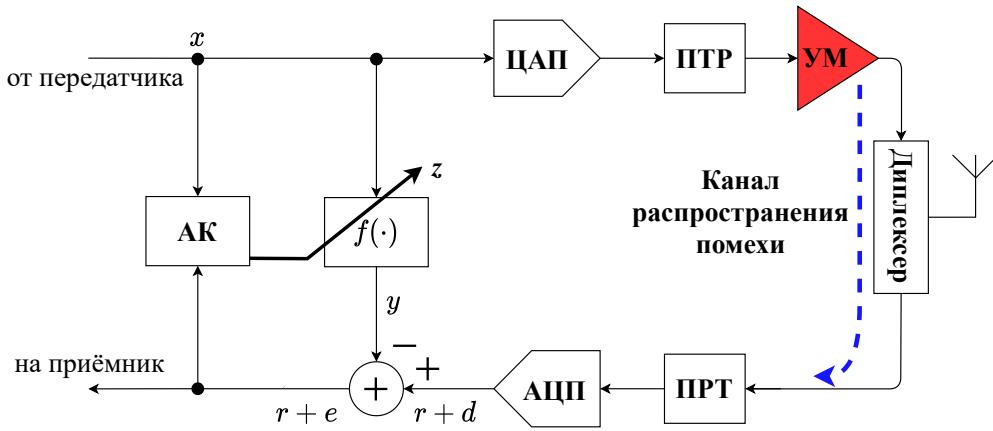


Рис. 3. Схема компенсации нелинейных помех в приёмнике приёмо-передающего устройства

сигнал  $r$ . Таким образом, на выходе АЦП – сумма  $r + d$ .

Блок адаптации АД производит идентификацию, подстраивая коэффициенты  $z$  нелинейной модели  $f(\cdot)$  посредством измерения отклонения  $e = d - y$  принятой приёмным трактом помехи от выхода нелинейной модели. В общем случае можно считать, что коэффициент корреляции отправляемого на передатчике сигнала  $x$  и полезного принятого сигнала  $r$  равен нулю. Поэтому в результате работы модуля компенсации результирующий сигнал на приёмнике будет стремиться к полезному принятому сигналу  $r + e \rightarrow r$ . Задачу идентификации паразитной помехи путём адаптации нелинейной модели по критерию минимума среднего квадрата сформулируем следующим образом:

$$\begin{cases} J = e^* e \rightarrow \min_z, \\ e = d - f(x, z) \end{cases} \quad (1.2.1)$$

### 1.3 Проблема возникновения нелинейных искажений сигнала передающего тракта

Ввиду нелинейности характеристики передающего тракта формируются нелинейные искажения, которые значительно влияют на качество радиочастотных сигналов, создавая помехи в канале связи. Такие нелинейные искажения приводят к увеличению битовой ошибки на приемнике, генерируют внутриполосные и внеполосные помеховые сигналы и ухудшают качество передачи сигналов соседних полос.

Как было отмечено ранее, наибольших вклад в искажения сигнала в передающем тракте вносят нелинейные усилители мощности. В результате прохождения через УМ формируются компоненты спектра внутри полосы передатчика, а также внеполосные искажения. На рис. 4 изображена спектральная плотность мощности сигнала передатчика до и после прохождения через УМ.

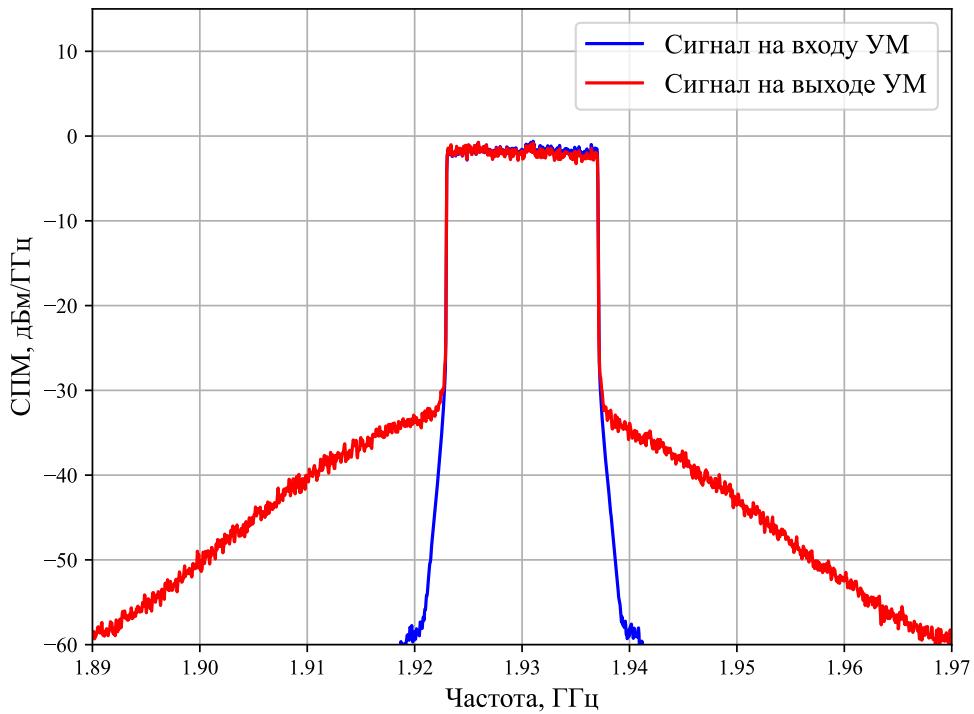


Рис. 4. Спектральная плотность мощности сигнала передатчика на входе и выходе усилителя мощности на частоте  $f_0 = 1.93$  ГГц

#### 1.4 Схема компенсации нелинейных искажений в передатчике приемо-передающего устройства

Для предотвращения искажений в современных базовых станциях и сотовых устройствах широко используются методы цифровой предискажения DPD [1]-[3].

Устройство DPD представлено блоком с обратной нелинейной характеристикой, изменяющим входной сигнал УМ так, чтобы минимизировать нелинейных искажений на выходе УМ (рис. 5).

Таким образом, DPD предыскаивает входной сигнал УМ таким образом, чтобы

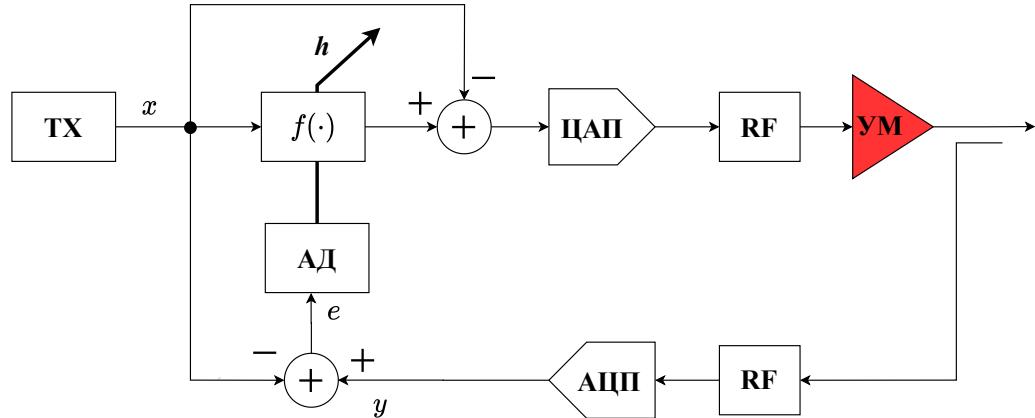


Рис. 5. Структура системы цифрового предыскажения

на выходе сформировался линейно искаженный сигнал. Такой подход позволяет работать в режимах высокой нелинейности УМ и, в результате, повысить КПД УМ. Задача оптимизации цифрового предискажения может быть представлена математическим выражением [4], описывающим минимизацию отклонения входа DPD от выхода УМ:

$$\|g(x - f(x, z)) - x\|_2^2 \rightarrow \min_z, \quad (1.4.1)$$

где  $x$  – входной сигнала УМ,  $g(\cdot)$  – нелинейная характеристика УМ,  $f(\cdot)$  – модель цифрового предискажения с адаптивными параметрами  $z$ . Предположим, что нелинейную характеристику УМ  $g(x)$  можно аппроксимировать линейной функцией в окрестности рабочей точки:

$$g(x - f(x, z)) \approx g(x) - g'_x(x)f(x, z), \quad (1.4.2)$$

где  $g'_x(x)$  – производная нелинейной функции УМ по входу. Поскольку DPD должен работать в таком режиме УМ, чтобы уровень нелинейных искажений оставался значительно ниже уровня передаваемого сигнала, то  $g'_x(x) \approx I$  – будет представлять собой единичную матрицу. Подставляя (1.4.2) в (1.4.1), получаем выражение:

$$\|f(x, z) - e\|_2^2 \rightarrow \min_z, \quad (1.4.3)$$

где  $e = g(x) - x$  – вектор ошибки.

## 1.5 Критерий компенсации нелинейных искажений в устройствах связи

Существует несколько способов обработки сигнала [6]. Симуляция алгоритмов как правило проводится на сохранных блоках данных. В этом случае имеется мощный вычислитель (компьютер) и возможность вести математическую обработку матриц и других объектов алгоритмов. Это блочный режим работы.

При аппаратной реализации блочный режим стараются заменить стохастическим в реальном времени [6], чтобы минимизировать задержки и сократить ресурсы. В этом случае векторы состояния, корреляционные матрицы, шаг алгоритма, а также другие объекты, обновляются каждый отсчет.

При рассмотрении методов компенсации паразитных помех будем считать, что на вход блока адаптации поступает вектор комплексных отсчетов передатчика  $\mathbf{x} \in \mathbb{C}^{N \times 1}$  длины  $N$ , вектор отсчетов с приёмника  $\mathbf{d} \in \mathbb{C}^{N \times 1}$  также длины  $N$ .

Задача блока адаптации заключается в том, чтобы так преобразовать исходный вектор  $\mathbf{x}$ , чтобы как можно лучше приблизить к вектору отсчетов на приёмнике  $\mathbf{d}$  путём минимизации нормы вектора ошибки  $\mathbf{e} = \mathbf{d} - f(\mathbf{x}, z) \in \mathbb{C}^{N \times 1}$ , где  $f(\cdot)$  – оператор преобразования исходного вектора в вектор отсчетов на выходе модели помехи. Оператор  $f(\cdot)$  определяется коэффициентами  $\mathbf{z} \in \mathbb{C}^{K \times 1}$ ,  $K$  – число адаптивных

параметров модели.

Рассмотрим требования к целевой функции (метрике) алгоритмов компенсации. Метрика величины отклонения выхода модели помехи от вектора отсчетов самой помехи должна быть вещественнозначной и неотрицательной для реализации процедуры поиска минимума данной метрики. Кроме того, желательно задать целевую функцию квадратичной и выпуклой для применения эффективных методов адаптации [7].

Такой вещественной, неотрицательной и квадратичной метрикой является средний квадрат ошибки (MSE - Mean Square Error). Физический смысл среднего квадрата ошибки – энергия отклонения выхода нелинейной модели от помехи, измеренной на приёмнике. При блочной обработке сигнала MSE имеет вид:

$$J = \|e\|_2^2 = e^H e. \quad (1.5.1)$$

Помимо MSE на практике применяется метрика - нормированный средний квадрат ошибки (NMSE - Normalized Mean Square Error):

$$J = \frac{\|e\|_2^2}{\|d\|_2^2} = \frac{e^H e}{d^H d}. \quad (1.5.2)$$

В данном случае энергия ошибки нормируется к энергии сигнала приёмника для того чтобы оценить уровень ошибки независимо от динамического диапазона сигнала на входе приемника.

Реальные сигналы, используемые в системах связи [8] являются мощностными, то есть обладают бесконечной энергией на бесконечном промежутке времени, однако в уравнениях (1.5.1), (1.5.2) фигурирует энергия ошибки и сигнала помехи, рассчитанная на конечной длине блока.

## 2. Методы первого порядка для адаптации моделей нелинейных искажений

Методы первого порядка [9] занимают центральное место в задачах адаптации нелинейных моделей, применяемых для компенсации искажений в трактах телекоммуникационных устройств. Их ключевое преимущество заключается в низкой вычислительной сложности и хорошей масштабируемости, что делает их особенно привлекательными для аппаратной реализации в системах с ограниченными ресурсами.

В отличие от более сложных второпорядковых алгоритмов, методы первого порядка – такие как стохастический градиентный спуск и его модификации – опираются лишь на локальную информацию о градиенте функции ошибки, обеспечивая устойчивую и быструю адаптацию даже при неполном знании статистики сигнала. Благодаря этому они хорошо подходят для работы в реальном времени и могут

эффективно функционировать при изменяющихся условиях канала или параметров нелинейности.

Кроме того, градиентные методы допускают реализацию в фиксированной точке, что позволяет использовать простые арифметические блоки и снижает энергопотребление. Такая особенность делает их оптимальным выбором для встроенных и энергоэффективных систем, где важно обеспечить баланс между скоростью сходимости и аппаратной сложностью. В контексте нелинейной компенсации это обеспечивает возможность динамического обновления коэффициентов модели с минимальными затратами ресурсов при сохранении высокой точности восстановления сигнала.

## 2.1 Метод градиентного спуска

Градиентный спуск для случая нелинейной модели с вещественными параметрами представляет собой следующий итеративный алгоритм:

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \mu(D_{\mathbf{z}_k} J(\mathbf{z}_k))^T, \mathbf{z}_k \in \mathbb{R}^{K \times 1}, \quad (2.1.1)$$

$$\left\| D_{\mathbf{z}^*} J(\mathbf{z}_k) - D_{\mathbf{z}^*} J(\mathbf{z}_{k+1}) \right\|_2 < L \left\| D_{\mathbf{z}^*} J(\mathbf{z}_k) - D_{\mathbf{z}^*} J(\mathbf{z}_k) \right\|_2, \quad \mu \leq L, \quad (2.1.2)$$

где  $D_{\mathbf{z}} J \in \mathbb{R}^{1 \times K}$  – оператор вычисления производной функции  $J$  по адаптивным параметрам  $\mathbf{z}$ . Условие Липшица [9] задаёт ограничение на скорость поиска минимума целевой функции. Для ускорения алгоритма достаточно увеличить шаг  $\mu$ , однако при слишком больших  $\mu$  – алгоритм расходится. Критическое значение, при котором наступает стагнация алгоритма –  $\mu_{kp} = \frac{L}{2}$ .

Ввиду того, что в устройствах связи работают с комплекснозначными сигналами, параметры моделей нелинейной обработки сигналов также выбирают комплекснозначными. Направление возрастания вещественной функции комплексного переменного определяется производной вещественной функции по комплексно-сопряженным параметрам [10]. В связи с этим, метод градиентного спуска для моделей с комплексными параметрами строится следующим образом:

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \mu(D_{\mathbf{z}_k^*} J(\mathbf{z}_k))^T, \mathbf{z}_k \in \mathbb{C}^{K \times 1}, \quad (2.1.3)$$

где  $D_{\mathbf{z}_k^*} J \in \mathbb{C}^{1 \times K}$  – оператор вычисления производной вещественной функции  $J$  по адаптивным комплексно-сопряженным параметрам  $\mathbf{z}^*$ .

Получим разностное уравнение метода градиентного спуска оптимизации вещественной скалярной функции потерь, зависящей от комплексных векторных параметров. Для этого разложим функцию потерь  $J$  в окрестности нуля в ряд Тейлора до членов первого порядка малости:

$$dJ = D_{\mathbf{z}} J d\mathbf{z} + D_{\mathbf{z}^*} J d\mathbf{z}^* \in \mathbb{R}, \quad (2.1.4)$$

где  $D_{\mathbf{z}} J \in \mathbb{C}^{1 \times K}$ ,  $D_{\mathbf{z}^*} J \in \mathbb{C}^{1 \times K}$ ,  $d\mathbf{z} \in \mathbb{C}^{K \times 1}$ ,  $d\mathbf{z}^* \in \mathbb{C}^{K \times 1}$ . Разложим выход нелинейной модели в Ряд Тейлора в окрестности нуля до членов первого порядка малости [10]:

$$d\mathbf{y} = D_{\mathbf{z}} \mathbf{y} d\mathbf{z} + D_{\mathbf{z}^*} \mathbf{y} d\mathbf{z}^* \in \mathbb{C}^{N \times 1}, \quad (2.1.5)$$

где  $D_{\mathbf{z}} \mathbf{y}$ ,  $D_{\mathbf{z}^*} \mathbf{y} \in \mathbb{C}^{N \times K}$  – матрицы Якоби, производные выхода модели по прямым и комплексно-сопряженным параметрам модели.

Распишем дифференциал среднего квадрата ошибки, введенного в (1.5.1) и подставим разложение выхода нелинейной модели  $\mathbf{y}$  до членов первого порядка малости (2.1.5):

$$\begin{aligned} dJ &= d(\mathbf{e}^H \mathbf{e}) = d\mathbf{e}^H \mathbf{e} + \mathbf{e}^H d\mathbf{e} = -\mathbf{e}^T d\mathbf{y}^* - \mathbf{e}^H d\mathbf{y} = \\ &= -\mathbf{e}^T D_{\mathbf{z}} \mathbf{y}^* d\mathbf{z} - \mathbf{e}^T D_{\mathbf{z}^*} \mathbf{y}^* d\mathbf{z}^* - \mathbf{e}^H D_{\mathbf{z}} \mathbf{y} d\mathbf{z} - \mathbf{e}^H D_{\mathbf{z}^*} \mathbf{y} d\mathbf{z}^* = \\ &= (-\mathbf{e}^T D_{\mathbf{z}} \mathbf{y}^* - \mathbf{e}^H D_{\mathbf{z}} \mathbf{y}) d\mathbf{z} + (-\mathbf{e}^T D_{\mathbf{z}^*} \mathbf{y}^* - \mathbf{e}^H D_{\mathbf{z}^*} \mathbf{y}) d\mathbf{z}^*. \end{aligned} \quad (2.1.6)$$

Сравнивая выражения перед дифференциалом  $\mathbf{z}^*$  в (2.1.4) и (2.1.6), получаем выражение производной функции потерь по комплексно-сопряженным параметрам  $\mathbf{z}^*$ :

$$(D_{\mathbf{z}^*} J)^T = (-\mathbf{e}^T D_{\mathbf{z}^*} \mathbf{y}^* - \mathbf{e}^H D_{\mathbf{z}^*} \mathbf{y})^T = -(D_{\mathbf{z}} \mathbf{y})^H \mathbf{e} - (D_{\mathbf{z}^*} \mathbf{y})^T \mathbf{e}^* \in \mathbb{C}^{K \times 1}. \quad (2.1.7)$$

Таким образом, подставляя (2.1.7) в (2.1.3) получаем разностное уравнение градиентного спуска для функции потерь MSE с вектором ошибки общего вида, то есть зависящим от прямых и сопряженных параметров  $\mathbf{e} = \mathbf{e}(\mathbf{x}, \mathbf{z}, \mathbf{z}^*)$ :

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \mu(D_{\mathbf{z}} \mathbf{y})^H \mathbf{e} + \mu(D_{\mathbf{z}^*} \mathbf{y})^T \mathbf{e}^* \in \mathbb{C}^{K \times 1} \quad (2.1.8)$$

Пусть вектор отклонения выхода модели от значений помехи  $\mathbf{e}(\mathbf{x}, \mathbf{z}) \in \mathbb{C}^{N \times 1}$  – голоморфная функция относительно  $\mathbf{z}$ , то есть не зависит от вектора комплексно-сопряженных параметров  $\mathbf{z}^*$ . В этом случае Якобиан выхода модели от сопряженных параметров равен нулю:

$$D_{\mathbf{z}^*} \mathbf{y} = \mathbf{0}. \quad (2.1.9)$$

Тогда из (2.1.8) разностное уравнение метода градиентного спуска может быть представлено как:

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \mu(D_{\mathbf{z}} \mathbf{y})^H \mathbf{e}. \quad (2.1.10)$$

В нелинейной адаптивной обработке сигналов условие (2.1.9) используется при построении нелинейных моделей, поскольку в этом случае снижается вычислительная сложность пересчета параметров адаптивных моделей, снижается требования по памяти, энергопотребление, а также занимаемая площадь на кристалле.

## 2.2 Модификации метода градиентного спуска

Классический метод градиентного спуска (2.1.3) является эффективным в случае с гладкими выпуклыми функциями потерь, однако в случае сильно искривлённого ландшафта или шумных градиентов он может сходиться медленно. Для ускорения сходимости и сглаживания колебаний используется метод тяжёлого шара (momentum), предложенный Поляком [11]. Основная идея заключается во введении вектора скорости, аккумулирующего информацию о предыдущих шагах:

$$\begin{cases} \mathbf{m}_{k+1} = \beta \mathbf{m}_k + \mu (D_{\mathbf{z}^*} J(\mathbf{z}_k))^T, \\ \mathbf{z}_{k+1} = \mathbf{z}_k - \mathbf{m}_{k+1}, \end{cases} \quad (2.2.1)$$

где  $0 \leq \beta < 1$  — коэффициент инерции. Такой подход позволяет снизить осцилляции и ускорить движение вдоль пологих направлений, однако градиент вычисляется в текущей точке, что может приводить к избыточным колебаниям.

Для устранения этого недостатка была предложена модификация Нестерова, известная как *ускоренный градиент Нестерова* [12] (Nesterov accelerated gradient, NAG). В этом подходе градиент вычисляется в «прогнозируемом» положении, что стабилизирует алгоритм адаптации:

$$\begin{cases} \mathbf{m}_{k+1} = \beta \mathbf{m}_k + \mu (D_{\mathbf{z}^*} J(\mathbf{z}_k - \beta \mathbf{v}_k))^T, \\ \mathbf{z}_{k+1} = \mathbf{z}_k - \mathbf{m}_{k+1}. \end{cases} \quad (2.2.2)$$

Таким образом, данный метод корректирует направление движения до обновления параметров, что обеспечивает более устойчивую и быструю сходимость для выпуклых задач.

Дальнейшее развитие концепции инерционного градиентного спуска привело к созданию алгоритма *Adam* (Adaptive Moment Estimation) [13], который сочетает механизмы накопления импульса градиента и аддитивной нормализации шага по каждой координате параметров. В отличие от классических схем с фиксированной скоростью обучения, Adam автоматически масштабирует величину шага в зависимости от первой и второй статистических моментов градиента, что обеспечивает баланс между скоростью сходимости и устойчивостью обновлений. Такая аддитивность позволяет алгоритму поддерживать высокую скорость обучения на ранних этапах и эффективно выходить из окрестностей локальных минимумов, что делает его одним из наиболее универсальных и применимых методов стохастической оптимизации:

$$\begin{cases} \mathbf{m}_{k+1} = \beta_1 \mathbf{m}_k + (1 - \beta_1) (D_{\mathbf{z}^*} J(\mathbf{z}_k))^T, \\ \mathbf{v}_{k+1} = \beta_2 \mathbf{v}_k + (1 - \beta_2) |(D_{\mathbf{z}^*} J(\mathbf{z}_k))^T|^2, \\ \hat{\mathbf{m}}_{k+1} = \frac{\mathbf{m}_{k+1}}{1 - \beta_1^{k+1}}, \quad \hat{\mathbf{v}}_{k+1} = \frac{\mathbf{v}_{k+1}}{1 - \beta_2^{k+1}}, \\ \mathbf{z}_{k+1} = \mathbf{z}_k - \mu \frac{\hat{\mathbf{m}}_{k+1}}{\sqrt{\hat{\mathbf{v}}_{k+1}} + \varepsilon}, \end{cases} \quad (2.2.3)$$

где  $\beta_1 \approx 0.9$ ,  $\beta_2 \approx 0.999$ ,  $\varepsilon \approx 10^{-8}$ . Adam объединяет преимущества момента (инерции) и адаптивных методов (нормализация шага по координатам), что обеспечивает стабильную сходимость даже при шумных или нестационарных градиентах.

Эволюция этих подходов отражает постепенное улучшение базового метода: от простого градиентного шага (2.1.3) к инерционному ускорению (2.2.1), затем к прогнозирующему ускорению Нестерова (2.2.2), и, наконец, к адаптивно-инерционной схеме (2.2.3).

### 3. Методы второго порядка для адаптации моделей нелинейных искажений

#### 3.1 Метод Ньютона

Несмотря на широкое распространение и простоту реализации методов первого порядка, их эффективность существенно снижается в задачах с жёсткими (плохо обусловленными) функциями, где направление антиградиента может существенно отличаться от направления наискорейшего убывания. Для ускорения сходимости и более точного учёта локальной геометрии целевой функции применяются методы второго порядка, использующие информацию не только о градиенте, но и о кривизне поверхности уровня.

Классическим представителем данного класса является *метод Ньютона*, основанный на аппроксимации функции её локальным квадратичным разложением в окрестности текущей точки. Основная идея заключается в том, что, если функция  $J(\mathbf{z}, \mathbf{z}^*)$  близка к квадратичной, то минимум аппроксимации можно найти аналитически, что обеспечивает сверхлинейную сходимость. В отличие от методов первого порядка, метод Ньютона использует матрицу Гессе (матрицу вторых производных) для корректировки направления поиска, что позволяет автоматически масштабировать шаги в зависимости от локальной кривизны поверхности.

Таким образом, метод Ньютона интерпретируется как переход от *направлений убывания* к *направлениям оптимального спуска*, обеспечивающим более точное приближение минимума.

Для формального описания метода Ньютона введем следующие обозначения. Производная второго порядка скалярной вещественной функции  $J$  по вектору па-

раметров описывается следующим образом:

$$D_{\mathbf{z}}(D_{\mathbf{z}}J)^T = H_{\mathbf{z}, \mathbf{z}}J \in \mathbb{C}^{K \times K} - \text{матрица Гессе.} \quad (3.1.1)$$

Для удобства записи алгоритма метода Ньютона введем вектор удвоенной длины, включающий вектор прямых и сопряженных параметров:

$$\tilde{\mathbf{z}}^T = \begin{pmatrix} \mathbf{z} & \mathbf{z}^* \end{pmatrix}^T \quad (3.1.2)$$

Используя введённые выше обозначения, дифференциал второго порядка функции  $J$  может быть описан следующим образом [**complex\_deriv**]:

$$d^2J = \begin{pmatrix} d\mathbf{z}^T & d(\mathbf{z}^*)^T \end{pmatrix} \begin{pmatrix} H_{\mathbf{z}, \mathbf{z}}J & H_{\mathbf{z}^*, \mathbf{z}}J \\ H_{\mathbf{z}, \mathbf{z}^*}J & H_{\mathbf{z}^*, \mathbf{z}^*}J \end{pmatrix} \begin{pmatrix} d\mathbf{z} \\ d\mathbf{z}^* \end{pmatrix} = \tilde{\mathbf{z}}^T H_{\tilde{\mathbf{z}}, \tilde{\mathbf{z}}}J \tilde{\mathbf{z}} \quad (3.1.3)$$

Таким образом, разностное уравнение метода Ньютона может быть представлен следующим образом:

$$\tilde{\mathbf{z}}_{k+1} = \tilde{\mathbf{z}}_k - \mu(H_{\tilde{\mathbf{z}}, \tilde{\mathbf{z}}}J)^{-1}(D_{\tilde{\mathbf{z}}^*}J)^T \quad (3.1.4)$$

Получим разностное уравнение метода Ньютона для функции потерь MSE (1.5.1). Градиент функции потерь по вектору параметров удвоенной длины может быть получен из выражения (2.1.7):

$$(D_{\tilde{\mathbf{z}}^*}J)^T = \begin{pmatrix} D_{\mathbf{z}}J & D_{\mathbf{z}^*}J \end{pmatrix}^T = \begin{pmatrix} (D_{\mathbf{z}^*}J)^* & D_{\mathbf{z}^*}J \end{pmatrix}^T, \quad (3.1.5)$$

где использовано свойство  $(D_{\mathbf{z}^*}J)^* = D_{\mathbf{z}}J^* = D_{\mathbf{z}}J$  ввиду вещественности функции потерь  $J \in \mathbb{R}$ .

Заметим также, что ввиду определения матрицы Гессе (3.1.1), достаточно найти две из четырех составляющих матрицы (3.1.3):

$$H_{\mathbf{z}, \mathbf{z}^*}J = D_{\mathbf{z}^*}(D_{\mathbf{z}}J)^T = (D_{\mathbf{z}}(D_{\mathbf{z}^*}J)^T)^* = (H_{\mathbf{z}^*, \mathbf{z}}J)^*, \quad (3.1.6)$$

$$H_{\mathbf{z}, \mathbf{z}}J = D_{\mathbf{z}}(D_{\mathbf{z}}J)^T = (D_{\mathbf{z}^*}(D_{\mathbf{z}^*}J)^T)^* = (H_{\mathbf{z}^*, \mathbf{z}^*}J)^* \quad (3.1.7)$$

Найдем теперь матрицы Гессе для MSE из выражения (2.1.7):

$$d(D_{\mathbf{z}^*}J)^T = -d \left[ (D_{\mathbf{z}}\mathbf{y})^H \mathbf{e} \right] - d \left[ (D_{\mathbf{z}^*}\mathbf{y})^T \mathbf{e}^* \right]. \quad (3.1.8)$$

Найдем первое слагаемое выражения (3.1.8):

$$\begin{aligned} d \left[ (D_{\mathbf{z}}\mathbf{y})^H \mathbf{e} \right] &= d \left[ \mathbf{I}_K(D_{\mathbf{z}}\mathbf{y})^H \mathbf{e} \right] = d \left[ (\mathbf{e}^T \otimes \mathbf{I}_K) \text{vec}(D_{\mathbf{z}}\mathbf{y})^H \right] = \\ &= (-d\mathbf{y}^T \otimes \mathbf{I}_K) \text{vec}(D_{\mathbf{z}}\mathbf{y})^H + (\mathbf{e}^T \otimes \mathbf{I}_K) d \text{vec}(D_{\mathbf{z}}\mathbf{y})^H = \\ &= -(D_{\mathbf{z}}\mathbf{y})^H d\mathbf{y} + (\mathbf{e}^T \otimes \mathbf{I}_K) d \text{vec}(D_{\mathbf{z}}\mathbf{y})^H, \end{aligned} \quad (3.1.9)$$

где  $\mathbf{I}_K \in \mathbb{R}^{K \times K}$  – единичная матрица,  $\otimes$  – произведение Кронекера,  $\text{vec}(\cdot)$  – оператор векторизации матрицы, вытягивает матрицу в единый столбец. Отметим, что в выкладках (3.1.9) использовано свойство  $\text{vec}(ABC) = (C^T \otimes A)\text{vec}(B)$  [10].

