



УДК 004.934

## РАЗРАБОТКА КОМПОНЕНТОВ ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ ПОТОКОВОЙ ФИЛЬТРАЦИИ АУДИОКОНТЕНТА НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ СКРЫТЫХ МАРКОВСКИХ МОДЕЛЕЙ

А. Н. Савин<sup>1</sup>, Н. Е. Тимофеева<sup>2</sup>, А. С. Гераськин<sup>3</sup>, Ю. А. Мавлютова<sup>4</sup>

<sup>1</sup>Савин Александр Николаевич, кандидат физико-математических наук, доцент кафедры дискретной математики и информационных технологий, Саратовский государственный университет им. Н. Г. Чернышевского, savinan@info.sgu.ru

<sup>2</sup>Тимофеева Надежда Евгеньевна, ассистент кафедры дискретной математики и информационных технологий, Саратовский государственный университет им. Н. Г. Чернышевского, timofeevane@yandex.ru

<sup>3</sup>Гераськин Алексей Сергеевич, кандидат педагогических наук, доцент кафедры теоретических основ компьютерной безопасности и криптографии, Саратовский государственный университет им. Н. Г. Чернышевского, gerascinas@mail.ru

<sup>4</sup>Мавлютова Юлия Альбертовна, студентка кафедры теоретических основ компьютерной безопасности и криптографии, Саратовский государственный университет им. Н. Г. Чернышевского, yuliyamav@yandex.ru

Приведены результаты разработки эффективных алгоритмов потокового распознавания речи с помощью стохастических моделей, основанных на использовании скрытых марковских моделях. В статье приводятся основные теоретические сведения для скрытой марковской модели дискретной системы, выделяются параметры, необходимые для ее определения, также рассмотрены три основные задачи, которые должны быть решены для успешного применения скрытой марковской модели в системах распознавания речи. Приводятся алгоритмы метода Баума–Велша, направленного на уточнение параметров модели и метода Витерби, подбора наиболее вероятной последовательности состояний системы. Эти два метода реализованы в среде графического программирования LabVIEW в виде программных модулей, реализующих построение скрытых марковских моделей отдельных слов с использованием метода Баума–Велша и распознавание этих слов на основе метода Витерби. Предполагается использование данных модулей для реализации потоковой фильтрации аудиоконтента в цифровых системах связи.

*Ключевые слова:* марковские цепи, распознавание речи, алгоритм метода Витерби, метод Баума–Велша.

DOI: 10.18500/1816-9791-2015-15-3-340-350

### ВВЕДЕНИЕ

Важными задачами при разработке систем передачи информации являются математическое моделирование их узлов и процессов, происходящих при передаче информации. Одной из таких задач является моделирование сигналов, с помощью которых осуществляется передача информации [1].

Сигналы по своей природе могут быть непрерывными (например, фрагменты речи, музыка и т. п.) или дискретными (символы конечного алфавита и т. п.). Если статистические свойства сигналов не изменяются во времени, то такие сигналы являются стационарными, а в противном случае — нестационарными [2]. Сигналы в системах передачи информации, как правило, искажаются различными шумами и помехами от других источников [1]. В этом случае модель сигнала можно использовать для создания системы, оптимально удаляющей эти шумы и искажения. В случае, когда получение сигнала от реального источника затруднительно, хорошая модель сигнала позволяет имитировать источник и соответственно разрабатывать оптимальные системы приёма информации для таких источников.

Математические модели сигналов широко используются также в системах передачи информации для решения задач идентификации, предсказания, распознавания и т. д. [3].

Для описания свойств некоторого заданного сигнала можно использовать детерминированные и стохастические модели [1, 2]. При этом в детерминированных моделях используются некоторые известные специфические свойства сигнала, например, то, что он является синусоидальным. Тогда для построения модели достаточно определить значения амплитуды, частоты и начальной фазы. Стохастические модели охарактеризовывают только статистические свойства сигналов. При этом считается, что сигнал может быть хорошо описан некоторым параметрическим случайным процессом (например,



гауссовским, пуассоновским, марковским и т. п.), а параметры этого процесса могут быть достаточно точно оценены каким-либо способом.

В современных цифровых системах связи задачу качественной передачи звуковой информации (разборчивость, низкий уровень шумов) при жестких требованиях к спектральному составу передаваемого сигнала успешно решают с помощью различных методов цифровой фильтрации [1, 2]. В настоящее время уровень развития вычислительной техники, используемой в цифровых системах звуковой связи, позволяет решать задачу контроля и фильтрации передаваемого контента на основе потокового распознавания речи. При этом распознавание речи может быть осуществлено с помощью стохастических моделей, основанных на использовании скрытых марковских моделей (СММ) [3].

Данная работа посвящена разработке компонентов программного комплекса для потоковой фильтрации аудиоконтента на основе СММ, решающих задачи обучения по заданной последовательности наблюдений параметров СММ, поиска последовательности состояний СММ по заданной последовательности наблюдений и вычисления вероятности того, что заданная последовательность наблюдений построена для данной СММ [4].

Для реализации компонентов была выбрана среда графического программирования LabVIEW компании National Instruments (NI), так как она обеспечивает быстрое создание эффективных программ, а также она исходно создавалась для обработки информационных сигналов систем передачи и соответственно в ней встроены основные алгоритмы цифровой обработки информации [5].

