SAVITSKY Yu.V., HVESHCHUK V.I., SAVITSKY A.Yu. Modification of algorithm of BPE for adaptive training of sigmoidalny neurons in architecture of the multilayered neural network

In article it is formulated and modification of algorithm of the return distribution of a mistake (BPE) for exact training of neural elements with sigmoidalny functions of activation in architecture of a multilayered neural network is mathematically proved. The offered rules of modification of synoptic communications of neurons of day off and the hidden layers can be used for creation of the effective algorithms providing reduction of temporary and computing complexity of process of training of multilayered neural network architecture.

УДК 581.3

Николайчук Я.Н., Ивасьев С.В., Якименко И.З., Касянчук М.Н.

МЕТОД ФАКТОРИЗАЦИИ МНОГОРАЗРЯДНЫХ ЧИСЕЛ НА ОСНОВЕ СВОЙСТВ КВАДРАТИЧНОСТИ ВЫЧЕТОВ В СИСТЕМЕ ОСТАТОЧНЫХ КЛАССОВ

Введение. Факторизация является одной из важнейших задач теории чисел [1] и современной асимметричной криптографии [2]. Ее суть заключается в разложении некоторого целого числа в произведение простых сомножителей. Известные методы факторизации, в зависимости от их производительности, разбиваются на две группы: экспоненциальные и субэкспоненциальные [3]. Все они достаточно громоздки, поэтому требуют значительных вычислительных ресурсов для обработки многоразрядных чисел. Однако теоретическое обоснование необходимой сложности таких вычислений или, другими словами, существование нижних оценок не доказано, поэтому вопрос о существовании алгоритма факторизации с полиномиальной сложностью является одной из открытых проблем современной теории чисел. Кроме того, актуальность проблемы факторизации продиктована также неопределенностью относительно теоретического обоснования устойчивости к раскрытию асимметричных криптосистем [4].

Наиболее распространенными для факторизации являются алгоритмы, основанные на теореме Ферма [3].

Анализ метода Ферма для факторизации многоразрядных чисел. Пусть P_0 – известное целое число, являющееся произведением двух простых чисел, которые нужно найти. Большинство современных методов факторизации основываются на идее, предложенной Пьером Ферма, которая заключается в поиске пар натуральных чисел A и B таких, что выполняется соотношение: $P_0 = A^2 - B^2$.

Метод Ферма описывается таким выражением:

$$\Delta_n = \sqrt{n^2 - P_0}\ , \qquad (1)$$

где $n = \left\lceil \sqrt{P_0} \right\rceil + k$, k=1, 2, 3, … .

Количество итераций k равняется значению, когда параметр Δ_n будет целым числом. Отсюда можно найти искомое разложение на множители:

$$P_0 = (n - \Delta_n)(n + \Delta_n)\ . \qquad (2)$$

Вычислительная сложность метода Ферма для многоразрядных чисел достаточно большая, поскольку количество итерацій k может быть порядка 2^{300}–2^{400} и только на единственном шагу возможно однозначное решение задачи факторизации. Причем операции необходимо виполнять над числами, разрядность которых примерно 300–500 бит.

Для упрощения этой задачи целесообразно использовать систему остаточных классов (СОК) [5], которая позволит выполнить распараллеливание процесса вычислений и будет выступать индикатором квадратичности вычетов.

Исходя из вышесказанного, целью данной работы является усовершенствование алгоритма факторизации на основании теоремы Ферма с использованием СОК для уменьшения вычислительной сложности при работе с многоразрядными числами.

Метод факторизации многоразрядных чисел на основании теоремы Ферма с помощью использования свойств квадратичности остатков. Известно, что квадраты целых чисел можно представить в виде суммы нечетных чисел, количество которых равняется данному числу [6]:

$$n^2 = \sum_{i=1}^{n} (2i - 1)\ . \qquad (3)$$

Поэтому, отыскав Δ_n по формуле (1) при k=1, следующие итерации выполняются согласно выражению $S_k = \sqrt{(\Delta_0)^2 + (2n - 1)}$. Итерации продолжаются до тех пор, пока параметр S_k не будет целым числом, причем количество итераций в обоих методах одинаково.