В выражении (3.1.9) разложим вектор в ряд Тейлора до слагаемых первого порядка малости:

$$d\text{vec}(D_{\mathbf{z}} \mathbf{y})^H = D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}} \mathbf{y})^H dz + D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^H dz^*. \quad (3.1.10)$$

Подставим (2.1.5), (3.1.10) в (3.1.9):

$$\begin{aligned} d \left[ (D_{\mathbf{z}} \mathbf{y})^H \mathbf{e} \right] &= -(D_{\mathbf{z}} \mathbf{y})^H (D_{\mathbf{z}} \mathbf{y}) dz - (D_{\mathbf{z}} \mathbf{y})^H (D_{\mathbf{z}^*} \mathbf{y}) dz^* + \\ &+ (\mathbf{e}^T \otimes \mathbf{I}_K) D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}} \mathbf{y})^H dz + (\mathbf{e}^T \otimes \mathbf{I}_K) D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^H dz^*. \end{aligned} \quad (3.1.11)$$

По аналогии с (3.1.11) получаем выражение для второго слагаемого дифференциала градиента потерь MSE:

$$\begin{aligned} d \left[ (D_{\mathbf{z}^*} \mathbf{y})^T \mathbf{e}^* \right] &= -(D_{\mathbf{z}^*} \mathbf{y})^T (D_{\mathbf{z}^*} \mathbf{y}^*) dz^* - (D_{\mathbf{z}^*} \mathbf{y})^T (D_{\mathbf{z}} \mathbf{y}^*) dz + \\ &+ (\mathbf{e}^H \otimes \mathbf{I}_K) D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^T dz + (\mathbf{e}^H \otimes \mathbf{I}_K) D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^T dz^*. \end{aligned} \quad (3.1.12)$$

Подставим выражения (3.1.11), (3.1.12) в (3.1.8) и приравняем выражения при дифференциалах  $dz$ ,  $dz^*$  с выражениями при дифференциалах в (2.1.4), получим искомые Гессианы:

$$\begin{aligned} H_{\mathbf{z}^*, \mathbf{z}} J &= D_{\mathbf{z}} (D_{\mathbf{z}^*} J)^T = (D_{\mathbf{z}} \mathbf{y})^H (D_{\mathbf{z}} \mathbf{y}) - (\mathbf{e}^T \otimes \mathbf{I}_K) D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}} \mathbf{y})^H + \\ &+ (D_{\mathbf{z}^*} \mathbf{y})^T (D_{\mathbf{z}} \mathbf{y}^*) - (\mathbf{e}^H \otimes \mathbf{I}_K) D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^T, \end{aligned} \quad (3.1.13)$$

$$\begin{aligned} H_{\mathbf{z}^*, \mathbf{z}^*} J &= D_{\mathbf{z}^*} (D_{\mathbf{z}^*} J)^T = (D_{\mathbf{z}} \mathbf{y})^H (D_{\mathbf{z}^*} \mathbf{y}) - (\mathbf{e}^T \otimes \mathbf{I}_K) D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}} \mathbf{y})^H + \\ &+ (D_{\mathbf{z}^*} \mathbf{y})^T (D_{\mathbf{z}^*} \mathbf{y}^*) - (\mathbf{e}^H \otimes \mathbf{I}_K) D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^T. \end{aligned} \quad (3.1.14)$$

Таким образом, выражения (3.1.13), (3.1.14), (3.1.6), (3.1.7) определяют полную матрицу вторых производных, а выражения (3.1.5), (2.1.7) полный градиент функции потерь MSE в разностном уравнении (3.1.4).

## 3.2 Модификации метода Ньютона

Как отмечалось ранее, метод Ньютона сходится квадратично вблизи точки оптимума. При старте алгоритма с удалённой от оптимума начальной точки итеративная процедура может расходиться [9]. Для обеспечения глобальной сходимости широко применяются модификации метода Ньютона.

Алгоритм Левенберга-Марквардта [9] является одной из модификаций метода Ньютона. Метод заключается в том, чтобы внести добавку к гессиану (3.1.4) для

улучшения численной устойчивости алгоритма при обращении матрицы:

$$\mathbf{z}_{k+1} = \mathbf{z}_k - (\mathbf{H}_J(\mathbf{z}_k) + \lambda_k \mathbf{I}_K)^{-1} D_{\mathbf{z}} J(\mathbf{z}_k), \quad (3.2.1)$$

где  $\lambda_k$  – действительное положительное число. Добавка  $\lambda_k \mathbf{I}_K$  – регуляризация, которая изменяет направление шага алгоритма в сторону антиградиента тем больше, чем больше норма  $\|\lambda_k \mathbf{I}_K\|$ :

$$\lim_{\lambda_k \rightarrow \infty} (\mathbf{H}_J(\mathbf{z}_k) + \lambda_k \mathbf{I}_K)^{-1} = \mu \mathbf{I}_K, \quad \mu \rightarrow 0.$$

Число  $\lambda_k$  подбирается каждую итерацию при помощи одномерной оптимизации. Такие алгоритмы подбора  $\lambda_k$  вычислительно проще обращения матрицы.

Кроме того, практическое применение находит демпфированный метод Ньютона [14], представленный алгоритмом 1. Идея модификации заключается в том, чтобы на каждой итерации уменьшать шаг алгоритма до тех пор, пока значение целевой функции не станет меньше, чем при исходном значении шага.

На вход демпфированного метода Ньютона поступают векторы сигнала передатчика  $\mathbf{x}$  и принимаемого  $\mathbf{d}$  сигналов, начальные значения параметров модели  $\mathbf{z}$ , исходное значение шага  $\mu$  и желаемое число итераций  $P$ . Функция  $\text{model}(\cdot)$  выдает вектор сигналов на выходе адаптивной модели на основе текущих параметров и входа модели.

Далее по методу Ньютона вычисляется вектор шага коэффициентов, после чего определяются новые значения коэффициентов с текущим значением шага  $\mu$ .

На основе новых параметров получаем выход модели и новое отклонение. Если значение критерия MSE стало меньше после обновления коэффициентов, то шаг  $\mu$  удваивается. Такое увеличение шага будет происходить пока он меньше единицы. Параметры запоминаются, алгоритм переходит в новую итерацию.

Если значение критерия MSE ухудшилось, алгоритм будет уменьшать значение шага  $\mu$  до тех пор, пока значение критерия не станет лучше исходного. После такой процедуры алгоритм запомнит значения параметров и запустится новая итерация.

Ввиду того, что в задачах нелинейной обработки сигналов задача вообще говоря не является квадратичной, методу Ньютона понадобится больше одной итерации для поиска точки оптимума.

### 3.3 Смешанный метод Ньютона

Смешанный метод Ньютона (Mixed Newton method – MNM) — модификация метода Ньютона (3.1.4) в условиях голоморфности вектора ошибки  $\mathbf{e} = \mathbf{e}(\mathbf{z}, \mathbf{x})$ . Иначе говоря смешанный метод Ньютона работает в преположении независимости вектора

---

**Algorithm 1:** Damped Newton method

---

```

Data:  $x, d, z, P, \mu$ 
Result: Obtain  $z$ 
1 for  $i \in \overline{0, P - 1}$  do
2    $y_{prev} = \text{model}(z, x);$ 
3    $e_{prev} = d - y_{prev};$ 
4    $\Delta z = H_J^{-1} \nabla J(z);$ 
5    $z_{tmp} = z + \mu \Delta z;$ 
6    $y_{curr} = \text{model}(z_{tmp}, x);$ 
7    $e_{curr} = d - y_{curr};$ 
8   if  $MSE(e_{prev}) - MSE(e_{curr}) \geq 0$  then
9      $\mu = \mu * 2;$ 
10    if  $\mu > 1$  then
11       $\mu = 1;$ 
12     $z = z_{tmp},$ 
13  else
14    flag = True;
15    while flag do
16       $\mu = \mu / 1.5;$ 
17       $z_{tmp1} = z + \mu \Delta z;$ 
18       $y_{div} = \text{model}(z_{tmp1}, x);$ 
19       $e_{div} = d - y_{div};$ 
20      if  $MSE(e_{prev}) - MSE(e_{div}) \geq 0$  then
21        flag = False;
22         $z = z_{div}$ 

```

---

ошибки от сопряженных параметров  $z^*$ :  $e = e(x, z)$ :

$$D_{z^*} e(x, z) = -D_{z^*} y(x, z) = 0. \quad (3.3.1)$$

Идея алгоритма заключается в том, чтобы использовать только смешанный гессиан  $H_{z^*, z} J$  для обучения параметров модели [mixed\_newton]. Разностное уравнения смешанного метода Ньютона может быть представлено следующим выражением:

$$z_{k+1} = z_k - \mu (H_{z^*, z} J)^{-1} (D_{z^*} J)^T, \quad (3.3.2)$$

где  $J$  — функция потерь MSE, определенная выражением (1.5.1).

В этом случае MNM обладает важнейшим свойством отталкивания от седловых точек [mixed\_newton] в то время, как метод Ньютона, использующий полную матрицу вторых производных (3.1.4), может застревать в локальных оптимумах и седловых точках без дополнительных модификаций шага и регуляризации, описанных в разделе 3.2.

Вычислим смешанный гессиан для рассматриваемой функции потерь (1.5.1) в

случае голоморфности вектора ошибки. Согласно выражению (3.1.13):

$$H_{z^*, z} J = (D_z \mathbf{y})^H D_z \mathbf{y}, \quad (3.3.3)$$

ввиду выражения (3.3.1), а также  $D_{z^*} \text{vec}(D_z \mathbf{y})^H = 0$ . Последнее выражение справедливо поскольку матрица  $(D_z \mathbf{y})^H$  зависит только от  $z^*$  и не зависит от  $z$  ввиду голоморфности  $\mathbf{y} = \mathbf{y}(\mathbf{x}, z)$ .

Таким образом, учитывая выражения (3.3.3), (2.1.10) и (3.3.2) алгоритм смешанного метода Ньютона можно представить следующим разностным уравнением:

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \mu((D_z \mathbf{y})^H D_z \mathbf{y})^{-1} (D_z \mathbf{y})^H \mathbf{e}. \quad (3.3.4)$$

Отдельно отметим, что сложность хранения гессиана может быть оценена как  $o(K^2)$  [15], если  $K \times K$  – размерность гессиана. Обращение гессиана может быть сведено к решению системы из  $M$  линейных уравнений. Вычислительная сложность такой операции может быть оценена как  $o(M^3)$  [15].

### 3.4 Метод LS

Метод LS – Least Squares представляет собой один шаг смешанного метода Ньютона с шагом адаптации  $\mu = 1$  (3.3.4). В случае голоморфной по параметрам модели  $z$  ошибки  $\mathbf{e} = \mathbf{e}(\mathbf{x}, z)$  и линейности по параметрам модели

$$\begin{aligned} \mathbf{e} &= \mathbf{e}(\mathbf{x}, z) = \mathbf{d} - \mathbf{y}(\mathbf{x}, z), \\ \mathbf{y} &= \mathbf{U}(\mathbf{x})\mathbf{z}, \quad \mathbf{U}(\mathbf{x}) – \text{матрица входных данных}, \end{aligned} \quad (3.4.1)$$

функция потерь MSE (1.5.1) является квадратичной по параметрам модели. В этом случае минимизация MSE сводится к решению линейной системы [7]. В связи с этим LS находит глобальный оптимум функции потерь за 1 шаг [7].

Поскольку глобальный оптимум может быть найден из любой начальной точки, включая  $\mathbf{z}_0 = \mathbf{0}$ , то финальный вид LS метода представлен выражением:

$$\mathbf{z}_{\text{opt}} = ((D_z \mathbf{y})^H D_z \mathbf{y})^{-1} (D_z \mathbf{y})^H \mathbf{d}, \quad (3.4.2)$$

где  $D_z \mathbf{y} = \mathbf{U}(\mathbf{x})$  ввиду линейности модели (3.4.1),  $\mathbf{e} = \mathbf{d} - \mathbf{U}\mathbf{z}_0 = \mathbf{d}$  – начальная ошибка адаптации.

## 4. Методы адаптации моделей нелинейных искажений не требующие явного вычисления матрицы Гессе

Методы второго порядка подразумевают необходимость накапливать матрицу вторых производных для каждого нового блока данных, хранить в памяти, а также обращать, обеспечивая при этом численную устойчивость процедуры инверсии матрицы [15]. Такие методы являются вычислительно сложными, кроме того достаточно трудными с точки зрения реализации в целочисленной арифметике. В связи с этим в теории нелинейной обработки сигналов предлагается использовать методы применяющие приближенное вычисление обратной матрицы вторых производных либо приближенное вычисление произведения обратной матрицы и градиента функции потерь.

### 4.1 Метод сопряженных градиентов

Рассмотрим шаг смешанного метода Ньютона как минимизацию второй нормы ошибки  $e(z)$  (1.5.1), разложенной до членов первого порядка малости. Ввиду голоморфности  $e$ :

$$\Delta z_k = \arg \min_z f(z) = \arg \min_z \left\| e(z)|_{z=z_k} + D_z e(z)|_{z=z_k} (z - z_k) \right\|_2^2. \quad (4.1.1)$$

Распишем вторую норму разложения ошибки в выражении (4.1.1) и приведем подобные слагаемые, получим следующее выражение для квадратичной формы:

$$\begin{aligned} f(z) &= (z - z_k)^H (D_z e(z))^H|_{z=z_k} D_z e(z)|_{z=z_k} (z - z_k) + \\ &+ e^H(z_k) D_z e(z)|_{z=z_k} (z - z_k) + (z - z_k)^H (D_z e(z))^H|_{z=z_k} e(z_k) + \\ &+ e^H(z_k) e(z_k) = \Delta z_k^H M \Delta z_k + b^H \Delta z_k + \Delta z_k^H b + c, \end{aligned} \quad (4.1.2)$$

где  $M = [(D_z e(z))^H D_z e(z)]|_{z=z_k} = H_{z^*, z} J|_{z=z_k}$  — неотрицательно определена матрица смешанных производных функции потерь MSE (1.5.1),  $b = (D_z e(z))^H e(z)|_{z=z_k} = (D_{z^*} J)^T$  — градиент функции потерь MSE (1.5.1),  $c = e^H(z_k) e(z_k)$  — константа. Отметим, что поскольку

$$\arg \min_z f(z) = \arg \min_z f(z) - c, \quad (4.1.3)$$

константу  $c$  можно отбросить при выводе метода сопряженных градиентов.

Для простоты выражений обозначим  $x \equiv \Delta z_k$  и перепишем квадратичную форму:

$$f(x) = x^H M x + b^H x + x^H b, \quad M \succcurlyeq 0, \quad (4.1.4)$$

Выведем метод сопряженных градиентов для минимизации квадратичной формы (4.1.4)

для поиска приращения параметров модели  $\mathbf{x}_k$  на шаге  $k$ .

Пусть множество векторов  $\{\mathbf{p}_j\}_{j=0}^{K-1} \in \mathbb{C}^{K \times 1}$  — сопряжены относительно матрицы  $\mathbf{M}$ , т.е. выполняется условие:

$$\mathbf{p}_j^H \mathbf{M} \mathbf{p}_l = \mathbf{0} \quad \forall k \neq l, \quad (4.1.5)$$

тогда вектора  $\{\mathbf{p}_j\}_{j=0}^{K-1}$  — система линейно-независимых векторов по аналогии с вещественным случаем [HestenesStiefel1952], т.е. представляют собой базис в пространстве  $\mathbb{C}^{K \times 1}$ . Будем искать приращение  $\mathbf{x}$  минимизирующее (4.1.4) методом простой итерации, раскладывая приращение на  $i$ -ом шаге по линейно-независимым векторам сопряженным относительно  $\mathbf{M}$ :

$$\mathbf{x}_{i+1} = \mathbf{x}_0 + \sum_{j=0}^i \alpha_j \mathbf{p}_j, \quad (4.1.6)$$

где  $\alpha_j \in \mathbb{C}$  — коэффициенты разложение по базису,  $\mathbf{x}_0 \in \mathbb{C}^{K \times 1}$  — начальное приближение решения  $\mathbf{x}$ . Подставим выражение (4.1.6) в квадратичную форму (4.1.4):

$$f(\mathbf{x}_{i+1}) = f(\mathbf{x}_0) + \sum_{j=0}^i \left( \alpha_j (\mathbf{M} \mathbf{x}_0 + \mathbf{b})^H \mathbf{p}_j + \alpha_j^* \mathbf{p}_j^H (\mathbf{M} \mathbf{x}_0 + \mathbf{b}) + |\alpha_j|^2 \mathbf{p}_j^H \mathbf{M} \mathbf{p}_j \right), \quad (4.1.7)$$

где  $\mathbf{p}_0 = \mathbf{M} \mathbf{x}_0 + \mathbf{b}$  — первый вектор множества сопряженных векторов, инициализируется как градиент функции  $f(\mathbf{x})$ ,  $\alpha_j^* \in \mathbb{C}$  — комплексно сопряжение коэффициента разложения  $\alpha_j$ . Найдём коэффициенты разложения  $\alpha_j$  из условия минимизации квадратичной формы по коэффициентам разложения  $D_{\alpha_j^*} f(\mathbf{x}, \alpha_j) \Big|_{\mathbf{x}=\mathbf{x}_{i+1}} = 0$ :

$$\mathbf{p}_j^H \mathbf{p}_0 + \alpha_j \mathbf{p}_j^H \mathbf{M} \mathbf{p}_j = \mathbf{0}, \quad (4.1.8)$$

$$\alpha_j = -\frac{\mathbf{p}_j^H \mathbf{p}_0}{\mathbf{p}_j^H \mathbf{M} \mathbf{p}_j}. \quad (4.1.9)$$

Таким образом рекурсивная формула обновления приближения решения  $\mathbf{x}$ :

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{p}_i = \mathbf{x}_i - \frac{\mathbf{p}_i^H \mathbf{p}_0}{\mathbf{p}_i^H \mathbf{M} \mathbf{p}_i} \mathbf{p}_i. \quad (4.1.10)$$

Отметим, что  $\mathbf{p}_i^H \mathbf{p}_0 = \mathbf{p}_i^H (\mathbf{M} \mathbf{x}_0 + \mathbf{b}) = \mathbf{p}_i^H (\mathbf{M} \mathbf{x}_i + \mathbf{b})$  по условию сопряженности векторов (4.1.5). Обозначим градиент функции  $f(\mathbf{x})$  в точке  $\mathbf{x}_i$  как  $\mathbf{r}_i \equiv D_{\mathbf{x}^*} f(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_i} = \mathbf{M} \mathbf{x}_i + \mathbf{b}$ , тогда рекурсивное выражение пересчета приближения решения (4.1.10) представляется как:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{p}_i = \mathbf{x}_i - \frac{\mathbf{p}_i^H \mathbf{r}_i}{\mathbf{p}_i^H \mathbf{M} \mathbf{p}_i} \mathbf{p}_i. \quad (4.1.11)$$

В теории метода сопряженного градиента для вещественных систем линейных уравнений методом математической индукции доказывается, что для системы  $\{\mathbf{p}_j\}_{j=0}^i$

векторов сопряженных относительно  $\mathbf{M}$  (4.1.5) для  $i \geq 0$  и  $j \leq i$  справедливо выражение [HestenesStiefel1952]:

$$\mathbf{r}_{i+1}^H \mathbf{p}_j = 0. \quad (4.1.12)$$

Выражение (4.1.12) представляет собой физический смысл векторов  $\mathbf{p}_j$ ,  $\mathbf{p}_0 = \mathbf{M}\mathbf{x}_0 + \mathbf{b}$ . Он заключается в том, что для упомянутой системы сопряженных векторов градиент квадратичной формы (4.1.4) в точке, определяемой итеративным алгоритмом (4.1.6) будет ортогонален всем ранее найденным сопряженным направлениям. Иначе говоря градиент в новой точке не будет содержать избыточности.

В связи с физическим смыслом сопряженных векторов (4.1.12) определим выбор сопряженных направлений следующим итеративным алгоритмом:

$$\mathbf{p}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1} \mathbf{p}_i, \quad (4.1.13)$$

где параметры  $\beta_i$  определяются также из условия сопряженности (4.1.5):

$$\beta_{i+1} = -\frac{\mathbf{p}_i^H \mathbf{M} \mathbf{r}_{i+1}}{\mathbf{p}_i^H \mathbf{M} \mathbf{p}_i}. \quad (4.1.14)$$

Отметим, что все направления найденные таким итеративным методом (4.1.13), (4.1.14) будут сопряженными, т.е. верно  $\mathbf{p}_j^H \mathbf{M} \mathbf{p}_i = 0 \forall i > 1, \forall j < i$  [HestenesStiefel1952].

Таким образом на основе рекурсивных выражений приближения решения (4.1.11), поиска сопряженных направлений (4.1.13), (4.1.14) метод сопряженных градиентов для приближенного поиска шага смешанного метода Ньютона (4.1.1) представлен алгоритмом 2:

Метод сопряжённых градиентов итерационно устраняет влияние наиболее значимых собственных значений матрицы  $\mathbf{M}$  при минимизации квадратичной формы (4.1.4) [HestenesStiefel1952]. В случае решения линейной задачи аддитивной фильтрации роль матрицы  $M$  играет оценка корреляционной матрицы входного сигнала фильтра. Для узкополосных сигналов число наиболее значимых собственных значений корреляционной матрицы невелико [16]. По этой причине метод сопряженных градиентов сходится за число итераций существенно меньше числа параметров модели тем самым решая проблему высокой вычислительной сложности смешанного метода Ньютона [17] обусловленную обращением матрицы вторых производных.

## 4.2 Метод DCD

Метод покоординатного спуска (англ. DCD – Dichotomous Coordinate Descent) является методом поиска решения систем линейных уравнений с квадратной матрицей и не относится к методам первого порядка [18].

Идея применения алгоритма, заключается также в том, чтобы аппроксимировать шаг метода Ньютона (3.3.4) путем решения оптимизационной задачи (4.1.4). Как было отмечено в разделе 4.1 поиск шага смешанного метода Ньютона эквива-

---

**Algorithm 2:** Метод сопряженных градиентов

---

**Require:**  $\mathbf{z}_0$  – начальные значения параметров модели,  $T$  – число обновлений параметров модели,  $L$  – число итераций метода сопряженных градиентов между обновлениями параметров модели,  $\mu$  – величина шага обновления параметров.

**Ensure:**  $\mathbf{z}_{T-1}$  – конечное значение параметров модели.

- 1: **for**  $t = 1$  **to**  $T$  **do**
- 2:     Получить текущие параметры модели  $\mathbf{z}_t$
- 3:     Инициализировать приращение  $\mathbf{x}_0$
- 4:     Вычислить смешанный гессиан MSE  $\mathbf{M}_t = H_{\mathbf{z}^*, \mathbf{z}} J$
- 5:     Вычислить градиент MSE  $\mathbf{b}_t = (D_{\mathbf{z}^*} J)^T$
- 6:     Вычислить градиент квадратичной формы  $\mathbf{r}_0 = \mathbf{M}_t \mathbf{x}_0 + \mathbf{b}_t$
- 7:     Инициализировать направление  $\mathbf{p}_0 = \mathbf{r}_0$
- 8:     **for**  $k = 0$  **to**  $L - 1$  **do**
- 9:          $\xi_k = \mathbf{M}_t \mathbf{p}_k$
- 10:          $\alpha_k = -\frac{\mathbf{p}_k^H \mathbf{r}_k}{\mathbf{p}_k^H \xi_k}$
- 11:          $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$
- 12:          $\mathbf{r}_{k+1} = \mathbf{r}_k + \alpha_k \xi_k$
- 13:          $\beta_{k+1} = -\frac{\xi_k^H \mathbf{r}_{k+1}}{\mathbf{p}_k^H \xi_k}$
- 14:          $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_{k+1} \mathbf{p}_k$
- 15:     **end for**
- 16:     Пересчитать параметры модели:  $\mathbf{z}_{t+1} = \mathbf{z}_t + \mu \mathbf{x}_{K-1}$
- 17: **end for**
- 18:
- 19: **return**  $\mathbf{z}_{T-1}$

---

лентен минимизации квадратичной формы (4.1.4), что в свою очередь эквивалентно решению системы линейных уравнений:

$$\mathbf{M} \mathbf{x} = -\mathbf{b}, \quad (4.2.1)$$

где  $\mathbf{M} = [(D_{\mathbf{z}} \mathbf{e}(\mathbf{z}))^H D_{\mathbf{z}} \mathbf{e}(\mathbf{z})] \Big|_{\mathbf{z}=\mathbf{z}_k} = H_{\mathbf{z}^*, \mathbf{z}} J \Big|_{\mathbf{z}=\mathbf{z}_k}$  – неотрицательно определена матрица смешанных производных функции потерь MSE (1.5.1),  $\mathbf{b} = (D_{\mathbf{z}} \mathbf{e}(\mathbf{z}))^H \mathbf{e}(\mathbf{z}) \Big|_{\mathbf{z}=\mathbf{z}_k} = (D_{\mathbf{z}^*} J)^T$  – градиент функции потерь MSE (1.5.1).

Введем вектор невязки следующим образом:

$$\mathbf{r} = \mathbf{M} \mathbf{x} + \mathbf{b}. \quad (4.2.2)$$

Идея алгоритма заключается в том, чтобы упростить процедуру обращения матрицы. Для этого на каждой итерации предлагается выделять наибольший по модулю элемент в векторе ошибки  $\mathbf{r}$ , обнулять его и пересчитывать вектор коэффициентов, двигаясь Таким образом, по направлению убывания среднего квадрата ошибки вдоль соответствующей координаты вектора коэффициентов. Метод DCD для вещественнозначных векторов и матриц описан в алгоритме 3 [18].

Константы  $K_1, K_2$  задаются пользователем.  $K_1$  определяет количество шагов в направлении уменьшения среднего квадрата ошибки,  $K_2$  задаёт число возможных

---

**Algorithm 3:** Real-valued leading DCD algorithm

---

**Data:**  $M, r, K_1, K_2, T, \mu$

**Result:**  $z_{T-1}, r$

```

1 for  $t \in \overline{0, T-1}$  do
2   Инициализация:
3    $\Delta z = 0, \alpha = H, m = 1$ 
4   for  $i \in \overline{0, K_1}$  do
5      $k = \arg \max_{p=\overline{0, P-1}} \{|r_p|\}$ , перейти на строку 9;
6      $m = m + 1, \alpha = \alpha/2;$ 
7     if  $m > K_2$  then
8       завершения алгоритма;
9     if  $|r_k| \leq (\frac{\alpha}{2})M_{k,k}$  then
10      перейти на строку 6;
11       $\Delta z_k = \Delta z_k - \text{sign}(r_k)\alpha;$ 
12       $r = r + \text{sign}(r_n)\alpha M_{:,k}$ 
13   Пересчитать параметры модели:  $z_{t+1} = z_t + \mu \Delta z;$ 

```

---

дроблений шага  $\alpha$  на каждой итерации перед внесением изменения в вектор приращения  $\Delta z$ .

Выбирается индекс  $k$  наибольшей по модулю координаты невязки  $r_k$ . Далее алгоритм уменьшает шаг  $\alpha$  до тех пор, пока условие строки 9 не станет ложным. Таким образом, шаг  $\alpha$  будет уменьшаться до тех пор, пока приращение невязки  $\alpha M_{:,k}$  не станет меньше модуля самой невязки. Это необходимо для устойчивости алгоритма.

Далее вносится приращение в вектор невязки  $r$ . Для того, чтобы преобразование было равносильным, из вектора приращения параметров  $\Delta z$  необходимо добавить поправку  $\text{sign}(r_k)\alpha$  по координате с индексом  $k$ .

Рассмотрим алгоритм DCD для комплекснозначных векторов и матриц. Такой алгоритм можно получить сведением системы комплекснозначных уравнений к системе вещественнозначных уравнений и воспользовавшись алгоритмом 3. Сгруппируем комплекснозначные вектора и матрицы следующим образом:

$$\begin{aligned} M &= M_1 + jM_2, \quad M \in \mathbb{C}^{K/2 \times K/2} \\ z &= z_1 + jz_2, \quad z \in \mathbb{C}^{K/2 \times 1} \\ b &= b_1 + jb_2, \quad b \in \mathbb{C}^{K/2 \times 1}, \end{aligned} \tag{4.2.3}$$

где  $M_1, M_2, z_1, z_2, b_1, b_2$  – вещественнозначные объекты. Тогда:

$$Mz = (M_1z_1 - M_2z_2) + j(M_2z_1 + M_1z_2) = c_1 + jc_2, \tag{4.2.4}$$

$$\begin{pmatrix} M_1z_1 - M_2z_2 \\ M_2z_1 + M_1z_2 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \tag{4.2.5}$$

$$\begin{pmatrix} \mathbf{M}_1 & -\mathbf{M}_2 \\ \mathbf{M}_2 & \mathbf{M}_1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = - \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}. \quad (4.2.6)$$

Из матричного уравнения (4.2.6) следует, что решение системы  $K/2$  комплекснозначных уравнений с  $K/2$  переменными эквивалентно решению  $K$  вещественнозначных уравнений с  $K$  переменными. Исходя из этого DCD алгоритм для комплекснозначной системы размерности  $K/2 \times K/2$  может быть представлен, как алгоритм 4 [18].

---

**Algorithm 4:** Complex-valued leading DCD algorithm

---

```

Data:  $\mathbf{M}, \mathbf{r}, K_1, K_2, T, \mu$ 
Result:  $\mathbf{z}_{T-1}, \mathbf{r}$ 
1 for  $t \in \overline{0, T-1}$  do
2   Инициализация:
3      $\Delta\mathbf{z} = \mathbf{0}, \alpha = H, m = 1$ 
4     for  $i \in \overline{0, K_1}$  do
5        $[k, s] = \arg \max_{p=\overline{0, (P-1)/2}} \{|Re(r_p)|, |Im(r_p)|\}$ , go to line 13;
6        $m = m + 1, \alpha = \alpha/2;$ 
7       if  $m > K_2$  then
8         завершение алгоритма;
9       if  $s = 1$  then
10         $r_{tmp} = Re(r_k)$ 
11      else
12         $r_{tmp} = Im(r_k)$ 
13      if  $|r_{tmp}| \leq (\frac{\alpha}{2})M_{k,k}$  then
14        перейти на строку 6;
15       $\Delta z_k = \Delta z_k - \text{sign}(r_k)s\alpha;$ 
16       $\mathbf{r} = \mathbf{r} + \text{sign}(r_n)s\alpha\mathbf{M}_{:,k}$ 
17   Пересчитать параметры модели:  $\mathbf{z}_{t+1} = \mathbf{z}_t + \mu\Delta\mathbf{z};$ 

```

---

Сначала производится поиск индекса  $k$  наибольшего модуля среди вещественных и мнимых частей элементов вектора отклонения  $\mathbf{r}$ . В переменную  $s$  записывается 1, если в наибольшей по модулю является вещественная часть одной из координат  $\mathbf{r}$ ,  $s = j$ , если мнимая. Эта переменная учитывает вещественный или мнимый характер приращения невязки и поправки на вектор коэффициентов.

Затем, как и для вещественнозначного случая, шаг алгоритма уменьшается до тех пор, пока приращение вектора невязки  $\mathbf{r}$  не станет меньше модуля самой невязки. После чего производится пересчёт вектора приращения коэффициентов  $\Delta\mathbf{z}$  и вектора невязки  $\mathbf{r}$ .

В результате алгоритм 4 выдаёт вектор невязки  $\mathbf{r}$  и вектор приращения коэффициентов  $\Delta\mathbf{z}$ , после чего производится обновление коэффициентов.

Алгоритм является методом приближенного решения системы линейных уравнений и работает итеративно. В связи с приближенным характером решения СЛАУ вблизи точки оптимума процедура DCD не гарантирует монотонной сходимости.

