



ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 102 за 2013 г.



Павловский В.Е., Кирков А.Ю.

Тональная мультимодальная
акустическая коммуникация
роботов

Рекомендуемая форма библиографической ссылки: Павловский В.Е., Кирков А.Ю.
Тональная мультимодальная акустическая коммуникация роботов // Препринты ИПМ
им. М.В.Келдыша. 2013. № 102. 32 с. URL: <http://library.keldysh.ru/preprint.asp?id=2013-102>

Ордена Ленина
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Российской академии наук

В.Е.Павловский, А.Ю.Кирков

**ТОНАЛЬНАЯ МУЛЬТИЧАСТОТНАЯ АКУСТИЧЕСКАЯ
КОММУНИКАЦИЯ РОБОТОВ**

Москва, 2013 г.

УДК 004.896::534.4

В.Е. Павловский, А.Ю.Кирков

Институт прикладной математики им. М.В.Келдыша РАН (ИПМ)

ТОНАЛЬНАЯ МУЛЬТИЧАСТОТНАЯ АКУСТИЧЕСКАЯ КОММУНИКАЦИЯ РОБОТОВ

АННОТАЦИЯ

Рассматривается задача коммуникации роботов на основе языка мультичастотных акустических сигналов. В работе приведена формальная модель языка, в котором каждому символу языка соответствует последовательность акустических мультичастотных сигналов, и рассмотрен алгоритм распознавания таких характерных звуков.

Ключевые слова и выражения: акустическая коммуникация, разговор роботов, распознавание речи

V.E.Pavlovsky, A.J.Kirkov

Keldysh Institute of Applied Mathematics of RAS (KIAM)

TONE MULTI-FREQUENCY ACOUSTIC COMMUNICATION OF ROBOTS

SUMMARY

A task of communication of robots on the basis of multi-frequency acoustic signals is the focus of this work. In the work a language model is presented, where each letter is associated with a pattern of multi-frequency tones, supplied with an algorithm of identification of these tones.

Key words and phrases: acoustic communication, conversation of robots, speech recognition

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ.	3
2. СТРУКТУРА ЯЗЫКА РОБОТОВ.....	3
3. СЛОВАРЬ И ШАБЛОН СЛОВА	5
4. АЛГОРИТМ РАСПОЗНАВАНИЯ ШАБЛОНА СЛОВА	6
4.1. Исходные понятия и схема распознавания.....	7
4.2. Вспомогательные функции и множества.....	14
4.3. Алгоритм подавления эхо.	15
4.4. Алгоритм подавления шума.....	17
4.5. Алгоритм корректировки смещения	20
4.6. Пошаговое описание алгоритма	23
5. АЛГОРИТМ ПОИСКА СЛОВА В СЛОВАРЕ ПО ШАБЛОНУ.....	25
6. ПРОГРАММНЫЙ КАРКАС СИСТЕМЫ КОММУНИКАЦИИ.	30
7. ЭКСПЕРИМЕНТАЛЬНЫЙ АНАЛИЗ РАБОТЫ СИСТЕМЫ	30
8. ЗАКЛЮЧЕНИЕ.	32
ЛИТЕРАТУРА.....	32

1. ВВЕДЕНИЕ. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Настоящая работа является продолжением работ [1-3] по созданию системы тональной акустической коммуникации роботов. В этих работах коммуникация роботов основана на базе языка одночастотных сигналов. Система показала свою работоспособность в многочисленных экспериментах. Проведенные опыты подтвердили устойчивую работу алгоритмов распознавания в условиях тишины на типичном расстоянии порядка двух метров между роботами.

Целью настоящей работы является создание усовершенствованной версии языка для системы тональной акустической коммуникации роботов. Базовый язык, представленный в работе, строится на более сложных, многочастотных сигналах. Целью усложнения языка является более устойчивая работа системы в условиях посторонних шумов. Более сложные конструкции языка должны уменьшить процент ложного распознавания.

В работе также рассмотрен словарь роботов. Словарь представляет собой некоторую базу данных слов, которыми пользуются роботы. Цель введения словаря заключается в исправлении ошибок, возникших в результате коммуникации роботов.

В работе представлен алгоритм распознавания конструкций указанного языка. Реализована программная среда для визуализации и анализа работы построенного алгоритма. Проведены эксперименты на основе созданных компьютерной и аппаратной моделей для анализа работоспособности построенных алгоритмов.

2. СТРУКТУРА ЯЗЫКА РОБОТОВ

В отличие от одночастотной системы, фрагмент в настоящей работе представляет собой не одночастотный, а многочастотный сигнал, определенной длительности и амплитуды. Будем рассматривать фрагменты, состоящие из одного и того же количества частот.

Введем некоторый параметр N_L , который будет определять количество частот каждого фрагмента языка. Параметр N_L будем называть частотностью языка роботов.

Рассмотрим некоторое подмножество множества из N_L различных неупорядоченных положительных чисел, которое будем называть множеством частот.

$$Frq \subset \{(frq_1, frq_2, \dots, frq_{N_L}) \mid frq_i \in N, i = \overline{1, N_L}, frq_i \neq frq_j \forall i, j = \overline{1, N_L}, i \neq j\}.$$

Каждый элемент этого множества будет представлять собой некоторый набор частот. Без ограничения общности, будем считать $frq_1 < frq_2 < \dots < frq_{N_L}$.

Фрагмент будет представлять элемент следующего множества:

$$Frg = \{(amp, frq, dur) \mid amp \in N, dur \in N, frq \in Frq\}.$$

Отметим еще раз, что в отличие от одночастотной системы фрагмент содержит некоторый набор частот, который характеризуется элементом множества Frq . Поскольку для рассматриваемого языка фрагменты играют роль неделимых частиц, атомов языка, систему будем называть многочастотной.

Для получения k -ой частоты фрагмента $frg = (amp, frq, dur)$ примем следующее обозначение $frg[k] = frq_k, frg = (frq_1, \dots, frq_k, \dots, frq_{N_f})$, для получения множества frq будем использовать следующую запись $frg[] = frq$. Длительность фрагмента будем обозначать $|frg| = dur$.

Любое конечное подмножество

$$F \subset \{(frg^1, frg^2, \dots, frg^n) | n \in N, frg^i \in \overline{Frg}, i = \overline{1, n}\}$$

множества конечных упорядоченных наборов элементов из Frg будем называть множеством фонем.

Таким образом, любая фонема представляет собой последовательность многочастотных сигналов. Пример аудиосигнала, соответствующего фонеме двухчастотного языка роботов, графически изображен на 1.

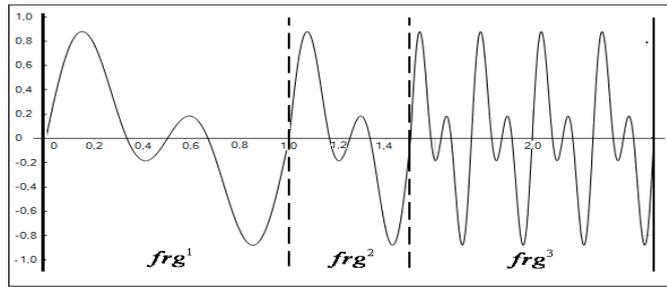


Рис.1. Пример аудиосигнала, соответствующего фонеме двухчастотного языка роботов

Длительностью фонемы будем считать сумму длительностей всех входящих в неё фрагментов. Длительность фонемы $f = (frg^1, \dots, frg^n)$ будем обозначать $|f| = \sum_{i=1}^n |frg^i|$. Для получения k -го фрагмента frg^k фонемы $f = (frg^1, \dots, frg^k, \dots, frg^n)$ будем использовать обозначение $f[k] = frg^k$.

Ограничения, наложенные на множество фонем для одночастотной системы, будут также иметь силу. А именно, множество фонем должно удовлетворять условиям ограничения вложенности и ограничения объединения фонем. Кроме этого, будем считать, что все фонемы имеют одинаковую длительность. Введем параметр D_f , который будем называть общей длительностью фонемы:

$$|f| = D_f, \forall f \in F.$$

Введем также множество используемых фрагментов

$$Frg_u = \{frg \in Frg | \exists f = (frg^1, \dots, frg^k, \dots, frg^n), f \in F, frg^k = frg\}$$

и множество используемых частот

$$Frq_u = \{frq \in frg[], frg \in Frg_u\}.$$

Аналогично одночастотной системе рассмотрим алфавит B , соответствие R между алфавитом и множеством фонем и пробел $S^* = \{S, dur_s\}$. Набор $L = (B, F, R, S^*)$ алфавита, множества фонем, соответствия между ними и пробела будем называть языком роботов.

Любой упорядоченный набор букв из множества B будем называть словом. Множество слов будем обозначать W :

$$W = \{(b_1, b_2, \dots, b_n), b_i \in B, i = \overline{1, n}, n \in N\}.$$

Длиной слова $w \in W$ будем называть количество букв в этом слове. Введем обозначение $|w|$ для длины слова.

Будем называть слово $w' \in W$ подсловом слова $w = (b_1, \dots, b_n) \in W$, если существуют такие индексы $i, j, 1 \leq i \leq j \leq n$, что $w' = (b_i, \dots, b_j)$. Введем обозначение $w' \subset w$. Подслово $w' = (b_i, \dots, b_j) \in W$ будем называть началом слова $w = (b_1, \dots, b_n)$, если $i = 1$, концом этого слова, если $j = n$.

3. СЛОВАРЬ И ШАБЛОН СЛОВА

Множество слов W бесконечно. Однако если провести аналогию с человеческим языком, набор слов, которые использует человек в своей речи, ограничен. Например, характерный словарный запас русского языка насчитывает примерно 200 тысяч слов, из которых 3000 являются наиболее употребительными.

Введем ограничения для словарного запаса языка роботов. Для этого рассмотрим некоторое конечное подмножество множества слов

$$D \subset W, |D| < \infty.$$

Множество D будем называть словарем роботов. Будем считать, что роботы могут проигрывать слова исключительно из словаря. В настоящей работе будем предполагать, что все роботы пользуются одним и тем же словарем.

Поиск слова в словаре будем осуществлять по шаблону. Шаблоном слова будем называть произвольный упорядоченный набор подмножеств множества B , в котором первый и последний элемент не являются пустыми множествами, т.е. элемент множества

$$W_T^B = \{(B_1, B_2, \dots, B_n), B_i \subset B, B_1, B_n \neq \emptyset, i = \overline{1, n}, n \in N\}.$$

Длиной шаблона слова $w_i^b \in W_T^B$ будем называть количество подмножеств B_i в этом шаблоне. Длину шаблона слова w_i^b будем обозначать $|w_i^b|$. Количество непустых множеств B_i шаблона w_i^b будем обозначать $\|w_i^b\|$. Множества $B_i, i = \overline{1, n}$ будем называть буквенными множествами шаблона w_i^b .