В таблице 1 представлен пример факторизации с помощью классического и усовершенствованного методов Ферма для P_0=3811.

Таким образом, получено разложение числа 3811 на простые сомножители:

$$3811 = 70^2 - 33^2 = (70 + 33)(70 - 33) = 103*37.$$

Из таблицы 1 видно, что в совершенствованном методе исключается операция возведения в квадрат. Кроме этого, арифметические действия выполняются над числами, размерность которых на несколько порядков меньше, чем в классическом методе. Однако следует отметить, что количество итераций в обоих случаях будет одинаково, а самой сложной остается операция проверки квадратичности остатка. Для уменьшения ее вычислительной сложности можно использовать СОК.

Исследование свойств квадратов в СОК. Рассмотрим остатки квадратов целых чисел по нескольким простым модулям p_j, то есть

$a_1(p_1, p_2, ..., p_m) = b_1^1, b_2^1, \ldots, b_m^1$,

$a_2(p_1, p_2, ..., p_m) = b_1^2, b_2^2, \ldots, b_m^2$,

$a_n(p_1, p_2, ..., p_m) = b_1^n, b_2^n, \ldots, b_m^n$, где $a_i = i^2$,

$b_j^i = a_i (\mathrm{mod}\, p_j), 1 \le i \le n,\ 1 \le j \le m$, m – количество модулей.

Используя свойство (3), искомые остатки можно получить с помощью рекурентной формулы $b_j^i = \left(b_j^{i-1} + z_i^j\right) \mathrm{mod}\, p_j$, где $z_j^i = z_i \bmod p_j$, z_i=2i-1.

Николайчук Ярослав Николаевич, *д.т.н., профессор, заведующий кафедрой специализированных компьютерных систем Тернопольского национального экономического университета.*

Ивасьев Степан Владимирович, *старший преподаватель кафедры компьютерной инженерии Тернопольского национального экономического университета.*

Якименко Игорь Зиновьевич, *к.т.н., доцент кафедры компьютерной инженерии Тернопольского национального экономического университета.*

Касянчук Михаил Николаевич*, к.ф.-м.н., доцент кафедры компьютерной инженерии Тернопольского национального экономического университета.*

Украина, 46009, г. Тернополь, Тернопольская область, ул. Львовская, 11.

Таблица 1. Пример факторизации с помощью классического и усовершенствованного методов Ферма для P_0=3811

k	n	$(\Delta_n)^2$, классический метод	$(\Delta_n)^2$, усовершенствованный метод
1	62	62^2-3811=33	62^2-3811=33
2	63	63^2-3811=158	33+125=158
3	64	64^2-3811=285	158+127=285
4	65	65^2-3811=414	285+129=414
5	66	66^2-3811=545	414+131=545
6	67	67^2-3811=678	545+133=678
7	68	68^2-3811=813	678+135=813
8	69	69^2-3811=950	813+137=950
9	70	70^2-3811=1089=33^2	950+139=1089=33^2

Таблица 2. Поиск остатков квадратов чисел по простым модулям

n	z_i	a_n	p_1=3		p_2=5		p_3=7		p_4=11	
			z_1^i	b_1^i	z_2^i	b_2^i	z_3^i	b_3^i	z_4^i	b_4^i
1	1	1	1	1	1	1	1	1	1	1
2	3	4	0	1	3	4	3	4	3	4
3	5	9	2	0	0	4	5	2	5	9
4	7	16	1	1	2	1	0	2	7	5
5	9	25	0	1	4	0	2	4	9	3
6	11	36	2	0	1	1	4	1	0	3
7	13	49	1	1	3	4	6	0	2	5
8	15	64	0	1	0	4	1	1	4	9
9	17	81	2	0	2	1	3	4	6	4
10	19	100	1	1	4	0	5	2	8	1
11	21	121	0	1	1	1	0	2	10	0