### 4.3 Квазиньютоновские методы

Как отмечалось ранее, метод Ньютона обладает локальной квадратичной сходимостью, высокой точностью решения. Кроме того он является аффинно инвариантным [14], что означает, что вид разностного уравнения и скорость сходимости не меняются при аффинном преобразовании координат. Однако, для реализации метода требуется хранение гессиана в памяти, также операция обращения гессиана, которая является вычислительно сложной.

Для снижения вычислительной сложности применяют методы, которые обладают скоростью сходимости выше, чем у градиентного спуска, имея при этом накладные расходы меньше, чем у метода Ньютона. Квазиньютоновские методы объединяют в себе простоту градиентного спуска с точки зрения вычислительной сложности и высокую скорость сходимости.

Как отмечать в разделе 4.1 смешанный метод Ньютона раскладывает целевую функцию в окрестности точки (4.1.2)  $\mathbf{z}_k$ :

$$\begin{aligned} p(\mathbf{z}_{k+1}) = J(\mathbf{z}_k) + (\mathbf{z}_{k+1} - \mathbf{z}_k)^H (D_{\mathbf{z}^*} J)^T + (\mathbf{z}_{k+1} - \mathbf{z}_k)^T (D_{\mathbf{z}^*} J)^H + \\ + \frac{1}{2} (\mathbf{z}_{k+1} - \mathbf{z}_k)^H H_{\mathbf{z}^*, \mathbf{z}} \nabla_J (\mathbf{z}_{k+1} - \mathbf{z}_k). \end{aligned} \quad (4.3.1)$$

Идея квазиньютоновских алгоритмов в том, чтобы заменить матрицу Гессе на её оценку:

$$H_{\mathbf{z}^*, \mathbf{z}} \rightarrow \mathbf{B}_k, \quad \mathbf{B}_k \succ 0. \quad (4.3.2)$$

Отметим, что искомая оценка гессиана должна быть положительно определённой. Выполним замену:

$$\mathbf{z}_{k+1} - \mathbf{z}_k = \Delta \mathbf{z}_k, \quad (4.3.3)$$

тогда из (4.3.1), (4.3.2) и (4.3.3) получим:

$$\begin{aligned} p(\Delta \mathbf{z}_k + \mathbf{z}_k) = J(\mathbf{z}_k) + \Delta \mathbf{z}_k^H (D_{\mathbf{z}_k^*} J)^T + \\ + \Delta \mathbf{z}_k^T (D_{\mathbf{z}_k^*} J)^H + \frac{1}{2} \Delta \mathbf{z}_k^H \mathbf{B}_k \Delta \mathbf{z}_k. \end{aligned} \quad (4.3.4)$$

По аналогии со смешанным методом Ньютона (3.3.2) квазиньютоновский метод будет иметь вид:

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \mu \mathbf{B}_k^{-1} \nabla_J (\mathbf{z}_k) = \mathbf{z}_k - \mu \mathbf{H}_k \nabla_J (\mathbf{z}_k), \quad (4.3.5)$$

где  $\mathbf{B}_k, \mathbf{H}_k$  – оценки прямого и обратного гессиана соответственно.

Требования, которые предъявляются к оценке гессиана:

- Быстрое обновление  $\mathbf{B}_k \rightarrow \mathbf{B}_{k+1}$  в условиях, когда доступны только градиенты
- Быстрый поиск направления:

$$\Delta \mathbf{z}_k = \mu \mathbf{B}_k^{-1} \nabla_J (\mathbf{z}_k) = \mu \mathbf{H}_k \nabla_J (\mathbf{z}_k) \quad (4.3.6)$$

- Компактное хранение  $\mathbf{B}_{k+1}$
- Сверхлинейная сходимость

Получим уравнение на поиск матрицы  $\mathbf{B}_{k+1}$ . Для этого сформулируем так называемое правило двух градиентов [19]:

$$\frac{\partial p(\Delta \mathbf{z}_{k+1} + \mathbf{z}_{k+1})}{\partial \Delta \mathbf{z}_{k+1}^H} \Big|_{\Delta \mathbf{z}_{k+1} = \mathbf{0}} = (D_{\mathbf{z}_{k+1}^*} J)^T, \quad (4.3.7)$$

$$\frac{\partial p(\Delta \mathbf{z}_{k+1} + \mathbf{z}_{k+1})}{\partial \Delta \mathbf{z}_{k+1}^H} \Big|_{\Delta \mathbf{z}_{k+1} = -\mu \Delta \mathbf{z}_k} = (D_{\mathbf{z}_k^*} J)^T. \quad (4.3.8)$$

Требование (4.3.7) следует напрямую из (4.3.4). Вычислим производную в левой части (4.3.8):

$$\frac{\partial p(\Delta \mathbf{z}_{k+1} + \mathbf{z}_{k+1})}{\partial \Delta \mathbf{z}_{k+1}^H} \Big|_{\Delta \mathbf{z}_{k+1} = -\mu \Delta \mathbf{z}_k} = (D_{\mathbf{z}_{k+1}^*} J)^T - \frac{1}{2} \mathbf{B}_{k+1} \Delta \mathbf{z}_{k+1} = (D_{\mathbf{z}_{k+1}^*} J)^T - \mu \mathbf{B}_{k+1} \Delta \mathbf{z}_k \quad (4.3.9)$$

Требование (4.3.8) приводит к квазиньютоновскому уравнению (англ. Secant equation):

$$(D_{\mathbf{z}_{k+1}^*} J)^T - (D_{\mathbf{z}_k^*} J)^T = \mu \mathbf{B}_{k+1} \Delta \mathbf{z}_k \quad (4.3.10)$$

Сделаем замену и получим квазиньютоновское уравнение в следующем виде:

$$\mathbf{s}_k = \mu (\mathbf{z}_{k+1} - \mathbf{z}_k), \quad (4.3.11)$$

$$\mathbf{f}_k = (D_{\mathbf{z}_{k+1}^*} J)^T - (D_{\mathbf{z}_k^*} J)^T, \quad (4.3.12)$$

$$\mathbf{B}_{k+1} \mathbf{s}_k = \mathbf{f}_k. \quad (4.3.13)$$

В уравнении (4.3.13) векторы  $\mathbf{s}_k, \mathbf{f}_k$  определяются в процессе вычислений. Квазиньютоновское уравнение является уравнением на поиск матрицы  $\mathbf{B}_{k+1}$ .

Отметим, что ввиду требования положительной определённости  $\mathbf{B}_{k+1}$  (4.3.2), уравнение (4.3.13) не будет иметь решение в случае  $\mathbf{s}_k^T \mathbf{f}_k < 0$ . При выборе шага важно следить за тем, чтобы это произведение было неотрицательным.

Кроме того, пусть система (4.3.13) имеет  $F$  уравнений, тогда она будет иметь  $\frac{F(F-1)}{2}$  неизвестных ввиду симметричности оценки гессиана. Это означает, что требуются дополнительные условия для обеспечения единственности решения.

Для устойчивости алгоритма соседние оценки гессиана  $\mathbf{B}_k, \mathbf{B}_{k+1}$  должно быть близки друг к другу.

Далее рассмотрим следующие примеры квазиньютоновских методов: DFP, BFGS и Barzilai-Borwein [20] [21].

#### 4.4 Метод Barzilai-Borwein

Данный метод аппроксимирует гессиан диагональной матрицей. Пусть шаг алгоритма – величина переменная, подстраиваемая на каждой новой итерации. Распишем алгоритм градиентного спуска:

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{z}_k - \mu_k (D_{\mathbf{z}_k^*} J)^T = \mathbf{z}_k - \mu_k \mathbf{I} (D_{\mathbf{z}_k^*} J)^T = \\ &= \mathbf{z}_k - \left( \frac{1}{\mu_k} \mathbf{I} \right)^{-1} (D_{\mathbf{z}_k^*} J)^T = \mathbf{z}_k - \mathbf{B}_k^{-1} (D_{\mathbf{z}_k^*} J)^T, \end{aligned} \quad (4.4.1)$$

Квазиньютоновское уравнение будет иметь вид приближенного равенства, поскольку  $\mathbf{s}_k, \mathbf{f}_k$  заранее известны, а  $\mathbf{B}_{k+1}$  приближается всего одним числом, чего не достаточно для строгого равенства:

$$\mu_k^{-1} \mathbf{s}_{k-1} \approx \mathbf{f}_{k-1}, \quad (4.4.2)$$

где приближенное равенство формализуется следующей задачей минимизации [20]:

$$\min_{\mu_k} \left\| \mathbf{s}_{k-1} - \mu_k \mathbf{f}_k \right\|_2 \Rightarrow \mu_k = \frac{\mathbf{s}_{k-1}^T \mathbf{f}_{k-1}}{\mathbf{f}_{k-1}^T \mathbf{f}_{k-1}}. \quad (4.4.3)$$

Выражение (4.4.3) задаёт способ обновления шага.

Данный метод не является градиентным спуском, поскольку учитывает два предыдущих вектора коэффициентов и градиента для получения нового вектора коэффициентов.

#### 4.5 Метод DFP

Метод DFP (англ. Davidon-Fletcher-Powell) использует поиск оценки матрицы Гёссе на каждой новой итерации [21]. Учтём ранее упомянутое требование, которое заключается в близости оценок гессианов на соседних итерациях. Формально такая задача может быть поставлена следующим образом [21]:

$$\begin{cases} \min_{\mathbf{B}_{k+1}} \left\| \mathbf{B}_k - \mathbf{B}_{k+1} \right\|_2, \\ \mathbf{B}_{k+1} = \mathbf{B}_{k+1}^T, \\ \mathbf{B}_{k+1} \mathbf{s}_k = \mathbf{f}_k. \end{cases} \quad (4.5.1)$$

Аналитическое решение задачи (4.5.1) имеет вид [19]:

$$\mathbf{B}_{k+1} = (\mathbf{I} - \rho_k \mathbf{f}_k \mathbf{s}_k^T) \mathbf{B}_k (\mathbf{I} - \rho_k \mathbf{s}_k \mathbf{f}_k^T) + \rho_k \mathbf{f}_k \mathbf{f}_k^T, \quad (4.5.2)$$

$$\rho_k = \frac{1}{\mathbf{f}_k^T \mathbf{s}_k}. \quad (4.5.3)$$

По формуле Шермана-Вудбери-Моррисона [22] (лемма об обращении матриц) решение (4.5.2) можно преобразовать:

$$\mathbf{B}_{k+1}^{-1} \equiv \mathbf{H}_{k+1} = \mathbf{H}_k - \frac{\mathbf{H}_k \mathbf{f}_k \mathbf{f}_k^T \mathbf{H}_k}{\mathbf{f}_k^T \mathbf{H}_k \mathbf{f}_k} + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{f}_k^T \mathbf{s}_k}. \quad (4.5.4)$$

Формула (4.5.4) задаёт аналитическое выражение для вычисления оценки обратного гессиана. Для вычисления вектора коэффициентов на новой итерации эта оценка подставляется в разностное уравнение (4.3.5).

Сложность хранения оценки обратной матрицы может быть оценена как  $o(K^2)$  – такая же как и для метода Ньютона. Оценим вычислительную сложность пересчета оценки гессиана. В выражении (4.5.4) сложность подсчета третьего слагаемого оценивается как  $o(K^2)$ , поскольку  $\mathbf{s}_k \mathbf{s}_k^T$  представляет собой векторное произведение, в результате которого получается матрица размерности  $K \times K$ .

Второе слагаемое также вычисляется за  $o(K^2)$ . Выражение  $\mathbf{H}_k \mathbf{f}_k$  вычисляется за  $o(K^2)$ . При этом  $\mathbf{f}_k^T \mathbf{H}_k = (\mathbf{H}_k \mathbf{f}_k)^T$  ввиду симметричности  $\mathbf{H}_k$ . Остается произвести векторное умножение в числителе, сложность которого  $o(K^2)$ , а в знаменателе скалярное умножение векторов, сложность которого  $o(K)$ .

## 4.6 Метод BFGS

Метод BFGS (англ. Broyden–Fletcher–Goldfarb–Shanno). Вместо того, чтобы искать оценку гессиана, напрямую поставим задачу поиска оценки обратного гессиана [21]:

$$\begin{cases} \min_{\mathbf{H}_{k+1}} \left\| \mathbf{H}_k - \mathbf{H}_{k+1} \right\|_2, \\ \mathbf{H}_{k+1} = \mathbf{H}_{k+1}^T, \\ \mathbf{H}_{k+1} \mathbf{f}_k = \mathbf{s}_k. \end{cases} \quad (4.6.1)$$

Аналитическое решение задачи (4.6.1) записывается в виде выражения [23]:

$$\mathbf{H}_{k+1} = (\mathbf{I} - \rho_k \mathbf{s}_k \mathbf{f}_k^T) \mathbf{H}_k (\mathbf{I} - \rho_k \mathbf{f}_k \mathbf{s}_k^T) + \rho_k \mathbf{s}_k \mathbf{s}_k^T, \quad (4.6.2)$$

$$\rho_k = \frac{1}{\mathbf{f}_k^T \mathbf{s}_k}. \quad (4.6.3)$$

BFGS обладает локальной сверхлинейной сходимостью.

На практике приведенный метод является наиболее предпочтительным квазиньютоновским методом по поскольку обладает свойством самокоррекции.

Сложность хранения и обращения гессиана оценивается как  $o(K^2)$ , что следует из формулы (4.6.2).

Отметим, что для вычисления коэффициентов на новой итерации необходима не сама матрица Гессе (или обратная к ней), а эффективная процедура умножения матрицы на вектор  $(D_{z_k^* J})^T$ . Кроме того, значения векторов  $\mathbf{f}$ ,  $\mathbf{s}$ , полученные на пер-

вых итерациях могут портить оценки  $\mathbf{B}$ ,  $\mathbf{H}$  на более поздних итерациях. Для подсчет оценки обратного гессиана предлагается хранить в очереди и использовать последние  $m \ll K$  значений векторов  $\mathbf{f}$ ,  $\mathbf{s}$ . На этой идее основана модификация Limited-Memory BFGS [24].

## 5. Математические модели линейных и нелинейных искажений в системах связи

В этом разделе изложена теория, необходимая для реализации рассмотренных алгоритмов адаптации. Рассмотренные модели нелинейных искажений делятся на голоморфные по параметрам модели  $\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{z})$  и неголоморфные  $\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{z}, \mathbf{z}^*)$ ,  $\mathbf{y} \in \mathbb{C}^{N \times 1}$  – выходные отсчеты модели.

К голоморфным моделям применимы метода адаптации на основе градиентного спуска и его модификации 2.1, 2.2, метод Ньютона и его модификации 3.1, 3.2, смешанный метод Ньютона 3.3 и методы построенные на его основе 4.1, 4.2, а также квазиньютоновские методы 4.3.

Для реализации смешенного метода Ньютона, а также методов, построенных на его основе достаточно вычислить матрицу  $D_{\mathbf{z}} \mathbf{y}$  – якобиан выхода голоморфной модели по параметрам модели (2.1.10), (3.3.3). В связи с этим данном разделе основной целью рассмотрения голоморфных моделей является аналитический вывод матрицы  $D_{\mathbf{z}} \mathbf{y}$ .

К неголоморфным моделям применимы методы адаптации на основе градиентного спуска и его модификации 2.1, 2.2, метод Ньютона и его модификации 3.1, 3.2, а также квазиньютоновские методы 4.3. Отметим также, что методы на основе аппроксимации шага метода Ньютона (метод сопряженных градиентов, покоординатный спуск и др.) реализуемы в рамках решения задачи адаптации неголоморфных моделей. Тем не менее, данная работа фокусируется на исследовании алгоритмов адаптации голоморфных моделей ввиду их малой вычислительной сложности. По этой причине методы аппроксимации метода Ньютона для неголоморфных моделей не рассматриваются.

Для реализации метода Ньютона для неголоморфных моделей (3.1.4) необходимо вычисление 6 матриц  $D_{\mathbf{z}} \mathbf{y}$ ,  $D_{\mathbf{z}^*} \mathbf{y}$ ,  $D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^T$ ,  $D_{\mathbf{z}} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^H$ ,  $D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^T$ ,  $D_{\mathbf{z}^*} \text{vec}(D_{\mathbf{z}^*} \mathbf{y})^H$ . Ввиду громоздкости аналитического вывода последних 4-х структур для неголоморфных моделей, в данном разделе рассматривается аналитический вывод матриц  $D_{\mathbf{z}} \mathbf{y}$ ,  $D_{\mathbf{z}^*} \mathbf{y}$ , вычисленных которых является достаточным для построения градиентных и квазиньютоновских методов.

### 5.1 Линейная модель искажений сигнала в системах связи

Одним из видов линейных по параметрам моделей в адаптивной обработке сигналов является КИХ-фильтр, фильтр с конечной импульской характеристикой. КИХ-

фильтр описывается набором комплексных коэффициентов  $\{w_k \in \mathbb{C}\}_{k=0}^{M-1}$ , которые можно представить в векторной форме  $\mathbf{w} = \{w_k\}_{k=0}^{M-1}$ , здесь  $M = 2D + 1$  – порядок фильтра. Выход КИХ-фильтра описывается линейной свёрткой [25]:

$$y_n = \sum_{k=-D}^D w_k x_{n-k}. \quad (5.1.1)$$

В векторном виде выход КИХ-фильтра будет иметь следующий вид:

$$y_n = \mathbf{u}_n^T \mathbf{w} = \begin{pmatrix} x_{n+D} & x_{n+D-1} & \cdots & x_{n-D} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_{M-1} \end{pmatrix}, \quad (5.1.2)$$

где  $\mathbf{u}_n$  – вектор состояния КИХ-фильтра  $M$ -ого порядка в момент времени  $n$ :

$$\mathbf{u}_n^T = \begin{pmatrix} x_{n+D} & x_{n+D-1} & \cdots & x_{n-D} \end{pmatrix}. \quad (5.1.3)$$

Рассмотрим последовательность из  $N$  комплексных отсчетов сигнала на входе фильтра:

$$\mathbf{x} = \{x_n \in \mathbb{C}\}_{n=0}^{N-1}. \quad (5.1.4)$$

Введём матрицу состояния [6] фильтра  $\mathbf{U} \in \mathbb{C}^{N \times M}$ :

$$\mathbf{U} = \begin{pmatrix} \mathbf{u}_0^T \\ \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_{N-2}^T \\ \mathbf{u}_{N-1}^T \end{pmatrix} = \begin{pmatrix} x_D & \cdots & x_1 & x_0 & 0 & \cdots & 0 \\ x_{D+1} & \cdots & x_2 & x_1 & x_0 & \cdots & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & \cdots & x_{N-1} & x_{N-2} & x_{N-3} & \cdots & x_{N-D-2} \\ 0 & \cdots & 0 & x_{N-1} & x_{N-2} & \cdots & x_{N-D-1} \end{pmatrix}. \quad (5.1.5)$$

Тогда отсчеты на выходе фильтра можно получить в виде векторно-матричного произведения:

$$\mathbf{y} = \mathbf{U} \mathbf{w}. \quad (5.1.6)$$

Матрица состояния фильтра в уравнении (5.1.6) учитывает задержку фильтра  $D_{FIR}$  равную половине длины фильтра. Такой способ выражения выходных отсчетов фильтра предлагается для того чтобы сохранить неизменными размерности векторов  $\mathbf{y} \in \mathbb{C}^{N \times 1}$  и  $\mathbf{x} \in \mathbb{C}^{N \times 1}$ , что в дальнейшем будет необходимо для реализации численных алгоритмов.

Ввиду того, что изначально при симуляции алгоритмов адаптации неизвестна задержка сигнала  $\mathbf{x}$  передатчика от сигнала помехи  $\mathbf{d}$ , то не имеет значение, какой именно задавать задержку  $D_{FIR}$ . Для выравнивания отсчётов передатчика относительно отсчётов приёмника по времени, предлагается задать  $D_{FIR}$  равной половине

длины фильтра, поскольку такая структура позволит описать как минимальнофазовые, так и линейно-фазовые каналы распространения помехи [26].

Якобиан выхода модели по параметрам вычисляется из выражения:

$$d\mathbf{y} = D_{\mathbf{w}} \mathbf{y} d\mathbf{w} = \mathbf{U} d\mathbf{w}. \quad (5.1.7)$$

Таким образом, для адаптивного КИХ-фильтра якобиан имеет вид матрицы состояния:

$$D_{\mathbf{w}} \mathbf{y} = \mathbf{U}. \quad (5.1.8)$$

Отметим, что метод LS 3.4 находит оптимальные параметры КИХ-фильтра за 1 шаг. При этом матрица Гессе представляет собой оценку корреляционной матрицы КИХ-фильтра:

$$H_{\mathbf{z}^*, \mathbf{z}} J = \check{\mathbf{R}}_{xx} \equiv \check{\mathbb{E}} \left( \mathbf{u}_n^* \mathbf{u}_n^T \right) = \mathbf{U}^H \mathbf{U}, \quad (5.1.9)$$

а градиент является оценкой кросс-корреляционного вектора сигнала и ошибки адаптации:

$$(D_{\mathbf{z}^*} J)^T = \check{\mathbf{r}}_{xe} \equiv \check{\mathbb{E}} \left( \mathbf{u}_n^* e_n \right) = \mathbf{U}^H \mathbf{e} \quad (5.1.10)$$

Уравнение (5.1.9), (5.1.10) отражают итеративный способ накопления корреляционной матрицы и кросс-корреляционного вектора:

$$\check{\mathbf{R}}_{xx,n} = \check{\mathbf{R}}_{xx,n-1} + \mathbf{u}_{n-1}^* \mathbf{u}_{n-1}^T, \quad (5.1.11)$$

$$\check{\mathbf{r}}_{xe,n} = \check{\mathbf{r}}_{xe,n-1} + \mathbf{u}_{n-1}^* e_{n-1} \quad (5.1.12)$$

В данном случае оценка тем ближе к истинному значению матожидания, чем больше отсчетов сигнала поступило на вход системы. Такой подход связан с реализацией стохастических методов и успешно применяется в случае потоковой обработки сигнала.

Отметим, также, что корреляционная матрица является эрмитовой, поскольку из (5.1.9) видно, что:

$$\mathbf{R}_{xx} = \mathbf{R}_{xx}^H \succcurlyeq 0 \quad (5.1.13)$$

Корреляционная матрица [6] является центральным объектом при построении алгоритмов адаптации линейных фильтров.

## 5.2 Полиномиальная нелинейных искажений в системах связи

Существует множество моделей, аппроксимирующих нелинейность усилителей мощности [27], таких как модель Винера, Гаммерштейна, Винера-Гаммерштейна, а также полиномиальные модели с памятью и без памяти [28]. Перечисленные модели являются следствием упрощения ряда Вольтерра.

Основной моделью паразитных нелинейных помех в приёмном тракте мобильного терминала является модель Гаммерштейна, описанная в разделе 5.5. Поскольку мо-

дель содержит КИХ-фильтр, который помимо задержек распространения помехи по различным путям от приёмника к передатчику учитывает инерционность усилителя мощности, то модель усилителя мощности можно выбирать безынерционной. Среди перечисленных моделей нелинейности усилителя безынерционной является полиномиальная модель без памяти.

В данном разделе рассматриваются две основные модели нелинейности: полиномиальная модель без памяти, а также метод улучшения численной устойчивости при адаптации этой модели, и модель кусочно-линейной аппроксимации.

Перед тем, как рассмотреть полиномиальную модель нелинейности усилителя мощности, рассмотрим общий способ описания нелинейной амплитудной характеристики.

На вход нелинейной модели поступает последовательность из  $N$  отсчетов модуля входного сигнала  $\{|x_n|\}|_{n=0}^{N-1}$ . Общая модель нелинейности описывается выражением [28]:

$$s_n = f(x_n) = g(|x_n|)x_n = \left[ \sum_{p=0}^{P-1} h_p \varphi_p(|x_n|) \right] x_n, \quad (5.2.1)$$

где  $\{\varphi_p(|x|)\}|_{p=0}^{P-1}$  – набор базисных функций модели нелинейности,  $P$  – порядок нелинейности. Выражение (5.2.1) учитывает тот факт, что нелинейное искажение сигнала на выходе усилителя рассматривается на несущей частоте сигнала передатчика, поэтому множитель  $x_n$  линейно входит в выражение для выхода нелинейности.

Рассмотрим полиномиальную модель без памяти, описывающую нелинейность усилителя мощности. Базисные функции полиномиальной модели без памяти имеют вид [28]:

$$\varphi_p(|x_n|) = |x_n|^p, \quad (5.2.2)$$

тогда модель нелинейности будет иметь вид:

$$g(|x_n|) = \sum_{p=0}^P h_p |x_n|^p. \quad (5.2.3)$$

В случае использования данной модели для описания нелинейной характеристики АМ-АМ [28] усилителя мощности на выходе нелинейной модели будут отсчеты:

$$s_n = g(|x_n|)x_n = \left[ \sum_{p=0}^P h_p |x_n|^p \right] x_n, \quad (5.2.4)$$

в векторной форме выходные отсчеты нелинейности будут иметь вид:

$$\begin{aligned} s_n &= \mathbf{h}_n^T \mathbf{v}_n = \begin{pmatrix} h_0 & h_1 & \cdots & h_{P-1} \end{pmatrix} \begin{pmatrix} x_n \varphi_0(|x_n|) & x_n \varphi_1(|x_n|) & \cdots & x_n \varphi_P(|x_n|) \end{pmatrix}^T = \\ &= \begin{pmatrix} h_0 & h_1 & \cdots & h_P \end{pmatrix} \begin{pmatrix} x_n & x_n |x_n| & \cdots & x_n |x_n|^{P-1} \end{pmatrix}^T, \end{aligned} \quad (5.2.5)$$

где  $\mathbf{v}$  – вектор состояния модели нелинейности, который вводится по аналогии с вектором состояния фильтра (5.1.3).

Матрицу состояния полиномиальной модели для случая блочной обработки по ограниченной выборке сигнала можно ввести по аналогии с матрицей состояния КИХ-фильтра через вектора состояния (5.1.5):

$$\mathbf{V} = \begin{pmatrix} \mathbf{v}_0^T \\ \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_{N-1}^T \end{pmatrix} = \begin{pmatrix} x_0 & x_0|x_0| & \cdots & x_0|x_0|^{P-1} \\ x_1 & x_1|x_1| & \cdots & x_1|x_1|^{P-1} \\ \vdots & & \ddots & \vdots \\ x_{N-1} & x_{N-1}|x_{N-1}| & \cdots & x_{N-1}|x_{N-1}|^{P-1} \end{pmatrix}. \quad (5.2.6)$$

Таким образом, выход нелинейной модели в матричном виде описывается выражением:

$$\mathbf{s} = \mathbf{V}\mathbf{h}. \quad (5.2.7)$$

В связи с этим, якобиан выхода модели по адаптивным параметрам может быть вычислен из выражения:

$$d\mathbf{s} = D_{\mathbf{h}}\mathbf{s}d\mathbf{h} = \mathbf{V}d\mathbf{h}. \quad (5.2.8)$$

Таким образом, для адаптивного КИХ-фильтра якобиан имеет вид матрицы состояния:

$$D_{\mathbf{h}}\mathbf{s} = \mathbf{V}. \quad (5.2.9)$$

Отметим, что матрица Гессе полиномиальной модели  $H_{\mathbf{h}^*, \mathbf{h}}J = \mathbf{V}^H \mathbf{V}$  является плохо обусловленной для высоких значений степеней полинома  $P \gg 1$ .

Для улучшения обусловленности гессиана модели используются ортогональные полиномы. Использование ортогональных полиномов позволяет повысить порядок полинома вплоть до  $P = 100$  с сохранением численной устойчивости алгоритмов адаптации. В случае полиномиальной модели без памяти базисные функции будут иметь вид [28]:

$$\varphi_p(|x_n|) = P_p(|x_n|^p). \quad (5.2.10)$$

Полиномы Лежандра [29]:

$$P_{p+1}(x) = \frac{2p+1}{p+1}xP_p(x) - \frac{p}{p+1}P_{p-1}(x), \quad (5.2.11)$$

$$P_0(x) = 1, P_1(x) = x,$$

являются ортогональными на отрезке  $[-1; 1]$ .

Полиномы Чебышёва 1-ого рода [29]:

$$T_{p+1}(x) = 2xT_p(x) - T_{p-1}(x), \quad (5.2.12)$$

$$T_0(x) = 1, T_1(x) = x,$$

являются ортогональными на отрезке  $[-1; 1]$ . Кроме того, полином Чебышёва 1-ого рода степени  $n$  меньше всего отклоняется от нуля на отрезке  $[-1; 1]$  среди полиномов степени  $n$ .

Полиномы Чебышёва 2-ого рода [29]:

$$\begin{aligned} U_{p+1}(x) &= 2xU_p(x) - U_{p-1}(x), \\ U_0(x) &= 1, U_1(x) = 2x, \end{aligned} \tag{5.2.13}$$

являются ортогональными на отрезке  $[-1; 1]$ . Кроме того, интеграл модуля полинома Чебышёва 2-ого рода степени  $n$  меньше всего отклоняется от нуля на отрезке  $[-1; 1]$  среди полиномов степени  $n$ .

Полиномы Эрмита [29]:

$$\begin{aligned} H_{p+1}(x) &= 2xH_p(x) - 2nH_{p-1}(x), \\ H_0(x) &= 1, H_1(x) = 2x, \end{aligned} \tag{5.2.14}$$

являются ортогональными на всей числовой оси.

### 5.3 Модель нелинейных искажений в системах связи на основе сплайновых полиномов

Полином на основе сплайнов первого порядка реализует кусочно-линейную интерполяцию при построении нелинейной модели. Данный вид базисных функций эффективен с точки зрения числа операций, осуществляемых в единицу времени при вычислении значений полинома. В связи с этим, метод кусочно-линейной аппроксимации активно используется в цифровой технике для аппроксимации выходных характеристик нелинейных компонент, таких как аналоговые усилители мощности, дуплексеры и другие [30].

Кусочно-линейная функция может быть задана, как на рис. 6. Здесь расстояние между отсчетами коэффициентов  $h_k$  равно 1.