Шаблон слова $w_i^{b'}$ будем называть составной частью шаблона $w_i^b = (B_1, \dots, B_n)$, если $w_i^{b'} = (B_i, \dots, B_j)$ для некоторых $i, j, 1 \leq i < j \leq n$. Введем обозначение $w_i^{b'} \subset w_i^b$ для составной части шаблона.

Рассмотрим также множество, элемент которого будет представлять упорядоченный набор подмножеств множества фонем F . При этом первый и последний элемент этого набора должны быть не пустыми множествами.

$$W_T^F = \{(F_1, F_2, \dots, F_n), F_i \subset F, F_1, F_n \neq \emptyset, i = \overline{1, n}, n \in N\}.$$

Элемент этого множества будем называть фонемным шаблоном слова.

В силу того, что язык роботов $L = (B, F, R, S^*)$ задает однозначное соответствие R между алфавитом B и множеством фонем F , каждому элементу множества W_T^B можно поставить в соответствие элемент множества W_T^F и наоборот. Обозначим это соответствие R_T^W .

Определенный выше словарь роботов D , а также множества W_T^B , W_T^F и соответствие между ними R_T^W будут использоваться в алгоритме распознавания.

4. АЛГОРИТМ РАСПОЗНАВАНИЯ ШАБЛОНА СЛОВА

В работах [1-3] был описан алгоритм поиска фонем в полученном роботом аудиосигнале для одночастотного языка роботов. Цель этого алгоритма заключалась в поиске всех отрезков времени, на которых была проиграна некоторая фонема. Отметим, что указанный алгоритм не использовал тот факт, что фонемы могут проигрываться не обособленно, а в составе некоторого слова. Однако, учитывая это свойство языка, можно из аудиофразы извлечь дополнительную информацию, которую потом использовать в алгоритме распознавания.

Согласно введенным в настоящей работе ограничениям, роботы могут говорить слова только из словаря роботов, что сильно ограничивает варианты слов для распознавания. Данный факт тоже будет полезно использовать в алгоритме распознавания.

В этом разделе рассматривается алгоритм, целью которого является поиск слов из словаря роботов. Алгоритм состоит из двух частей. Первая часть заключается в поиске отрезков времени, на которых было проиграно некоторое слово, и в определении шаблона, соответствующего этому слову. Задача второй части алгоритма заключается в поиске слов из словаря роботов по полученному шаблону.

Такое разделение алгоритма распознавания обусловлено тем, что в процессе распознавания некоторые буквы слова могут быть не распознаны или, наоборот, вместо одной буквы в результате процесса распознавания могут быть получены два или более символа. Например, при проигрывании слова «привет» на выходе алгоритма можно получить слово «првет» без буквы «и» в середине слова. При этом между интервалами, на которых были распознаны буквы «р» и «в», будет «промежуток» длиной, равной длительности фонемы. Совершенно очевидно, что на этом месте была пропущена какая-то буква. И слово «првет» на самом деле является

некоторым словом «пр?вет», где знаком «?» помечен некоторый символ языка роботов. С учетом использования словаря из подобного шаблона нетрудно получить исходное слово.

В начале раздела вводится ряд базовых понятий и описывается общая схема работы алгоритма. Далее рассматривается ряд вспомогательных алгоритмов, и в конце раздела приводится пошаговое описание алгоритма распознавания шаблона слова.

4.1. Исходные понятия и схема распознавания

Аудиосигнал будем рассматривать в качестве функции $s(t) \in L_2[t_1, t_2]$, заданной на некотором отрезке времени $[t_1, t_2]$. Задача алгоритма заключается в определении всех интервалов $[t', t''] \subset [t_1, t_2]$, на которых было проиграно некоторое слово, и шаблона $w_t^b \subset W_T^B$, соответствующего этому слову.

Поскольку между множествами W_T^B и W_T^F существует однозначное соответствие, обусловленное языком роботом, можно перефразировать задачу следующим образом. Определить все интервалы $[t', t''] \subset [t_1, t_2]$, на которых некоторое слово входит в аудиосигнал, и фонемный шаблон $w_t^f \subset W_T^F$, соответствующий этому слову.

Рассмотрим критерии, согласно которым будем определять вхождение слова, соответствующего некоторому шаблону $w_t^f \subset W_T^F$, в аудиосигнал на некотором интервале времени $[t', t''] \subset [t_1, t_2]$. Для этого введем ряд дополнительных определений и функций.

Аналогично алгоритму распознавания для одночастотной системы, будем рассматривать разбиение T отрезка точками $t^0, t^1, \dots, t^{s-1}, t^s$, $t^i = t_1 + i \cdot \tau, i = \overline{0, s-1}$, $t^s = t_2, t^s - t^{s-1} \leq \tau$ на операционные интервалы $\Delta^i = [t^{i-1}, t^i], i = \overline{1, s}$.

Аналогичным образом на каждом операционном интервале $\Delta^i, i = \overline{1, s}$ будем рассматривать функции амплитуды фонемы $a_i(f)$, текущего фрагмента фонемы $frg_c^i(f)$, функцию смещения $\delta_i(f)$, множество выделенных фонем S_i , функцию текущего времени распознавания фонемы $T_i^c(f)$.

Введем ряд дополнительных параметров.

Параметр $I_i^w, i = \overline{1, s}$ будет характеризовать, проигрывается ли на интервале Δ^i некоторое слово. Параметр будет принимать значения из множества $\{0, 1\}$. Значение I_i^w будет равно единице на операционных интервалах, следующих за первой распознанной фонемой в слове, и до окончания этого слова. Если I_i^w будет равен единице, будем говорить, что система находится в режиме распознавания слова. Положим $I_1^w = 0$.

Рассмотрим параметр $T_i^W, i = \overline{0, s-1}$, который будем называть текущим временем распознавания слова. Параметр T_i^W будет равен длине того промежутка времени, в течение которого непрерывно проигрывается некоторое слово до момента времени t^i . Ниже приведена формула для определения T_i^W .

$$T_i^W = \begin{cases} 0, I_{i+1}^W = 0, i \geq 1 \\ D_f, I_{i+1}^W = 1, I_i^W = 0, i \geq 1 \\ T_i^W = T_{i-1}^W + \tau, I_{i+1}^W = I_i^W = 1, i \geq 1 \\ 0, i \in \{0\} \cup \{s\} \end{cases} \quad (4.1)$$

Иными словами, текущее время распознавания слова к моменту времени t^i будет равно нулю в том случае, если на операционном интервале Δt^{i+1} система не находится в режиме распознавания слова. В момент перехода в режим распознавания слова T_i^W будет равно общей длительности фонемы, что соответствует тому, что система переходит в режим распознавания слова после распознавания первой фонемы этого слова. В режиме распознавания слова параметр T_i^W на каждом следующем операционном интервале будет увеличиваться на длину этого отрезка.

В режиме распознавания слова интервалы, на которых могут быть распознаны фонемы, определены точно. Эти интервалы следуют за отрезком времени, на котором была распознана первая фонема слова, и равны общей длительности фонемы D_f . Такие отрезки времени будем называть фонемными интервалами слова.

Поскольку в режиме распознавания слова фонемы могут быть проиграны только на фонемных интервалах слова, если в начале такого интервала фонема не входила в аудиосигнал, то на последующих отрезках разбиения этого интервала она также не может входить в аудиосигнал.

Вследствие этого рассмотрим множество $P_i^W \subset F, i = \overline{1, s}$, состоящее из набора фонем, которые потенциально могут входить в аудиосигнал на операционном интервале Δt^i . Множество P_i^W будем называть потенциальным множеством фонем на операционном интервале Δt^i .

$$P_i^W = \begin{cases} S_{i-1}, I_i^W = 1, \text{mod}_{D_f}(T_{i-1}^W) \neq 0, i \geq 2 \\ F, \text{mod}_{D_f}(T_{i-1}^W) = 0 \\ F, I_i^W = 0 \end{cases} \quad (4.2)$$

Таким образом, в режиме распознавания слова потенциальными фонемами будут те фонемы, которые были выделенными на предыдущем операционном интервале. В том случае, если операционный интервал является первым для фонемного интервала слова, потенциальными фонемами считается все множество фонем. На операционных интервалах, во время которых система не находится в режиме распознавания слова, множество P_i^W будет совпадать с множеством всех фонем F .

Определим критерии вхождения фонемы в аудиосигнал на каждом операционном интервале. Для этого введем параметр $m_i, i = \overline{1, s}$, который будем называть относительным порогом на операционном интервале Δt^i .

Кроме этого, на каждом операционном интервале $\Delta t^i, i = \overline{1, s}$ рассмотрим множества E_i и N_i . Эти множества будут состоять из фонем, вхождение которых в аудиосигнал будет являться эхом и шумом соответственно.

Используя введенные множества и параметры, рассмотрим определение выделенной фонемы.

Фонему $f \in F$ будем называть выделенной на операционном интервале $\Delta t^i, i = \overline{1, s}$ по параметрам $a_{\min} \in N, 0 < \sigma < 1$, если одновременно выполняются следующие условия:

1. $f \notin E_i \cup N_i$
2. $f \in P_i^W$
3. $a_i(f) \geq a_{\min}$
4. $a_i(f) \geq \sigma \cdot m_i$

Иными словами, фонему будем считать выделенной на операционном интервале, если она не является эхом или шумом на этом интервале, является потенциальной фонемой и её амплитуда выше двух порогов: абсолютного и относительного. Целью введения этих порогов является фильтрация фонем. Абсолютный порог отсекает все “тихие” фонемы, тогда как относительный порог выделяет из «громких» «самые громкие».

Вышеуказанные критерии будут являться основными условиями вхождения фонемы в аудиосигнал на операционном интервале. Множество выделенных фонем на отрезке $\Delta t^i, i = \overline{1, s}$ будем обозначать S_i .

На каждом операционном интервале рассмотрим параметр $m_i^N, i = \overline{1, s}$, который будем называть относительным порогом с шумом. Используя этот параметр, введем следующее определение.

Фонему $f \in F$ будем называть выделенной с шумом на операционном интервале $\Delta t^i, i = \overline{1, s}$ по параметрам $a_{\min} \in N, 0 < \sigma < 1$, если одновременно выполняются следующие условия:

1. $f \notin E_i$
 2. $f \in P_i^W$
 3. $a_i(f) \geq a_{\min}$
 4. $a_i(f) \geq \sigma \cdot m_i^N$
- (4.3)

Данное определение отличается от понятия выделенной фонемы тем, что фонема может быть выделенной и содержаться во множестве N_i и в качестве относительного порога используется новый параметр.

Определим текущее время распознавания фонемы $f \in F$ к моменту времени $t^i \in T, i = \overline{0, s}$.

$$T_i^c(f) = \begin{cases} T_{i-1}^c(f) + \tau, & f \in S_i, I_{i-1}^W = I_i^W \text{ или } I_i^W = 0, i \geq 1 \\ 0, & \text{в остальных случаях} \end{cases} \quad (4.4)$$

Текущее время распознавания фонемы к моменту времени t^i равно длине отрезка времени до t^i , на котором фонема непрерывно входила в аудиосигнал, т.е. была выделенной на каждом операционном интервале, входящем в этот отрезок.