Таблица 3. Нахождение ключа факторизации

k	n	z_i	$(\Delta_n)^2$	p_1=3			p_2=5			p_3=7			p_4=11			Y_n
				z_1^i	Δ_n^1	y_n^1	z_2^i	Δ_n^2	y_n^2	z_3^i	Δ_n^3	y_n^3	z_4^i	Δ_n^4	y_n^4	
1	62	123	33	0	0	1	3	3	0	4	5	0	2	0	1	0
2	63	125	158	2	2	0	0	3	0	6	4	1	4	4	1	0
3	64	127	285	1	0	1	2	0	1	1	5	0	6	10	0	0
4	65	129	414	0	0	1	4	4	1	3	1	1	8	7	0	0
5	66	131	545	2	2	0	1	0	1	5	6	0	10	6	0	0
6	67	133	678	1	0	1	3	3	0	0	6	0	1	7	0	0
7	68	135	813	0	0	1	0	3	0	2	1	1	3	10	0	0
8	69	137	950	2	2	0	2	0	1	4	5	0	5	4	1	0
9	70	139	1089	1	0	1	4	4	1	6	4	1	7	0	1	1

Соответствующие результаты по модулям 3, 5, 7, 11 представлены в таблице 2, из которой видно, что количество квадратичных вычетов для каждого модуля составляет $(p_j+1)/2$ (включая 0). Это следует из равенства $n_2 \bmod p_j=(-n_2) \bmod p_j=(p_j-n)^2 \bmod p_j$.

Нахождение ключа факторизации. Для выявления, есть ли число квадратичным вычетом, используется символ Лежандра или его обобщение – символ Якоби [1]. Их поиск требует выполнения операций факторизаци и нахождения остатков от деления многоразрядных чисел. Для упрощения данной задачи предлагается использовать свойство, что квадрат числа есть квадратичным вычетом по любому простому модулю. Соответственно, найдя остатки данного числа по простому модулю pj, можно получить для него одно из значений частичного ключа факторизации y_n^j, которое равняется 0 или 1, причем значению 1 отвечает случай, когда $\Delta_n^j=(\Delta_n)^2 \times$ $\times \bmod p_j=\left(\Delta_{n-1}^j+z_j^i\right) \bmod p_j$ есть квадратичным вычетом по p_j и тогда Δ_n может быть целым числом. Значение ключа факторизации для числа n определяется так: $Y_n=y_n^1 \wedge y_n^2 \wedge \ldots \wedge y_n^m$. Полученные результаты представлены в таблице 3, а на рисунке 1 – блок-схема нахождения ключа факторизации.

Вектор $Y(Y_1, Y_2, \ldots, Y_n)$, в котором Y_i=0 или 1, образует общий ключ факторизации. В данном примере Y(000000001). Это означает, что на девятой итераци возможен случай, когда Δ_n будет целым числом, что и подтверждается вычислением: $\sqrt{1089}=33$.

Следует отметить, что частичный ключ факторизации y_n^j владеет свойством цикличности, период которой равен модулю p_j, поэтому при рассчетах нет необходимости определять все текущие значения $(\Delta_n)^2$. Число $(\Delta_n)^2$, квадратный корень из которого может быть целым числом (Y_n=1), получается согласно формуле вычисления суммы арифметической прогрессии, которую образует последовательность z_i, с помощью такого выражения:

$$(\Delta_n)^2=\left(z_{i\ \min}+k\right)(k-1)+(\Delta_n)_{\min}^2. \qquad (4)$$

Кроме этого, можно использовать другую формулу, полученную из (1) для классического алгоритма Ферма:

$$(\Delta_n)^2=(n_{\min}+k-1)^2-P_0, \qquad (5)$$

но вычислення согласно (5) должны выполняться над намного большими числами.