Обозначим  $\Delta_n$  – расстояние между модулем входного отсчета  $|x_n|$  и ближайшим целым  $p < |x_n|$  по оси абсцисс,  $h_p$  – отсчет с координатой  $p$  по оси абсцисс. Тогда

$$\frac{\Delta_n}{1} = \frac{g(|x_n|) - h_p}{h_{p+1} - h_p}, \tag{5.3.1}$$

$$g(|x_n|) = \Delta_n h_{p+1} + (1 - \Delta_n)h_p. \tag{5.3.2}$$

Подставим выражение (5.3.2) в выражение для выходных отсчетов нелинейности

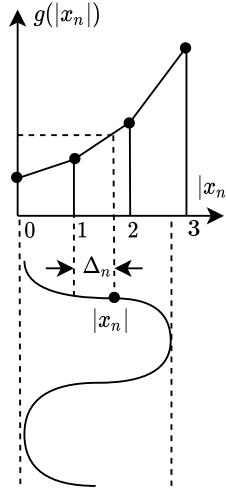


Рис. 6. Кусочно-линейная функция  $g(|x_n|)$  сплайнового полинома

(5.2.1) и перепишем в виде скалярного произведения векторов:

$$s_n = \mathbf{v}_n^T \mathbf{h} = \begin{pmatrix} 0 & \dots & 0 & (1 - \Delta_n)x_n & \Delta_n x_n & 0 & \dots & 0 \end{pmatrix} \cdot \begin{pmatrix} h_0 & \dots & h_{p-1} & h_p & h_{p+1} & h_{p+2} & \dots & h_{P-1} \end{pmatrix}^T, \quad (5.3.3)$$

где  $\mathbf{v}_n^T$  – строка матрицы состояния модуля, которая представляет собой вектор состояния модуля кусочно-линейной аппроксимации:

$$\mathbf{v}_n^T = \begin{pmatrix} 0 & \dots & 0 & (1 - \Delta_n)x_n & \Delta_n x_n & 0 & \dots & 0 \end{pmatrix}. \quad (5.3.4)$$

Полная матрица состояния может быть записана следующим образом:

$$\mathbf{V} = \begin{pmatrix} \mathbf{v}_0^T \\ \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_{N-1}^T \end{pmatrix} = \begin{pmatrix} 0 & \dots & 0 & (1 - \Delta_0)x_n & \Delta_0 x_n & 0 & \dots & 0 \\ 0 & \dots & (1 - \Delta_1)x_n & \Delta_1 x_n & 0 & 0 & \dots & 0 \\ \vdots & & & \ddots & & & & \vdots \\ 0 & \dots & 0 & (1 - \Delta_{N-1})x_n & \Delta_{N-1} x_n & 0 & \dots & 0 \end{pmatrix} \quad (5.3.5)$$

Вектор выходных отсчетов  $\mathbf{z}$  может быть записан:

$$\mathbf{s} = \mathbf{V} \mathbf{h}. \quad (5.3.6)$$

Отметим также, что кусочно-линейная функция, аппроксимирующая нелинейную амплитудную характеристику усилителя мощности может быть представлена через базисные функции (5.2.1) следующего вида [30]:

$$\varphi_p(|x_n|) = \begin{cases} |x_n| - (p - 1), & \text{если } (p - 1 \leq |x_n| < p) \vee (0 < p \leq P - 1), \\ -|x_n| + (p + 1), & \text{если } (p \leq |x_n| < p + 1) \vee (0 \leq p < P - 1). \end{cases} \quad (5.3.7)$$

## 5.4 Модель Вольтерра нелинейных искажений в системах связи

Модель Вольтерра представляет собой наиболее общую модель нелинейной адаптивной обработки сигналов и является функциональным обобщением линейной свертки [volterra1959theory, schetzen1980volterra, matthews1991adaptive]. Для комплекснозначных сигналов выходной отсчёт  $s_n$  выражается через входной сигнал  $x_n$  в виде обобщённого функционального ряда:

$$s_n = \sum_{p=0}^P \sum_{q=0}^p \sum_{d_1=0}^{D-1} \cdots \sum_{d_p=0}^{D-1} h_{p,q,d_1,\dots,d_p} \prod_{j=1}^q x_{n-d_j} \prod_{j=q+1}^p x_{n-d_j}^*, \quad (5.4.1)$$

где:  $h_{p,q,d_1,\dots,d_p}$  – ядро Вольтерры  $(p, q)$ -го порядка,  $P$  – порядок нелинейности, то есть максимальная степень нелинейных членов,  $D$  – глубина памяти системы, то есть максимальная задержка,  $x_{n-d_j}$  – задержанные входные отсчёты, индекс  $q$  определяет количество не сопряжённых членов,  $(p - q)$  – количество сопряжённых членов.

Структура ряда включает:

- **Нулевой порядок** ( $p = 0$ ):  $h_{0,0}$  – постоянная составляющая
- **Первый порядок** ( $p = 1$ ):
  - $q = 1$ :  $\sum_{d_1=0}^{D-1} h_{1,1}(d_1) x_{n-d_1}$  – линейная фильтрация
  - $q = 0$ :  $\sum_{d_1=0}^{D-1} h_{1,0}(d_1) x_{n-d_1}^*$  – линейная фильтрация сопряжённого сигнала
- **Второй порядок** ( $p = 2$ ): включает члены  $x^2$ ,  $xx^*$  и  $(x^*)^2$
- **Высшие порядки** ( $p \geq 3$ ): описывают более сложные нелинейные эффекты

По аналогии с (5.2.1) базисная функция модели Вольтерра представляет собой следующее выражение:

$$\varphi_{p,q,d_1,\dots,d_p}(x_n) = \prod_{j=1}^q x_{n-d_j} \prod_{j=q+1}^p x_{n-d_j}^*. \quad (5.4.2)$$

Такая структура позволяет полностью описать нелинейные системы с памятью включая эффекты взаимного влияния квадратурных составляющих.

## 5.5 Модель Гаммерштейна нелинейных искажений в системах связи

Модель Гаммерштейна является частным случаем модели Вольтерра (5.4.1) и представляет собой двухслойную модель. Первый слой модели описывает нелинейные искажения сигнала, обусловленные нелинейной характеристикой передатчика,

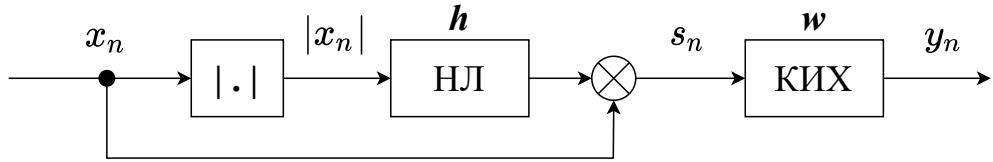


Рис. 7. Модель Гаммерштейна

второй слой описывает линейные искажения, обусловленные многолучевым распространением сигнала и передатчика в приёмнике. В связи с этим, модель является эффективной в задачах компенсации нелинейных помех в приёмнике приемо-передающих устройств [17].

Модель состоит из последовательно соединенных нелинейного блока и адаптивного КИХ-фильтра. Нелинейный блок может быть представлен любой моделью, нелинейной по входному сигналу, включая полиномиальную модель без памяти (5.2.5), сплайновый полином (5.3.3) и другие. Схема модели Гаммерштейна представлена на рис. 7.

Выход модели Гаммерштейна представлен выражением:

$$y_n = \sum_{m=-D}^D w_m \sum_{p=0}^{P-1} h_k x_{n-m} \varphi_p(|x_{n-m}|), \quad (5.5.1)$$

где  $\mathbf{h} \in \mathbb{C}^{P \times 1}$  – адаптивные параметры блока нелинейности,  $P$  – порядок полинома, параметры адаптивного КИХ-фильтра **[haykin]**  $\mathbf{w} \in \mathbb{C}^{M \times 1}$ ,  $M = 2D + 1$  – число адаптивных коэффициентов фильтра,  $\{\varphi_p(\cdot)\}_{p=0}^{P-1}$  – базисные функции нелинейности.

Представим общий вектор параметров модели Гаммерштейна следующим образом:

$$\mathbf{z} = \begin{pmatrix} \mathbf{h} \\ \mathbf{w} \end{pmatrix}, \mathbf{z}^* = \begin{pmatrix} \mathbf{h}^* \\ \mathbf{w}^* \end{pmatrix} \in \mathbb{C}^{(P+L) \times 1}. \quad (5.5.2)$$

Якобиан выходного вектора модели Гаммерштейна по параметрам представлен выражением:

$$D_{\mathbf{z}} \mathbf{y} = \begin{pmatrix} D_{\mathbf{h}} \mathbf{y} & D_{\mathbf{w}} \mathbf{y} \end{pmatrix} \in \mathbb{C}^{N \times (P+L)}. \quad (5.5.3)$$

Таким образом, для подсчета матрицы Гёссе и градиента целевой функции необходимо вычислить производную выходного вектора модели по параметрам слоя нелинейности и КИХ-фильтра.

Согласно теории адаптивной фильтрации **[haykin]** вектор выходных отсчётов КИХ-фильтра может быть выражен через матрицу состояния  $\mathbf{U}$  (5.1.6), где матрица  $\mathbf{U}$  заполнена задержанными отсчетами  $s_n = g(|x_n|)x_n$  выхода нелинейного слоя (5.2.1). Тогда производная выхода модели по параметрам фильтра выражается через матрицу состояния фильтра (5.1.8):

$$D_{\mathbf{w}} \mathbf{y} = \mathbf{U} = \mathbf{U}(s). \quad (5.5.4)$$

Согласно (5.2.7), (5.3.6) выход нелинейной модели выражается через матрично-векторное произведение матрицы состояния  $\mathbf{V}$  и параметров нелинейности  $\mathbf{h}$ . Тогда выход модели Гаммерштейна можно представить как:

$$\mathbf{y} = \text{conv}_{\mathbf{w}}(\mathbf{s}) = \text{conv}_{\mathbf{w}}(\mathbf{V}\mathbf{h}), \quad (5.5.5)$$

где  $\text{conv}_{\mathbf{w}}(\cdot)$  — одномерный линейный оператор свёртки. Запишем матрицу состояния через её столбцы:

$$\mathbf{V} = \begin{pmatrix} \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{P-1} \end{pmatrix}, \quad (5.5.6)$$

тогда входной вектор фильтра (5.2.7), (5.3.6)  $\mathbf{s}$  может быть выражен как:

$$\mathbf{s} = h_0 \mathbf{v}_0 + h_1 \mathbf{v}_1 + \cdots + h_{P-1} \mathbf{v}_{P-1}. \quad (5.5.7)$$

Используя свойство линейности свёртки из уравнений (5.5.7) и (5.5.5) получаем выход модели Гаммерштейна через матрично-векторное умножение:

$$\begin{aligned} \mathbf{y} = \text{conv}_{\mathbf{w}}(h_0 \mathbf{v}_0 + \cdots + h_{P-1} \mathbf{v}_{P-1}) &= h_0 \text{conv}_{\mathbf{w}}(\mathbf{v}_0) + \cdots + h_{P-1} \text{conv}_{\mathbf{w}}(\mathbf{v}_{P-1}) = \\ &= \left( \text{conv}_{\mathbf{w}}(\mathbf{v}_0) \ \cdots \ \text{conv}_{\mathbf{w}}(\mathbf{v}_{P-1}) \right) \mathbf{h} \equiv \mathbf{V}_f \mathbf{h}. \end{aligned} \quad (5.5.8)$$

Таким образом, выходной вектор модели Гаммерштейна выводится через произведение матрицы состояния нелинейности, фильтрованной по столбцам, и вектора параметров нелинейности. Отметим, что матрица  $\mathbf{V}_f$  не зависит от параметров полинома  $\mathbf{h}$ .

В результате, используя уравнение (5.5.8), получаем производную выхода модели по параметрам нелинейности:

$$D_{\mathbf{h}} \mathbf{y} = D_{\mathbf{h}}(\mathbf{V}_f \mathbf{h}) = \mathbf{V}_f. \quad (5.5.9)$$

Подставляя (5.5.4), (5.5.9) в (5.5.3) получаем финальное выражение производной выхода модели Гаммерштейна по общем вектору параметров (5.5.2):

$$D_{\mathbf{z}} \mathbf{y} = \begin{pmatrix} \mathbf{U} & \mathbf{V}_f \end{pmatrix} \in \mathbb{C}^{N \times (P+M)}. \quad (5.5.10)$$

## 5.6 Модель Винера нелинейных искажений в системах связи

Модель Винера представляет собой двухслойную модель, описывающую нелинейность с памятью. Первым слоем является КИХ-фильтр описывающий инерционные свойства передающего тракта, второй слой представляет собой нелинейность без памяти, описывающую нелинейный характер передающего тракта. Схема модели Гаммерштейна представлена на рис. 8.

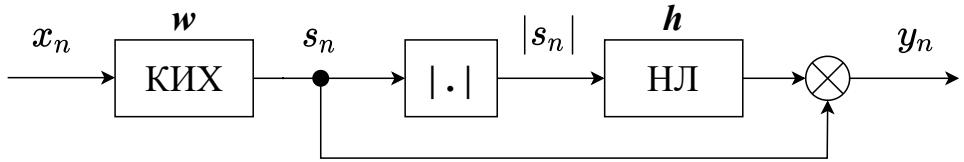


Рис. 8. Модель Винера

Выход модели Винера представлен выражением:

$$\begin{aligned} y_n &= \sum_{p=0}^{P-1} h_p s_n \varphi_p(|s_n|), \\ s_n &= \sum_{m=-D}^D w_m x_{n-m} \end{aligned} \quad (5.6.1)$$

где  $\mathbf{h} \in \mathbb{C}^{P \times 1}$  – адаптивные параметры блока нелинейности,  $P$  – порядок полинома, параметры адаптивного КИХ-фильтра **[haykin]**  $\mathbf{w} \in \mathbb{C}^{M \times 1}$ ,  $M = 2D + 1$  – число адаптивных коэффициентов фильтра,  $\{\varphi_p(\cdot)\}|_{p=0}^{P-1}$  – базисные функции нелинейности.

Представим параметры модели в виде вектора содержащего параметры фильтра и нелинейности (5.5.3) по аналогии с моделью Гаммерштейна.

Заметим, выход модели Винера не является голоморфным относительно параметров КИХ-фильтра  $\{w_m\}|_{-D}^D$  ввиду наличия операции взятия модуля перед блоком нелинейности:

$$\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{h}, \mathbf{w}, \mathbf{w}^*), \quad (5.6.2)$$

$$d\mathbf{y} = D_{\mathbf{h}} \mathbf{y} d\mathbf{h} + D_{\mathbf{w}} \mathbf{y} d\mathbf{w} + D_{\mathbf{w}^*} \mathbf{y} d\mathbf{w}^*. \quad (5.6.3)$$

В связи с этим (5.6.3) для реализации алгоритмов на основе градиентного спуска (2.1.8) необходимо вычислить матрицы производных по прямым и сопряженным параметрам модели:

$$\begin{aligned} D_{\mathbf{z}} \mathbf{y} &= \begin{pmatrix} D_{\mathbf{h}} \mathbf{y} & D_{\mathbf{w}} \mathbf{y} \end{pmatrix} \in \mathbb{C}^{N \times (P+M)}, \\ D_{\mathbf{z}^*} \mathbf{y} &= \begin{pmatrix} \mathbf{0} & D_{\mathbf{w}^*} \mathbf{y} \end{pmatrix} \in \mathbb{C}^{N \times (P+M)}, \end{aligned} \quad (5.6.4)$$

Поскольку нелинейность является выходным слоем модели Винера, то производная выхода модели по параметрам нелинейности вычисляется как:

$$D_{\mathbf{h}} \mathbf{y} = \mathbf{V} = \mathbf{V}(\mathbf{s}), \quad (5.6.5)$$

где  $\mathbf{V}$  – матрица состояния нелинейности, заполненная отсчетами вида  $s_n \varphi_p(|s_n|)$ ,  $s_n$  – выходные отсчеты фильтра,  $\varphi_p(\cdot)$  –  $p$ -я базисная функция нелинейности.

Распишем дифференциал выхода (5.6.3) модели используя структуру модели Ви-

нера (5.6.1):

$$\begin{aligned} d\mathbf{y} &= \begin{pmatrix} d[s_n \varphi_0(|s_n|)] & \cdots & d[s_n \varphi_{P-1}(|s_n|)] \\ \vdots & \ddots & \vdots \\ d[s_{n-N+1} \varphi_0(|s_{n-N+1}|)] & \cdots & d[s_{n-N+1} \varphi_{P-1}(|s_{n-N+1}|)] \end{pmatrix} \mathbf{h} + \mathbf{V} d\mathbf{h} \\ &= \mathbf{D}\mathbf{h} + \mathbf{V} d\mathbf{h}. \end{aligned} \quad (5.6.6)$$

Из равенства выражений (5.6.3) и (5.6.6) следует, что первое слагаемое в (5.6.6) содержит компоненты соответствующие производным по параметрам фильтра. Далее будем рассматривать именно это слагаемое для поиска  $D_{\mathbf{w}}\mathbf{y}$ ,  $D_{\mathbf{w}^*}\mathbf{y}$ .

Рассмотрим один элемент матрицы дифференциалов  $\mathbf{D}$  (5.6.6):

$$\begin{aligned} d[s_n \varphi_j(|s_n|)] &= ds_n \varphi_j(|s_n|) + s_n d\varphi_j(|s_n|) = \\ &= \varphi_j(|s_n|) \mathbf{u}_n^T d\mathbf{w} + s_n d\varphi_j(|s_n|), \end{aligned} \quad (5.6.7)$$

где  $s_n = \mathbf{u}_n^T \mathbf{w}$  – выходной отсчет фильтра, представленный в виде скалярного произведения вектора состояния (одной строки матрицы состояния  $\mathbf{U}$ ) и параметров фильтра по аналогии с (5.1.6).

Из (5.6.7) следует, что матрица дифференциалов  $\mathbf{D}$  разбивается на сумму двух матриц:

$$\mathbf{D} = \mathbf{D}_1 + \mathbf{D}_2, \quad (5.6.8)$$

где  $\mathbf{D}_1$  состоит из элементов  $\varphi_j(|s_n|) \mathbf{u}_n^T d\mathbf{w}$ ,  $\mathbf{D}_2$  состоит из элементов  $s_n d\varphi_j(|s_n|)$ .

Из (5.6.6) и (5.6.8) следует, что  $\mathbf{D}\mathbf{h} = \mathbf{D}_1\mathbf{h} + \mathbf{D}_2\mathbf{h}$ . Рассмотрим первое слагаемое:

$$\begin{aligned} \mathbf{D}_1\mathbf{h} &= \begin{pmatrix} \varphi_0(|s_n|) \mathbf{u}_n^T d\mathbf{w} & \cdots & \varphi_{P-1}(|s_n|) \mathbf{u}_n^T d\mathbf{w} \\ \vdots & \ddots & \vdots \\ \varphi_0(|s_{n-N+1}|) \mathbf{u}_{n-N+1}^T d\mathbf{w} & \cdots & \varphi_{P-1}(|s_{n-N+1}|) \mathbf{u}_{n-N+1}^T d\mathbf{w} \end{pmatrix} \mathbf{h} = \\ &= \begin{pmatrix} \varphi_n^T \mathbf{h} \mathbf{u}_n^T \\ \vdots \\ \varphi_{n-N+1}^T \mathbf{h} \mathbf{u}_{n-N+1}^T \end{pmatrix} d\mathbf{w} = \begin{pmatrix} (y_n/s_n) \mathbf{u}_n^T \\ \vdots \\ (y_{n-N+1}/s_{n-N+1}) \mathbf{u}_{n-N+1}^T \end{pmatrix} d\mathbf{w} = \\ &= \text{diag}(\mathbf{y} \odot \frac{1}{s}) \mathbf{U} d\mathbf{w}. \end{aligned} \quad (5.6.9)$$

В выкладках (5.6.9) использовались обозначения

$$\left( \varphi_0(|s_n|) \ \cdots \ \varphi_{P-1}(|s_n|) \right) \mathbf{h} = \boldsymbol{\varphi}_n \mathbf{h} = (y_n/s_n), \quad (5.6.10)$$

$\mathbf{u}_n \in \mathbb{C}^{M \times 1}$  – вектор состояния фильтра,  $\mathbf{u}_n^T \in \mathbb{C}^{1 \times M}$  – строка матрицы состояния фильтра  $\mathbf{U}$ ,  $\odot$  – поэлементное произведение векторов,  $\text{diag}(\mathbf{x})$  – преобразование вектора  $\mathbf{x}$  в диагональную матрицу.

Рассмотрим теперь второе слагаемое в выражении (5.6.7):

$$s_n d\varphi_j(|s_n|) = s_n D_{|s_n|} \varphi_j(|s_n|) d|s_n| = s_n r_{j,n} d|s_n|, \quad (5.6.11)$$

где  $r_{j,n}$  – производная базисной функции нелинейности по ее входу. Вычислим дифференциал модуля выходного отсчета фильтра (5.6.11):

$$d|s_n| = d\sqrt{s_n^* s_n} = \frac{s_n ds_n^* + s_n^* ds_n}{2|s_n|} = \frac{s_n \mathbf{u}_n^H d\mathbf{w}^* + s_n^* \mathbf{u}_n^T d\mathbf{w}}{2|s_n|} \quad (5.6.12)$$

Подставим выражение (5.6.12) в (5.6.12):

$$s_n d\varphi_j(|s_n|) = \frac{s_n^2 r_{j,n} \mathbf{u}_n^H d\mathbf{w}^*}{2|s_n|} + \frac{|s_n| r_{j,n} \mathbf{u}_n^T d\mathbf{w}}{2} \quad (5.6.13)$$

Получим выражение для второго слагаемого суммы (5.6.8) используя выражение (5.6.13):

$$\mathbf{D}_2 \mathbf{h} = \text{diag}(\mathbf{s}^2 \odot \frac{1}{2|\mathbf{s}|} \odot \mathbf{R} \mathbf{h}) \mathbf{U}^* d\mathbf{w}^* + \frac{1}{2} \text{diag}(|\mathbf{s}| \odot \mathbf{R} \mathbf{h}) \mathbf{U} d\mathbf{w}, \quad (5.6.14)$$

где  $\mathbf{s}^2$ ,  $|\mathbf{s}|$ ,  $1/\mathbf{s}$  – поэлементное возведение в квадрат, взятие модуля и деление соответственно,  $\mathbf{R}$  – матрица производных базисных функций по их аргументу:

$$\mathbf{R} = \begin{pmatrix} D_{x_n} \varphi_0(x_n) & \cdots & D_{x_n} \varphi_{P-1}(x_n) \\ \vdots & \ddots & \vdots \\ D_{x_{n-N+1}} \varphi_0(x_{n-N+1}) & \cdots & D_{x_{n-N+1}} \varphi_{P-1}(x_{n-N+1}) \end{pmatrix}. \quad (5.6.15)$$

Объединяя выражения (5.6.9), (5.6.14) в (5.6.8) и учитывая (5.6.3) получаем выражения для производных выхода модели Винера по прямым и спороженным параметрам фильтра:

$$\begin{aligned} D_{\mathbf{w}} \mathbf{y} &= \text{diag}(\mathbf{y} \odot \frac{1}{\mathbf{s}} + \frac{1}{2} |\mathbf{s}| \odot \mathbf{R} \mathbf{h}) \mathbf{U} \\ D_{\mathbf{w}^*} \mathbf{y} &= \text{diag}(\mathbf{s}^2 \odot \frac{1}{2|\mathbf{s}|} \odot \mathbf{R} \mathbf{h}) \mathbf{U}^*. \end{aligned} \quad (5.6.16)$$

Тогда производные по всем прямым и сопряженным параметрам модели (5.6.4) могут быть представлены следующим образом:

$$\begin{aligned} D_{\mathbf{z}} \mathbf{y} &= \left( \mathbf{V} \quad \text{diag}(\mathbf{y} \odot \frac{1}{\mathbf{s}} + \frac{1}{2} |\mathbf{s}| \odot \mathbf{R} \mathbf{h}) \mathbf{U} \right) \in \mathbb{C}^{N \times (P+M)}, \\ D_{\mathbf{z}^*} \mathbf{y} &= \left( \mathbf{0} \quad \text{diag}(\mathbf{s}^2 \odot \frac{1}{2|\mathbf{s}|} \odot \mathbf{R} \mathbf{h}) \mathbf{U}^* \right) \in \mathbb{C}^{N \times (P+M)}. \end{aligned} \quad (5.6.17)$$

Отметим, что в выражениях (5.6.17) произведение диагональной и квадратной матрицы может быть представлено в виде численно-эффективной операции по-стобцовому

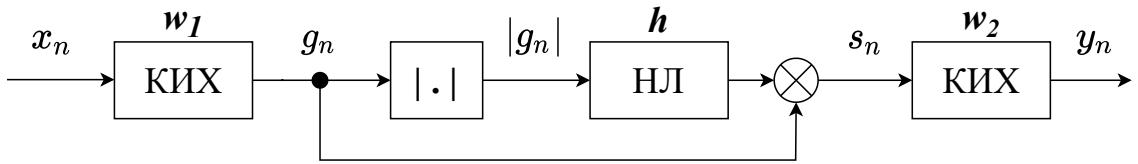


Рис. 9. Модель Винера-Гаммерштейна

произведения диагонали и квадратной матрицы:

$$\text{diag}(\mathbf{a})\mathbf{B} = \mathbf{a} \odot \mathbf{B}, \quad (5.6.18)$$

где  $\odot$  представляет собой поэлементное произведение вектора  $\mathbf{a}$  с каждым столбцом матрицы  $\mathbf{B}$ . Данная операция не требует хранения диагональной матрицы  $\text{diag}(\mathbf{a})$ , состоящей преимущественно из нулей.

## 5.7 Модель Винера-Гаммерштейна нелинейных искажений в системах связи

Модель Винера-Гаммерштейна объединяет в себе преимущества модели Винера 5.6 и модели Гаммерштейна 5.5. Модель является трех-слойной и включает в себя последовательно соединенные адаптивный КИХ-фильтр, описывающий инерционные свойства канала передатчика, безынерционную нелинейность, а также выходной адаптивный КИХ-фильтр, описывающий многолучевое распространение помехи в канале передатчик-приёмник. Модель Винера-Гаммерштейна является эффективной структурой с точки зрения описания как нелинейных искажений возникающих в передатчике приемо-передающего устройства, так и паразитных помех, возникающих в приемнике приемо-передающего устройства. Схема модели Гаммерштейна представлена на рис. 9.

Выходной отсчет модели Винера-Гаммерштейна представлен выражением:

$$y_n = \sum_{m=-D_2}^{D_2} w_{2,m} \sum_{p=0}^{P-1} h_p g_{n-m} \varphi_p(|g_{n-m}|),$$

$$g_n = \sum_{m=-D_1}^{D_1} w_{1,m} x_{n-m}, \quad (5.7.1)$$

где  $\mathbf{h} \in \mathbb{C}^{P \times 1}$  – адаптивные параметры блока нелинейности,  $P$  – порядок полинома,  $\mathbf{w}_1 \in \mathbb{C}^{M_1 \times 1}$  – параметры входного адаптивного КИХ-фильтра,  $M_1 = 2D_1 + 1$  – число адаптивных коэффициентов входного фильтра,  $\mathbf{w}_2 \in \mathbb{C}^{M_2 \times 1}$  – параметры выходного адаптивного КИХ-фильтра,  $M_2 = 2D_2 + 1$  – число адаптивных коэффициентов выходного фильтра,  $\{\varphi_p(\cdot)\}_{p=0}^{P-1}$  – базисные функции нелинейности.

Представим параметры модели в виде вектора содержащего параметры фильтра

и нелинейности по аналогии с моделью Винера и моделью Гаммерштейна:

$$\mathbf{z} = \begin{pmatrix} \mathbf{h} \\ \mathbf{w}_1 \\ \mathbf{w}_2 \end{pmatrix}, \mathbf{z}^* = \begin{pmatrix} \mathbf{h}^* \\ \mathbf{w}_1^* \\ \mathbf{w}_2^* \end{pmatrix} \in \mathbb{C}^{(P+M_1+M_2) \times 1}. \quad (5.7.2)$$

Заметим, выход модели Винера-Гаммерштейна не является голоморфным относительно параметров входного КИХ-фильтра  $\{w_{1,m}\}|_{-D_1}^{D_1}$  ввиду наличия операции взятия модуля перед блоком нелинейности:

$$\mathbf{y} = \mathbf{y}(\mathbf{x}, \mathbf{h}, \mathbf{w}_1, \mathbf{w}_1^*, \mathbf{w}_2), \quad (5.7.3)$$

$$d\mathbf{y} = D_{\mathbf{h}} \mathbf{y} d\mathbf{h} + D_{\mathbf{w}_1} \mathbf{y} d\mathbf{w}_1 + D_{\mathbf{w}_1^*} \mathbf{y} d\mathbf{w}_1^* + D_{\mathbf{w}_2} \mathbf{y} d\mathbf{w}_2. \quad (5.7.4)$$

В связи с этим (5.7.4) также как и для модели Винера для реализации алгоритмов на основе градиентного спуска (2.1.8) необходимо вычислить матрицы производных по прямым и сопряженным параметрам модели:

$$\begin{aligned} D_{\mathbf{z}} \mathbf{y} &= \begin{pmatrix} D_{\mathbf{h}} \mathbf{y} & D_{\mathbf{w}_1} \mathbf{y} & D_{\mathbf{w}_2} \mathbf{y} \end{pmatrix} \in \mathbb{C}^{N \times (P+M_1+M_2)}, \\ D_{\mathbf{z}^*} \mathbf{y} &= \begin{pmatrix} \mathbf{0} & D_{\mathbf{w}_1^*} \mathbf{y} & \mathbf{0} \end{pmatrix} \in \mathbb{C}^{N \times (P+M_1+M_2)}, \end{aligned} \quad (5.7.5)$$

Выход модели Винера-Гаммерштейна может быть представлен следующим образом:

$$\mathbf{y} = \text{conv}_{\mathbf{w}_2}(\mathbf{s}) = \text{conv}_{\mathbf{w}_2}(\mathbf{V}\mathbf{h}), \quad (5.7.6)$$

тогда дифференциал выхода модели выражается как:

$$d\mathbf{y} = \mathbf{U}_2 d\mathbf{w}_2 + \mathbf{V}_{f,\mathbf{w}_2} d\mathbf{h} + \text{conv}_{\mathbf{w}_2}(ds), \quad (5.7.7)$$

где  $\mathbf{V}_{f,\mathbf{w}_2}$  – матрица состояния нелинейности со столбцами фильтрованными выходным фильтром, получена из условия линейности свертки по аналогии с (5.5.8). Тогда производная выхода модели по параметрам выходного КИХ-фильтра и нелинейности вычисляется аналогично модели Гаммерштейна:

$$D_{\mathbf{w}_2} \mathbf{y} = \mathbf{U}_2 = \mathbf{U}_2(\mathbf{s}), \quad (5.7.8)$$

$$D_{\mathbf{h}} \mathbf{y} = \mathbf{V}_{f,\mathbf{w}_2}, \quad (5.7.9)$$

где  $\mathbf{U}_2$  – матрица состояния выходного фильтра, заполненная отсчетами  $s_n$ ,  $\mathbf{V}_{f,\mathbf{w}_2}$  – заполнена выходными отсчетами входного фильтра  $g_n$ .