## 1. СКРЫТЫЕ МАРКОВСКИЕ МОДЕЛИ

### 1.1. Определение скрытой марковской модели

Основы теории скрытых марковских моделей приведены в работах Баума в конце 1960-х годов [3]. В последнее время СММ стали очень популярны, так как математическая структура СММ очень богата и позволяет решать математические проблемы различных областей науки. При этом грамотно спроектированная модель выдаёт на практике хорошие результаты работы. Результаты применения СММ в приложениях, связанных с распознаванием речи, опубликованы Бейкером и Елинеком в 1970-х годах [3].

В обзорной работе Л. Р. Рабинера [3] излагаются основы теории СММ для дискретных систем, а также приведены рекомендации по практическому использованию её результатов для решения задач автоматического распознавания речи. Далее будут рассматриваться именно такие СММ.

Рассмотрим дискретную систему, имеющую конечное множество из  $N$  состояний —  $S = \{s_1, \dots, s_N\}$ , в каждом из которых она может принимать одно из  $M$  значений из набора наблюдаемых параметров  $V = \{v_1, \dots, v_M\}$  — алфавита. Состояние системы в момент времени  $t$  —  $q_t$ , принимающее одно из  $N$  значений множества  $S$ , зависит только от её состояния в момент времени  $t - 1$  —  $q_{t-1}$ , а значение наблюдаемого параметра в момент времени  $t$  —  $o_t$  зависит только от состояния  $q_t$ , т. е. не зависит от времени. Вероятности переходов между состояниями системы задаются матрицей  $A$ . Вероятности выпадения каждого из  $M$  значений наблюдаемого параметра системы в каждом из  $N$  состояний системы задаются набором векторов  $B$ . Вероятность появления некоторого начального состояния системы задаётся вектором  $\pi$ . При этом последовательность состояний, в которых пребывает система —  $Q = \{q_1, \dots, q_T\}$ , внешнему наблюдателю не видна, а видит он только последовательность наблюдений —  $O = \{o_1, \dots, o_T\}$  (здесь  $T$  — длина последовательности), т. е. система ведёт себя как «чёрный ящик». Модель такой системы получила название скрытой марковской модели.

Для полного определения СММ дискретной системы должны быть заданы следующие параметры:

- Число состояний в множестве состояний модели  $S = \{s_1, \dots, s_N\}$  —  $N$ ;
- Алфавит модели —  $V = \{v_1, \dots, v_M\}$  и число символов в нём —  $M$ ;
- Вероятности переходов между состояниями модели —  $A = \{a_{ij}\}$ , где  $a_{ij}$  — вероятность перехода

модели из состояния  $s_i$  в состояние  $s_j$ :

$$a_{ij} = p(q_{t+1} = s_j \mid q_t = s_i), \quad 1 \leq i \leq N, \quad 1 \leq j \leq N. \quad (1)$$

Элементы  $A$  должны удовлетворять стохастическим ограничениям:

$$a_{ij} \geq 0, \quad 1 \leq i \leq N, \quad 1 \leq j \leq N \quad \text{и} \quad \sum_{j=1}^N a_{ij} = 1, \quad 1 \leq i \leq N. \quad (2)$$

- Вероятности выпадения каждого из  $M$  значений наблюдаемого параметра модели в каждом из  $N$  состояний модели —  $B = \{b_j(k)\}$ , где  $b_j(k)$  — вероятность выпадения  $k$ -го значения наблюдаемого параметра модели в состоянии  $s_j$ :

$$b_j(k) = p(o_t = v_k \mid q_t = s_j), \quad 1 \leq j \leq N, \quad 1 \leq k \leq M. \quad (3)$$

Элементы  $B$  должны удовлетворять следующим ограничениям:

$$b_j(k) \geq 0, \quad 1 \leq j \leq N, \quad 1 \leq k \leq M \quad \text{и} \quad \sum_{k=1}^M b_j(k) = 1, \quad 1 \leq j \leq N. \quad (4)$$

- Вероятность появления некоторого начального состояния модели —  $\pi = \{\pi_i\}$ , где  $\pi_i$  — вероятность того, что в начальный момент модель окажется в состоянии  $s_i$ :

$$\pi_i = p(q_1 = s_i), \quad 1 \leq i \leq N. \quad (5)$$

Элементы  $\pi$  должны удовлетворять следующим ограничениям:

$$\pi_i \geq 0, \quad 1 \leq i \leq N \quad \text{и} \quad \sum_{i=1}^N \pi_i = 1, \quad 1 \leq i \leq N. \quad (6)$$

В компактной записи СММ дискретной системы обозначается следующим образом:

$$\lambda = (A, B, \pi). \quad (7)$$

## 1.2. Три основных задачи скрытой марковской модели

Согласно описанию СММ, изложенному в предыдущем параграфе, существует три основных задачи, которые должны быть решены для успешного применения модели в какой-либо области, в том числе и при распознавании речи.

**Задача 1.** Дано: наблюдаемая последовательность  $O = \{o_1, \dots, o_T\}$  и модель  $\lambda = (A, B, \pi)$ . Необходимо вычислить  $p(O \mid \lambda)$  — вероятность того, что данная наблюдаемая последовательность построена именно для данной модели.

Это задача оценки модели, которая заключается в вычислении вероятности того, что модель соответствует заданной наблюдаемой последовательности. К этой задаче можно подойти и с другой стороны: насколько выбранная СММ соответствует заданной наблюдаемой последовательности. Такой подход имеет большую практическую ценность. Например, если стоит вопрос выбора наилучшей модели из набора уже существующих, то решение первой задачи даёт ответ на этот вопрос.