В отличие от описания этой функции для одночастотной системы, текущее время распознавания каждой фонемы обнуляется при переходе системы в режим распознавания слова. Данное условие характеризует то, что фонемы не могут проигрываться одновременно. Соответственно, после проигрывания первой фонемы слова никакая другая фонема не может находиться в стадии проигрывания.

Используя понятие текущего времени распознавания фонемы, параметры I_i^W и T_i^W , определим текущий фрагмент фонемы. Текущий фрагмент фонемы характеризует, какой фрагмент этой фонемы звучит на операционном интервале, если сама фонема на нём проигрывается.

Для определения текущего фрагмента фонемы рассмотрим функцию, которая равна остатку от деления текущего времени распознавания фонемы $T_i^c(f)$ на общую длительность фонемы D_f , если $I_{i+1}^W = 0$, и остатку от деления текущего времени распознавания слова T_i^W на D_f в противном случае.

$$\tilde{T}_i^c(f) = \begin{cases} \text{mod}_{D_f}(T_i^c(f)), & I_{i+1}^W = 0 \\ \text{mod}_{D_f}(T_i^W), & I_{i+1}^W = 1 \end{cases}$$

Текущим фрагментом фонемы будем называть первый фрагмент фонемы, для которого его длительность плюс длительность всех предыдущих фрагментов будет не меньше $\tilde{T}_i^c(f)$. Ниже приводится общая формула для расчета текущего фрагмента.

$$frg_c^i(f) = f[k], \text{ где } k = \min_{g(j) \geq 0} (j), g(j) = \sum_{l=1}^j |frg^l| - \tilde{T}_{i-1}^c(f).$$

Определим функцию смещения $\delta_i(f)$. Функция $\delta_i(f)$ определяет смещение начала проигрывания фонемы относительно ближайшей точки разбиения слева.

Введем параметр $\delta_i^W, i = \overline{1, s}$, который будем называть смещением текущего слова. Этот параметр для каждого операционного интервала $\Delta t^i, i = \overline{1, s}$ будет равен смещению первой фонемы слова, которое проигрывается на этом отрезке.

В работе [1] определена функция $\delta_i^*(frq)$, которая характеризует начало проигрывания конкретной частоты. Используя эту функцию и параметр δ_i^W , определим функцию смещения.

$$\delta_i(f) = \begin{cases} \min_{frq \in frg[]} \delta_i^*(frq), frg = f[1], T_{i-1}^C(f) = 0, I_i^W = 0, \\ \min_{frq \in frg[]} amp s(t) > a_{\min} \text{ или} \\ \min_{frq \in frg[]} amp s(t) > a_{\min} \\ \delta_{i-1}(f), T_{i-1}^C(f) > 0, I_i^W = 0 \\ \delta_i^W, I_i^W = 1 \\ 0, T_{i-1}^C(f) = 0, I_i^W = 0, \min_{frq \in frg[]} amp s(t) \leq a_{\min}, j = \overline{i, i+1} \end{cases}$$

В режиме распознавания слова текущим смещением каждой фонемы $f \in F$ будем считать значение параметра δ_i^W .

Вне режима распознавания слова текущее смещение $\delta_i(f)$ фонемы f будет равно минимальному значению функции $\delta_i^*(frq)$ среди частот первого фрагмента в том случае, если на соответствующем операционном интервале Δ^i фонема f начинает проигрываться. Будем считать, что фонема f начинает проигрываться на операционном интервале Δ^i , если текущее время распознавания фонемы f к началу этого интервала равно нулю и минимальное значение амплитуд, соответствующих частотам первого фрагмента, рассчитанных на отрезке Δ^i или на следующем, будет больше значения параметра a_{\min} .

В том случае, если фонема f проигрывалась на предыдущем операционном интервале, т.е. текущее время распознавания этой фонемы к началу текущего отрезка будет больше нуля, значение $\delta_i(f)$ берется из предыдущего шага. Если ни одно из вышеуказанных условий не выполняется, функция $\delta_i(f)$ будет равна нулю.

Введем обозначение для операционных интервалов со смещением.

$$\Delta_\delta^i(f) = [t^{i-1} + \delta_i(f), t^i + \delta_i(f)], i = \overline{1, s}, f \in F.$$

Теперь на основе введенных понятий определим относительный порог m_i . Для этого введем функцию $a_i^*(frq)$, которая будет равна наибольшему значению амплитуды, соответствующей частоте frq , на операционных интервалах со смещением $\Delta_\delta^i(f)$ для тех фонем $f \in F \setminus \{E_i \cup N_i\}$, текущий фрагмент которых содержит частоту frq .

$$a_i^*(frq) = \max_{frq \in frg_C^i(f) \cap \{f \in F \setminus \{E_i \cup N_i\}\}} amp s(t), i = \overline{1, s}.$$

На каждом операционном интервале рассмотрим множество частот всех текущих фрагментов. Это множество обозначим Frq_i^C

$$Frq_i^C = \bigcup_{frg = frg_C^i(f), f \in F \setminus \{E_i \cup N_i\}} frg[], i = \overline{1, s}$$

Параметр m_i будет определяться исходя из частотности N_L системы и равен наибольшему N_L -ому значению функции $a_i^*(frq)$ в порядке убывания среди частот множества Frq_i^C .

Для вывода формулы m_i рассмотрим последовательность множеств $Frq_i^k \subseteq Frq_i^C, k \in N$. Будем считать $Frq_i^1 = Frq_i^C$, $Frq_i^{k+1} = Frq_i^k \setminus \{frq_i^k\}$, где frq_i^k - частота, на которой достигается максимум функции $a_i^*(frq)$ среди множества Frq_i^k , $a_i^*(frq_i^k) = \max_{frq \in Frq_i^k} a_i^*(frq)$. Т.е. каждое следующее множество Frq_i^k будет состоять из тех же частот, что и предыдущее, за исключением элемента, на котором достигается максимум функции $a_i^*(frq)$.

Определим m_i как максимальное значение $a_i^*(frq)$ среди множества $Frq_i^{N_i}$.

$$m_i = \max_{frq \in Frq_i^{N_i}} a_i^*(frq), i = \overline{1, s}.$$

Параметр $m_i^{N_i}$ будем определять аналогично параметру m_i , используя в качестве N_i пустое множество.

Теперь определим функцию амплитуды фонемы. Будем считать амплитудой $a_i(f)$ фонемы $f \in F$ на операционном интервале Δ^i минимальное значение амплитуды, соответствующее частоте текущего фрагмента этой фонемы на операционном интервале со смещением $\Delta_s^i(f)$.

$$a_i(f) = \min_{frq \in frq_i^C \cap \Delta_s^i(f), frq} amp s(t), i = \overline{1, s}.$$

Таким образом, рассмотрены основные понятия и функции необходимые для ввода определения распознанной фонемы на отрезке. Отметим, что параметры I_i^W , δ_i^W , а также множества E_i и N_i пока остаются неопределенными.

Фонему $f \in F$ будем называть распознанной на отрезке $[t^k, t^l], t^k, t^l \in T$, $t^l - t^k = |f|$, если текущее время распознавания фонемы к моменту времени t^l положительно и кратно длительности фонемы f , т.е. одновременно выполняются следующие два условия:

$$\begin{aligned} 1. & T_i^C(f) > 0, \\ 2. & \text{mod}_{|f|}(T_i^C(f)) = 0. \end{aligned} \tag{4.5}$$

Данное определение будем рассматривать в качестве критерия вхождения фонемы в аудиосигнал.

Рассмотрим множества R_i^* , которые для каждой точки $t^i, i = \overline{0, s}$ разбиения T будут содержать все фонемы $f \in F$, распознанные на некотором отрезке $[t^k, t^i], t^k \in T$, конец которого совпадает с точкой t^i . Очевидно, что фонема $f \in R_i^*$, если для неё выполняются условия (4.5) определения распознанной фонемы.

Используя понятие распознанной фонемы на отрезке и множества R_i^* , определим параметры I_i^W , δ_i^W . Будем считать параметр I_i^W равным единице, если с момента последней распознанной фонемы прошло не больше

длительности пробела. Иными словами, параметр I_i^W будет равен единице, если существует интервал $[t^k, t^l], t^k, t^l \in T$, на котором некоторая фонема будет распознанной, или $R_i^* \neq \emptyset$, что то же самое, и расстояние между концом t^l этого отрезка и точкой t^{i-1} меньше длительности пробела. В противном случае параметр будет равен нулю.

$$I_i^W = \begin{cases} 1, \exists j \leq i-1, |t^j, t^{i-1}| < dur_s, R_j^* \neq \emptyset \\ 0, \text{ в противном случае} \end{cases}$$

Таким образом, система будет находиться в режиме распознавания слова сразу после распознавания первой фонемы слова и до тех пор, пока в течение времени, равного длительности пробела, ни одна фонема не будет распознана.

Используя параметр I_i^W , определим смещение текущего слова δ_i^W . При переходе в режим распознавания слова, параметр δ_i^W будет рассчитываться как минимальное значение функции смещения $\delta_{i-1}(f)$ среди фонем, распознанных на предыдущем операционном интервале.

В режиме распознавания слова параметр δ_i^W будет равен его значению на предыдущем операционном интервале. Вне режима распознавания слова параметр будет равен нулю.

$$\delta_i^W = \begin{cases} \min_{f \in R_{i-1}^*} \delta_{i-1}(f), I_{i-1}^W = 0, I_i^W = 1, i \geq 2 \\ \delta_{i-1}^W, I_{i-1}^W = I_i^W = 1, i \geq 2 \\ 0, I_i^W = 0 \text{ или } i = 1 \end{cases}$$

Теперь введем ключевое понятие распознанного шаблона слова. Данное понятие будет являться основным критерием вхождения слова, соответствующего некоторому шаблону $w_i^f \subset W_T^F$, в аудиосигнал на некотором интервале времени $[t', t''] \subset [t_1, t_2]$.

Фонемный шаблон слова $w_i^f = (F_1, \dots, F_n) \in W_T^F$, $F_i \subset F, i = \overline{1, n}$ будем называть распознанным на отрезке $[t^k, t^l], t^k, t^l \in T$, если существует разбиение отрезка $[t^k, t^l]$ точками $t_0', \dots, t_n', t_i' \in T$, $t_0' = t^k, t_n' = t^l$, такое, что на каждом отрезке $\Delta t_i' = [t_{i-1}', t_i']$ фонемы из множества F_i являются распознанными на этом отрезке.

Иными словами, шаблон слова $w_i^f = (F_1, \dots, F_n) \in W_T^F$ будет распознанным на отрезке $[t^k, t^l], t^k, t^l \in T$, если этот отрезок можно разбить на интервалы, на каждом из которых фонемы из соответствующего множества F_i будут распознанными.