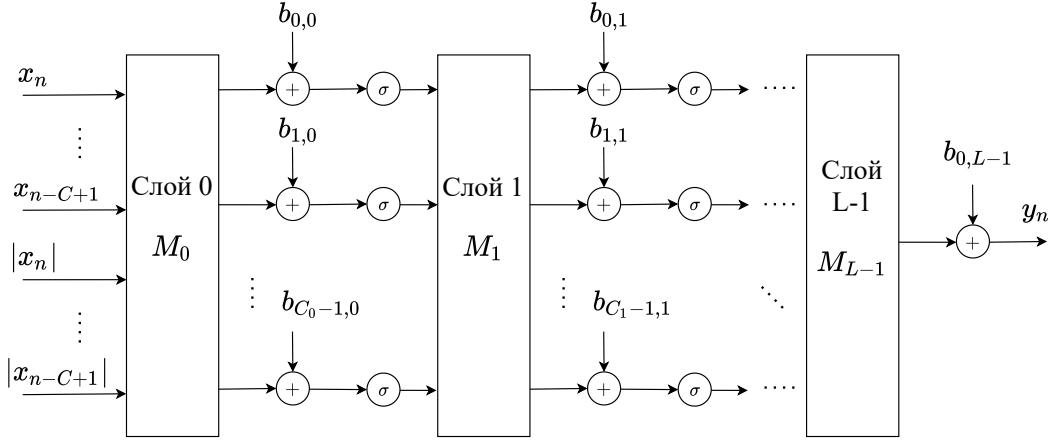


Рис. 10. Модель нейросетевой модели на основе полносвязного перцептрона

Распишем третье слагаемое суммы (5.7.7):

$$\begin{aligned} \text{conv}_{\mathbf{w}_2}(ds) &= \text{conv}_{\mathbf{w}_2}(D_{\mathbf{w}_1} s d\mathbf{w}_1) + \text{conv}_{\mathbf{w}_2}(D_{\mathbf{w}_1^*} s d\mathbf{w}_1^*) = \\ &= (D_{\mathbf{w}_1} s)_{f, \mathbf{w}_2} d\mathbf{w}_1 + (D_{\mathbf{w}_1^*} s)_{f, \mathbf{w}_2} d\mathbf{w}_1^* = \\ &= (D_{\mathbf{w}_1} \mathbf{y}) d\mathbf{w}_1 + (D_{\mathbf{w}_1^*} \mathbf{y}) d\mathbf{w}_1^*. \end{aligned} \quad (5.7.10)$$

Заметим, что \$D\_{\mathbf{w}\_1} \mathbf{s}\$, \$D\_{\mathbf{w}\_1^\*} \mathbf{s}\$ – производные выхода нелинейного слоя по параметрам внутреннего фильтра, уже найдены в разделе, посвященном модели Винера (5.6.17). Согласно (5.6.17), (5.7.10) матрицы Якоби выхода модели Винера-Гаммерштейна по параметрам внутреннего фильтра могут быть найдены следующим образом:

$$\begin{aligned} D_{\mathbf{w}_1} \mathbf{y} &= \left[ \text{diag}(\mathbf{s} \odot \frac{1}{g} + \frac{1}{2}|g| \odot \mathbf{R}\mathbf{h}) \mathbf{U}_1 \right]_{f, \mathbf{w}_2} \\ D_{\mathbf{w}_1^*} \mathbf{y} &= \left[ \text{diag}(g^2 \odot \frac{1}{2|g|} \odot \mathbf{R}\mathbf{h}) \mathbf{U}_1^* \right]_{f, \mathbf{w}_2}. \end{aligned} \quad (5.7.11)$$

Тогда производные по всем прямым и сопряженным параметрам модели (5.7.5) выражаются как:

$$\begin{aligned} D_{\mathbf{z}} \mathbf{y} &= \left( \mathbf{V}_{f, \mathbf{w}_2} \quad \left[ \text{diag}(\mathbf{s} \odot \frac{1}{g} + \frac{1}{2}|g| \odot \mathbf{R}\mathbf{h}) \mathbf{U}_1 \right]_{f, \mathbf{w}_2} \quad \mathbf{U}_2 \right) \in \mathbb{C}^{N \times (P+M_1+M_2)}, \\ D_{\mathbf{z}^*} \mathbf{y} &= \left( \mathbf{0} \quad \left[ \text{diag}(g^2 \odot \frac{1}{2|g|} \odot \mathbf{R}\mathbf{h}) \mathbf{U}_1^* \right]_{f, \mathbf{w}_2} \quad \mathbf{0} \right) \in \mathbb{C}^{N \times (P+M_1+M_2)}, \end{aligned} \quad (5.7.12)$$

## 5.8 Применение нейросетевых структур для аппроксимации нелинейных искажений приемо-передающего тракта

Схема нейросетевой модели на основе полносвязного перцептрона представлена на рис. 10. Пусть на входе модели матрица \$\mathbf{x} \in \mathbb{C}^{2C \times M}\$, где \$C\$ – число входных каналов \$x\_{n-d}\$ и \$|x\_{n-d}|\$, \$M\$ – число отсчетов на одно обновление параметров, то есть

длина блока:

$$\mathbf{x} = \begin{pmatrix} x_n & x_{n-1} & \cdots & x_{n-M+1} \\ x_{n-1} & x_{n-2} & \cdots & x_{n-M} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n-C+1} & x_{n-C} & \cdots & x_{n-C-M+2} \\ |x_n| & |x_{n-1}| & \cdots & |x_{n-M+1}| \\ |x_{n-1}| & |x_{n-2}| & \cdots & |x_{n-M}| \\ \vdots & \vdots & \ddots & \vdots \\ |x_{n-C+1}| & |x_{n-C}| & \cdots & |x_{n-C-M+2}| \end{pmatrix} \in \mathbb{C}^{2C \times M}. \quad (5.8.1)$$

Обозначим число выходных каналов каждого слоя  $C_j$ , где  $j = \overline{0, L-1}$ ,  $L$  – число слоев модели. Таким образом, полносвязная модель определяется следующей системой:

$$\mathbf{y}_j = \begin{cases} \mathbf{M}_j \mathbf{y}_{j-1} + \mathbf{b}_j, & j = L-1 \\ \sigma(\mathbf{M}_j \mathbf{y}_{j-1} + \mathbf{b}_j), & j = \overline{0, L-2} \end{cases}, \quad (5.8.2)$$

где  $\mathbf{M}_j \in \mathbb{C}^{C_j \times C_{j-1}}$ ,  $\mathbf{b}_j \in \mathbb{C}^{C_j \times 1}$   $j = \overline{0, L-1}$ . При этом в (5.8.2)  $\mathbf{b}_j$  складывается с каждым столбцом матрицы  $\mathbf{M}_j \mathbf{y}_{j-1}$ . Кроме того, отметим, что  $C_{L-1} = 1$ , т.е.  $\mathbf{M}_{L-1} \in \mathbb{C}^{1 \times C_{L-2}}$ ,  $\mathbf{b}_j \in \mathbb{C}$  – скаляр, поэтому  $\mathbf{y} \equiv \mathbf{y}_{L-1} \in \mathbb{C}^{1 \times M}$ . Также отметим, что нелинейная функция активации  $\sigma(\cdot)$  применяется поэлементно.

## 5.9 Применение канонического тензорного разложения для аппроксимации многомерных структур

## Глава II

# Исследование эффективности методов адаптивной компенсации нелинейных искажений в устройствах связи

### 1. Структура тестовой платформы

В данном разделе описывается тестовая установка, на которой сформирован сигнал, имитирующий нелинейно искаженный сигнал усилителем мощности в передатчике устройства связи.

Измерительная установка состоит из персонального компьютера (ПК), генератора сигналов (ГС) R&S SMW200A, анализатора спектра (АС) R&S FSW85 и испытуемого усилителя мощности ZKY66291-11.

ПК загружает синфазную и квадратурную составляющие базовой полосы (IQ-данные) в ГС, который осуществляет модуляцию, переносит сигнал на несущую частоту  $f_{\text{гет}} = 1,8 \text{ ГГц}$  и передает сигнал на вход УМ. Выходной сигнал УМ с нелинейными искажениями поступает на АЦП анализатора спектра с частотой дискретизации  $f_s = 245.76 \text{ МГц}$ , выбранной в соответствии со стандартом 5G [31]. После этого IQ-данные базовой полосы передаются в ПК для последующей обработки сигнала на языке программирования Python.

Регулировка входной мощности УМ осуществлялась с помощью ГС. В экспериментах использовался комплексный 20-МГц ортогональный сигнал с частотным мультиплексированием (OFDM) сигнал со 100 ресурсными блоками.

Выделение нелинейно искаженного сигнала на выходе УМ  $\text{PA}(\mathbf{x})$  из потока цифрового сигнала происходит путём подсчета взаимной корреляции сигнала  $\text{PA}(\mathbf{x})$  и нелинейного терма передатчика  $\mathbf{x} \odot |\mathbf{x}|$ , как изображено на рис. 12, где  $\mathbf{x} \odot |\mathbf{x}|$  – вектор  $j$ -ым элементом которого является отсчет  $x_{n-j}|x_{n-j}|$ .

Таким образом получены сигнал передатчика  $\mathbf{x}$  и сигнал на выходе УМ  $\text{PA}(\mathbf{x})$ . Спектральные плотности мощности этих сигналов изображены на рис. 13а, 13б.

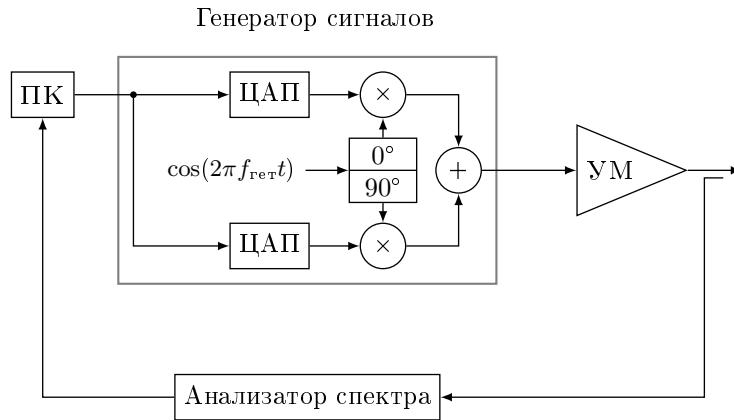


Рис. 11. Установка формирования нелинейных искажений в передатчике устройства связи.

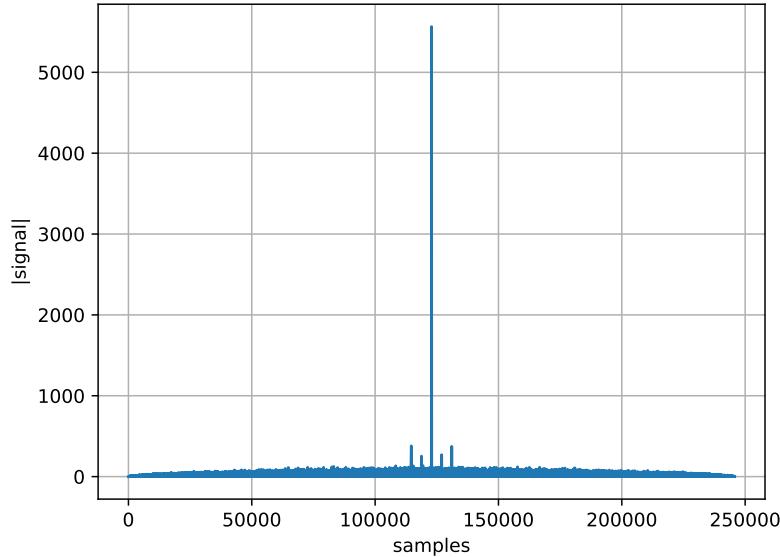


Рис. 12. Модуль функции взаимной корреляции сигнала на выходе УМ РА( $x$ ) и нелинейного терма  $x \odot |x|$ .

## 2. Формирование тестовых наборов данных

Общая схема формирования нелинейных искажений в сигнале передатчика представлена в разделе 1. В данном разделе рассматриваются схемы формирования данных под конкретные задачи компенсации искажений в устройствах связи.

### 2.1 Формирование обучающей выборки для задачи компенсации нелинейной помехи в приёмнике полнодуплексной системы связи

Полнодуплексная система связи подразумевает передачу и приём данных в одной и той же полосе частот. Поскольку в реальных устройствах связи существуют

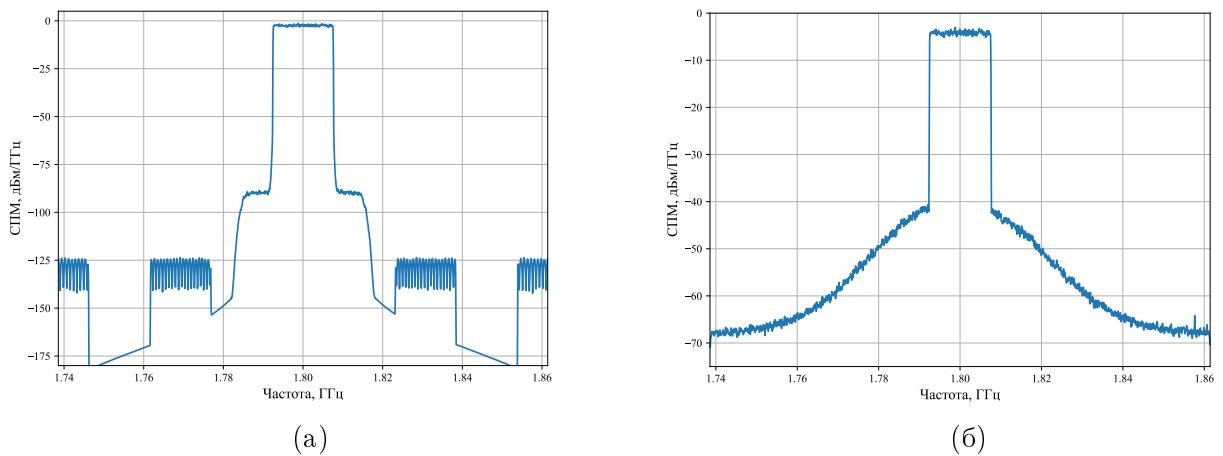


Рис. 13. Спектральная плотность мощности: (а) сигнал, формируемый на передатчике для отправки в канал связи  $x$ ; (б) сигнал  $\text{PA}(x)$ , образованный после прохождения сигнала передатчика через усилитель мощности

проблемы в обеспечении изоляции передатчика и приёмника, сигнал передатчика высокой мощности попадает в канал приёмника и ухудшает качество приёма.

Будем считать, что канал распространения помехи передатчик-приёмник линейный. Частотная характеристика этого канала сформирована при помощи программного обеспечения Python в соответствии с данными о реальном канале распространения помехи и представлена на рис. 14а

Для симуляции прохождения нелинейной помехи по различным путям распространения от передатчика к приёмнику производится свёртка массива отсчётов  $\text{PA}(x)$  с коэффициентами  $w$  импульсной характеристики канала:

$$\mathbf{d} = \text{conv}_{\mathbf{w}} \text{PA}(\mathbf{x}) \quad (2.1.1)$$

Таким образом, получаем сигнал  $\mathbf{d}$ , имитирующий сигнал паразитной помехи передатчика на приёмнике мобильного терминала. Спектральная плотность мощности сигнала паразитной помехи  $\mathbf{d}$  изображена на рис. 14б.

## 2.2 Формирование обучающей выборки для задачи компенсации второй гармоники в приёмнике системы связи

Ввиду сложности обеспечения изоляции дуплексного фильтра приемо-передатчики, работающие в режиме частотного разделения каналов, подвержены утечке передаваемого сигнала в приемный тракт. Утечка сигнала в сочетании с нелинейностью малошумящего усилителя (МШУ) и смесителя понижения частоты приемника может привести к генерации интермодуляционных искажений второго порядка, что значительно снижает чувствительность приемника на частоте базовой полосы.

Схема генерации нелинейных искажений второго порядка представлена на рис. 15. Экспериментальная установка состоит из компьютера (ПК), усилителя мощности

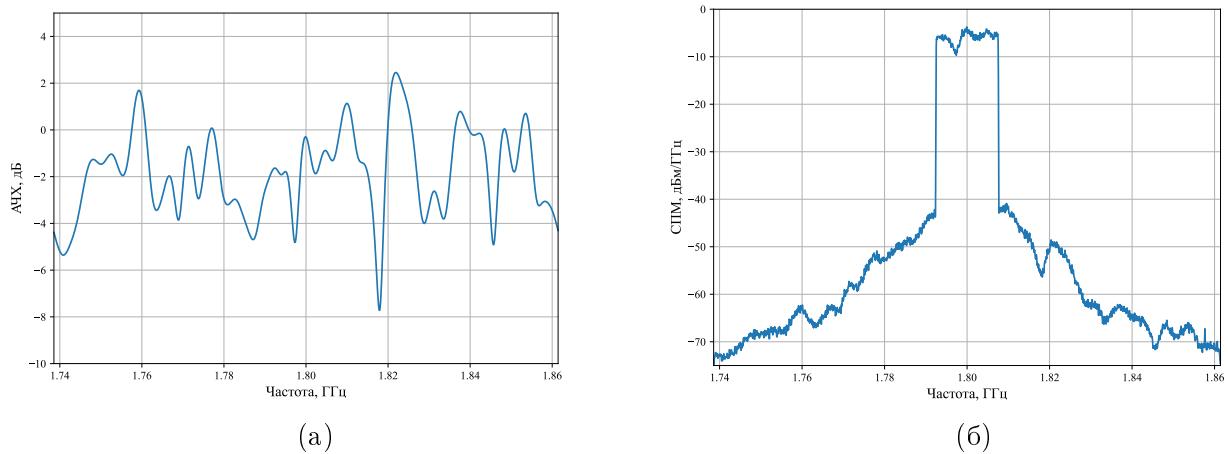


Рис. 14. (а) Амплитудно-частотная характеристика канала распространения второй гармоники от передатчика к приёмнику; (б) паразитная помеха на приёмнике  $d$

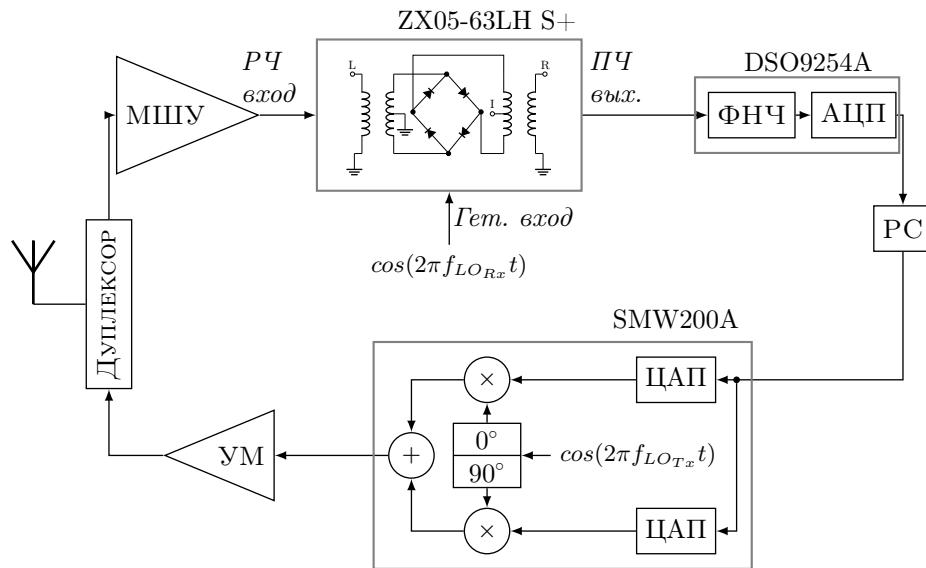


Рис. 15. Установка формирования нелинейных искажений второго порядка в приёмнике устройства связи.

(УМ) ZRL-3500+. Выход УМ подключен к полосовому фильтру, который представляет собой дуплексный фильтр ( $\Delta\Phi$ ) с подавлением 30 dB в полосе заграждения. Выход  $\Delta\Phi$  подключен к малошумящему усилителю (МШУ) ZRL-3500+ с коэффициентом усиления 26 dB. Для преобразования частоты использовался частотный смеситель ZX05-63LH-S+. В передающей и приемной частях дополнительные аналоговые фильтры не использовались.

Для экспериментов использовался комплексный сигнал OFDM с шириной полосы 5 МГц,  $f_{Tx} = 814$  МГц,  $f_{Rx} = 859$  МГц, дуплексное разделение 45 МГц (5G NR Band 26). Сигнал LTE передается генератором сигналов R&S SMW 200A и усиливается усилителем мощности (УМ). Передаваемый сигнал после нелинейного УМ проникает через полосу заграждения дуплексера с частотным сдвигом 45 МГц относительно сигнала гетеродина и усиливается МШУ. Усиленный сигнал генерирует интермодуляционные помехи на выходе частотного смесителя. После понижения частоты сигнал

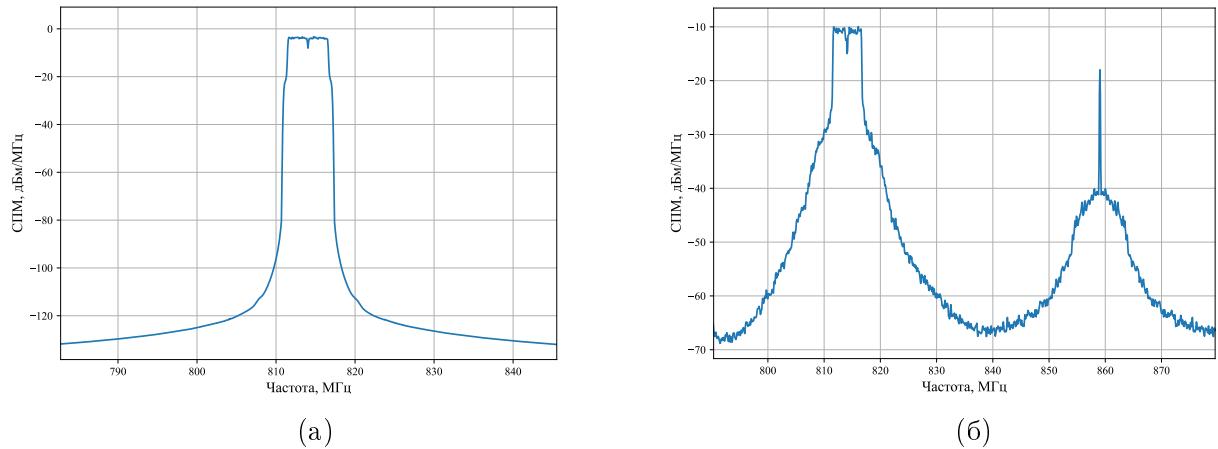


Рис. 16. Спектральная плотность мощности: (а) сигнала передатчика до прохождения нелинейных компонент; (б) на выходе нелинейного УМ РА( $x$ )

захватывается цифровым осциллографом DSO9254A. Сигнал гетеродина мощностью 10 дБм для ZX05-63LH-S+ формируется генератором сигналов R&S SMW 200A.

Передаваемая мощность на выходе УМ установлена на уровне  $P_{Tx} = 8$  дБм, что в сочетании с аттенюацией дуплексного фильтра 30 дБ (при  $f_{Tx} = 814$  МГц) и усилением МШУ 26 дБ обеспечивает мощность паразитного сигнала на входе смесителя понижения частоты  $P = 8$  дБм – 30 дБ + 26 дБ  $\approx 4$  дБм.

Сигнал передатчика  $x$  на нулевой частоте, а также сигнал на выходе нелинейного УМ РА( $x$ ) изображены на рис. 16 . Заметим, что нелинейное искажение второго порядка  $x^2$  формируется засчет нелинейности компонент смесителя. Этот процесс можно формально разделить на 2 шага: формирование 2-ой гармоники  $x^2$ , а затем перенос на несущую частоту передатчика  $x^2 e^{j2\pi f_{Tx} t}$ . **Уточнить процесс формирования второй гармоники.**

### 2.3 Формирование обучающей выборки для задачи цифровых предыскажений в передатчике двухканальной системы связи

Система цифрового предыскажения для двухканальной системы связи изображена на рис. 17. Цифровые сигналы каналов А и В на нулевой частоте:  $x_A$  и  $x_B$  соответственно, - проходят через цифровой повышающий преобразователь (ЦПП), преобразовываются в аналоговый сигнал. Затем суммарный двухканальный сигнал переносится на несущую частоту  $f_{гет} = 1990$  МГц. Частотный разнос канала составляет 300 МГц. Таким образом сигналы каналов А и В находятся на несущих 1840 МГц и 2140 МГц соответственно [32]. Спектральная плотность мощности суммарного узкополосного сигнала передатчика на выходе УМ изображена на рис. 18.

Суммарный узкополосный сигнал далее проходит через нелинейный УМ и отправляется в петлю обратной связи через направленный ответвитель. Спектральная

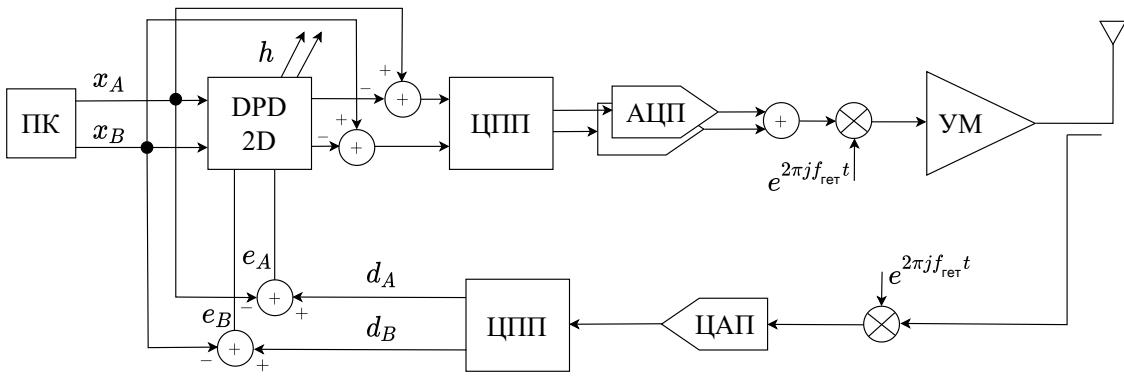


Рис. 17. Схема компенсации нелинейных искажений в передатчике двухканальной системы связи.

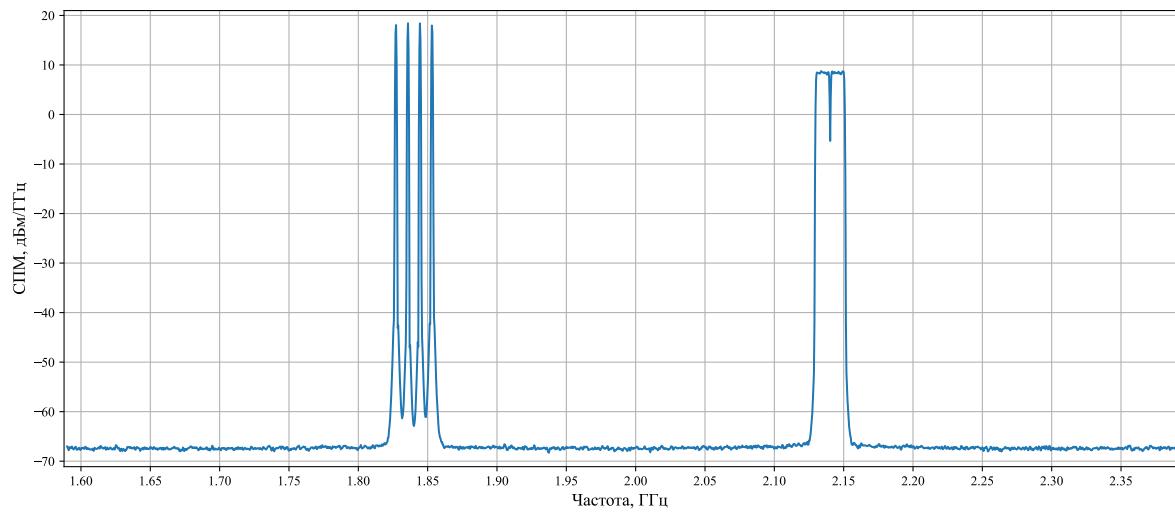


Рис. 18. Спектральная плотность мощности двухканального узкополосного сигнала передатчика до прохождения нелинейного УМ.

плотность мощности узкополосного сигнала на выходе УМ изображена на рис. 19.

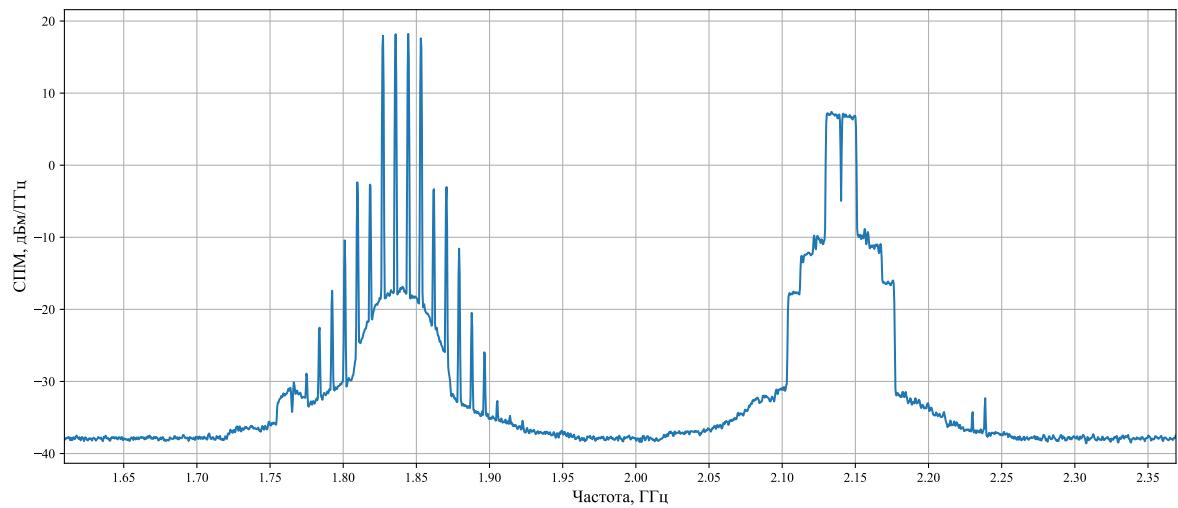


Рис. 19. Спектральная плотность мощности двухканального узкополосного сигнала передатчика после прохождения нелинейного УМ.

В петле обратной связи узкополосный сигнал переносится на нулевую частоту, оцифровывается и проходит через цифровой понижающий преобразователь (ЦПП). ЦПП разделяет каналы и переносит их на нулевую частоту. В результате получаем 2 цифровых нелинейно искаженных сигнала передатчика  $d_A, d_B$  на нулевой частоте.

В разделе 1.4 главы I показано, что задача компенсации нелинейных искажений в передатчике сводится к идентификации нелинейных искажений на основе исходного сигнала передатчика [3]. Таким образом, обучающая выборка состоит из исходных сигналов передатчика на нулевой частоте  $x_A, x_B$ , а также сигналов отклонения выхода УМ от входа:

$$e_A = d_A - x_A$$

$$e_B = d_B - x_B.$$

Спектральные плотности мощности нелинейных искажений на нулевой частоте  $e_A, e_B$  изображены на рис. 20.