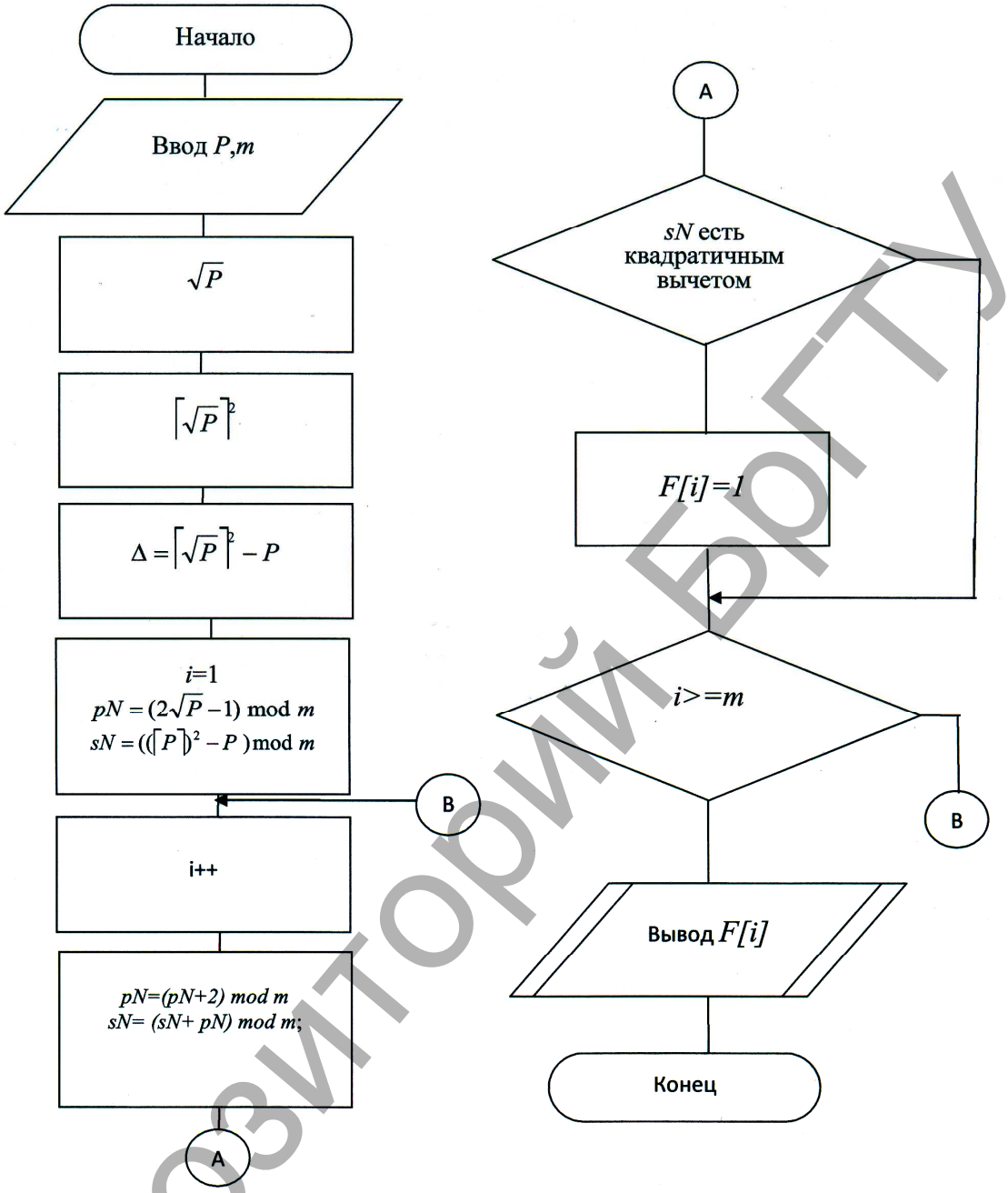

Рис. 1. Блок-схема нахождения ключа факторизации

Итак, разработанный алгоритм факторизации позволяет выполнять операции в системе остаточных классов над остатками небольшой разрядности, которые не превышают модуль.