**Задача 2.** Дано: наблюдаемая последовательность  $O = \{o_1, \dots, o_T\}$  и модель  $\lambda = (A, B, \pi)$ . Необходимо подобрать последовательность состояний системы  $Q = \{q_1, \dots, q_T\}$ , которая лучше всего соответствует наблюдаемой последовательности, т. е. «объясняет» наблюдаемую последовательность.



Задача 2 позволяет понять, что же происходит в скрытой части модели, т. е. найти «правильную» последовательность, которую проходит модель. Абсолютно точно нельзя определить эту последовательность. Говорить можно лишь о предположениях с соответственной степенью достоверности. При этом для приближенного решения этой проблемы приходится использовать различные критерии оценки для определения последовательности состояний. Данные, полученные при решении задачи 2, используются для изучения поведения построенной модели, нахождения оптимальной последовательности её состояний и т. п.

**Задача 3.** Подобрать параметры модели  $\lambda = (A, B, \pi)$  таким образом, чтобы максимизировать  $p(O | \lambda)$ .

Решение задачи 3 состоит в оптимизации модели таким образом, чтобы она как можно лучше описывала реальную наблюдаемую последовательность. Наблюдаемая последовательность, по которой оптимизируется СММ, называется обучающей, поскольку с помощью её модель «обучается». Задача обучения СММ — один из важнейших при проектировании СММ, поскольку она заключается в оптимизации параметров СММ, т. е. создается модель, наилучшим образом описывающая реальные процессы.

Для построения программного комплекса потоковой фильтрации аудиоконтента необходимо разработать программные модули, решающие эти три задачи.

### 1.3. Алгоритм метода Баума–Велша обучения СММ системы

Исходными данными для алгоритма метода Баума–Велша являются СММ со случайными параметрами  $\lambda = (A, B, \pi)$  и заданная обучающая последовательность наблюдений  $O = \{o_1, \dots, o_T\}$ . Алгоритм позволяет уточнять параметры модели таким образом, чтобы вероятность  $p(O | \lambda)$  увеличивалась. Итеративное применение алгоритма до схождения в одной точке позволяет максимизировать  $p(O | \lambda)$ , т. е. настроить модель на данную последовательность и соответственно решить задачу 3.

Рассмотрим алгоритм подробнее. Он состоит из нескольких шагов.

- **Прямая процедура** определяет «прямые» переменные — вероятности получения начальной части (от момента времени 1 до  $t$ ) заданной последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  при условии, что движение начинается в момент времени 1 и заканчивается в момент времени  $t$  в состоянии  $s_i$ :

$$\alpha_t(i) = p(o_1, \dots, o_t, q_t = s_i | \lambda). \tag{8}$$

Эти вероятности можно вычислить рекурсивно, используя следующее выражение:

$$\alpha_{t+1}(j) = b_j(o_{t+1}) \sum_{i=1}^N \alpha_t(i) a_{ij}, \quad 1 \leq j \leq N, \quad 1 \leq t \leq T - 1, \tag{9}$$

где

$$\alpha_1(j) = \pi_j b_j(o_1), \quad 1 \leq j \leq N. \tag{10}$$

Используя рекурсию, можно найти

$$\alpha_T(i), \quad 1 \leq i \leq N.$$

Теперь с учётом (8) требуемую для решения задачи 1 вероятность можно вычислить через «прямые» переменные:

$$p(O | \lambda) = \sum_{i=1}^N \alpha_T(i). \tag{11}$$

- **Обратная процедура** является дополнением прямой процедуры и позволяет вычислить «обратные» переменные — вероятности получения оставшейся части (от момента времени  $t + 1$



до  $T$ ) заданной последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  при условии, что движение начинается в момент времени  $t$  из исходного состояния  $s_i$  и заканчивается в момент времени  $T$ :

$$\beta_t(i) = p(o_{t+1}, \dots, o_T | q_t = s_i, \lambda). \quad (12)$$

Эти вероятности можно вычислить рекурсивно, используя следующее выражение:

$$\beta_t(i) = \sum_{j=1}^N \beta_{t+1} a_{ij} b_j(o_{t+1}), \quad 1 \leq i \leq N, \quad 1 \leq t \leq T-1, \quad (13)$$

где

$$\beta_T(i) = 1, \quad 1 \leq i \leq N. \quad (14)$$

Используя рекурсию, можно найти

$$\beta_1(i), \quad 1 \leq i \leq N.$$

Теперь с учётом (10) и (12) требуемую для решения задачи 1 вероятность можно вычислить через «обратные» переменные:

$$p(O | \lambda) = \sum_{i=1}^N \alpha_1(i) \beta_1(i) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i). \quad (15)$$

С другой стороны, из (8) и (12) следует:

$$p(O, q_t = s_i | \lambda) = \alpha_t(i) \beta_t(i), \quad 1 \leq i \leq N, \quad 1 \leq t \leq T. \quad (16)$$

Используя (16), можно вычислить требуемую для решения задачи 1 вероятность через «прямые» и «обратные» переменные:

$$p(O | \lambda) = \sum_{i=1}^N p(O, q_t = s_i | \lambda) = \sum_{i=1}^N \alpha_t(i) \beta_t(i), \quad 1 \leq t \leq T. \quad (17)$$