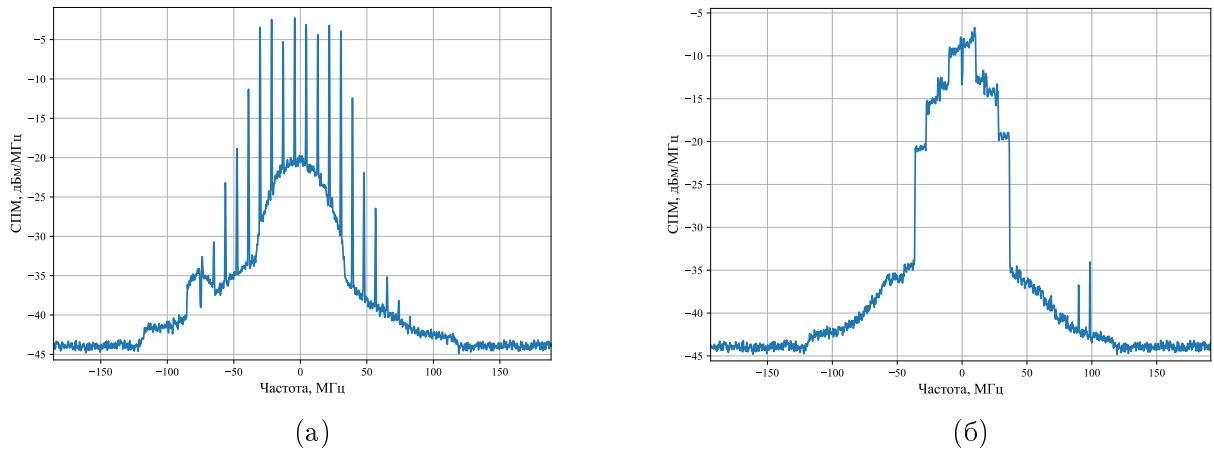


Рис. 20. Спектральная плотность мощности сигнала нелинейных искажений: (а) канал А; (б) канал В

## 2.4 Формирование обучающей выборки для задачи цифровых предыскажений в передатчике одноканальной системы связи в условиях динамически меняющегося режима работы усилителя мощности

В реальных системах связи для соответствия требованиям протоколов связи [2] производится динамическое распределение ресурсных блоков в рамках кадра данных в соответствии с реальным трафиком. Это распределение вызывает кратковременные изменения уровня мощности, что, в свою очередь, снижает коэффициент полезного действия (КПД) усилителя мощности. Указанный факт противоречит предположению о работе алгоритма цифрового предыскажения в стационарном режиме. Данная

проблема исследуется в работе [33].

В то же время, для повышения энергоэффективности требуется увеличение динамического диапазона входной мощности УМ. Компенсация нелинейных искажений, индуцированных сигналами в широком динамическом диапазоне, требует повторной калибровки коэффициентов модели DPD в реальном времени, что на практике неэффективно с точки зрения распределения ресурсов системы.

Кроме того, переходные процессы, сопровождающие переключение режимов мощности УМ, генерируют нестационарные искажения. Эти искажения, передаваемые по каналу связи, являются причиной кратковременных нарушений протоколов связи и деградации спектральной маски излучаемого сигнала. Как следствие, прямое применение параметров модели DPD, полученных для одного режима мощности УМ, к сигналам другого режима приводит к значительной деградации характеристик системы.

Обучающая выборка для исследования динамического режима работы усилителя мощности состоит из комплексного OFDM сигнала шириной полосы 20 МГц со 100 ресурсными блоками на несущей частоте  $f_{LO} = 1.8$  ГГц. Выборка включает в себя  $C = 61$  исходных сигналов передатчика в динамическом диапазоне 15 дБ:

$$\begin{aligned} & 25.1 \text{ мкВт}, 37.9 \text{ мкВт}, 50.8 \text{ мкВт}, \dots, 794.3 \text{ мкВт}, \\ & 14.0 \text{ дБмк}, 15.8 \text{ дБмк}, 17.1 \text{ дБмк}, \dots, 29.0 \text{ дБмк}. \end{aligned} \quad (2.4.1)$$

Им соответствуют  $C = 61$  сигналов на выходе УМ в динамическом диапазоне 11.2 дБ:

$$\begin{aligned} & 0.069 \text{ Вт}, 0.107 \text{ Вт}, 0.143 \text{ Вт}, \dots, 0.912 \text{ Вт} \\ & -11.6 \text{ дБВт}, -9.7 \text{ дБВт}, -8.4 \text{ дБВт}, \dots, -0.4 \text{ дБВт} \end{aligned} \quad (2.4.2)$$

На рис. 21 изображены распределения входной и выходной мощности УМ в линейном масштабе. Как видно из рис. 21а мощность входного сигнала УМ распределена равномерно. Для исключения смещения модели в процессе её обучения в пользу какого-либо уровня мощности, входная мощность УМ имеет равномерное распределение во всем динамическом диапазоне. Заметим, что согласно рис. 21б распределение выходной мощности УМ отличается от равномерного ввиду нелинейности характеристики УМ.

На рис. 22 изображены спектральные плотности мощности нормированного входного сигнала УМ 22а и нормированного выходного сигнала УМ 22б для 4-х случаев выходной мощности: -0.4 дБВт, -4.6 дБВт, -8.5 дБВт, -11.6 дБВт. Заметим, что 4-м случаям выходной мощности соответствуют неискаженные сигналы передатчика с мощностями 29.0 дБмк, 21.9 дБмк, 17.1 дБмк, 14.0 дБмк соответственно.

Отметим, что графики СПМ входных сигналов УМ различной мощности совпадают с СПМ на рис. 22а с точностью до смещения в константу раз.

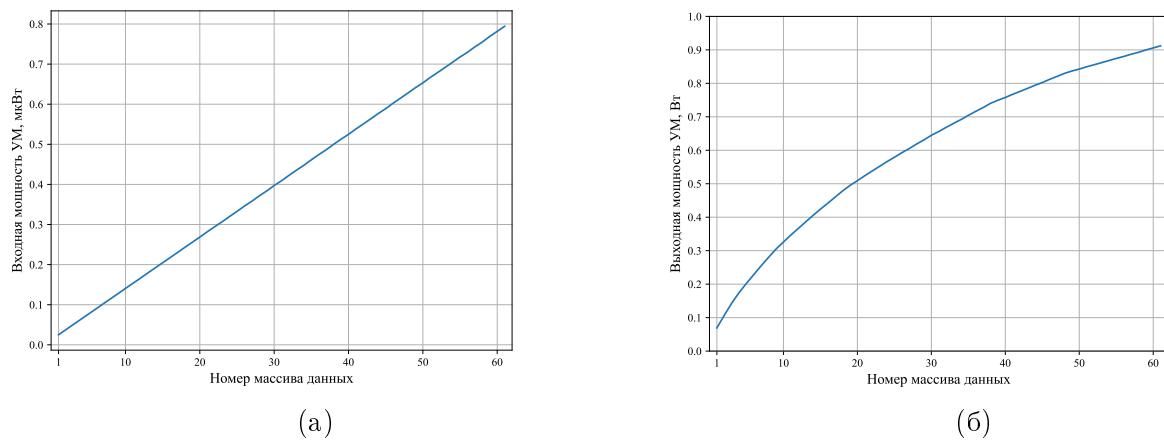


Рис. 21. Динамический диапазон мощности УМ: (а) входная мощность УМ; (б) выходная мощность УМ

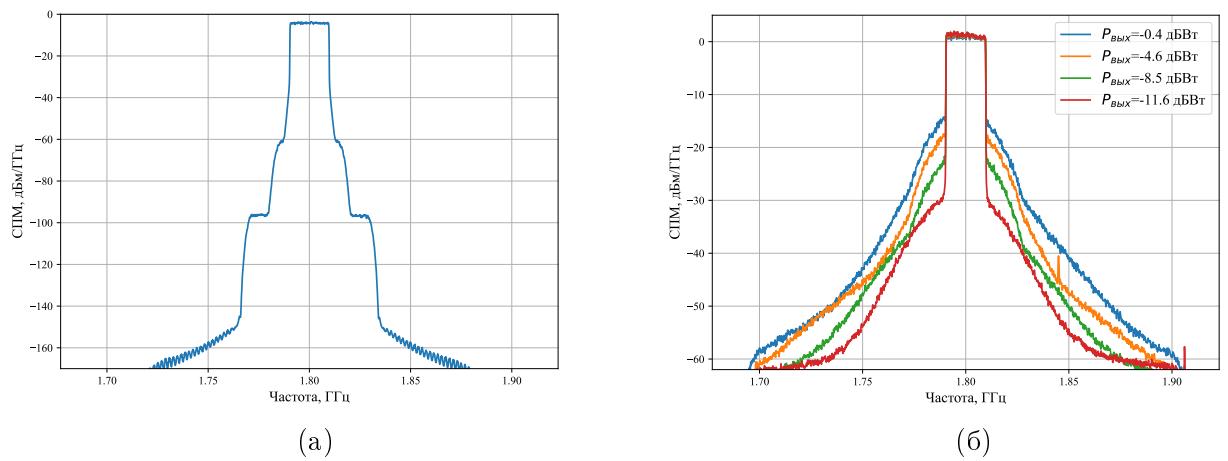


Рис. 22. Спектральная плотность мощности: (а) входного сигнала УМ; (б) выходного сигнала УМ

### 3. Исследование алгоритмов и адаптивных моделей компенсации паразитной помехи на приёмнике устройств связи

#### 3.1 Компенсация паразитной помехи на приёмнике полнодуплексной системы связи методом адаптации модели Гаммерштейна

В данном разделе рассматривается задача компенсации паразитных помех, возникающих в приёмнике полнодуплексной системы связи. Процесс возникновения помехи, а также тестовая установка формирования паразитных помех данного типа описаны в разделах 1, 2.1 главы II.

Адаптивная модель [behav\_model] компенсации паразитной помехи строится таким образом, чтобы отражать физические процессы возникновения данного рода

помех. Согласно схемы возникновения и компенсации паразитных помех на приёмнике на рис. 3 в сигнал передатчика искажается вследствие нелинейности характеристик усилителя передатчика и дуплексера. Затем сигнал искажается вследствие распространения по каналу TX-RX.

Таким образом, модель Винера-Гаммерштейна [**behav\_model**], представленная на рис. 9 отражает процесс возникновения собственной помехи в примённом тракте и может быть использована для решения задачи идентификации помехи.

Согласно результатам эксперимента, представленным в текущем разделе, эффекты инерционности УМ проявляются слабо. В связи с этим в данном эксперименте использована модель Гаммерштейна (рис. 7), которая является упрощением модели Винера-Гаммерштейна (рис. 9) для случая нелинейных эффектов без памяти.

В качестве нелинейного блока модели Гаммерштейна (5.5.1) выбраны линейные сплайновые полиномы, описанные в разделе 5.3 главы I. Линейно-сплайновые полиномы в данном эксперименте выбраны ввиду его низкой вычислительной сложности и простоты аппаратной реализации [**lut\_dpd, 28**].

Для экспериментов была выбрана модель Гаммерштейна с адаптивным фильтром с  $L = 45$  числом параметров, а также кусочно-линейной функцией, содержащей  $P = 8$  сплайнов первого порядка.

Полный набор данных состоит из 80000 отсчётов, 50% из которых используется как тренировочная выборка, остальные отсчёты используются как тестовая выборка.

В данном эксперименте проводится сравнение блочного градиентного спуска (BGD, Block Gradient Descent), стохастического градиентного спуска (SGD, Stochastic Gradient Descent), описанных в разделе 2.1 главы I со смешанным методом Ньютона (раздел 3.3 главы I). Отметим, что ввиду голоморфности выхода модели Гаммерштейна (5.5.1), метод градиентного спуска для адаптации данной модели по методу наименьших квадратов представлен выражением (2.1.10). Отметим также, что в данном эксперименте отличие блочного и стохастического градиентного спуска заключается в формировании блока данных для шага адаптации.

Шаг адаптации смешанного метода Ньютона и BGD пересчитывается на всей тренировочной последовательности. Другими словами, якобиан выхода модели по параметрам модели  $D_z y$ , представленный выражением (5.5.10), а также вектор ошибки (2.1.10) вычисляются для  $N = 40000$  отсчётов.

С другой стороны, в случае стохастического градиентного спуска вся тренировочная последовательность делится на неперекрывающиеся блоки, которые упаковываются в 50 блоков по 200 отсчётов каждый. Таким образом, за одну эпоху (одно прохождение всей тренировочной последовательности) модель тренируется на 4 блоках. Градиенты вычисляются для каждого блока (по 200 отсчётов) и усредняются (по 50 блоков). Размер блока был выбран экспериментально с точки зрения максимизации результирующего уровня качества модели.

Также следует обратить внимание, что в случае блочного градиентного спуска и

смешанного метода Ньютона параметры обновляются один раз за эпоху, то есть после подсчёта шага на полной последовательности тренировочных данных. В то время как при реализации стохастического градиентного спуска параметры пересчитываются после вычисления стохастического градиента на каждом блоке ( $50 \times 200$  отсчётов).

Кроме того, в данном численном эксперименте рассмотрены ускоренные версии градиентного спуска, представленные оптимизаторами Adam и Momentum, описанными в разделе 2.2 главы I.

В следующих симуляциях нормированная среднеквадратичная ошибка NMSE (1.5.2) отслеживается на каждой эпохе на тренировочной и тестовой последовательности, что отражено на рис. 23, 24.

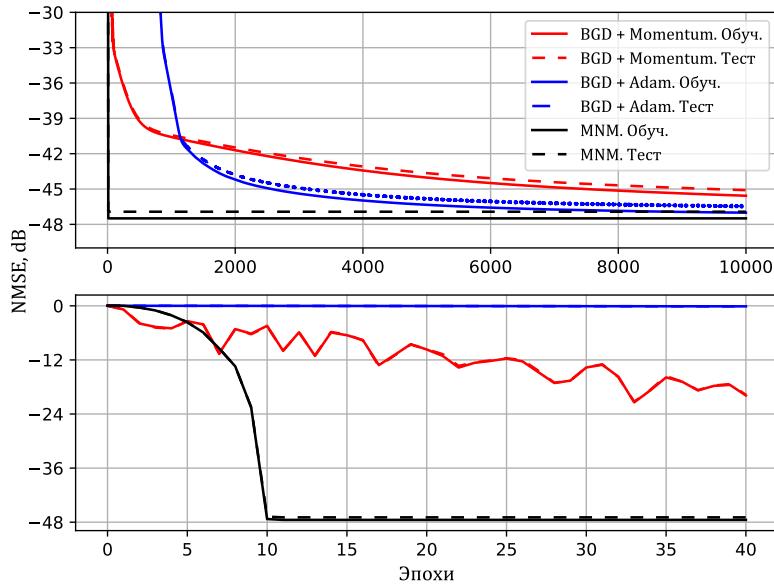


Рис. 23. Кривые сходимости на тренировочной и тестовой последовательностях. Сравнение BGD с оптимизаторами Momentum, Adam и смешанного метода Ньютона. BGD – блочный градиентный спуск, MNM – смешанный метод Ньютона, NMSE – нормированная среднеквадратичная ошибка, Adam, Momentum – оптимизаторы градиентного спуска.

Сравнение скорости сходимости блочного градиентного спуска с оптимизаторами Momentum и Adam, а также смешанного метода Ньютона показано на рис. 23. Смешанный метод Ньютона требует 30 эпох для достижения результирующего значения уровня компенсации, в то время как блочным градиентным спуском с оптимизаторами требуют  $\sim 10000$  эпох для достижения такого же значения NMSE.

На рис. 24 представлены кривые обучения, полученные для стохастического градиентного спуска с оптимизаторами Momentum и Adam и смешанного метода Ньютона. Графики сходимости отражают тот факт, что SGD также требует приблизительно 10000 эпох для достижения уровня компенсации паразитной помехи, полученного при помощи смешанного метода Ньютона.

Отметим, что параметры оптимизаторов и темп обучения градиентного спуска, были выбраны с точки зрения наилучшего результирующего качества модели и вы-

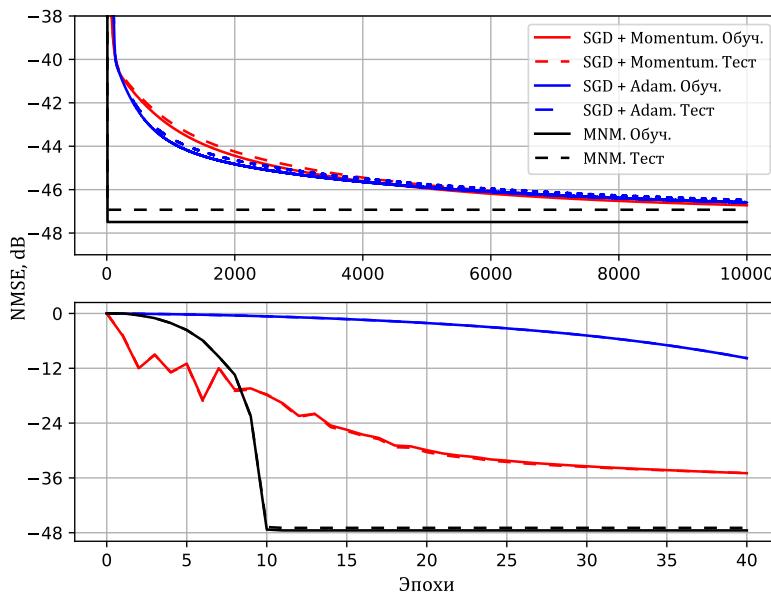


Рис. 24. Кривые сходимости на тренировочной и тестовой последовательностях. Сравнение SGD с оптимизаторами Momentum, Adam и смешанного метода Ньютона. SGD – стохастический градиентный спуск, MNM – смешанный метод Ньютона, NMSE – нормированная среднеквадратичная ошибка, Adam, Momentum – оптимизаторы градиентного спуска.

сокой скорости сходимости.

Спектральные плотности мощности (СПМ) исходной и компенсированных различными методами помех изображены на рис. 25. Графики СПМ построены на основе тестовой последовательности.

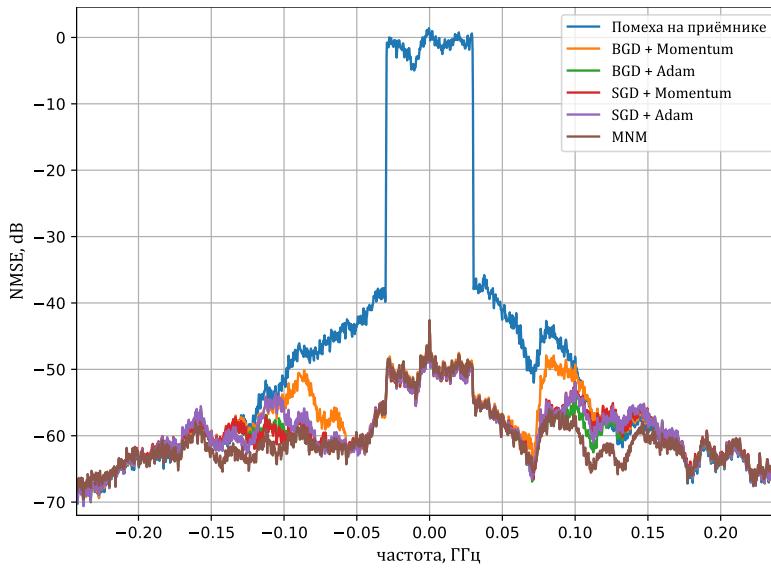


Рис. 25. Спектральные плотности мощности исходной и компенсированных помех. BGD – блочный градиентный спуск, SGD – стохастический градиентный спуск, Adam, Momentum – оптимизаторы градиентного спуска, MNM – смешанный метод Ньютона, NMSE – нормированная среднеквадратичная ошибка.

Таблица 1 отражает заметное увеличение скорости сходимости смешанного метода Ньютона по сравнению с классическими градиентными методами. Для достиже-

ния сопоставимого уровня компенсации для смешанного метода Ньютона требуется всего 30 эпох. Несмотря на то, что при использовании метода второго порядка требуется существенно меньшее число шагов оптимизации, каждый шаг адаптации вычисляется  $\sim 5$  в раз больше. Тем не менее, общее время, затрачиваемое на обучение модели, существенно меньше по сравнению с методами первого порядка.

Таблица 1. Сравнение уровня компенсации помехи и скорости сходимости

Алгоритм	BGD Moment.	BGD Adam	SGD Moment.	SGD Adam	MNM
Номер эпохи	10000	10000	10000	10000	30
Время за эпоху, $10^{-2}$ с	3.8	4.0	3.7	4.1	21
Общее время работы, с	380	403	386	412	6.2
NMSE, dB	-45.1	-46.7	-46.6	-46.5	-46.9

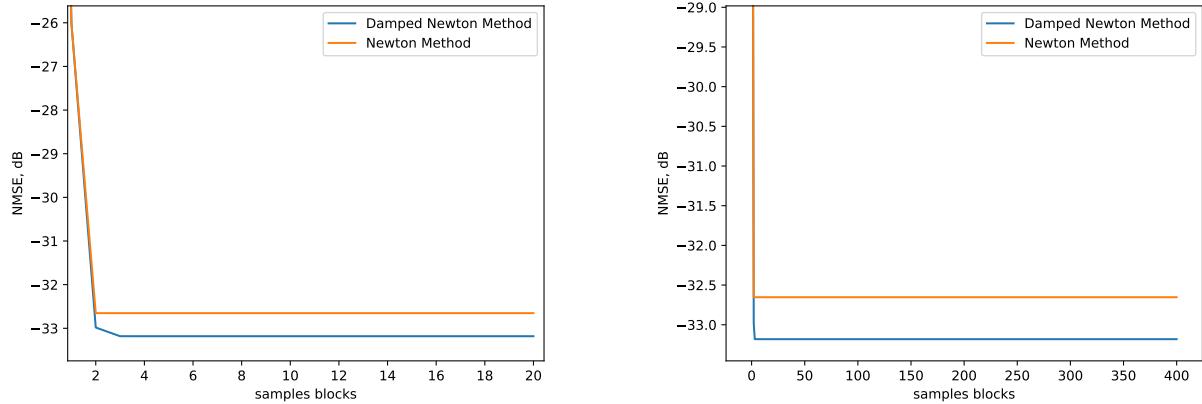
Как было отмечено ранее, итерация смешанного метода Ньютона подразумевает обращение матрицы Гессе (3.3.4). В связи с этим вычислительная сложность шага смешанного метода Ньютона [34] высока по сравнению с алгоритмами на основе градиентного спуска [35], [13]. Тем не менее, число шагов, необходимых для достижения такого же уровня компенсации, значительно меньше для смешанного Ньютона (таблица 1). Таким образом, смешанный метод Ньютона отлично подходит для исследования возможного уровня качества моделей. Кроме того, смешанный метод Ньютона можно использовать для онлайн обучения моделей, так как в реальных приложениях требуется накопление матрицы Гессе и, как следствие, частота шагов оптимизации существенно меньше по сравнению с методами первого порядка.

Таким образом, несмотря на то, что вычисление шага оптимизации смешанного метода Ньютона требует в  $\sim 5$  раз больше времени, общая длительность обучения уменьшена до 30 эпох, по сравнению 10000 эпох необходимых для классических методов оптимизации первого порядка, для достижения того же уровня компенсации 46.9 dB.

### 3.2 Компенсация паразитной помехи 2-ого порядка на основе адаптации классических и нейросетевых структур

Сравнение уровня компенсации паразитной помехи различными алгоритмами будем проводить относительно критерия нормированного среднего квадрата ошибки (1.5.2), поскольку NMSE позволяет оценить уровень отклонения выхода модели от помехи на приёмнике независимо от динамического диапазона помехи.

Методы второго порядка обеспечивают быстрое приближение к точке оптимума по сравнению с методами первого порядка. В связи с этим целесообразно использо-



(а) Сигналы помехи и на передатчике состоят из 20 блоков длиной 122880 отсчётов

(б) Сигналы помехи и на передатчике состоят из 400 блоков длиной 122880 отсчётов

Рис. 26. Кривые адаптации модели Гаммерштейна методом Ньютона и демпфированным методом Ньютона. Случай {RB25,  $path_0$ }

вать значения NMSE, полученные в результате работы методов второго порядка, в качестве опорных уровней подавления помехи для дальнейшего сравнения с ними значений NMSE, полученных градиентными методами первого порядка.

Следует отметить, что метод Ньютона может стагнировать при определенных условиях. Это может произойти, например, в случае если коэффициенты модели примут значения, соответствующие точке в окрестности седловой точки целевой функции. В этом случае глобальный минимум не может быть достигнут в результате работы метода Ньютона.

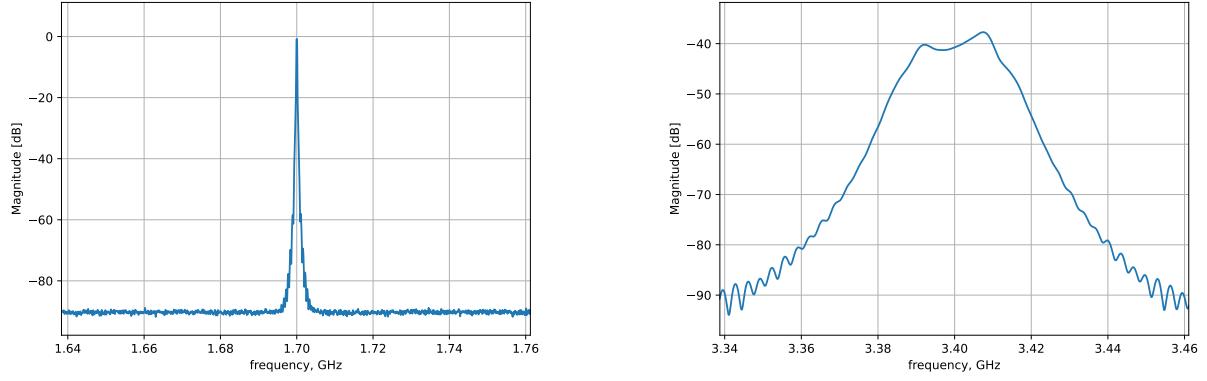
На рис. 26 изображены кривые адаптации метода Ньютона и демпфирированного метода Ньютона в случае, когда сигнал передатчика имеет ширину полосы равную 4.5 МГц (RB25), а канал распространения помехи  $path_0$ . При этом сигналы передатчика и помехи состоят из 20 блоков (рис. 26а) и 400 блоков (рис. 26б) длиной 122880 отсчётов.

Таким образом, предпочтительней использовать значения критерия NMSE, полученные в результате работы демпфирированного метода Ньютона в качестве опорных значений.

В дальнейшем при исследовании работы стохастических алгоритмов будем сравнивать значения критерия NMSE, получаемые в результате работы алгоритмов SGD-SGD, SGD-DCD со значениями NMSE, получаемыми в результате работы демпфирированного метода Ньютона.

Вычислим опорные значения критерия NMSE, полученные при помощи демпфирированного метода Ньютона для каждого случая, отображённого в таблице ??.

Рассмотрим подавление паразитной помехи в случае {RB1,  $path_0$ } (рис. 27), когда ширина полосы сигнала передатчика определяется одним ресурсным блоком, 180 кГц (рис. 27а), а канал распространения определяется частотной характеристикой  $path_0$  (рис. 27б).



(а) Спектральная плотность мощности сигнала передатчика с одним ресурсным блоком

(б) Амплитудно-частотная характеристика канала распространения помехи  $path_0$

Рис. 27. Условия формирования сигнала помехи на приёмнике  $\{RB1, path_0\}$

На рис. 28 изображены спектральные плотности мощности паразитной помехи, сигнала на выходе модели Гаммерштейна после адаптации демпфированным методом Ньютона, а также отклонения выхода модели Гаммерштейна от сигнала помехи. Значение критерия NMSE в данном случае составляет  $-23.8$  dB.

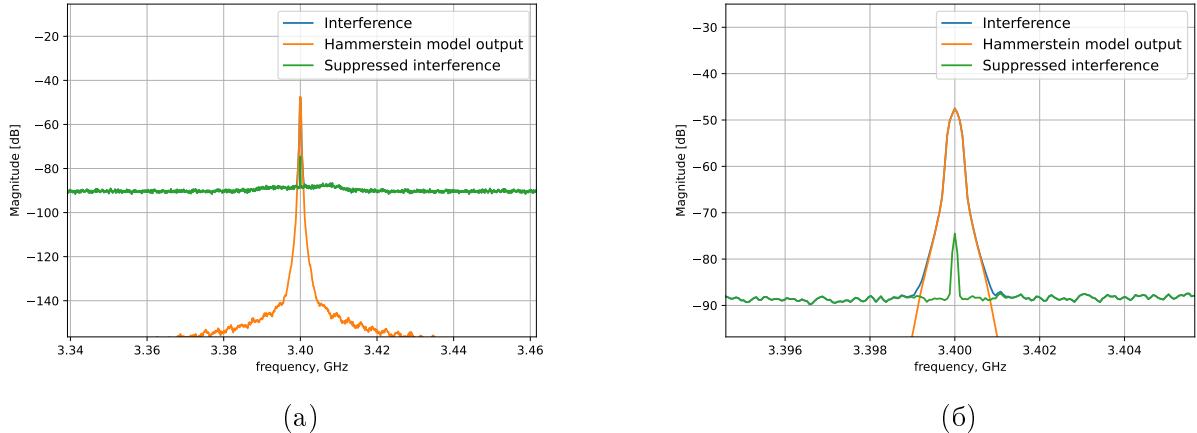
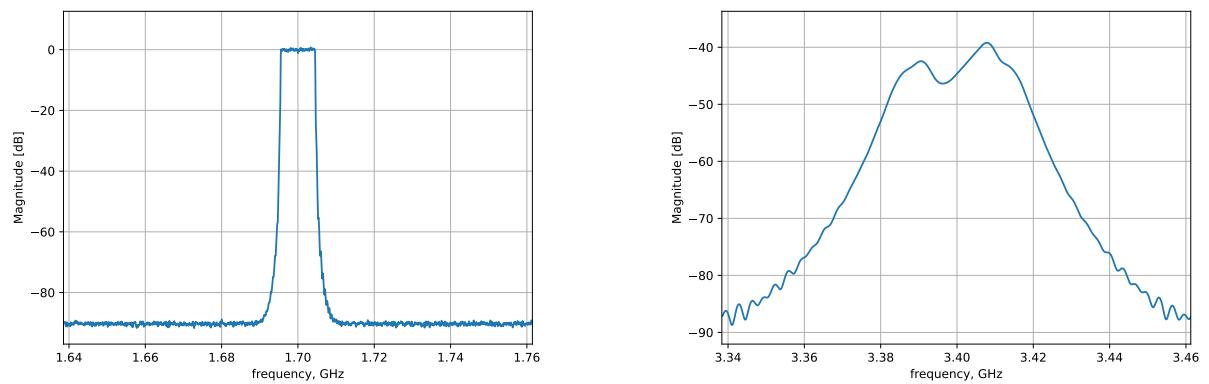


Рис. 28. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом Ньютона. Случай  $\{RB50, path_1\}$

Рассмотрим подавление паразитной помехи в случае  $\{RB50, path_1\}$  (рис. 29), когда ширина полосы сигнала передатчика определяется 50-ю ресурсными блоками, 9 МГц (рис. 29а), а канал распространения определяется частотной характеристикой  $path1$  (рис. 29б).

На рис. 30 изображены спектральные плотности мощности паразитной помехи, сигнала на выходе модели Гаммерштейна после адаптации демпфированным методом Ньютона, а также отклонения выхода модели Гаммерштейна от сигнала помехи. Значение критерия NMSE в данном случае составляет  $-36.2$  dB.