Отметим, что данное определение было введено с использованием множеств E_i и N_i , которые пока не определены.

4.2. Вспомогательные функции и множества

Введем ряд дополнительных множеств и функций, которые понадобятся при рассмотрении алгоритмов подавления эхо и подавления шума.

Определим функции $\tilde{T}_i^C(f)$, $i = \overline{0, s}$ следующим образом:

$$\tilde{T}_i^C(f) = \begin{cases} \text{mod}_{D_f}^*(T_i^C(f)), I_i^W = 0 \\ \text{mod}_{D_f}^*(T_i^W), I_i^W = 1 \end{cases}$$

где $\text{mod}_y^*(x)$ равна остатку от деления x на y , за тем исключением, что, если остаток равен нулю и $x \neq 0$, она равна y .

Вне режима распознавания слова функция $\tilde{T}_i^C(f)$ будет равна максимальной длине отрезка до момента времени t^i , на котором фонема f входила в аудиосигнал. В режиме распознавания слова эта функция равна расстоянию от начала текущего фонемного интервала слова до момента времени t^i .

Используя $\tilde{T}_i^C(f)$, определим функцию $t_i^C(f)$, которую будем называть временем распознавания текущего фрагмента фонемы к моменту времени t^i .

$$t_i^C(f) = \tilde{T}_i^C(f) - \sum_{j=1}^k | \text{frg}^j |, f[k+1] = \text{frg}_C^i(f).$$

Функция $t_i^C(f)$ будет определять, какое время непрерывно звучит текущий фрагмент фонемы к концу операционного интервала Δt^i , $i = \overline{1, s}$, в том случае, если на этом отрезке фонема проигрывается.

Введем множества F_P^i , $i = \overline{2, s}$, которые будут состоять из тех фонем, текущий фрагмент которых на предыдущем операционном интервале Δt^{i-1} был полностью проигран к концу этого интервала.

$$F_P^i = \begin{cases} \{f \in P_{i-1}^W \mid t_{i-1}^C(f) = | \text{frg}_C^{i-1}(f) | \}, i \geq 2 \\ \emptyset, i = 1 \end{cases}$$

Фонема $f \in F_P^i$, если она является потенциальной на предыдущем операционном интервале Δt^{i-1} и время распознавания текущего фрагмента этой фонемы к моменту времени t^{i-1} будет равно длительности этого фрагмента.

Аналогично рассмотрим множества F_L^i , которые будут состоять из тех фонем, для которых текущий операционный интервал является последним для полного проигрывания текущего фрагмента.

$$F_L^i = \begin{cases} \{f \in P_{i-1}^W \mid t_{i-1}^C(f) = | \text{frg}_C^{i-1}(f) | - \tau \}, i \geq 2 \\ \emptyset, i = 1 \end{cases}$$

4.3. Алгоритм подавления эхо

Во время проигрывания фрагмента амплитуда, соответствующая каждой его частоте, резко увеличивается. Однако после окончания проигрывания фрагмента, она "не спешит" резко падать. Как показывают эксперименты, в течение 100-250 мс амплитуда, соответствующая частотам проигранного фрагмента, остается достаточно высокой. Подобное поведение аудиосигнала называется эффектом эхо и является следствием отражения звуковых волн.

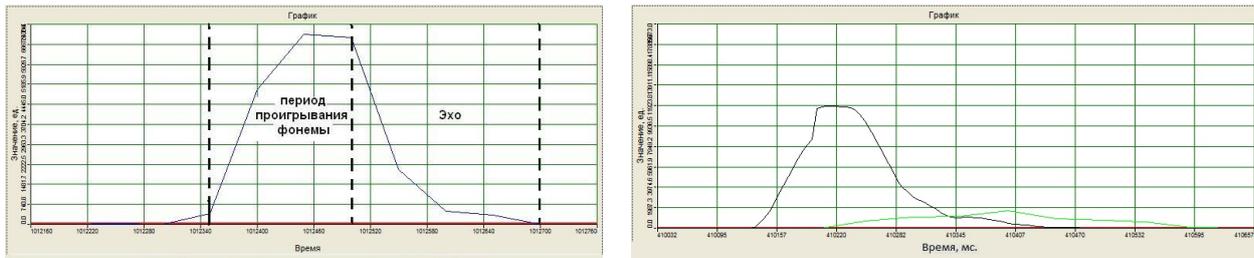


Рис.2. Демонстрация эффекта эхо

На 2 слева изображен график изменения амплитуды, соответствующей частоте 500 Гц, во время проигрывания фонемы, состоящей из одного фрагмента длительностью 150 мс, амплитудой 100 и частотой 500 Гц. Амплитуда вычислялась по интервалу 50 мс через каждые 50 мс. Пунктирными линиями график разделен на две области. Левая область соответствует времени, в течение которого была проиграна фонема. Правая область соответствует эху этого сигнала.

Эффект эхо негативно влияет на процесс распознавания. К примеру, если фонема, соответствующая букве «а», состоит из одного фрагмента, то при проигрывании только буквы «а» можно получить две-три буквы в результате распознавания.

На 2 справа изображены графики амплитуд двух фрагментов, которые были проиграны один за другим. Графики показывают, что фрагмент, который был проигран вторым (зеленый график), частично оказался менее громким, чем эхо от первого фрагмента (черный график).

Для решения таких проблем был построен алгоритм подавления эхо, который позволил избежать подобных ситуаций.

При описании критерия вхождения фонемы в аудиосигнал (определение выделенной фонемы) были введены множества E_i , состоящие из фонем, вхождение которых на соответствующем операционном интервале является эхом. Задача алгоритма подавления эхо заключается в определении этих множеств.

Будем рассматривать эхо каждой частоты в отдельности. Для этого введем множества E_i^* , состоящие из всех частот, вхождение которых в аудиосигнал на соответствующем операционном интервале Δ^i является эхом. Определим связь E_i и E_i^* следующим образом:

$$E_i = \{f \in F \mid \exists frq \in E_i^*, frq \in frg[], frg = frg_C^i(f)\}.$$

Иными словами, будем считать вхождение фонемы f в аудиосигнал эхом на отрезке Δt^i , если хотя бы одна частота текущего фрагмента этой фонемы на этом отрезке является эхом.

Введем параметр γ , который будем называть коэффициентом эхо. Значение γ будем рассматривать как параметр алгоритма. Используя этот параметр, рассмотрим критерии, по которым будем определять, является аудиосигнал, соответствующий произвольной частоте, эхом. Для этого введем следующие определения.

Частоту $freq \in N$ будем называть коротким эхом на отрезке $[t^{i-1}, t^{i+2}]$, $i \geq 3, i \leq s-2$, если амплитуда аудиосигнала, соответствующая частоте $freq$ на операционном интервале Δt^{i+2} , будет меньше произведения γ на среднее значение аналогичной амплитуды на двух отрезках Δt^{i-2} и Δt^{i-1} , т.е.

$$amp_{freq, \Delta t^{i+2}} s(t) < \gamma \cdot \sum_{j=i-2}^{i-1} amp_{freq, \Delta t^j} s(t) / 2. \quad (4.6)$$

Множество частот, удовлетворяющих этому определению на отрезке $[t^k, t^l]$, будем обозначать $E_{k,l}^S$.

Частоту $freq \in N$ будем называть длинным эхом на отрезке $[t^{i-1}, t^{i+5}]$, $i \geq 3, i \leq s-2$, если на отрезке $[t^{i-1}, t^{i+2}]$ частота $freq$ удовлетворяет условиям определения короткого эхо и амплитуды аудиосигнала, соответствующие частоте $freq$ на трех отрезках разбиения $\Delta t^j, j = \overline{i+3, i+5}$, будут меньше произведения γ на среднее значение аналогичной амплитуды на двух отрезках $\Delta t^{i-1}, \Delta t^{i-2}$, т.е. одновременно выполняются условия

1. $freq \in E^S[t^{i-1}, t^{i+2}]$;
2. $amp_{freq, \Delta t^j} s(t) < \gamma \cdot \sum_{j=i-2}^{i-1} amp_{freq, \Delta t^j} s(t) / 2, j = \overline{i+3, i+5}$.

(4.7)

Множество частот, удовлетворяющих вышеуказанному определению на отрезке $[t^k, t^l]$, будем обозначать $E_{k,l}^L$.

Иными словами, будем считать частоту эхом на определенном отрезке, если громкость этой частоты на этом отрезке заметно ниже средней громкости этой частоты на предыдущем отрезке.

Поскольку эхо возникает после проигрывания очередного фрагмента, на каждом операционном интервале $\Delta t^i, i = \overline{1, s}$ рассмотрим множество F_p^i , состоящее из тех фонем, некоторый фрагмент которых был проигран полностью на предыдущем операционном интервале.

Введем множество Frq_p^i , состоящее из частот текущих фрагментов фонем из множества F_p^i на предыдущем операционном интервале $\Delta t^{i-1}, i = \overline{2, s}$.

$$Frq_p^j = \{frq \in N \mid frq \in frg[], frg = frg_C^{i-1}(f), f \in F_p^i\}, i = \overline{2, s}.$$

Во множество Frq_p^j будут входить частоты тех фрагментов, которые были проиграны полностью на предыдущем отрезке Δt^{i-1} . Частоты из этого множества будем рассматривать в качестве потенциального эхо на некотором отрезке $[t^{i-1}, t^j]$.

Рассмотрим множество $\hat{E}_i^S, i = \overline{1, s}$, которое будет состоять из тех частот $frq \in Frq_p^j$, которые на отрезке $[t^{i-1}, t^{i+2}]$ удовлетворяют условиям (4.6) определения короткого эхо. При $i \in \{1, 2\}$ будем считать $\hat{E}_i^S = \emptyset$.

$$\hat{E}_i^S = \begin{cases} Frq_p^j \cap E_{i-1, i+2}^S, & i \geq 3 \\ \emptyset, & i \in \{1, 2\} \end{cases}$$

Во множество $\hat{E}_i^S, i = \overline{3, s}$ будут входить частоты тех фрагментов, которые были полностью проиграны на предыдущем операционном интервале Δt^{i-1} и которые являются коротким эхо на последующем отрезке $[t^{i-1}, t^{i+2}]$.

Аналогично введем множества $\hat{E}_i^L, i = \overline{1, s}$. Множества $\hat{E}_i^L, i = \overline{3, s}$ будет состоять из тех частот $frq \in Frq_p^j$, которые на отрезке $[t^{i-1}, t^{i+5}]$ удовлетворяют условиям (4.7) определения длинного эхо. При $i \in \{1, 2\}$ будем считать $\hat{E}_i^L = \emptyset$.

$$\hat{E}_i^L = \begin{cases} Frq_p^j \cap E_{i-1, i+5}^L, & i \geq 3 \\ \emptyset, & i \in \{1, 2\} \end{cases}$$

Во множества $\hat{E}_i^L, i = \overline{3, s}$ будут входить частоты тех фрагментов, которые были полностью проиграны на предыдущем операционном интервале Δt^{i-1} и которые являются длинным эхо на последующем отрезке $[t^{i-1}, t^{i+5}]$.