- **Вычисление вспомогательной переменной  $\xi$**  — вероятности того, что при заданной последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  модель в моменты времени  $t$  и  $t+1$  будет находиться в состояниях  $s_i$  и  $s_j$  соответственно:

$$\xi_t(i, j) = p(q_t = s_i, q_{t+1} = s_j | O, \lambda) = \frac{p(q_t = s_i, q_{t+1} = s_j, O | \lambda)}{p(O | \lambda)}. \quad (18)$$

Используя «прямые» (8) и «обратные» (12) переменные, можно записать (18) следующим образом:

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} \beta_{t+1}(j) b_j(o_{t+1})}{\sum_{i_s=1}^N \sum_{j_s=1}^N \alpha_t(i_s) a_{i_s j_s} \beta_{t+1}(j_s) b_{j_s}(o_{t+1})}, \quad 1 \leq t \leq T-1. \quad (19)$$

- **Вычисление вспомогательной переменной  $\gamma$**  — вероятности того, что при заданной последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  модель в момент времени  $t$  будет находиться в состоянии  $s_i$ :

$$\gamma_t(i) = p(q_t = s_i | O, \lambda) = \frac{p(O, q_t = s_i | \lambda)}{p(O | \lambda)}. \quad (20)$$

Используя «прямые» (8) и «обратные» (12) переменные и учитывая соотношения (16), (17), можно записать (20) следующим образом:

$$\gamma_t(i) = \frac{\alpha_t(i) \beta_t(i)}{\sum_{i_s=1}^N \alpha_t(i_s) \beta_t(i_s)}, \quad 1 \leq t \leq T. \quad (21)$$



Из выражений (19) и (20) видно, что вспомогательные переменные  $\xi_t(i, j)$  и  $\gamma_t(i)$  связаны между собой следующим образом:

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j), \quad 1 \leq i \leq N, \quad 1 \leq t \leq T-1. \quad (22)$$

- **Уточнение параметров исходной СММ  $\lambda = (A, B, \pi)$  с целью увеличения  $p(O | \lambda)$ .**

В соответствии с алгоритмом метода Баума – Велша параметры уточнённой СММ  $\lambda^* = (A^*, B^*, \pi^*)$  определяются с помощью следующих соотношений:

- 1) уточнённая вероятность того, что в начальный момент модель окажется в состоянии  $s_i - \pi_i^*$ :

$$\pi_i^* = \gamma_1(i), \quad 1 \leq i \leq N, \quad (23)$$

- 2) уточнённая вероятность перехода модели из состояния  $s_i$  в состояние  $s_j - a_{ij}^*$ :

$$a_{ij}^* = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, \quad 1 \leq i \leq N, \quad 1 \leq j \leq N, \quad (24)$$

- 3) уточнённая вероятность выпадения  $k$ -го значения наблюдаемого параметра модели в состоянии  $s_j - b_j^*(k)$ :

$$b_j^*(k) = \frac{\sum_{t=1: o_t=v_k}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}, \quad 1 \leq j \leq N, \quad 1 \leq k \leq M, \quad (25)$$

здесь в числителе суммируются только те  $\gamma_t(j)$ , для которых  $o_t = v_k$ .

- **Вычисление для уточнённой СММ  $\lambda^* = (A^*, B^*, \pi^*)$  вероятности  $p^*(O | \lambda)$ .**

Используя новые значения параметров СММ  $\lambda^* = (A^*, B^*, \pi^*)$ , по одному из соотношений (11), (15) или (17) вычисляется вероятность того, что заданная наблюдаемая последовательность  $O = \{o_1, \dots, o_T\}$  построена именно для данной модели —  $p(O | \lambda^*)$ . При этом алгоритм метода Баума – Велша обеспечивает неуменьшение этой вероятности, т. е. правдоподобность модели после уточнения выше, чем до уточнения во всех случаях, когда модель можно оптимизировать.

- **Проверка сходимости итерационного процесса обучения СММ  $\lambda = (A, B, \pi)$ .**

Итерационный процесс обучения СММ  $\lambda = (A, B, \pi)$  по заданной последовательности наблюдений —  $O = \{o_1, \dots, o_T\}$  — максимизация вероятности  $p(O | \lambda)$  — заканчивается, когда разница между вероятностями  $p(O | \lambda^*)$  и  $p(O | \lambda)$  становится менее некоторой заданной точности  $\varepsilon$ , т. е. выполняется следующее условие:

$$p(O | \lambda^*) - p(O | \lambda) < \varepsilon. \quad (26)$$

#### 1.4. Алгоритм вычисления последовательности состояний системы, описываемой СММ, по последовательности наблюдений методом Витерби

Задача 2 — подбор наиболее вероятной последовательности состояний  $Q = \{q_1, \dots, q_T\}$  системы, описываемой СММ  $\lambda = (A, B, \pi)$ , соответствующей последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  — может быть решена с помощью метода Витерби [4].