Заключение. В данной работе разработан усовершенствованный метод факторизации многоразрядных чисел на основе теоремы Ферма с помощью СОК, в котором исключается операция возведения в квадрат и, кроме этого, арифметические действия будут выполняться над числами, которые меньше выбранного модуля. Это позволяет изменить зону разрядностей вычислительных ресурсов на несколько порядков ниже и заменить операцию нахождения квадратного корня, на которой базируется вычислительная сложность алгоритма Ферма, на генерирование бинарного ключа факторизации.

СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Бухштаб, А.А. Теория чисел / А.А. Бухштаб. – М.: Просвещение, 1966. – 384 с.
2. Задирака, В.К. Компьютерные технологии криптографической защиты информаци на специальных цифровых носителях: учебное пособие / В.К. Задирака, А.М. Кудин, В.О. Людвиченко, А.С. Олексюк. – Киев-Тернополь: Учебники и пособия, 2007. – 272 с.
3. Ишмухаметов, Ш.Т. Методы факторизации натуральных чисел: учебное пособие / Ш.Т. Ишмухаметов.– Казань: Казан. ун., 2011.– 190 с.
4. Задирака, В.К. Компьютерная криптология / В.К. Задирака, А.С. Олексюк. – Тернополь: ТАНХ, 2002. – 504 с.
5. Николайчук, Я.Н. Теория источников информации / Я.Н. Николайчук. – Тернополь: ТНЭУ, 2008. – 536 с.
6. Грэхем, Р. Конкретная математика. Основания информатики / Р. Грэхем, Д. Кнут, О. Паташник; пер. с англ. – М.: Мир, 1998. – 703 с.

Материал поступил в редакцию 09.12.15

NYKOLAYCHUK YA.N., IVASIEV S.V., YAKYMENKO I.Z., KASIANCHUK M.N. Method of factorization of multi-digital numbers based on the properties of quadratic remainders in the system of residual classes

This work is devoted to develop of advanced method of factorization of multi-bit numbers based on Fermat's theorem with using of the system of residual classes, This method is excluded the operation of squaring and, besides that, arithmetic operations are performed on numbers which are smaller than the selected module. Last one allows to shifted zone of bit computing resources on several orders to deeper side and replace the operation of finding the square root, which is caused of computational complexity of the Fermats' algorithm onto generating a binary key of factorization.

УДК 004.94

Коваленко В.Ю., Костюк Д.А.

ВИРТУАЛИЗОВАННАЯ ФЕРМА ДЛЯ ТЕСТИРОВАНИЯ И ДЕМОНСТРАЦИИ ПРИЛОЖЕНИЙ ПЛАТФОРМЫ ANDROID С ВЕБ-ДОСТУПОМ

Введение. На сегодняшний день клиент-серверные приложения с веб-интерфейсом приобрели широчайшее распространение и по ряду направлений стали заменять классические настольные приложения. К числу причин следует отнести наличие веб-браузера на всех платформах и архитектурах, а также достижение современными браузерами достаточной производительности исполнения кода JavaScript, на котором строится клиентская часть веб-приложений. Универсальная доступность (в т.ч. на платформах с сенсорным интерфейсом и других так называемых «слабых клиентах», т.е. на мобильных и портативных устройствах, нацеленных преимущественно на использование облачных сервисов) делает работу через браузер наиболее удобной точкой входа для конечного пользователя приложения/сервиса. Особенно актуально это, когда основная вычислительная нагрузка ложится на сервер и/или другие узлы в сети, а сам клиент служит лишь для управления и доступа к ресурсам (обратный случай, к сожалению, на текущий момент неэффективен, так как реализация на JavaScript подразумевает заметно бóльшие накладные расходы по сравнению с традиционными языками серверных платформ).

Одной из задач, способных получить существенный выигрыш при их клиент-серверной реализации с использованием частного облака, является тестирование и отладка мобильных приложений.