Используя множества \hat{E}_i^S и \hat{E}_i^L , определим множество E_i^* как объединение множеств $\hat{E}_j^S, j = \overline{i-2, i}$ и множеств $\hat{E}_j^L, j = \overline{i-5, i}$.

$$E_i^* = [\bigcup_{j=i-2}^i \hat{E}_j^S] \cup [\bigcup_{j=i-5}^i \hat{E}_j^L].$$

Отметим, что принадлежность частоты $frq \in E_i^*$ множеству E_i^* гарантирует, что найдется интервал $[t^k, t^l]$, который содержит операционный интервал Δt^i и на котором частота frq является коротким или длинным эхо.

Таким образом, определены множества $E_i^*, i = \overline{1, s}$, а вместе с ними и искомые множества $E_i, i = \overline{1, s}$.

4.4. Алгоритм подавления шума

Вместе с полезным сигналом микрофон принимает различные шумы – разговор людей, шум бытовой техники, машин, шелест листьев и т. д. Подобные звуки оказывают отрицательное воздействие на качество работы алгоритма распознавания.

На 3 изображены график и спектр аудиосигнала, который был получен при проигрывании слова «привет» в тихом помещении и в комнате с включенной музыкой. Как видно из рисунков, график и спектр аудиосигнала сильно исказились при проигрывании слова при посторонних шумах.

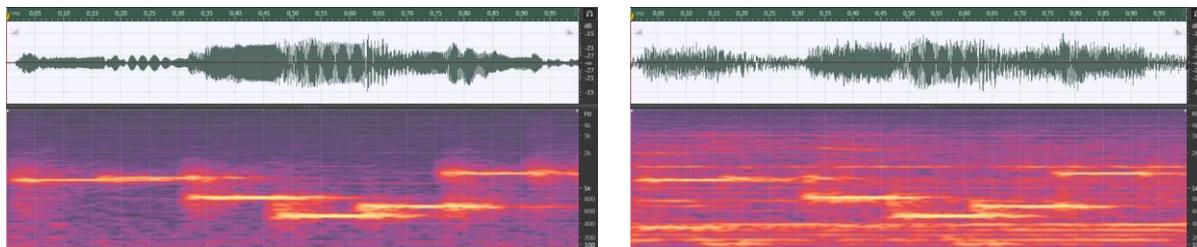


Рис.3. Аудиосигнал, соответствующий слову «привет», без шума (слева) и с шумом (справа)

Вследствие этого оказался необходимым алгоритм подавления шума. Задача данного алгоритма заключается в анализе аудиосигнала с целью поиска в нём звуков, не соответствующих фонемам языка робота.

При описании критериев вхождения фонемы в аудиосигнал были введены множества N_i , которые для каждого операционного интервала Δt^i должны содержать фонемы, вхождение которых в аудиосигнал на этом отрезке является шумом. Задача алгоритма подавления шума заключается в определении этих множеств.

Определим критерии, согласно которым будем выделять из аудиосигнала посторонние звуки, рассматриваемые как шум. Для этого введем ряд определений.

Рассмотрим некоторый параметр a_n , который будем называть уровнем шума. Значение данного параметра будем рассматривать как параметр алгоритма.

Частоту $freq \in Frq_n$ будем называть одиночным всплеском шума на отрезке $[t^{i-1}, t^i] = \Delta t^i, i = \overline{2, s-1}$, если одновременно выполняются следующие условия:

1. $amps(t)_{freq, \Delta t^i} > a_n$,
 2. $amps(t)_{freq, \Delta t^j} < a_n, j = \{i-1\} \cup \{i+1\}$.
- (4.8)

Иными словами, частоту будем называть одиночным всплеском шума на операционном интервале, если на нем амплитуда, соответствующая этой частоте, выше уровня шума a_n , а справа и слева от этого интервала аналогичная амплитуда будет ниже этого уровня.

Множество частот, удовлетворяющих на отрезке $[t^k, t^l]$ условиям (4.8) вышеуказанного определения, будем обозначать $N_{k,l}^1$.

Частоту $frq \in Frq_u$ будем называть двойным всплеском шума на отрезке $[t^{i-1}, t^{i+1}] = \Delta t^i \cup \Delta t^{i+1}, i = \overline{2, s-2}$, если одновременно выполняются следующие условия:

$$\begin{aligned} 1. \quad & \underset{frq, \Delta t^j}{amp s(t)} > a_n, j = \overline{i, i+1}, \\ 2. \quad & \underset{frq, \Delta t^j}{amp s(t)} < a_n, j = \{i-1\} \cup \{i+2\}. \end{aligned} \quad (4.9)$$

Другими словами, частоту будем называть двойным всплеском шума на отрезке $[t^{i-1}, t^{i+1}]$, если этот отрезок состоит из двух операционных интервалов и на каждом из этих интервалов амплитуда, соответствующая этой частоте, выше уровня шума a_n , а справа и слева от этого отрезка аналогичная амплитуда будет ниже этого уровня.

Множество частот, удовлетворяющих на отрезке $[t^k, t^l]$ условиям (4.9) вышеуказанного определения, будем обозначать $N_{k,l}^2$.

Частоту $frq \in Frq_u$ будем называть множественным всплеском шума на отрезке $[t^{i-1}, t^j], t^i, t^j \in T, i = \overline{2, s-3}, j = \overline{i+2, i+3}$, если одновременно выполняются следующие условия:

$$\begin{aligned} 1. \quad & \underset{frq, \Delta t^k}{amp s(t)} > a_n, \forall k = \overline{i, j}, \\ 2. \quad & \underset{frq, \Delta t^j}{amp s(t)} < a_n, j = \{i-1\} \cup \{j+1\}. \end{aligned} \quad (4.10)$$

Иными словами, частоту будем называть множественным всплеском шума на отрезке $[t^{i-1}, t^j]$, если этот отрезок состоит из трех или четырех операционных интервалов и на каждом из них амплитуда, соответствующая этой частоте, выше уровня шума a_n , а справа и слева от этого отрезка аналогичная амплитуда будет ниже этого уровня.

Множество частот, удовлетворяющих на отрезке $[t^k, t^l]$ условиям (4.10) вышеуказанного определения, будем обозначать $N_{k,l}^{3,4}$.

Введенные определения будем рассматривать в качестве критериев шума. Отметим, что указанные критерии будут иметь смысл, если в качестве уровня шума взять значение, не превышающее значение параметра a_{\min} , используемое в определении выделенной фонемы, и предположить, что длительность любого фрагмента не меньше 3τ (длины трех операционных интервалов). В таком случае, любая частота, удовлетворяющая на некотором отрезке условиям определения одиночного или двойного шума, не может являться частотой некоторого фрагмента, проигранного на этом отрезке.

Частоты, удовлетворяющие условиям (4.10) определения множественного всплеска шума на некотором отрезке, могут являться частотой фрагмента, проигранного на этом отрезке. Однако в том случае, когда этот отрезок попадает на стык проигрывания двух фрагментов, аудиосигнал, соответствующий этим частотам, может негативно повлиять на

распознавание этих фрагментов. В этом случае полезно будет избавиться от подобных частот при обработке аудиосигнала на таких отрезках.

Рассмотрим множества \hat{N}_i^1 , \hat{N}_i^2 , $\hat{N}_i^{3,4}$, которые будут состоять из частот, которые на некотором отрезке $[t^j, t^k]$, содержащем операционный интервал Δt^i , будут являться одиночным, двойным и множественным всплесками шума соответственно.

Очевидно, что множество \hat{N}_i^1 совпадает с множеством $N_{i-1,i}^1$ при $i = \overline{2, s-1}$. Вне этого интервала определим $\hat{N}_i^1 = \emptyset$.

$$\hat{N}_i^1 = \begin{cases} \{N_{i-1,i}^1, i = \overline{2, s-1}\} \\ \emptyset, i = \{1\} \cup \{s\} \end{cases}.$$

Аналогично будем считать, что множество \hat{N}_i^2 совпадает с объединением множеств $N_{i-2,i}^2$ и $N_{i-1,i+1}^2$ при $i = \overline{2, s-1}$ и \emptyset вне этого интервала.

$$\hat{N}_i^2 = \begin{cases} N_{i-2,i}^2 \cup N_{i-1,i+1}^2, i = \overline{2, s-1} \\ \emptyset, i = \{1\} \cup \{s\} \end{cases}.$$

Следующая формула определяет множества $\hat{N}_i^{3,4}$.

$$\hat{N}_i^{3,4} = \begin{cases} \{frq \in Frq_u \mid \exists [t^j, t^k], S_j \neq \emptyset, \Delta t^i \subset [t^j, t^k], |[t^j, t^k]| \leq l_N^j, \\ l_N^j = \max_{f \in S_j} (|frg_C^j(f)| - t_j^C(f)) + \min_{frg \in Frq_u} |frg| - \tau, \\ frq \in N_{j,k}^{3,4}\}, i = \overline{2, s-1} \\ \emptyset, \text{ в остальных случаях} \end{cases}$$

Множество $\hat{N}_i^{3,4}$ состоит из тех частот, для которых найдется отрезок $[t^j, t^k]$, содержащий операционный интервал Δt^i , на котором они являются множественным всплеском шума. Множество выделенных фонем S_j на операционном интервале, предшествующем этому отрезку, должно быть не пустым. Длина этого отрезка должна быть меньше времени, которое не доиграл текущий фрагмент каждой фонемы из множества S_j , плюс наименьшая длительность фрагментов.

Используя множества \hat{N}_i^1 , \hat{N}_i^2 , $\hat{N}_i^{3,4}$, определим искомые множества фонем N_i как множества фонем, хотя бы одна частота текущего фрагмента которых входит в объединение этих множеств.

$$N_i = \{f \in F \mid frq \in \hat{N}_i^1 \cup \hat{N}_i^2 \cup \hat{N}_i^{3,4}, frq \in frg[], frg = frg_C^i(f)\}, i = \overline{1, s}.$$

4.5. Алгоритм корректировки смещения

При определении начала проигрывания фонемы рассчитывается функция смещения $\delta_i(f)$. Если фонема f входит в аудиосигнал, на следующем шаге значение $\delta_{i+1}(f)$ для этой фонемы берется из предыдущего

шага, и так до тех пор, пока на некотором шаге фонема f не будет выделенной на соответствующем операционном интервале.

Таким образом, если в начале проигрывания фонемы значение $\delta_i(f)$ было рассчитано неправильно, оно же и будет использоваться в течение последующего периода, на котором фонема f непрерывно входит в аудиосигнал. Данная погрешность в расчете функции $\delta_i(f)$ может негативно повлиять на процесс распознавания.