Алгоритм метода Витерби состоит из следующих шагов:

- **Вычисление вспомогательной переменной  $\delta$  — максимальной вероятности того, что при заданной последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  модель в момент времени  $t$  (при условии, что движение начинается в момент времени 1 и заканчивается в момент времени  $t$ ) будет находиться в состоянии  $s_i$ :**

$$\delta_t(i) = \max_{q_1, \dots, q_{t-1}} p(q_1, \dots, q_{t-1}, q_t = s_i, o_1, \dots, o_{t-1} | \lambda). \quad (27)$$

Эти вероятности можно вычислить рекурсивно, используя следующее выражение:

$$\delta_{t+1}(j) = b_j(o_{t+1}) \cdot \max_{1 \leq i \leq N} (\delta_t(i) \cdot a_{ij}), \quad 1 \leq j \leq N, \quad 1 \leq t \leq T-1, \quad (28)$$

где

$$\delta_1(j) = \pi_j b_j(o_1), \quad 1 \leq j \leq N. \quad (29)$$

При этом надо запоминать не только значения вероятностей  $\delta_t(j)$ , но и состояния  $\psi_t(j)$ , при которых  $\delta_t(j)$  достигает максимума:

$$\psi_{t+1}(j) = \arg \max_{1 \leq i \leq N} (\delta_t(i) \cdot a_{ij}), \quad 1 \leq j \leq N, \quad 1 \leq t \leq T-1. \quad (30)$$

Используя рекурсию, можно найти

$$\delta_T(i), \quad 1 \leq i \leq N. \quad (31)$$

• **Вычисление наиболее вероятного конечного состояния системы  $q_t$ .**

После шага  $T-1$  вычисляется наиболее вероятное конечное состояние системы:

$$q_T = \arg \max_{1 \leq i \leq N} (\delta_T(i)), \quad (32)$$

где  $\max_{1 \leq i \leq N} (\delta_T(i))$  — наибольшая из максимальных вероятностей нахождения системы в момент времени  $T$  в состоянии  $s_i$ .

• **Вычисление наиболее вероятной последовательности состояний системы  $Q = \{q_1, \dots, q_T\}$ .**

Вычисление наиболее вероятной последовательности состояний  $Q = \{q_1, \dots, q_T\}$  системы, соответствующей последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  — решение задачи 2, осуществляется обратным проходом по массиву состояний  $\psi_t$ , при которых вероятности  $\delta_t(i)$  максимальны, начиная с наиболее вероятного конечного состояния  $q_T$ :

$$q_t = \psi_{t+1}(q_{t+1}), \quad 1 \leq t \leq T-1. \quad (33)$$

**2. РЕАЛИЗАЦИЯ АЛГОРИТМА «ПРЯМОЙ ПРОЦЕДУРЫ»**

Вычисление  $p(O | \lambda)$  — вероятности того, что наблюдаемая последовательность  $O = \{o_1, \dots, o_T\}$  построена для СММ  $\lambda = (A, B, \pi)$  (т.е. решение задачи 1), может быть осуществлено с помощью «прямой процедуры», приведённой в 1.3 при описании алгоритма метода Баума – Велша.

Эта «прямая процедура» была реализована и отлажена в среде графического программирования LabVIEW в виде отдельного программного модуля.

Входными параметрами для модуля являются:

- СММ  $\lambda = (A, B, \pi)$  с числом состояний  $N$  и числом значений  $M$ , параметры которой удовлетворяют ограничениям (2), (4), (6). Алфавитом  $V = \{v_1, \dots, v_M\}$  СММ являются номера значений от 1 до  $M$ .
- Последовательность наблюдений  $O = \{o_1, \dots, o_T\}$  длиной  $T$ .

Выходными параметрами модуля являются:

- Вычисленное значение  $p(O | \lambda)$  — вероятности того, что наблюдаемая последовательность  $O = \{o_1, \dots, o_T\}$  построена для СММ  $\lambda = (A, B, \pi)$ .

На рис. 1 представлены результаты вычисления с помощью программного модуля, реализующего «прямую процедуру»,  $p(O | \lambda)$  — вероятности того, что наблюдаемая последовательность построена для СММ  $\lambda = (A, B, \pi)$ .