Рассмотрим подавление паразитной помехи в случае  $\{RB100, path_2\}$  (рис. 31), когда ширина полосы сигнала передатчика определяется 100 ресурсными блоками,



(а) Спектральная плотность мощности сигнала передатчика с 50-ю ресурсными блоками

(б) Амплитудно-частотная характеристика канала распространения помехи  $path_1$

Рис. 29. Условия формирования сигнала помехи на приёмнике  $\{RB50, path_1\}$

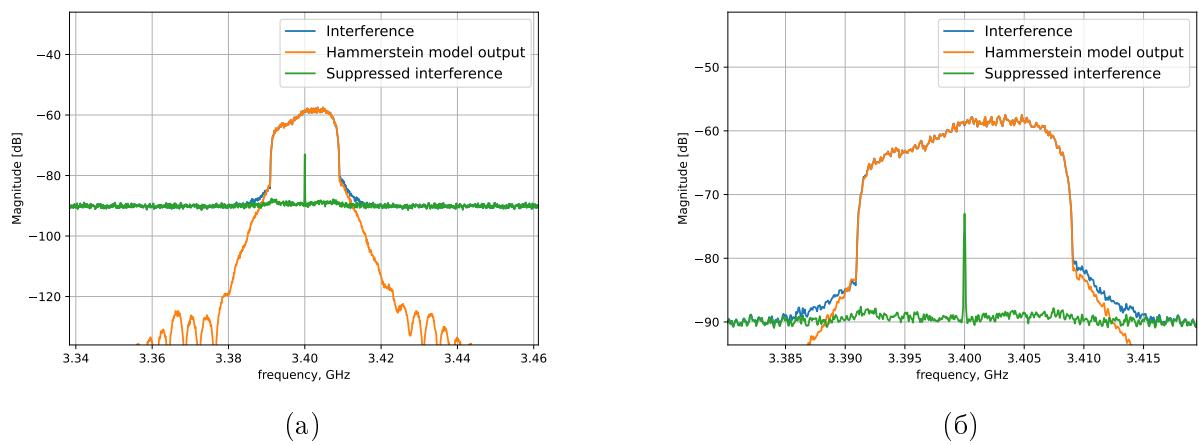
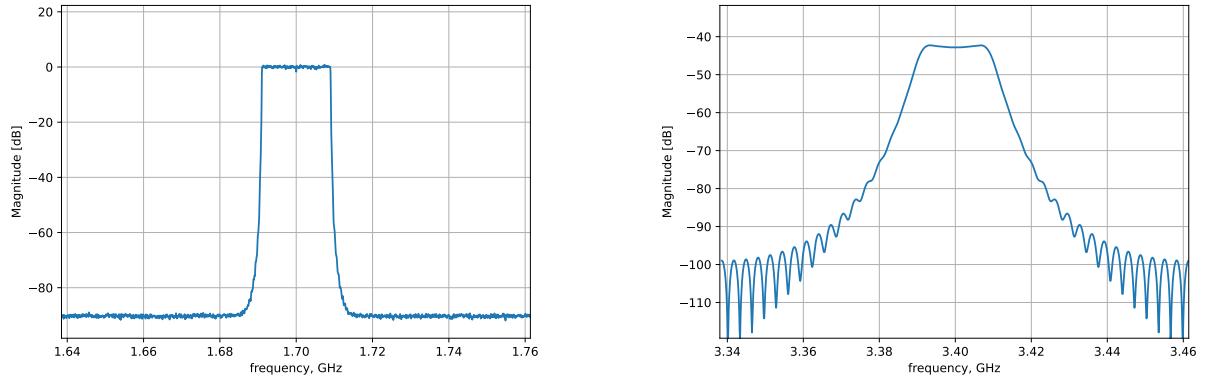


Рис. 30. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом Ньютона. Случай  $\{RB50, path_1\}$



(а) Спектральная плотность мощности сигнала передатчика с 100 ресурсными блоками

(б) Амплитудно-частотная характеристика канала распространения помехи  $path_2$

Рис. 31. Условия формирования сигнала помехи на приёмнике {RB100,  $path_2$ }

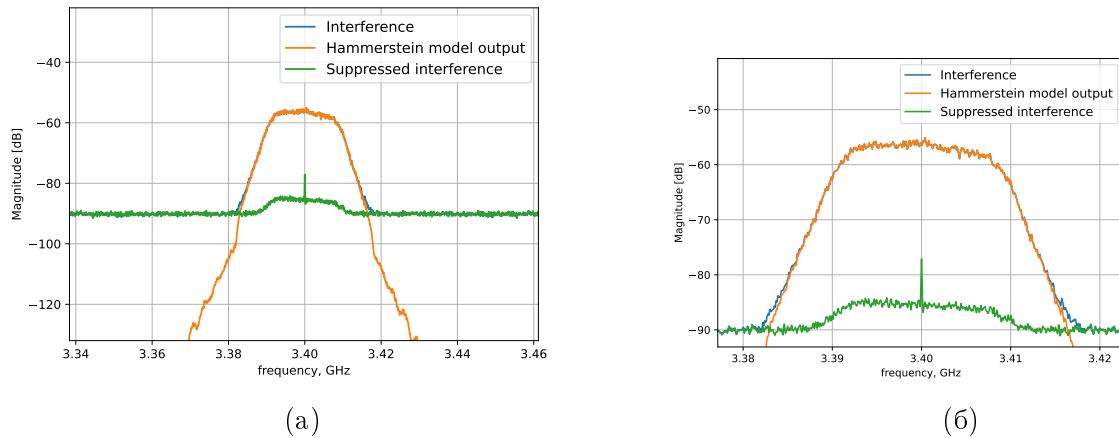


Рис. 32. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом Ньютона. Случай {RB100,  $path_2$ }

18 МГц (рис. 31а), а канал распространения определяется частотной характеристикой  $path_2$  (рис. 31б).

На рис. 32 изображены спектральные плотности мощности паразитной помехи, сигнала на выходе модели Гаммерштейна после адаптации демпфированным методом Ньютона, а также отклонения выхода модели Гаммерштейна от сигнала помехи. Значение критерия NMSE в данном случае составляет  $-34.3$  dB.

Спектральные плотности мощности помехи до и после подавления паразитной помехи демпфированным методом Ньютона для других случаев, приведённых в таблице ??, представлены в приложении А1.

Таблица 2 отражает опорные значения критерия NMSE dB, получаемые в результате адаптации модели Гаммерштейна демпфированным методом Ньютона для всех случаев формирования паразитной помехи, представленных в таблице ??.

Таким образом, в результате работы демпфированного метода Ньютона подавление паразитной помехи для сигнала шириной полосы 180 кГц не хуже 22 dB для

	RB1	RB25	RB50	RB75	RB100
$path_0$	-24.1	-33.2	-31.7	-31.8	-29.7
$path_1$	-22.2	-31.7	-31.6	-31.3	-29.2
$path_2$	-25.4	-34.1	-31.8	-32.0	-30.0

Таблица 2. Опорные значения критерия NMSE dB, полученные в результате адаптации демпфированным методом Ньютона для каждого случая формирования паразитной помехи

трех исследованных характеристик линейного канала распространения.

Подавление помехи в случае сигналов передатчика с шириной полосы 4.5 МГц, 9 МГц и 13.5 МГц не хуже 31 дБ для трёх рассмотренных оценок канала распространения помехи.

В случае сигнала передатчика с шириной полосы 18 МГц подавление помехи демпфированным методом Ньютона не хуже 29 дБ также для всех рассмотренных оценок канала распространения.

## 4. Исследование алгоритмов и адаптивных моделей компенсации нелинейных помех в передатчике устройств связи

- 4.1 Компенсация нелинейных искажений в передатчике устройств связи в условиях динамического изменения выходной мощности нелинейного усилителя
- 4.2 Компенсация нелинейных искажений в передатчике двухканальной системы связи

## Глава III

# Практическая реализация нелинейных корректоров в системах связи

## 1. Исследование компенсации нелинейных искажений методами не требующими явного обращения матрицы Гёссе

Проведём сравнение кривых адаптации, а также значений критерия NMSE, полученных в результате работы блочной реализации градиентного спуска и стохастических методов SGD-SGD, SGD-DCD.

Метод SGD-SGD подразумевает адаптацию обоих слоёв модели Гаммерштейна методом стохастического градиентного спуска. Метод SGD-DCD подразумевает адаптацию блока нелинейности амплитудной характеристики усилителя мощности методом стохастического градиентного спуска и адаптацию КИХ-фильтра стохастической реализацией метода DCD.

Будем строить кривые адаптации по значениям критерия NMSE (1.5.2), который подсчитывается после адаптации на блоке сигнала длиной 122880 отсчётов.

На рис. 33 приведены кривые адаптации данных алгоритмов для случая, представленного на рис. 27, при котором ширина полосы сигнала передатчика определяется одним ресурсным блоком, 180 кГц (рис. 27а), а канал распространения определяется частотной характеристикой  $path_0$  (рис. 27б).

На рис. 33 по оси абсцисс отмечены порядковые номера блоков длиной 122880 отсчётов.

Помимо кривых адаптации градиентных методов первого порядка на рис. 33 приведена кривая адаптации демпфированного метода Ньютона, отражающая опорные значения критерия NMSE.

Рис. 33 отражает тот факт, что наибольшей скоростью сходимости среди градиентных методов обладает алгоритм SGD-DCD для данных условий формирования

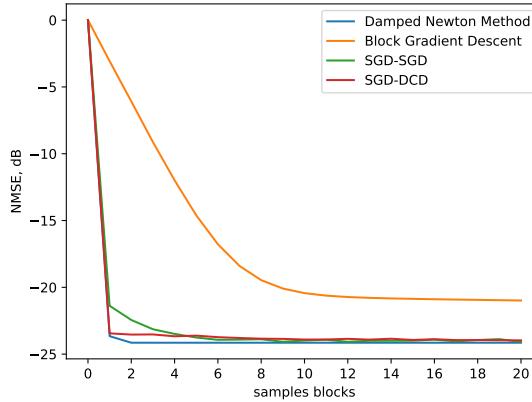


Рис. 33. Кривые адаптации модели Гаммерштейна для случая сигнала передатчика с шириной полосы равной 180 кГц и канала распространения помехи  $path_0$

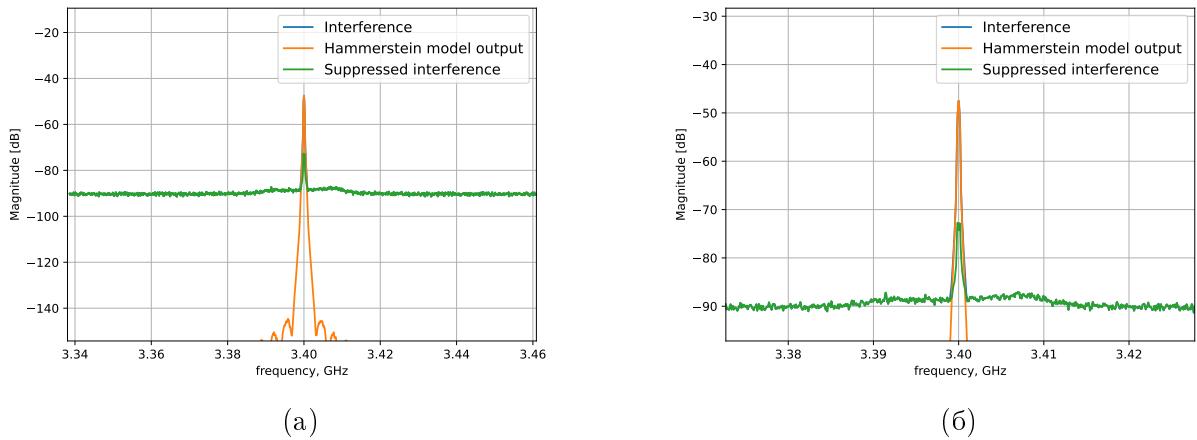


Рис. 34. Спектральная плотность мощности сигнала паразитной помехи после адаптации блочным методом градиентного спуска. Случай  $\{RB1, path_0\}$

помехи. Кроме того, данный метод обеспечивает наилучшее подавление близкое к значениям NMSE, полученным в результате работы демпфированного метода Ньютона.

На рис. 34–36 изображены спектральные плотности мощности паразитной помехи, сигнала на выходе модели Гаммерштейна после адаптации, а также отклонения выхода модели Гаммерштейна от сигнала помехи. Значения критериев NMSE в данном случае составляют  $-21.0$  dB,  $-24.1$  dB и  $-24.0$  dB для блочного метода градиентного спуска, SGD-SGD и SGD-DCD соответственно.

Кривые адаптации данных алгоритмов для случая  $\{RB50, path_1\}$  (рис. 29) приведены на рис. 37. Ширина полосы сигнала передатчика определяется 50-ю ресурсными блоками, 9 МГц (рис. 29а), а канал распространения определяется частотной характеристикой  $path_1$  (рис. 29б).

Рис. 37 отражает тот факт, что наибольшей скоростью сходимости среди градиентных методов обладает алгоритм SGD-SGD для данных условий формирования

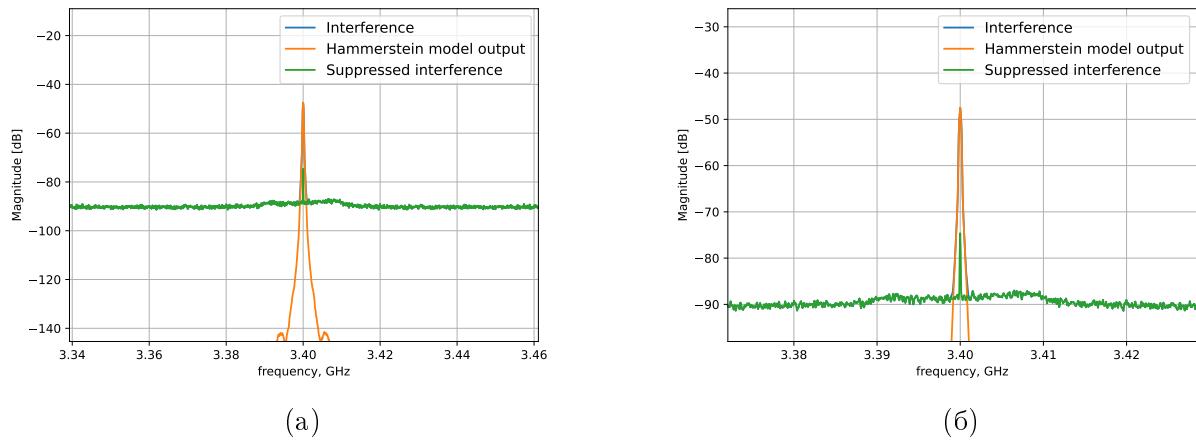


Рис. 35. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом SGD-SGD. Случай  $\{RB1, path_0\}$

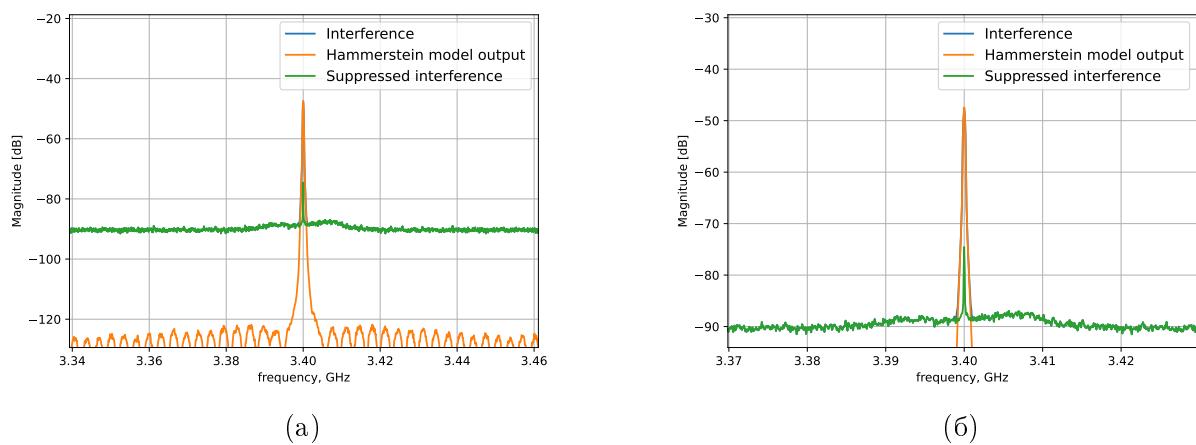


Рис. 36. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом SGD-DCD. Случай {RB1, path<sub>0</sub>}

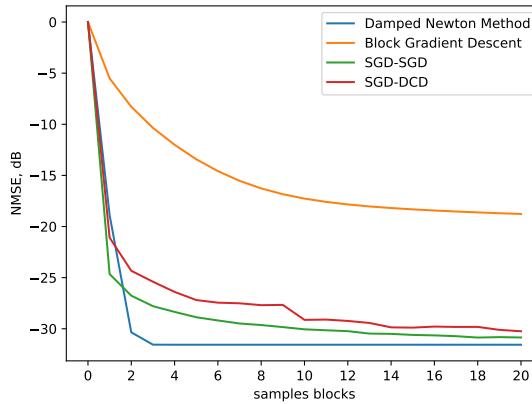


Рис. 37. Кривые адаптации модели Гаммерштейна для случая сигнала передатчика с шириной полосы равной 9 МГц и канала распространения помехи  $path_1$

помехи. Кроме того, данный метод обеспечивает наилучшее подавление близкое к значениям NMSE, полученным в результате работы демпфированного метода Ньютона. При этом алгоритм алгоритм SGD-SGD сходится к опорным значениям NMSE более плавно по сравнению с методом SGD-DCD.

Отметим также, что блочная реализация градиентного спуска адаптирует модель Гаммерштейна существенно медленнее, чем алгоритмы SGD-SGD и SGD-DCD.

На рис. 38–40 изображены спектральные плотности мощности паразитной помехи, сигнала на выходе модели Гаммерштейна после адаптации, а также отклонения выхода модели Гаммерштейна от сигнала помехи. Значения критериев NMSE в данном случае составляют  $-18.8$  dB,  $-30.9$  dB и  $-30.2$  dB для блочного метода градиентного спуска, SGD-SGD и SGD-DCD соответственно.

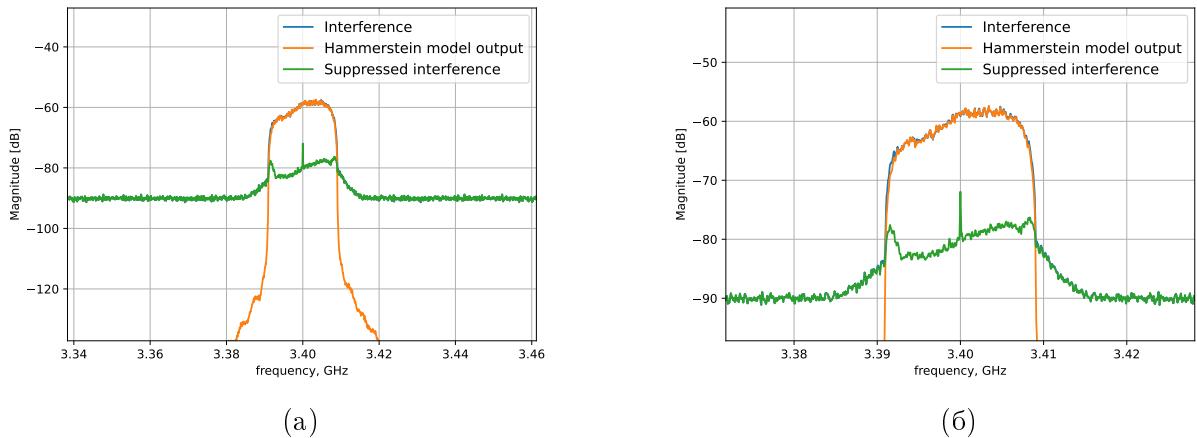


Рис. 38. Спектральная плотность мощности сигнала паразитной помехи после адаптации блочным методом градиентного спуска. Случай  $\{RB50, path_1\}$

Кривые адаптации данных алгоритмов для случая  $\{RB100, path_2\}$  (рис. 31) приведены на рис. 41. Ширина полосы сигнала передатчика определяется 100 ресурсными блоками, 18 МГц (рис. 31а), а канал распространения определяется частотной

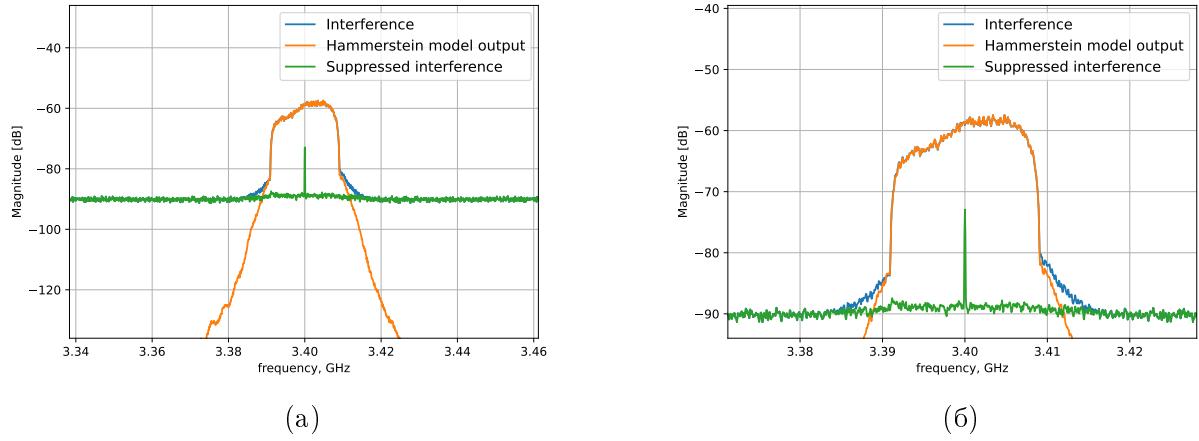


Рис. 39. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом SGD-SGD. Случай {RB50, path<sub>1</sub>}

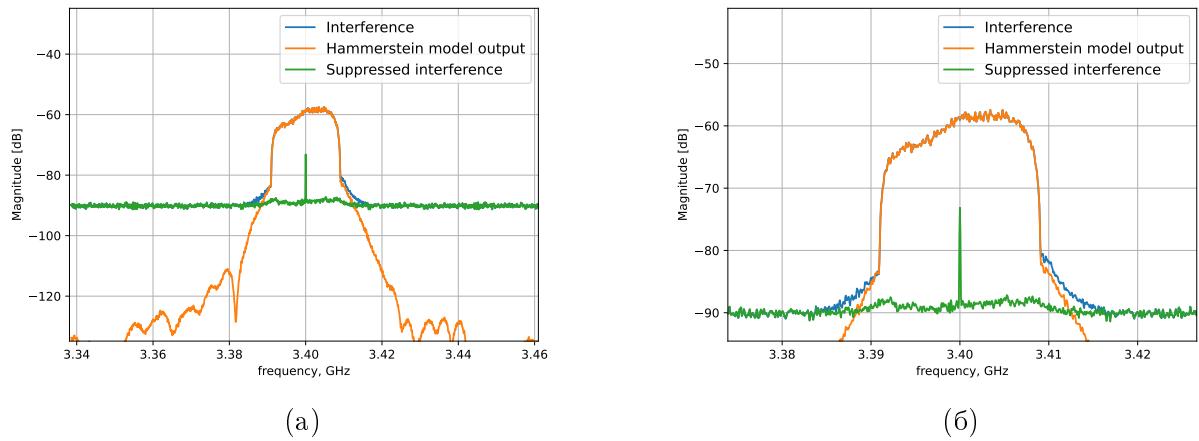


Рис. 40. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом SGD-DCD. Случай {RB50, path<sub>1</sub>}

характеристикой  $path_2$  (рис. 31б).

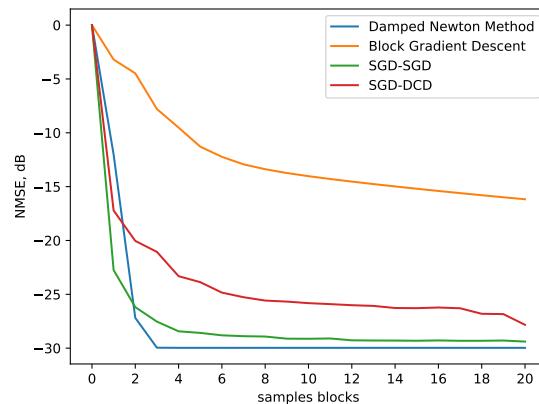


Рис. 41. Кривые адаптации модели Гаммерштейна для случая сигнала передатчика с шириной полосы равной 18 МГц и канала распространения помехи  $path_2$

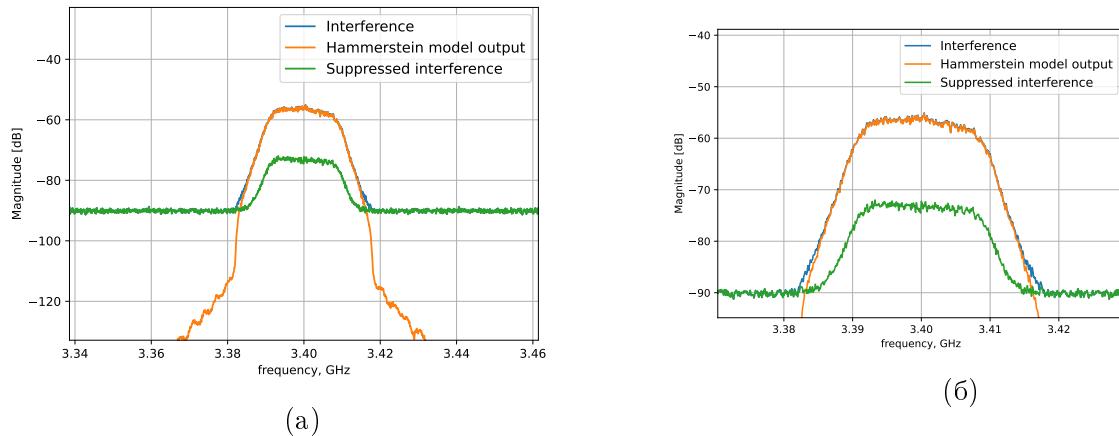


Рис. 42. Спектральная плотность мощности сигнала паразитной помехи после адаптации блочным методом градиентного спуска. Случай {RB100, path<sub>2</sub>}

Рис. 41 отражает тот факт, что наибольшей скоростью сходимости среди градиентных методов обладает алгоритм SGD-SGD для данных условий формирования помехи. Кроме того, данный метод обеспечивает наилучшее подавление близкое к значениям NMSE, полученным в результате работы демпфированного метода Ньютона.

На рис. 42–44 изображены спектральные плотности мощности паразитной помехи, сигнала на выходе модели Гаммерштейна после адаптации, а также отклонения выхода модели Гаммерштейна от сигнала помехи. Значения критериев NMSE в данном случае составляют  $-16.2$  dB,  $-29.4$  dB и  $-27.8$  dB для блочного метода градиентного спуска, SGD-SGD и SGD-DCD соответственно.

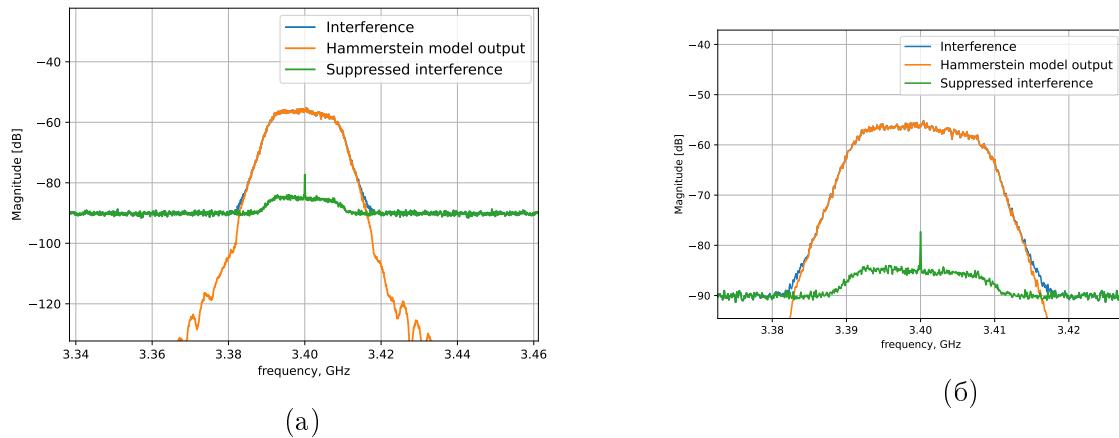


Рис. 43. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом SGD-SGD. Случай  $\{RB100, path_2\}$

Кривые адаптации блочного метода градиентного спуска, SGD-SGD, SGD-DCD, а также спектральные плотности мощности помехи после адаптации данными методами для всех случаев формирования помехи из таблицы ?? представлены в приложении А2.

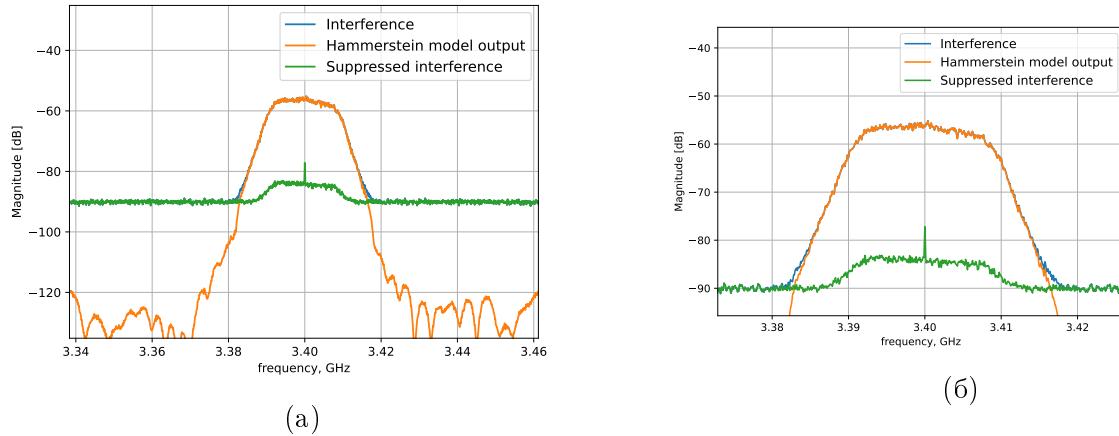


Рис. 44. Спектральная плотность мощности сигнала паразитной помехи после адаптации методом SGD-DCD. Случай {RB100, path<sub>2</sub>}

Таблица 3 отражает значения критерия NMSE dB, полученные в результате адаптации модели Гаммерштейна блочным методом градиентного спуска на сигналах стационарной помехи длительностью 2457600 отсчётов для каждого случая формирования паразитной помехи (таблица. ??).