В связи с этим был построен алгоритм, который корректирует значение функции $\delta_i(f)$ в случае необходимости. Ниже приводится описание двух подобных алгоритмов: алгоритма корректировки смещения влево и алгоритма корректировки смещения вправо. Оба алгоритма корректируют значение функции смещения на стыке фрагментов и фонем.

Корректировка смещения влево. Рассмотрим следующую ситуацию. На одном роботе проигрывается некоторая фонема $f = (frag^1, frag^2 \dots frag^n) \in F$. При этом фонема преобразуется в последовательность фрагментов $frag_1, frag_2 \dots frag_n, frag_i \in Frag, i = \overline{1, n}$. Второй робот слушает входящий поток аудиоданных и пытается определить фрагменты, проигранные первым роботом.

Предположим, при вычислении значения функции смещения $\delta_i(f)$ для фонемы f оно было определено некорректно. При этом будем считать, что реальное значение $\delta_f < \delta_i(f)$.

На 4 изображен график и спектр полученного аудиосигнала из подобного эксперимента. На графике толстыми жирными линиями обозначены границы фрагментов. При этом черная черта соответствует реальным значениям, красная черта – рассчитанным, используя функцию $\delta_i(f)$. Менее жирными линиями помечены границы операционных интервалов со смещением.

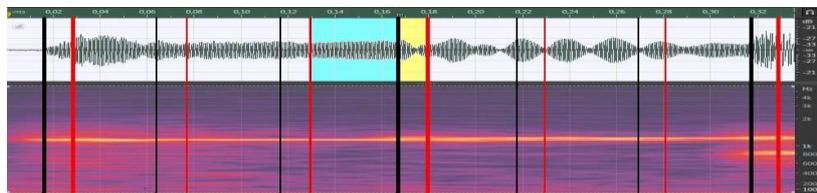


Рис.4. График и спектр аудиосигнала из эксперимента

Каждый фрагмент в описываемом эксперименте состоит из трех операционных интервалов. Рассмотрим операционный интервал со смещением, который является последним для первого фрагмента фонемы f . На рисунке область, соответствующая этому интервалу, помечена синим и желтым цветами.

Поскольку значение $\delta_i(f)$ было определено неправильно и реальное значение этой функции меньше, чем рассчитанное, этот интервал захватил

часть следующего фрагмента. Область этого интервала, соответствующая второму фрагменту фонемы f , отмечена желтым цветом.

В силу того, что амплитуда $a_i(f)$ фонемы f вычисляется как минимальное значение амплитуды аудиосигнала, соответствующее частотам текущего фрагмента этой фонемы, на операционном интервале со смещением, её значение будет вычислено некорректно. Значение $a_i(f)$ будет ниже реального, поскольку текущий фрагмент фонемы будет входить в аудиосигнал только на части операционного интервала со смещением.

Более того, если для некоторой фонемы $f' \in F$ текущим фрагментом будет второй фрагмент фонемы f , то значение $a_i(f')$ будет выше реального. В таком случае множество выделенных фонем S_i может быть определено некорректно, и, тем самым, проигранная фонема не будет распознана.

Рассмотрим алгоритм, который будет корректировать значение функции смещения $\delta_i(f)$ в подобных ситуациях.

На каждом операционном интервале $\Delta t^i, i = \overline{1, s}$ рассмотрим множества $F_L^i \subset F$. Эти множества состоят из фонем, для которых отрезок Δt^i является последним для полного вхождения текущего фрагмента фонемы в аудиосигнал.

Пусть для некоторого $k = \overline{1, s}$ множество $F_L^k \neq \emptyset$, однако ни одна фонема из F_L^k не является выделенной на отрезке Δt^k , т.е. множество S_k не пересекается с F_L^k .

$$F_L^k \cap S_k \neq \emptyset.$$

Введем параметр ξ_L , значение которого будем рассматривать в качестве параметра алгоритма.

Рассмотрим функцию $\delta'_k(f, \xi), \xi \in [0, \xi_L]$, которая будет равна функции смещения $\delta_k(f)$ минус ξ для всех фонем, для которых время распознавания к моменту t^{k-1} не равно нулю. Для всех остальных фонем она будет равна функции смещения $\delta_k(f)$.

$$\delta'_k(f, \xi) = \begin{cases} \delta_k(f) - \xi, & T_{k-1}^C(f) > 0 \\ \delta_k(f) & \text{для остальных фонем } f \in F \end{cases}$$

Отметим, что если фонема $f \in F_L^k$, то текущее время распознавания этой фонемы к моменту времени t^{k-1} будет больше нуля, т.е. $T_{k-1}^C(f) > 0$.

Рассмотрим множества $S'_k(\xi)$, которые будут отличаться от множества S_k тем, что при определении этих множеств вместо функции смещения будет использоваться функция $\delta'_k(f, \xi)$.

В том случае, если пересечение множеств $S'_k(\xi) \cap F_L^k \neq \emptyset$ хотя бы для одного $\xi \in [0, \xi_L]$, рассмотрим наименьшее значение ξ^* , для которого $S'_k(\xi^*) \cap F_L^k \neq \emptyset$

$$\xi^* = \min_{S'_k(\xi) \cap F_L^k \neq \emptyset, \xi \in [0, \xi_L]} (\xi).$$

В этом случае новой функцией смещения будем считать функцию $\delta_k(f) = \delta_k^i(f, \xi^*)$. Использование этой функции в качестве функции смещения на операционном интервале Δt^k будет гарантировать, что хотя бы одна фонема $f \in F_L^k$ будет выделенной на этом отрезке.

Корректировка смещения направо. Алгоритм корректировки смещения направо аналогичен алгоритму корректировки смещения влево за тем исключением, что корректировка смещения производится для тех фонем, некоторый фрагмент которых был проигран на предыдущем операционном интервале, и коррекция осуществляется в сторону увеличения.

4.6. Пошаговое описание алгоритма

Процесс обработки аудиосигнала начинается с прослушивания аудиоданных длительностью 6τ . Такой отрезок необходим для работы алгоритма подавления эхо. Далее аудиосигнал обрабатывается через каждый интервал длительностью τ .

В качестве исходных данных на каждом операционном интервале Δt^i будем рассматривать функцию $T_{i-1}^C(f)$ текущего времени распознавания фонемы к началу этого интервала, текущее время распознавания слова T_{i-1}^W к моменту t^{i-1} , параметр I_i^W , множества P_i^W , E_i , N_i , а также множество распознанных фонем R_{i-1}^* . Отметим, что на первом операционном интервале значения исходных данных, $T_0^C(f) = 0$, $T_0^W = 0$, $I_1^W = 0$, $P_1^W = F$, $E_1 = \emptyset$, $N_1 = \emptyset$, $R_0^* = \emptyset$, известны.

Обработка аудиосигнала на каждом операционном интервале Δt^i будет заключаться в определении множества выделенных фонем S_i , множества выделенных фонем с шумом S_i^N и множества распознанных фонем R_i^* к концу этого интервала и исходных данных для следующего операционного интервала Δt^{i+1} .

Рассмотрим процесс обработки аудиосигнала на произвольном операционном интервале Δt^k . Как отмечалось, к началу этого отрезка известна функция $T_{k-1}^C(f)$, параметры T_{k-1}^W , I_k^W и множества P_k^W , E_k , N_k , R_{k-1}^* .

Определение множества выделенных фонем S_k . Согласно алгоритму распознавания для определения множества S_k выделенных фонем на интервале Δt^k необходимо последовательно проделать ряд следующих действий.

1. Определить текущий фрагмент $frag_k^k(f)$ каждой фонемы $f \in F$.
2. Вычислить значение параметра δ_k^W .
3. Вычислить значение функции смещения $\delta_k(f)$ для каждой $f \in F$.
4. Определить значение параметров m_k .

5. Вычислить значение функции амплитуды фонемы $a_k(f)$ для каждой $f \in F$.
6. Определить промежуточное множество выделенных фонем S_k .
7. Скорректировать значение функции смещения $\delta_k(f)$ и множество выделенных фонем S_k посредством выполнения алгоритма корректировки смещения слева и алгоритма корректировки смещения направо.

Определение множества распознанных фонем R_k^* . Для определения множества R_k^* необходимо вычислить текущее время распознавания $T_k^C(f)$ каждой фонемы $f \in F$ к концу отрезка Δt^k . После этого проверяем условия (4.5) определения распознанной фонемы. В том случае, если для некоторой фонемы $f \in F$ условия удовлетворяются, считаем фонему f распознанной на интервале $[t^k - |f|, t^k]$, и соответственно, $f \in R_k^*$.

Определение множества S_k^N . Отметим, что в процессе получения множества R_k^* было определено множество выделенных фонем S_k . Однако множество выделенных фонем с шумом S_k^N пока еще остается не определенным.

Для определения S_k^N рассмотрим множество N_k . Множество выделенных фонем с шумом S_k^N будет совпадать с множеством S_k в том случае, если $N_k = \emptyset$. В противном случае переходим к определению значения параметра m_k^N . Вычисление данного параметра полностью совпадает с определением параметра m_k за тем исключением, что в качестве множества N_k берется пустое множество.

Определив параметр m_k^N для каждой фонемы $f \in F$, проверяем условия определения (4.3) выделенной фонемы с шумом. В том случае, если условия удовлетворяются, считаем $f \in S_k^N$.

Определение исходных данных интервала Δt^{k+1} . Получив множества S_k , S_k^N , R_k^* , переходим к определению исходных данных на следующем операционном интервале Δt^{k+1} . Отметим, что только что полученные значения функции $T_k^C(f)$ и множество R_k^* тоже рассматриваются в качестве исходных данных на этом интервале. Следовательно, осталось последовательно определить параметры I_{k+1}^W , T_k^W , а также множества F_L^{k+1} , F_P^{k+1} , P_{k+1}^W , E_{k+1} и N_{k+1} , которые определяются согласно соответствующим формулам.

Определение шаблона слова. Определив множество распознанных фонем R_k^* , а также исходные данные на следующем операционном интервале

Δt^{k+1} , переходим определению фонемного шаблона слова.

В том случае, если на отрезке Δt^k система не находилась в режиме распознавания слова и к моменту времени t^k была распознана хотя бы одна фонема, т.е. множество $R_k^* \neq \emptyset$ оказалось не пустым, согласно алгоритму распознавания система перейдет в режим распознавания слова.

В момент перехода в режим распознавания слова рассмотрим фонемный шаблон слова, в который добавим множество только что распознанных фонем R_k^*

$$w_t^f = (R_k^*).$$

После перехода в режим распознавания слова начинаются фонемные интервалы, которые будут идти друг за другом до тех пор, пока система не выйдет из режима распознавания слова. Длительность каждого фонемного интервала равна общей длительности фонемы D_f .