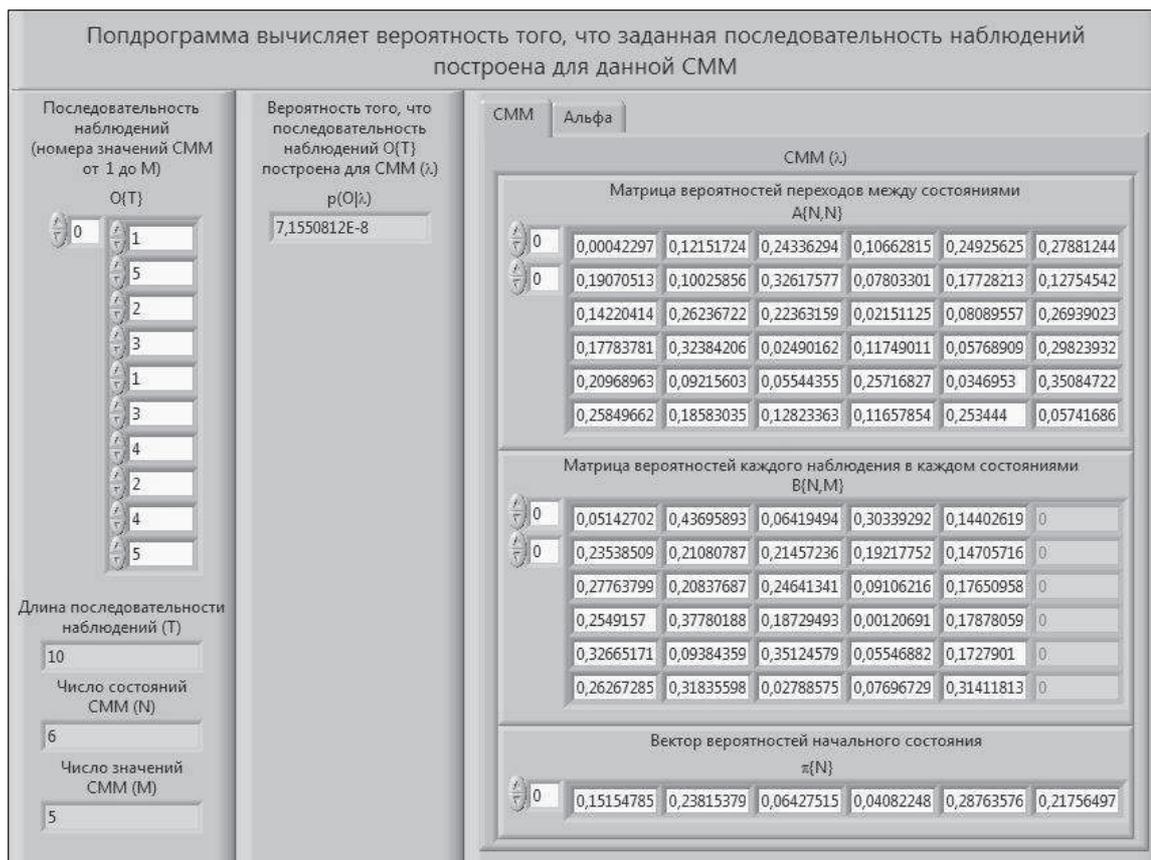


Рис. 1. Лицевая панель программного модуля, реализующего «прямую процедуру»

### 3. РЕАЛИЗАЦИЯ АЛГОРИТМА МЕТОДА БАУМА – ВЕЛША

На основе представленного выше в 1.3 алгоритма метода Баума – Велша в среде графического программирования LabVIEW был разработан и отлажен программный модуль, предназначенный для обучения СММ  $\lambda = (A, B, \pi)$  по заданной последовательности наблюдений  $O = \{o_1, \dots, o_T\}$ .

Входными параметрами для модуля являются:

- Исходная СММ  $\lambda = (A, B, \pi)$  с числом состояний  $N$  и числом значений  $M$ , параметры которой выбраны случайным образом, но при этом удовлетворяют ограничениям (2), (4), (6). Алфавитом  $V = \{v_1, \dots, v_M\}$  СММ являются номера значений от 1 до  $M$ ;
- Последовательность наблюдений  $O = \{o_1, \dots, o_T\}$  длиной  $T$ ;
- Заданная точность  $\epsilon$ ;
- Максимально возможное число итераций.

Выходными параметрами модуля являются:

- Обученная по последовательности наблюдений  $O = \{o_1, \dots, o_T\}$  СММ  $\lambda^* = (A^*, B^*, \pi^*)$ ;
- Значение вероятности  $p(O | \lambda)$  исходной СММ  $\lambda = (A, B, \pi)$ ;
- Значение вероятности  $p(O | \lambda^*)$  обученной СММ  $\lambda^* = (A^*, B^*, \pi^*)$ ;
- Число итераций, потребовавшееся для обучения.

Лицевая панель программного модуля, реализующего алгоритм метода Баума – Велша, с исходной СММ  $\lambda = (A, B, \pi)$  и полученной в результате обучения СММ  $\lambda^* = (A^*, B^*, \pi^*)$ , приведены на рис. 2 и рис. 3 соответственно.



Попрограмма обучает СММ по заданной последовательности наблюдений методом Баума-Велша

Последовательность наблюдений (номера значений СММ от 1 до M)  
O{T}

Максимальное число итераций: 1000

Точность (ε): 1E-12

p(O|i) исходной СММ: 7,1550812E-8

Число итераций: 14

p(O|i\*) обученной СММ: 0,015625

Исходная СММ | Альфа | Вета | Кси | Гамма | Обученная СММ

Исходная СММ (i)

Матрица вероятностей переходов между состояниями A{N,N}

0	0,00042297	0,12151724	0,24336294	0,10662815	0,24925625	0,27881244
0	0,19070513	0,10025856	0,32617577	0,07803301	0,17728213	0,12754542
	0,14220414	0,26236722	0,22363159	0,02151125	0,08089557	0,26939023
	0,17783781	0,32384206	0,02490162	0,11749011	0,05768909	0,29823932
	0,20968963	0,09215603	0,05544355	0,25716827	0,0346953	0,35084722
	0,25849662	0,18583035	0,12823363	0,11657854	0,253444	0,05741686

Матрица вероятностей каждого наблюдения в каждом состояниями B{N,M}

0	0,05142702	0,43695893	0,06419494	0,30339292	0,14402619	0
0	0,23538509	0,21080787	0,21457236	0,19217752	0,14705716	0
	0,27763799	0,20837687	0,24641341	0,09106216	0,17650958	0
	0,2549157	0,37780188	0,18729493	0,00120691	0,17878059	0
	0,32665171	0,09384359	0,35124579	0,05546882	0,1727901	0
	0,26267285	0,31835598	0,02788575	0,07696729	0,31411813	0

Вектор вероятностей начального состояния π{N}

0	0,15154785	0,23815379	0,06427515	0,04082248	0,28763576	0,21756497
---	------------	------------	------------	------------	------------	------------

Рис. 2. Лицевая панель программного модуля, реализующего алгоритм метода Баума – Велша с исходной СММ

Попрограмма обучает СММ по заданной последовательности наблюдений методом Баума-Велша

Последовательность наблюдений (номера значений СММ от 1 до M)  
O{T}

Максимальное число итераций: 1000

Точность (ε): 1E-12

p(O|i) исходной СММ: 7,1550812E-8

Число итераций: 14

p(O|i\*) обученной СММ: 0,015625

Исходная СММ | Альфа | Вета | Кси | Гамма | Обученная СММ

Обученная СММ (i\*)