Проблему тестирования и отладки программного кода в условиях сильной фрагментации целевых аппаратных платформ нельзя назвать новой, однако на сегодняшний день мобильная платформа Android является одной из наиболее фрагментированных. Разработчику необходимо держать на своей машине набор образов для эмулятора с различными версиями операционной системы и запускать их по очереди. При этом, даже если штатные средства разработки Android

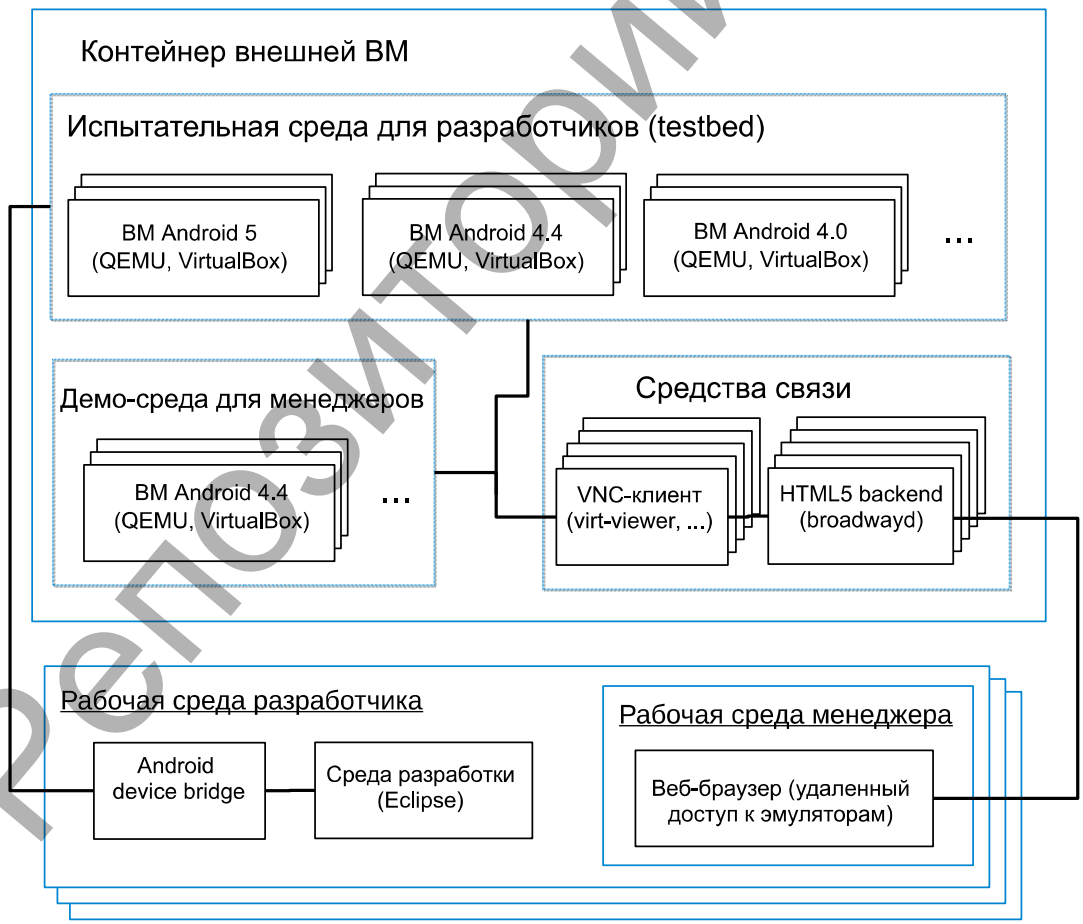

Рис. 1. Структура тестовой фермы Android-приложений

Коваленко Владимир Юрьевич, *старший преподаватель кафедры ЭВМиС Брестского государственного технического университета.*
Костюк Дмитрий Александрович, *к.т.н., доцент кафедры ЭВМиС Брестского государственного технического университета.*
Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.