До окончания каждого фонемного интервала множества R_j^* будут пустыми. К концу очередного фонемного интервала слова будем добавлять множество $R_{k_i}^*$ к фонемному шаблону слова w_t^f . Отметим, что множество $R_{k_i}^*$ может быть и пустым.

$$w_t^f = (R_{k_1}^*, R_{k_2}^*, \dots, R_{k_n}^*).$$

После выхода системы из режима распознавания слова из фонемного шаблона слова w_t^f удаляем все пустые множества фонем справа. Отметим, что такие множества обязательно будут в силу того, что длительность пробела dur_s больше общей длительности фонемы D_f .

Полученный фонемный шаблон $w_t^f = (R_{k_1}^*, R_{k_2}^*, \dots, R_{k_n}^*)$ согласно определению будет распознанным на интервале, объединяющем все отрезки, на которых были распознаны множества фонем $R_{k_i}^*$. Шаблон $w_t^b \in W_T^B$, соответствующий фонемному шаблону w_t^f , будет использоваться в качестве входного значения для алгоритма поиска слова в словаре по шаблону.

5. АЛГОРИТМ ПОИСКА СЛОВА В СЛОВАРЕ ПО ШАБЛОНУ

Цель данного алгоритма – определить минимально возможный набор слов $D^* \subset D$ из словаря, который будет соответствовать рассматриваемому шаблону слова $w_t^b \in W_T^B$, полученному в результате работы предыдущего алгоритма. В том случае, если множество D^* окажется непустым, будем считать, что одно из слов этого множества было проиграно некоторым роботом.

Критерии соответствия. Для определения множества D^* для произвольного шаблона $w_t^b \in W_T^B$ рассмотрим критерии, по которым будем определять соответствие слова $w \in D$ из словаря шаблону w_t^b . Для этого введем ряд следующих определений.

Будем говорить, слово $w \in D$ будет соответствовать шаблону $w_i^b = (B_1, \dots, B_n) \in W_T^B$, если для некоторого подслова $w' = (b_1, \dots, b_n) \subset w$, $|w'| = |w_i^b|$, длина которого будет совпадать с длиной шаблона w_i^b , каждая его буква будет содержаться в соответствующем множестве B_i шаблона w_i^b , либо это множество будет пустым, т.е. будет выполняться одно из следующих условий:

1. $b_i \in B_i$,
 2. $B_i = \emptyset$.
- (5.1)

Множество слов из словаря D , соответствующих шаблону w_i^b обозначим $D_T^0(w_i^b)$.

Если существует подслово w' , удовлетворяющее условиям определения (5.1), которое является началом слова w , будем говорить, что слово w соответствует шаблону w_i^b с начала. Множество слов, соответствующих шаблону w_i^b с начала, будем обозначать $D_T^B(w_i^b)$.

В том случае, если w' будет концом слова w , будем говорить, что оно соответствует шаблону w_i^b до конца. Множество слов, соответствующих шаблону w_i^b до конца, будем обозначать $D_T^E(w_i^b)$.

Если само слово w удовлетворяет условиям определения (5.1), будем говорить, что слово w полностью соответствует шаблону w_i^b . Множество слов, полностью соответствующих шаблону w_i^b , будем обозначать $D_T^F(w_i^b)$.

Оператор соответствия. Рассмотрим оператор $D_T(w_i^b)$, который каждому шаблону $w_i^b \in W_T^F$ будет ставить в соответствие некоторое подмножество слов из словаря следующим образом:

$$D_T(w_i^b) = \begin{cases} D_T^F(w_i^b), D_T^F(w_i^b) \neq \emptyset \\ D_T^B(w_i^b), |w_i^b| \geq 3, D_T^B(w_i^b) \neq \emptyset \\ D_T^E(w_i^b), |w_i^b| \geq 3, D_T^B(w_i^b) = \emptyset, D_T^E(w_i^b) \neq \emptyset \\ D_T^0(w_i^b), |w_i^b| \geq 3, D_T^B(w_i^b) = \emptyset, D_T^E(w_i^b) = \emptyset, D_T^0(w_i^b) \neq \emptyset \\ \emptyset, \text{ в остальных случаях} \end{cases}$$

$D_T(w_i^b)$ будет возвращать слова, которые полностью соответствуют шаблону w_i^b . Если таких слов не будет и длина шаблона не меньше трех, он будет возвращать слова, начало которых соответствует шаблону, потом – конец. Если и таких слов не будет, $D_T(w_i^b)$ вернет слова, которые просто соответствуют шаблону.

Оператор сопряжения. Шаблон w_i^b соответствует некоторому фонемному шаблону $w_i^f = (F_1, \dots, F_n) \in W_T^F$, который является распознанным на некотором отрезке $[t', t'']$. Для фонемного шаблона w_i^f определим сопряженный шаблон $\hat{w}_i^f \in W_T^F$.

Согласно определению распознанного шаблона существует разбиение

отрезка $[t', t'']$ точками $t'_0, \dots, t'_n, t'_i \in T$, $t'_0 = t', t'_n = t''$, такое, что на каждом отрезке $\Delta'_i = [t'_{i-1}, t'_i]$ фонемы из множества F_i являются распознанными. На каждом отрезке $\Delta'_i = [t'_{i-1}, t'_i]$ этого разбиения определим сопряженное множество фонем F_i^* .

В том случае, если множества $F_i \neq \emptyset$ не пустые, сопряженным множеством F_i^* будем считать само множество F_i .

В противном случае рассмотрим две функции $\rho_i(f)$, $\rho_i^N(f)$, $f \in F$. Данные функции будут равны количеству операционных интервалов, содержащихся в интервале Δ'_i , на которых фонема f является выделенной и выделенной с шумом соответственно. Отметим, что эти функции могут принимать значения в интервале $\overline{0, \rho_{\max}, \rho_{\max}} = [D_f / \tau]$, где ρ_{\max} равно максимальному количеству операционных интервалов, приходящих на интервал длиной, равной общей длительности фонемы.

Рассмотрим множества фонем F_ρ , для которых функция $\rho_i(f)$ принимает одно и то же положительное значение $\rho = \overline{1, \rho_{\max} - 1}$.

$$F_\rho \subset F, \forall f_1, f_2 \in F_\rho, f_3 \in F \setminus F_\rho \quad \rho_i(f_3) \neq \rho_i(f_1) = \rho_i(f_2) = \rho > 0.$$

Аналогично множества фонем F_ρ^N , для которых функция $\rho_i^N(f)$ принимает одно и то же положительное значение $\rho = \overline{1, \rho_{\max} - 1}$:

$$F_\rho^N \subset F, \forall f_1, f_2 \in F_\rho^N, f_3 \in F \setminus F_\rho^N \quad \rho_i^N(f_3) \neq \rho_i^N(f_1) = \rho_i^N(f_2) = \rho > 0.$$

В том случае, если все множества $F_\rho = \emptyset$, $F_\rho^N = \emptyset$, $\rho = \overline{1, \rho_{\max} - 1}$ являются пустыми, будем считать сопряженным множеством фонем F_i^* пустое множество.

В противном случае рассмотрим последовательность из всех непустых множеств F_ρ , F_ρ^N , $\rho = \overline{1, \rho_{\max} - 1}$. Отсортируем эту последовательность в порядке убывания приоритета. Будем считать более приоритетным то множество, для которого ρ будет больше. В том случае, если для двух множеств F_ρ и F_ρ^N значение ρ будет одинаковым, множество F_ρ , определенное с помощью функции $\rho_i(f)$, будем считать более приоритетным. В качестве сопряженного множества фонем F_i^* возьмем первое (самое приоритетное) множество из этой последовательности.

Установив сопряженное множество F_i^* для каждого множества F_i фонемного шаблона $w_i^f = (F_1, \dots, F_n)$, определим сопряженный фонемный шаблон $\hat{w}_i^f = (F_1^*, \dots, F_n^*)$, в котором вместо множеств F_i будут использоваться множества F_i^* .

Иными словами, сопряженный фонемный шаблон вместо пустых множеств содержит множества фонем, которые больше всего были выделенными на соответствующих интервалах, если таковые имеются.

Поскольку между множествами W_T^F и W_T^B существует однозначное соответствие, обусловленное языком роботов, для шаблона $w_i^b = (B_1, \dots, B_n)$,

соответствующего фонемному шаблону w_i^f , определим сопряженный шаблон $\hat{w}_i^b = (B_1^*, \dots, B_n^*)$.

Рассмотрим множество $W_T^{B'} \subset W_T^B$, состоящее из шаблонов, буквенные множества которых совпадают либо с $B_i, i = \overline{1, n}$, либо с пустыми множествами.

$$W_T^{B'} = \{w_i^{b'} = (B_1', \dots, B_n'), B_i' \in \{\emptyset, B_1, \dots, B_n\}, i = \overline{1, n}, B_1' \neq \emptyset, B_n' \neq \emptyset\}.$$

Над этим множеством рассмотрим оператор сопряжения $Q^*(w_i^{b'})$, который для каждого шаблона $w_i^{b'} \in W_T^{B'}$, $w_i^{b'} = (B_1', \dots, B_n')$ будет возвращать шаблон $\hat{w}_i^{b'} = (B_1^*, \dots, B_n^*)$, где $B_i^* = B_j^*$, если $B_i' = B_j$ для некоторого j и $B_i^* = \emptyset$, если $B_i' = \emptyset$.

Сопряженный оператор соответствия. Используя операторы соответствия $D_T(w_i^{b'})$ и сопряжения $Q^*(w_i^{b'})$, определим сопряженный оператор соответствия $D_T^*(w_i^{b'})$ над множеством $W_T^{B'}$. Этот оператор будет возвращать множество слов $D_T(w_i^{b'})$ в том случае, если оно непустое. Иначе, если $\|w_i^{b'}\| \geq 3$, он будет возвращать $D_T(\hat{w}_i^{b'})$, $\hat{w}_i^{b'} = Q^*(w_i^{b'})$. В противном случае он будет возвращать пустое множество.

$$D_T^*(w_i^{b'}) = \begin{cases} D_T(w_i^{b'}), D_T(w_i^{b'}) \neq \emptyset \\ D_T(\hat{w}_i^{b'}), \hat{w}_i^{b'} = Q^*(w_i^{b'}), \|w_i^{b'}\| \geq 3, D_T(w_i^{b'}) \neq \emptyset \\ \emptyset, \text{ в остальных случаях} \end{cases}$$

Базовый алгоритм. В том случае, если оператор $D_T^*(w_i^{b'})$ для исходного фонемного шаблона w_i^b возвращает непустое множество, это множество рассматривается в качестве искомого множества D^* и работа алгоритма на этом прекращается. В противном случае рассмотрим алгоритм разделения слова.

Алгоритм разделения слова. Данный алгоритм будет делить исходный шаблон на две части и пытаться определить слова, соответствующие каждой части исходного шаблона. Необходимость этого алгоритма обусловлена тем, что иногда два подряд проигранных слова распознаются в виде одного шаблона. Такое может возникнуть, например, если между этими словами будет распознан какой-нибудь лишний символ.