Матрица вероятностей переходов между состояниями A\*{N,N}

0	0	0	0	0,16971051	0	0,83028949
0	0	0	1	0	0	0
	0,5	0,5	0	0	0	0
	0	1	0	0	0	0
	0	0	0	1	0	0
	1	0	0	0	0	0

Матрица вероятностей каждого наблюдения в каждом состояниями B\*{N,M}

0	0	0	0	1	0	0
0	0,5	0,5	0	0	0	0
	0	0	1	0	0	0
	0	0	0	0	1	0
	1	0	0	0	0	0
	0	0,6021996	0	0	0,3978004	0

Вектор вероятностей начального состояния π\*{N}

0	0	0	0	0	1	0
---	---	---	---	---	---	---

Рис. 3. Лицевая панель программного модуля, реализующего алгоритм метода Баума – Велша с обученной СММ



В результате применения метода Баума – Велша (см. рис. 2, 3) значение вероятности  $p(O | \lambda) = 7.1550812 \cdot 10^{-8}$  исходной СММ  $\lambda = (A, B, \pi)$  увеличилось до значения вероятности  $p(O | \lambda^*) = 1.15625 \cdot 10^{-2}$  обученной СММ  $\lambda^* = (A^*, B^*, \pi^*)$ . При этом параметры обученной СММ так же как и исходной СММ удовлетворяют ограничениям (2), (4), (6), а для достижения заданной точности  $\varepsilon = 10^{-12}$  потребовалось всего 14 итераций. Дальнейшее повышение точности (уменьшение значения  $\varepsilon$ ) значение вероятности  $p(O | \lambda^*)$  не изменяло, т.е. действительно был найден максимум  $p(O | \lambda^*)$ . Однако надо учитывать, что обучение СММ методом Баума – Велша обеспечивает поиск локального максимума [4], и при других начальных параметрах СММ  $\lambda = (A, B, \pi)$  результаты обучения могут быть лучше.

#### 4. РЕАЛИЗАЦИЯ АЛГОРИТМА МЕТОДА ВИТЕРБИ

На основе представленного выше в 1.4 алгоритма метода Витерби в среде графического программирования LabVIEW был разработан и отлажен программный модуль, предназначенный для вычисления наиболее вероятной последовательности состояний  $Q = \{q_1, \dots, q_T\}$  системы, описываемой СММ  $\lambda = (A, B, \pi)$ , соответствующей последовательности наблюдений  $O = \{o_1, \dots, o_T\}$ .

Входными параметрами для модуля являются:

- СММ  $\lambda = (A, B, \pi)$  с числом состояний  $N$  и числом значений  $M$ , параметры которой удовлетворяют ограничениям (2), (4), (6). Алфавитом  $V = \{v_1, \dots, v_M\}$  СММ являются номера значений от 1 до  $M$ ;
- Последовательность наблюдений  $O = \{o_1, \dots, o_T\}$  длиной  $T$ .

Выходными параметрами модуля являются:

- Наиболее вероятная последовательность состояний  $Q = \{q_1, \dots, q_T\}$  системы, описываемой СММ  $\lambda = (A, B, \pi)$ , соответствующая последовательности наблюдений  $O = \{o_1, \dots, o_T\}$ .

Лицевая панель программного модуля, реализующего алгоритм метода Витерби с заданной последовательностью наблюдений  $O = \{o_1, \dots, o_T\}$  и полученной в результате вычисления наиболее вероятной последовательностью состояний  $Q = \{q_1, \dots, q_T\}$  СММ  $\lambda = (A, B, \pi)$ , приведена на рис. 4.