	RB1	RB25	RB50	RB75	RB100
$path_0$	-21.0	-22.8	-19.9	-16.4	-18.8
$path_1$	-20.0	-21.2	-18.8	-15.0	-16.3
$path_2$	-21.6	-23.9	-20.0	-15.3	-16.2

Таблица 3. Значения критерия NMSE dB, полученные в результате адаптации блочным методом градиентного спуска для каждого случая формирования паразитной помехи

Таблица 4 отражает значения критерия NMSE dB, полученные в результате адаптации модели Гаммерштейна методом SGD-SGD на сигналах стационарной помехи длительностью 2457600 отсчётов для каждого случая формирования паразитной помехи (таблица. ??).

Таблица 5 отражает значения критерия NMSE dB, полученные в результате адаптации модели Гаммерштейна методом SGD-DCD на сигналах стационарной помехи длительностью 2457600 отсчётов для каждого случая формирования паразитной помехи (таблица. ??).

Из таблицы 3 следует, что за время работы алгоритмов адаптации равное 2457600 отсчётов блочная реализация метода градиентного спуска обеспечивает подавление паразитной помехи не меньше, чем на 20 дБ для случаев сигнала передатчика с

	RB1	RB25	RB50	RB75	RB100
$path_0$	-24.1	-32.3	-31.4	-31.4	-29.1
$path_1$	-22.2	-30.5	-30.9	-30.4	-28.6
$path_2$	-25.3	-32.7	-31.5	-31.2	-29.4

Таблица 4. Значения критерия NMSE dB, полученные в результате адаптации методом SGD-SGD для каждого случая формирования паразитной помехи

	RB1	RB25	RB50	RB75	RB100
$path_0$	-24.0	-32.5	-31.2	-27.7	-27.5
$path_1$	-22.0	-30.5	-30.2	-26.1	-26.8
$path_2$	-25.0	-32.8	-30.6	-27.1	-27.8

Таблица 5. Значения критерия NMSE dB, полученные в результате адаптации методом SGD-DCD для каждого случая формирования паразитной помехи

ширины полосы равной 180 кГц и 4.5 МГц для рассмотренных случаев оценки канала распространения помехи.

Для случая сигнала передатчика с шириной полосы 9 МГц подавление паразитной нелинейной помехи составляет как минимум 18 дБ.

При этом для случая сигнала передатчика с шириной полосы 13.5 МГц и 28 МГц подавление помехи не хуже 15 дБ.

Из таблицы 5 следует, что за 2457600 отсчётов стохастический алгоритм SGD-DCD обеспечивает компенсацию помехи для случая сигнала передатчика с шириной полосы 180 кГц как минимум на 21 дБ для всех рассмотренных каналов распространения, что на 1 дБ больше, чем в случае блочного градиентного спуска.

Для случая сигналов передатчика с шириной полосы 4.5 МГц и 9 МГц компенсация помехи не хуже 30 дБ, что больше на 9 дБ и 12 дБ соответственно, чем в случае блочного градиентного спуска.

Компенсация паразитной помехи в случае сигнала передатчика шириной полосы 13.5 и 18 МГц не меньше, чем 26 дБ, что больше на 11 дБ и 10 дБ соответственно, чем в случае блочной реализации градиентного спуска.

Из таблицы 4 следует, что за 2457600 отсчётов стохастический алгоритм SGD-SGD компенсирует паразитную помеху не хуже, чем на 22 дБ для всех оценок канала

распространения помехи, что на 1 дБ лучше, чем в случае алгоритма SGD-DCD. При этом из графиков кривых адаптаций на рис. 33, ??, ?? следует, что скорость сходимости алгоритма SGD-SGD для данного случая выше, чем для SGD-DCD и блочного градиентного спуска.

Подавление нелинейной помехи в случае сигнала передатчика с шириной полосы 4.5 МГц, 9 МГц не меньше, чем 30 дБ, что совпадает с результатом работы алгоритма SGD-DCD за 2457600 отсчётов для всех рассмотренных оценок канала распространения помехи. При этом из графиков кривых адаптаций на рис. ??, ??, ?? для сигнала передатчика RB25 и на рис. ??, 37, ?? для сигнала передатчика RB50 следует, что скорость сходимости алгоритма SGD-SGD для данных случаев выше, чем скорость сходимости SGD-DCD и блочной реализации метода градиентного спуска.

Компенсация нелинейной помехи в случае сигнала передатчика с шириной полосы 13.5 МГц не меньше, чем 30 дБ, что больше, чем в случае алгоритма SGD-DCD на 4 дБ. Кроме того из графиков кривых адаптаций на рис. ??, ??, ?? следует, что скорость сходимости алгоритма SGD-SGD выше, чем скорость сходимости SGD-DCD и блочного метода градиентного спуска.

Для случая сигнала передатчика с шириной полосы 18 МГц подавление паразитной помехи не хуже 28 дБ, что на 1.5 дБ больше, чем в случае алгоритма SGD-DCD. Графики кривых адаптации на рис. ??, ??, 41 отражают тот факт, что скорость сходимости SGD-SGD выше, чем в случае SGD-DCD и блочной реализации градиентного спуска.

Таким образом подавление паразитной помехи методами SGD-SGD, SGD-DCD и блочной реализацией градиентного спуска на сигнале длительностью 2457600 отсчётов отличается от опорных значений компенсации, сформированных демпфированным методом Ньютона меньше, чем на 1 дБ, 4 дБ и 15 дБ соответственно, для рассмотренных случаев формирования помехи, представленных в таблице ??.

Отметим также, что за время работы алгоритмов адаптации равное 2457600 отсчётов среди рассмотренных градиентных методов первого порядка наилучшую компенсацию паразитной помехи, а также наибольшую скорость сходимости обеспечивает стохастический метод SGD-SGD для рассмотренных случаев формирования паразитной нелинейной помехи.

## 2. Сокращение ресурсов при реализации моделей нелинейных искажений засчет канонического тензорного разложения

## 3. Сокращение ресурсов при реализации моделей нелинейных искажений путем оптимизации гиперпараметров модели

### 3.1 Метод Orthogonal Least Squares

### 3.2 Метод Policy Gradient

## 4. Методы экономии ресурсов при реализации нейросетевых моделей компенсации нелинейных искажений

### 4.1 Методы экономии ресурсов при реализации цифровых фильтров

В предыдущих разделах была рассмотрена компенсация паразитной помехи в приёмнике мобильного устройства путём адаптации модели Гаммерштейна в реальном времени. Стохастические алгоритмы позволяют потактово обновлять коэффициенты модели.

Предположим, что импульсный отклик канала распространения паразитной помехи от передатчика к приёмнику меняется незначительно в процессе работы мобильного устройства. В таком случае коэффициенты КИХ-фильтра можно зафиксировать после адаптации, а в реальном времени будут адаптироваться только коэффициенты слоя нелинейности амплитудной характеристики усилителя.

Поскольку в таком случае коэффициенты фильтра могут быть зафиксированы после адаптации, то есть возможность организовать экономию таких ресурсов, как мощность потребляемая цифровой схемой, а также площадь, занимаемая фильтром на кристалле RF-чипсета.

В экспериментах, приведённых в предыдущих разделах, использовались длинные КИХ-фильтры порядка  $M = 38$  и выше, что соответствует аппаратной реализации на кристалле RF-чипсета не менее 39 умножителей.

Реализация умножителей является наиболее трудной с точки зрения используемых ресурсов по сравнению с другими элементами прямой формы реализации КИХ-

фильтра, то есть сумматорами и элементами задержки. Это связано с тем, что мощность, потребляемая умножителем, а также площадь, занимаемая на кристалле, расходует квадратично в зависимости от разрядности умножителя, в то время как потребление этих же ресурсов сумматорами и элементами задержки растёт линейно в зависимости от разрядности.

В связи с этим предлагается использовать метод квантования коэффициентов КИХ-фильтра, позволяющий сократить количество используемых умножителей за счёт выражения одних коэффициентов фильтра через произведение других коэффициентов и степеней двойки.

С точки зрения аппаратной реализации умножители заменяются на сумматоры и битовые сдвиги. Приведём пример из статьи [36]. Пусть имеется КИХ-фильтр, реализованный в прямой форме, со следующими коэффициентами:

$$\mathbf{h} = \begin{pmatrix} h_0 & h_1 & h_2 & h_3 & h_4 \end{pmatrix}^T = \begin{pmatrix} 1 & 2 & 4 & 3 & 6 \end{pmatrix}^T. \quad (4.1)$$

В данном примере коэффициенты  $h_0, h_1, h_3$  точно выражаются через коэффициенты  $h_2, h_4$ , однако в общем случае такой возможности нет. Алгоритм, приведённый в статье [36] формирует импульсную характеристику, соответствующую уменьшенному числу умножителей таким образом, чтобы минимизировать отклонение исходных коэффициентов от тех, что формируются алгоритмом. При этом желаемое количество умножителей задаётся пользователем.

Приведём примеры работы алгоритма для сокращения числа умножителей линейно-фазового Фильтра Нижних Частот, линейно-фазового Полосового Фильтра, а также минимально фазового Фильтра Нижних Частот.

Рассмотрим линейно-фазовый Фильтр Нижних Частот порядка  $N = 138$ . Пусть коэффициенты его импульсной характеристики  $\mathbf{h} = \{h_n\}, n \in \overline{0, N-1}$ . Пусть при этом желаемое количество умножителей  $M = 25$ . Импульсную характеристику сформированного алгоритмом фильтра обозначим  $\mathbf{g} = \{g_n\}, n \in \overline{0, N-1}$ .

На рис. 45 изображена импульсная характеристика исходного фильтра низких частот  $h_n$ , а также отклонение исходных коэффициентов от коэффициентов фильтра, упрощенного с точки зрения аппаратной реализации  $h_n - g_n$ . Это отклонение мало и составляет порядка  $10^{-3}$ .

На рис. 46 изображена частотная характеристика исходного фильтра  $H(\omega)$ , фильтра, построенного на 25 умножителях  $G(\omega)$ , а также разность частотных характеристик  $E(\omega) = H(\omega) - G(\omega)$ .

Из рис. 46 видно, что величина отклонения частотной характеристики фильтра, построенного на 25 умножителях от частотной характеристики исходного фильтра порядка  $N = 138$ , реализованного в прямой форме, составляет -60 dB.

Рассмотрим линейно-фазовый Полосовой Фильтр порядка  $N = 138$ . Пусть коэффициенты его импульсной характеристики  $\mathbf{h} = \{h_n\}, n \in \overline{0, N-1}$ . Пусть при этом желаемое количество умножителей  $M = 25$ . Импульсную характеристику сформи-

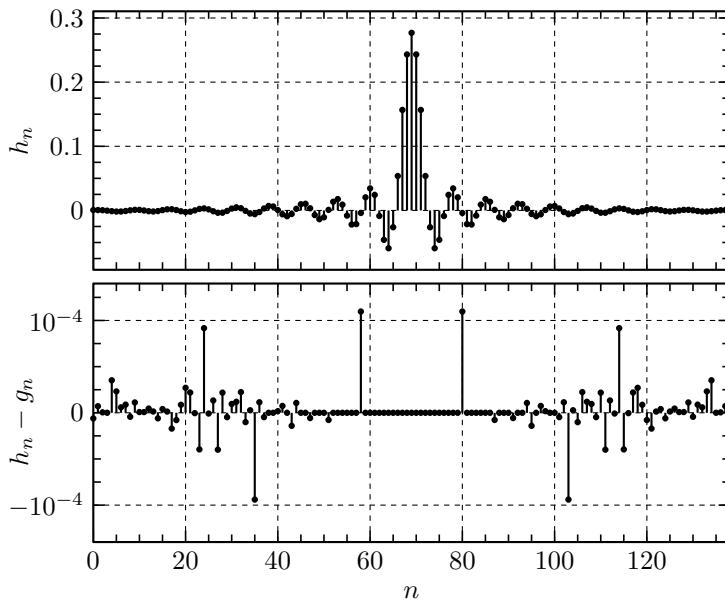


Рис. 45. Линейно-фазовый Фильтр Низких Частот. Исходная импульсная характеристика  $h_n$  и отклонение  $h_n - g_n$

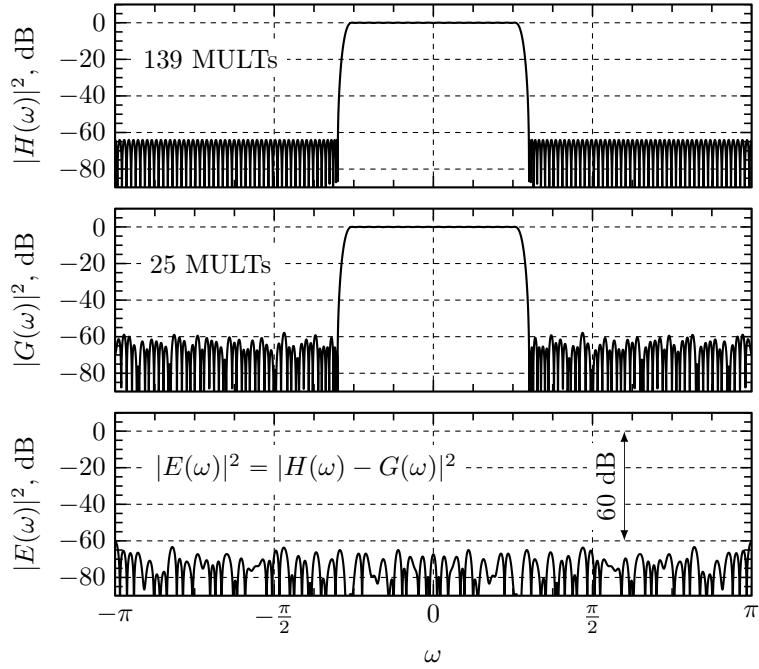


Рис. 46. Линейно-фазовый Фильтр Низких Частот. Исходная частотная характеристика  $H(\omega)$ , частотная характеристика  $G(\omega)$  и отклонение  $E(\omega)$

рованного алгоритмом фильтра также обозначим  $\mathbf{g} = \{g_n\}, n \in \overline{0, N-1}$ .

На рис. 47 изображена импульсная характеристика исходного полосового фильтра  $h_n$ , а также отклонение исходных коэффициентов от коэффициентов фильтра, упрощенного с точки зрения аппаратной реализации  $h_n - g_n$ . Это отклонение мало и составляет порядка  $10^{-5}$ .

На рис. 48 изображена частотная характеристика исходного фильтра  $H(\omega)$ , фильтра, построенного на 25 умножителях  $G(\omega)$ , а также разность частотных характеристи-

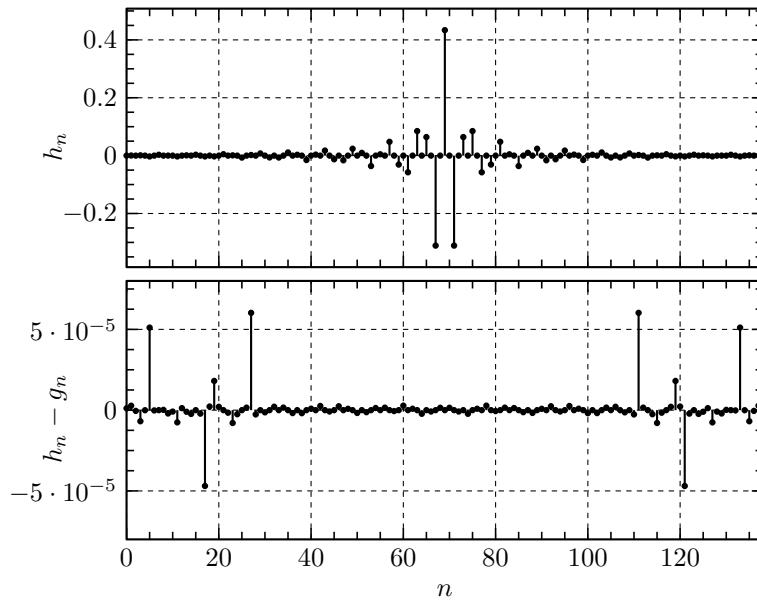


Рис. 47. Линейно-фазовый Полосовой Фильтр. Исходная импульсная характеристика  $h_n$  и отклонение  $h_n - g_n$

стик  $E(\omega) = H(\omega) - G(\omega)$ .

Как видно из рис. 48 величина отклонения частотной характеристики фильтра, построенного на 25 умножителях от частотной характеристики исходного фильтра порядка  $N = 138$ , реализованного в прямой форме, составляет, так же как и для линейно-фазового ФНЧ, -60 dB.

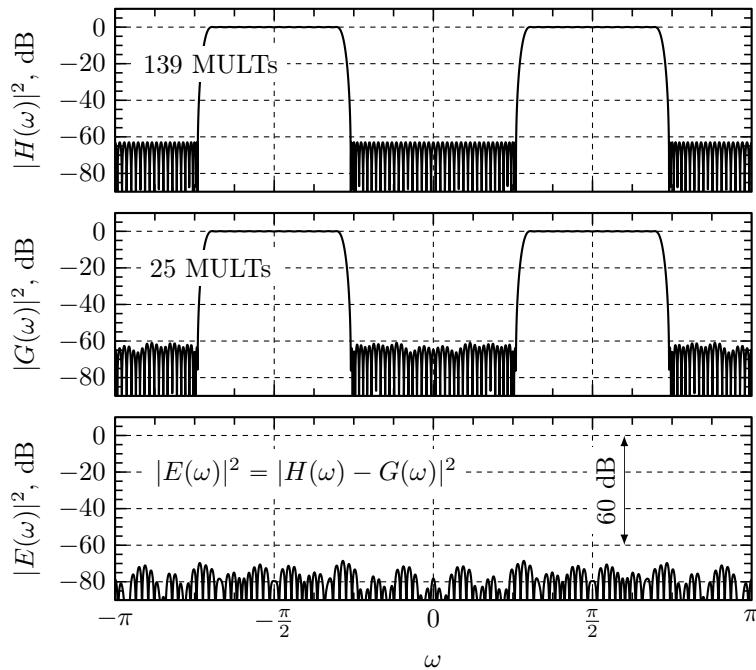


Рис. 48. Линейно-фазовый Полосовой Фильтр. Исходная частотная характеристика  $H(\omega)$ , частотная характеристика  $G(\omega)$  и отклонение  $E(\omega)$

Рассмотрим минимально фазовый Фильтр Низких Частот порядка  $N = 138$ . Пусть коэффициенты его импульсной характеристики  $\mathbf{h} = \{h_n\}, n \in \overline{0, N-1}$ .  $\mathbf{g} =$

$\{g_n\}, n \in \overline{0, N-1}$  – импульсная характеристика сформированного алгоритмом фильтра. Рассмотрим случай уменьшения количества умножителей до  $M = 25$  и до  $M = 70$ .

Данный пример интересен в связи с тем, что в данном случае импульсная характеристика исходного фильтра  $h = \{h_n\}$  не является симметричной.

На рис. 49 изображена импульсная характеристика исходного фильтра низких частот  $h_n$  и отклонение  $h_n - g_n$  исходных коэффициентов от коэффициентов фильтра, построенного на 25 умножителях. Это отклонение составляет порядка  $10^{-4}$ .

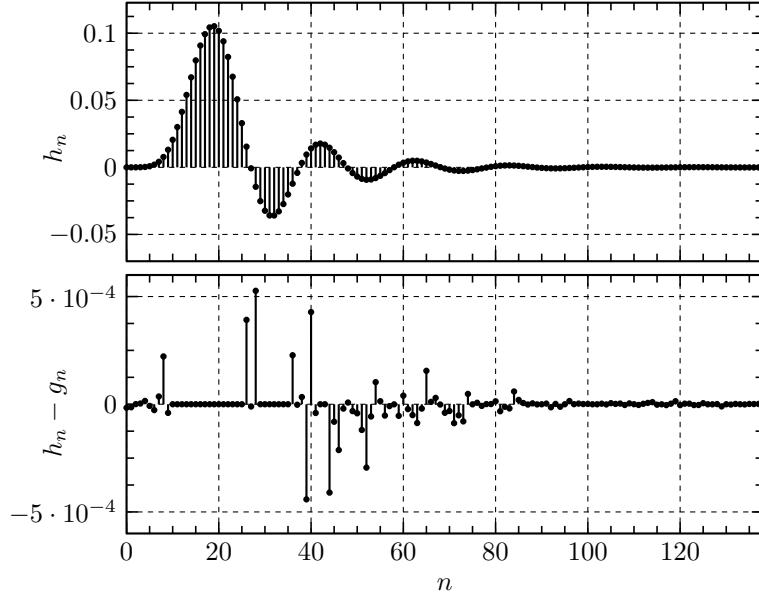


Рис. 49. Минимально фазовый Фильтр Низких Частот. Исходная импульсная характеристика  $h_n$  и отклонение  $h_n - g_n$

На рис. 50 изображена частотная характеристика исходного фильтра  $H(\omega)$ , фильтра, построенного на 25 и 70 умножителях  $G(\omega)$ , а также разность частотных характеристик  $E(\omega) = H(\omega) - G(\omega)$  для обоих случаев.

Из рис. 50 видно, что величина отклонения частотной характеристики фильтра, построенного на 25 умножителях от частотной характеристики исходного фильтра порядка  $N = 138$ , реализованного в прямой форме, составляет -50 dB.

При этом отклонения частотной характеристики фильтра, построенного на 70 умножителях от частотной характеристики исходного фильтра порядка  $N = 138$  составляет -70 dB.

Таким образом, алгоритм, приведённый в статье [36], позволяет существенно сократить мощность потребляемую цифровым КИХ-фильтром, а также занимаемую площадь кристалла RF-чипсета, за счёт замены части умножителей более простыми с точки зрения аппаратной реализации и затрачиваемых ресурсов битовыми сдвигами и дополнительными сумматорами.

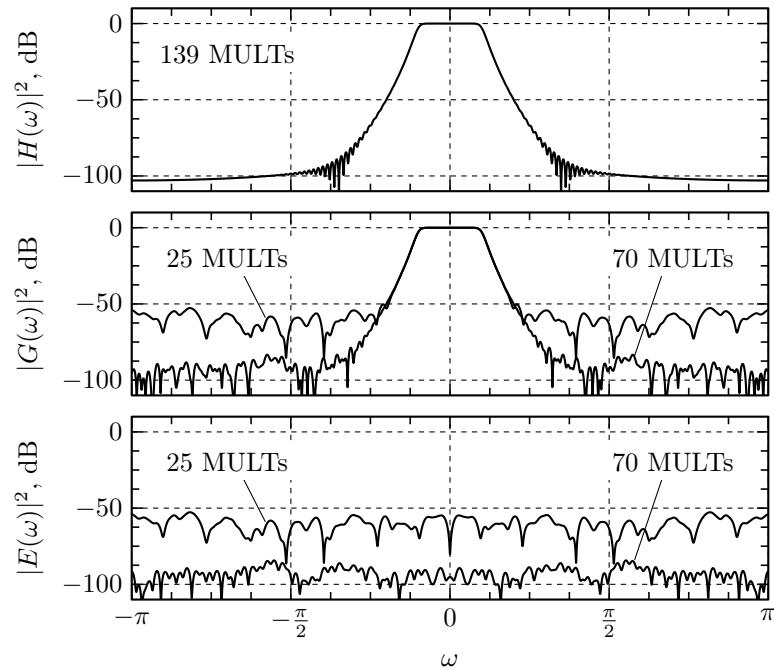


Рис. 50. Минимально фазовый Фильтр Нижних Частот. Исходная частотная характеристика  $H(\omega)$ , частотная характеристика  $G(\omega)$  и отклонение  $E(\omega)$

#### 4.2 Сокращение ресурсов при реализации свёрточных сетей для задачи компенсации нелинейных искажений

# Заключение

# Список литературы

- [1] Fa-Long Luo. *Digital Front-End in Wireless Communications and Broadcasting. Circuits and Signal Processing*. 1-е изд. New York: Cambridge University Press, 2011.
- [2] 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception; Standard. 650 Route des Lucioles - Sophia Antipolis Valbonne - FRANCE: European Telecommunications Standards Institute, 2008.
- [3] Ali Soltani Tehrani. "Behavioral modeling of wireless transmitters for distortion mitigation". Thesis for the degree of doctor of philosophy. Chalmers University, 2012.
- [4] 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception; Standard. 650 Route des Lucioles - Sophia Antipolis Valbonne - FRANCE: European Telecommunications Standards Institute, 2015.
- [5] П. М. Грант К. Ф. Н. Коуэн. *Адаптивные фильтры*. 1-е изд. Москва: "Мир", 1988.
- [6] Simon Haykin. *Adaptive filters theory*. 5-е изд. Edinburgh Gate, Harlow, Essex CM20 2JE, England: Pearson Education, 2014.
- [7] Lieven Vandenberghe Stephen Boyd. *Convex Optimization*. 7-е изд. The Edinburgh Building, Cambridge, CB2 8RU, UK: Cambridge University Press, 2009.
- [8] Бернард Склар. *Цифровая Связь. Теоретические Основы и Практическое Применение*. 2-е изд. Москва, Санкт-Петербург, Киев: Издательские дом: "Вильямс", 2003.
- [9] Б.Т. Поляк. *Введение в Оптимизацию*. 1-е изд. 117071, Москва, В-71, Ленинский проспект, 15: Издательство "Наука". Главная редакция физико-математической литературы, 2002.
- [10] Are Hjorungnes. *Complex-Valued Matrix Derivatives. With Applications in Signal Processing and Communications*. 1-е изд. The Edinburgh Building, Cambridge, CB2 8RU, UK: Cambridge University Press, 2011.

- [11] B. T. Polyak. *Some methods of speeding up the convergence of iteration methods*. USSR Computational Mathematics and Mathematical Physics (in Russian). 1964.
- [12] Yurii E. Nesterov. “A method for solving a convex programming problem with convergence rate  $O(1/k^2)$ ”. *Doklady Akademii Nauk SSSR* **269** 3 (1983), c. 543–547.
- [13] Diederik P. Kingma, Jimmy Ba. “Adam: A Method for Stochastic Optimization”. *arXiv preprint arXiv:1412.6980* (2014).
- [14] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. 1-е изд. 101 Philip Drive, Assinippi Park, Norwell, Massachusetts 02061 USA: Kluwer Academic Publishers, 2004.
- [15] David S. Watkins. *Fundamentals of Matrix Computation*. 2-е изд. New York: A John Wiley и Sons, Inc., 2002.
- [16] R. C. de Lamare. “Low-Rank Signal Processing: Design, Algorithms for Reduced-Rank Signal Processing”. *arXiv preprint arXiv:1508.00636* (2015).
- [17] Дегтярев Александр, Бахурин Сергей. “Компенсация собственных нелинейных помех на основе смешанного метода Ньютона”. *Компьютерные исследования и моделирование* (2024), c. 1579–1592. DOI: 10.20537/2076-7633-2024-16-7-1579-1592.
- [18] Ali Soltani Tehrani. “DCD Algorithm: Architectures, FPGA Implementations and Applications”. Thesis for the degree of doctor of philosophy. Communications Research Group, Department of Electronics, University of York, 2008.
- [19] R. Fletcher. *Practical Methods of Optimization*. 2-е изд. The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom: John Wiley и Sons Ltd, 1987.
- [20] J.M. Borwein J. Barzilai. “Two-Point Step Size Gradient Methods”. *IMA Journal of Numerical Analysis* 8 (1988), c. 141–148. DOI: 10.1093/imanum/8.1.141.
- [21] М. Райт Ф. Гилл У. Мюррей. *Практическая Оптимизация*. 1-е изд. Москва: Издательство "Мир", 1985.
- [22] Charles F. Van Loan Gene H. Golub. *Matrix Computations*. 3-е изд. 2715 North Charles Street, Baltimore, Maryland 21218-4319, The Johns Hopkins Press Ltd., London: The Johns Hopkins University Press, 1996.
- [23] R. Fletcher. “A New Approach To Variable Metric Algorithms”. *The Computer Journal* **13** 3 (1969), c. 317–322. DOI: 10.1093/comjnl/13.3.317.
- [24] Dong C. Liu, Jorge Nocedal. “On The Limited Memory BFGS Method for Large Scale Optimization”. *Mathematical Programming* **45** 3 (1989), c. 503–528. DOI: 10.1007/BF01589116.

- [25] Ричард Лайонс. *Цифровая обработка сигналов*. 2-е изд. Москва: "БИНОМ", 2006.
- [26] John R. Buck Alan V. Oppenheim Ronald W. Schafer. *Discrete-Time Signal Processing*. 2-е изд. Upper Saddle River, New Jersey 07458: Prentice-Hall, 1998.
- [27] D. Ronnow M. Isaksson D. Wisell. "A Comparative Analysis of Behavioral Models for RF Power Amplifiers". *IEEE Transactions on Microwave Theory and Techniques* **54** 1 (2006), c. 348—359. DOI: 10.1109/TMTT.2005.860500.
- [28] Mohamed Helaoui Fadhel M. Ghannouchi Oualid Hammi. *Behavioral Modelling and Predistortion of Wideband Wireless Transmitters*. 1-е изд. The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom: John Wiley и Sons Ltd, 2015.
- [29] Ф. Лёш Е. Янке Ф. Эмде. *Специальные функции (Формулы, графики, таблицы)*. 6-е изд. Москва: Наука, 1964.
- [30] K. Azadet A. Molina K. Rajamani. "Digital Predistortion Using Lookup Tables With Linear Interpolation and Extrapolation: Direct Least Squares Coefficient Adaptation". *IEEE Transactions on Microwave Theory and Techniques* **65** 3 (2017), c. 980—987. DOI: 10.1109/TMTT.2016.2627562.
- [31] *User Equipment (UE) conformance specification; Radio transmission and reception; Part 2: Range 2 Standalone*; Standard. 650 Route des Lucioles - Sophia Antipolis Cedex - FRANCE: European Telecommunications Standards Institute, 2019.
- [32] National Communications Commission, Taiwan. *Technical Specifications for New Radio Base Station Radio Frequency Equipment of Mobile Broadband Business*. Standard IS2051. Version 1.0. National Communications Commission (NCC), Taiwan, 2020.
- [33] Hang Yin, Anding Zhu. "Adaptive Kernel Function Sharing for Digital Predistortion of RF Power Amplifiers with Dynamic Resource Block Allocation". *2024 IEEE/MTT-S International Microwave Symposium - IMS 2024*. 2024, c. 645—648. DOI: 10.1109/IMS40175.2024.10600215.
- [34] Sergey Bakhurin и др. "Optimization in Complex Spaces with the Mixed Newton Method". *Journal of Global Optimization* **88** 3 (2024), c. 637—668. DOI: 10.1007/s10898-023-01355-z.
- [35] Boris T. Polyak. "Some Methods of Speeding up the Convergence of Iteration Methods". *USSR Computational Mathematics and Mathematical Physics* **4** 5 (1964), c. 1—17. DOI: 10.1016/0041-5553(64)90137-5. URL: <https://www.sciencedirect.com/science/article/pii/0041555364901375> (дата обр. 01.10.2024).
- [36] S. Bakhurin A. Degtyarev K. Saifullin. "High Order FIR Filter Hardware Implementation Complexity Reduction". *2022 24th International Conference on Digital Signal Processing and its Applications (DSPA)*. Accepted for publication. IEEE. 2022.