Для описания данного алгоритма рассмотрим ряд дополнительных операторов.

Оператор $L_i^W(w_i^{b'})$, $w_i^{b'} = (B_1', \dots, B_n') \in W_T^{B'}$, $i = \overline{2, n-1}$ будет возвращать составную часть шаблона $w_i^{b'}$ до i -го элемента не включительно. При этом справа будут убираться все пустые множества.

$$L_i^W(w_i^{b'}) = (B_1', \dots, B_j'), j < i, B_j \neq \emptyset, B_k = \emptyset, k = \overline{j+1, i-1}.$$

Аналогично оператор $R_i^W(w_i^{b'})$, $w_i^{b'} = (B_1^i, \dots, B_n^i) \in W_T^{B^i}$, $i = \overline{2, n-1}$ будет возвращать составную часть шаблона $w_i^{b'}$ после i -го элемента не включительно. При этом слева будут убираться все пустые множества.

$$R_i^W(w_i^{b'}) = (B_j^i, \dots, B_n^i), j > i, B_j \neq \emptyset, B_k = \emptyset, k = \overline{i-1, j+1}.$$

Для каждого пустого буквенного множества $B_i = \emptyset$ исходного шаблона слова w_i^b , начиная с конца, будем рассматривать шаблон $w_i^L \subset w_i^b$, полученный применением оператора $L_i^W(w_i^b)$. Для этого шаблона получим множество слов D_T^L , используя сопряженный оператор соответствия $D_T^*(w_i^L)$. В том случае, если $D_T^L = \emptyset$ пустое, переходим к следующему пустому множеству букв $B_i = \emptyset$ исходного шаблона w_i^b .

Если для некоторого пустого буквенного множества $B_i = \emptyset$ исходного шаблона слова w_i^b множество D_T^L будет не пустым, рассмотрим шаблон w_i^R , полученный применением соответствующего оператора $R_i^W(w_i^b)$. В противном случае в качестве w_i^R возьмем исходный шаблон w_i^b .

Далее будем обрабатывать шаблон w_i^R . Для каждого пустого буквенного множества $B_j = \emptyset$ этого шаблона, начиная с начала, будем рассматривать шаблон $w_i^{R_1} \subset w_i^R$, полученный применением оператора $R_j^W(w_i^R)$ к шаблону w_i^R . Для шаблона $w_i^{R_1}$ получим множество слов D_T^R , применяя оператор $D_T^*(w_i^{R_1})$ к этому шаблону. В том случае, если множество D_T^R непустое, прекращаем итерацию.

Если объединение множеств $D_T^L \cup D_T^R$ будет непустым, это множество будем рассматривать в качестве искомого множества D^* . В противном случае, если $\|w_i^b\| < 4$, считаем $D^* = \emptyset$. Иначе переходим к алгоритму исправления одинарной ошибки.

Алгоритм исправления одинарной ошибки. Необходимость данного алгоритма обусловлена тем, что иногда возникают ситуации, когда шаблон слова распознается с ошибками. Данный алгоритм по исходному шаблону слова строит шаблон, в котором вместо одного буквенного множества берет пустое множество, и по полученному шаблону ищет слова в словаре.

Для описания данного алгоритма рассмотрим оператор $H_i^W(w_i^{b'})$, $w_i^{b'} = (B_1^i, \dots, B_n^i) \in W_T^{B^i}$, $i = \overline{2, n-1}$. $H_i^W(w_i^{b'})$ будет возвращать шаблон $w_i^{b'}$, в котором вместо i -го множества букв будет пустое множество.

$$H_i^W(w_i^{b'}) = (B_1^i, \dots, B_{i-1}^i, \emptyset, B_{i+1}^i, B_n^i).$$

Для каждого непустого буквенного множества $B_i \neq \emptyset$, $i = \overline{2, n-1}$ исходного шаблона слова w_i^b будем рассматривать шаблон w_i^H , полученный из w_i^b применением оператора $H_i^W(w_i^b)$. Для шаблона w_i^H применением сопряженного оператора соответствия $D_T^*(w_i^{R_1})$ получим множество слов D_T^H .

В том случае, если множество D_T^H , берем это множество в качестве искомого множества D^* . Работа алгоритма на этом прекращается.

Если для всех непустых буквенных множеств $B_i \neq \emptyset$, $i = \overline{2, n-1}$ исходного шаблона слова, полученные из множества D_T^H , будут пустыми, в качестве D^* берем пустое множество.

6. ПРОГРАММНЫЙ КАРКАС СИСТЕМЫ КОММУНИКАЦИИ

Для системы тональной акустической коммуникации разработан программный каркас. Каркас представляет собой набор повторно используемых компонентов для решения прикладных задач с применением системы тональной акустической коммуникации.

Компоненты каркаса оформлены в виде классов на языке программирования C++. Некоторые классы реализованы окончательно. Часть классов являются абстрактными и представляют прогнозируемые точки роста каркаса, в которых осуществляется его доработка.

Применение программного каркаса системы связано с желанием сократить время разработки и обслуживания программного обеспечения, упростить его модификацию в различных приложениях (например, [4]).

На основе каркаса создана программная среда, которая позволяет осуществлять и анализировать работу рассматриваемых алгоритмов для системы, состоящей из двух роботов.

7. ЭКСПЕРИМЕНТАЛЬНЫЙ АНАЛИЗ РАБОТЫ СИСТЕМЫ

Тестирование проводилось для одночастотной, двухчастотной и трехчастотной систем. В качестве алфавита было рассмотрено множество букв русского алфавита. Каждой букве поставлено в соответствие множество, состоящее из одной фонемы, которая в свою очередь состоит из одного фрагмента длительностью 150 мс и амплитудой 100. Частоты фрагментов, которые использовались для каждой буквы, приведены в таблице 7.1. Размер словаря роботов составлял 150 слов.

Параметры алгоритма: $\tau = 50$, $a_{\min} = 50$, $\sigma = 0.9$, $\alpha = 0.7$, $\gamma = 0.5$, $a_n = 50$. Эти параметры вместе с выбором языка отвечали наилучшему качеству распознавания.

Таблица 7.1.

Соответствие между буквами алфавита и частотами фрагментов

Буква	Одночастотная система, Гц	Двухчастотная система, Гц	Трехчастотная система, Гц
а	400	400, 1000	400, 800, 1100
б	450	400, 1100	400, 800, 1200
в	500	400, 1200	400, 800, 1300
г	550	400, 1300	400, 900, 1100
д	600	400, 1400	400, 900, 1200
е	650	400, 1500	400, 900, 1300

ж	700	500, 1000	400,1000,1100
з	750	500, 1100	400,1000,1200
и	800	500, 1200	400,1000,1300
й	850	500, 1300	500,800,1100
к	900	500, 1400	500,800,1200
л	950	500, 1500	500,800,1300
м	1000	600, 1000	500,900,1100
н	1050	600, 1100	500,900,1200
о	1100	600, 1200	500,900,1300
п	1150	600, 1300	500,1000,1100
р	1200	600, 1400	500,1000,1200
с	1250	600, 1500	500,1000,1300
т	1300	700, 1000	600,800,1100
у	1350	700, 1100	600,800,1200
ф	1400	700, 1200	600,800,1300
х	1450	700, 1300	600,900,1100
ц	1500	700, 1400	600,900,1200
ч	1550	700, 1500	600,900,1300
ш	1600	800, 1000	600,1000,1100
щ	1650	800, 1100	600,1000,1200
ъ	1700	800, 1200	600,1000,1300
ы	1750	800, 1300	700,1000,1100
ь	1800	800, 1400	700,1000,1200
э	1850	800, 1500	700,1000,1300
ю	1900	900, 1000	700,800,1100
я	1950	900, 1100	700,800,1200

Эксперименты проводились в помещении общей площадью 20 кв.м. в условиях тишины и с шумом. В качестве шума использовался человеческий голос средней громкости. На разных расстояниях между микрофоном "распознающей" машины и динамиком "проигрывающей" машины десять раз проигрывался звуковой сигнал, соответствующей фразе «я робот я умею разговаривать на тональном акустическом языке я общаюсь с другими роботами используя мультисигнальные аудио сигналы». После чего результат распознавания сравнивался с исходной фразой. Критерием качества работы алгоритма служил процент правильно распознанных слов. Результаты эксперимента представлены в таблице 7.2.

Таблица 7.2.

Результаты экспериментов

Расстояние, м.	Одночастотная система, %	Двухчастотная система, %	Трехчастотная система, %
БЕЗ ШУМА			
1	100	100	100
2	98.5	99	99.5
3	97	97.5	98
4	95	96	97.5
5	93	94.5	95.5
6	91.5	93	94

7	89.5	90.5	92
<i>С ШУМОМ</i>			
1	76	82	85
2	75	80	83.5
3	72.5	77.5	81
4	71	75	79
5	68	72.5	76.5
6	66.5	70	73
7	64	68	71

Из результатов видно, что в условиях тишины качество распознавания систем с различной частотностью примерно одинаковое. В условиях шумов система с более высокой частотностью оказалась надежнее.

8. ЗАКЛЮЧЕНИЕ

В работе выполнено построение модели акустического тонального мультисигнального языка роботов со словарем и разработан алгоритм выделения звуков, соответствующих фонемам этого языка.

Выполнена реализация программной среды моделирования. Проведенные эксперименты подтвердили более устойчивую работу системы в условиях посторонних шумов по сравнению с моночастотной системой [1-3].

Передача акустических сигналов по надежности и скорости уступает передаче данных, например, по радиоканалу. Однако предложенная система может использоваться в тех случаях, где неприменимы радиосигналы, например, в водной среде [4].

В дальнейшем планируется создать версии акустического языка, которые будут работать на частотах ультразвука. Цель – создание таких версий языков, которые будут более устойчивы к помехам.

ЛИТЕРАТУРА

1. А.Ю.Кирков, В.Е.Павловский. Акустический тональный язык коммуникации роботов // НТИ, Информационные процессы и системы. ISSN 0548-0027. Серия 2, № 2, 2013, с.8-15.
2. А.Ю.Кирков, В.Е.Павловский. Искусственный тональный язык акустической коммуникации роботов // Искусственный интеллект. ISSN 1561-5359. № 4, 2012, с. 440-458.
3. В.Е.Павловский, А.Ю.Кирков. Тональная акустическая коммуникация роботов // Препринты ИПМ им.М.В.Келдыша. 2012. № 85. 24 с. URL: <http://library.keldysh.ru/preprint.asp?id=2012-85>
4. В.Е.Павловский, К.В.Герасимов, А.Ю.Кирков. Надводный беспилотник // Тр. 6-й Всероссийской мультikonференции по проблемам управления МКПУ-2013. 30.09.-05.10.2013. Дивноморское, Россия. Изд-во ЮФУ. т. 2. с.197-200.