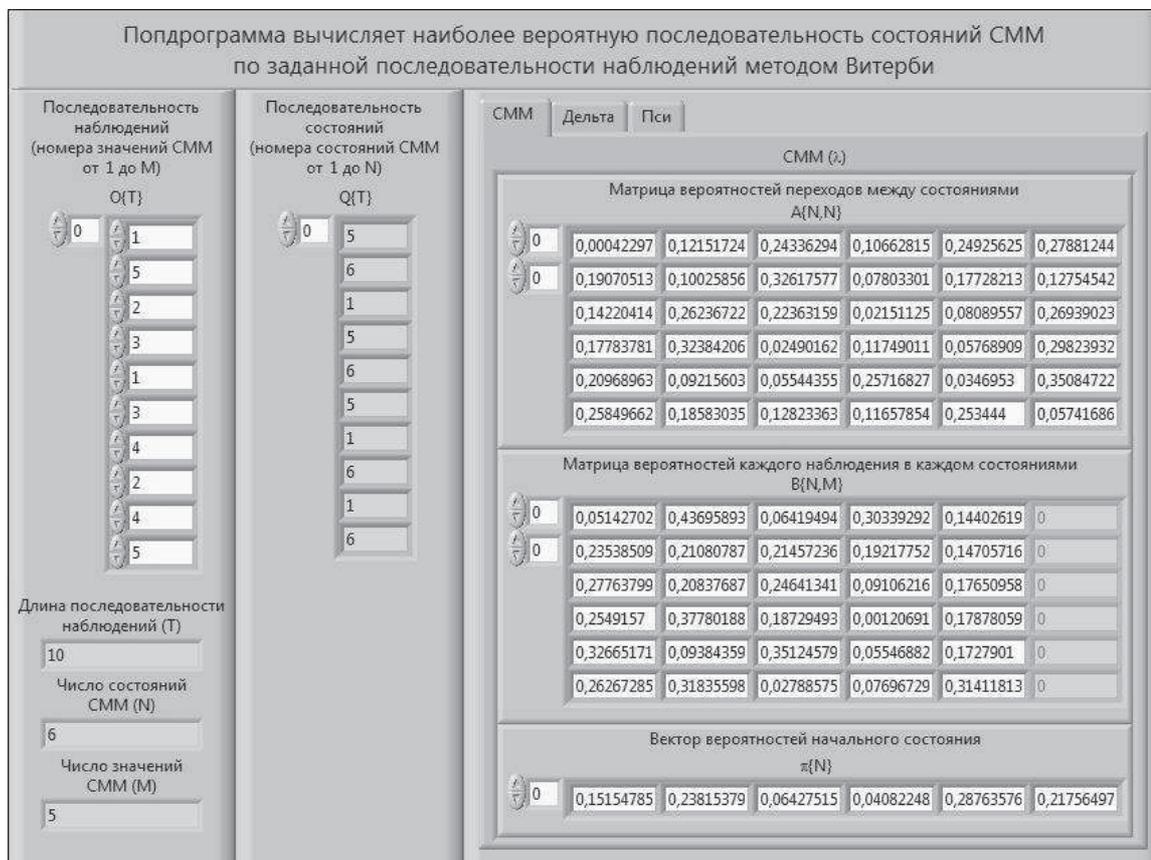


Рис. 4. Лицевая панель программного модуля, реализующего алгоритм метода Витерби



## ЗАКЛЮЧЕНИЕ

Разработанные программные модули, как показало тестирование, эффективно решают задачи обучения СММ по экспериментальным последовательностям наблюдений, соответствующим коротким словам и распознавание этих слов. Что позволяет использовать данные модули для реализации потоковой фильтрации аудиоконтента в цифровых системах связи.

## Библиографический список

1. Гоноровский И. С., Демин М. П. Радиотехнические цепи и сигналы. М. : Дрофа, 2006. 719 с.
2. Сергиенко А. Б. Цифровая обработка сигналов. СПб. : Питер, 2007. 750 с.
3. Рабинер Л. Р. Скрытые Марковские модели и их применение в избранных приложениях при распознавании речи // ТИИЭР. 1989. Т. 77. С. 86–120.
4. Narada Warakagoda A Hybrid ANN-HMM ASR system with NN based adaptive preprocessing M. Sc. thesis. URL: <http://jedlik.phy.bme.hu/gerjanos/HMM/hoved.html> (Accessed: 21.12.2012).
5. Портал компании National Instruments Russia. URL: <http://www.labview.ru> (дата обращения: 25.12.2012).

## The Development of Software Components for Streaming Audio Content Filtering Through the Use of Hidden Markov Models

A. N. Savin<sup>1</sup>, N. E. Timofeeva<sup>2</sup>, A. S. Geraskin<sup>3</sup>, Yu. A. Mavlutova<sup>4</sup>

<sup>1</sup>Savin Alexander Nikolaevich, Saratov State University, 83, Astrakhanskaya st., 410012, Saratov, Russia, savinan@info.sgu.ru

<sup>2</sup>Timofeeva Nadezhda Evgenievna, Saratov State University, 83, Astrakhanskaya st., 410012, Saratov, Russia, timofeevane@yandex.ru

<sup>3</sup>Geraskin Aleksej Sergeevich, Saratov State University, 83, Astrakhanskaya st., 410012, Saratov, Russia, gerascinas@mail.ru

<sup>4</sup>Mavlutova Yuliya Albertovna, Saratov State University, 83, Astrakhanskaya st., 410012, Saratov, Russia, yuliyamav@yandex.ru

The results of the development of efficient algorithms for streaming voice recognition using stochastic models based on the use of hidden Markov models are shown in this work. The article provides basic theoretical information for the hidden Markov model of the discrete system and the necessary parameters to define it are distinguished. Also there are three main tasks considered that need to be solved for the successful application of hidden Markov models in speech recognition systems. The algorithm of the method of Baum – Welch aimed at clarifying the parameters of the model and the Viterbi algorithm of selection of the most likely sequence of states of the system are given. These two methods are implemented in the environment of graphical programming LabVIEW in the form of software modules that implement the construction of the hidden Markov models of individual words, using the method of Baum – Welch and recognition of these words on the basis of the Viterbi method. It is supposed to use these modules to implement streaming audio content filtering in digital communication systems.

*Key words:* hidden Markov models, recognition of these words, Viterbi method, method of Baum – Welch.

## References

1. Gonorovskij I. S., Demin M. P. *Radiotekhnicheskie tsepi i signaly* [Radio Circuits and Signals]. Moscow, Drofa, 2006, 719 p. (in Russian).
2. Sergienko A. B. *Tsifrovaia obrabotka signalov* [Digital signal processing]. Saint Petersburg, Piter, 2007, 750 p. (in Russian).
3. Rabiner L. R. Hidden Markov models and their application in selected applications in speech recognition. *PIEEE*, 1989, vol. 77, pp. 86–120.
4. Narada Warakagoda A Hybrid ANN-HMM ASR system with NN based adaptive preprocessing M. Sc. thesis. Available at: <http://jedlik.phy.bme.hu/gerjanos/HMM/hoved.html> (Accessed: 21, December, 2012).
5. Portal of company National Instruments Russia. Available at: <http://www.labview.ru> (Accessed: 25, December, 2012).