УДК 681.324

М. Плонковски, аспирант; П.П. Урбанович, профессор

КРИПТОГРАФИЧЕСКОЕ ПРЕОБРАЗОВАНИЕ ИНФОРМАЦИИ НА ОСНОВЕ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ

In article the new method of synchronization of neural networks which distinctive feature is use during training complex numbers is considered and analyzed. Thus the purpose is pursued - to guarantee a required level of safety of the information for the account concerning short time of training of the networks basing known architecture TPM (tree parity machine).

1. Постановка задачи.

Широкий круг задач, решаемых нейронными сетями (НС), не позволяет в настоящее время создавать универсальные, мощные сети, вынуждая разрабатывать специализированные НС, функционирующие по различным алгоритмам.

Основу каждой НС составляют относительно простые, в большинстве случаев – однотипные, элементы (ячейки), имитирующие работу нейронов мозга. Далее под нейроном будет подразумеваться искусственный нейрон, то есть ячейка НС. Каждый нейрон характеризуется своим текущим состоянием по аналогии с нервными клетками головного мозга, которые могут быть возбуждены или заторможены. Он обладает группой синапсов – однонаправленных входных связей, соединенных с выходами других нейронов, а также имеет аксон – выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов. Каждый синапс характеризуется величиной синаптической связи или ее весом, который по физическому смыслу эквивалентен электрической проводимости.

Очевидно, что процесс функционирования НС, то есть сущность действий, которые она способна выполнять, зависит от величин синаптических связей. Поэтому, задавшись опре-деленной структурой НС, отвечающей какой-либо задаче, разработчик сети должен найти оптимальные значения всех переменных весовых коэффициентов (некоторые синаптические связи могут быть постоянными). Этот этап называется обучением НС, и от того, насколько качественно он будет выполнен, зависит способность сети решать поставленные перед ней задачи во время эксплуатации.

На этапе обучения, кроме качества подбора весов, важную роль играет время обучения. Как правило, эти два параметра связаны обратной зависимостью и их приходится выбирать на основе компромисса.

Взаимные обучение двух сетей ведет к синхронизации их векторов весов.

Две синхронизированные сети могут использоваться для передачи информации, в частности секретных сообщений. Этот процесс заключается в следующем. В обычной криптографии необходимо, чтобы отправитель секретного сообщения (A) каким-либо образом сообщил получателю (B) и ключ для расшифрования. И это самое узкое место. Поскольку «подслушивающий» объект (C) может перехватить ключ, то вместе с ним он получит несанкционированный доступ и к содержанию секретных сообщений. Синхронизированные нейронные сети устраняют именно это узкое место, поскольку никакого ключа вообще не требуется. Его роль выполняют скрытые статистические веса их межнейронных соединений (известный метод Кинцеля).

В данной статье рассматривается и анализируется новый метод синхронизации НС, отличительной особенностью которого является использование в процессе обучения комплексных чисел. При этом преследуется цель – гарантировать требуемый уровень безопасности информации за счет относительно короткого времени обучения сетей, базирующихся на известной архитектуре TPM (tree parity machine) [1]. Проанализируем вначале особенности этой архитектуры.

2. Архитектура TPM.

TPM состоит из двух уровней. Первый составляет K независимых персептронов, из которых каждый характеризуется N-элементным вектором весов ($[w_{k,1}, w_{k,2}, \ldots, w_{k,N}]$, где $1 \le k \le K$). Коэффициенты этих векторов – это целые числа с интервала $[-L, L]$. Входы персептронов составляет K N-элементных векторов ($[x_{k,1}, x_{k,2}, \ldots, x_{k,N}]$; k часто отождествляется с одним $N*K$-элементным вектором $[x_1, x_2, \ldots, x_{kN}]$), выбираемым из двухэлементного массива целых чисел $\{-1, 1\}$. Выходы нейронов – это также целые числа (относятся к тому же массиву $\{-1, 1\}$), обозначим через $y_1, y_2, \ldots, y_K$. Выход O архитектуры TPM (сети A или B) вычисляется как про-

изведение выходов персептронов в соответствии со следующей формулой [2]:

$$O^{A/B} = \prod_{k=1}^{K} y_k^{A/B} =$$

$$= \prod_{k=1}^{K} \sigma(\sum_{j=1}^{N} \omega_{kj}^{A/B} x_{kj}),$$

где σ – это модифицированная функция знака, которая определяется следующим образом:

$\sigma(\alpha_k^{A/B}) = 1$, при $\alpha_k^{A/B}$ положительном; если же $\sigma(\alpha_k^{A/B}) = -1$, то $\alpha_k^{A/B}$ должно быть отрицательным.

В процессе обучения (например, по методу Хебба) обе сети обмениваются между собой значениями параметров на выходах ($O^{A/B}$), оставляя одновременно «в тайне» внутренние (как и начальные) состояния векторов весов. Входной вектор X генерируется случайным образом.

Активизация весов происходит только тогда, когда выходные значения обеих сетей одинаковы ($O^A = O^B$). Кроме того, активизируются только весы тех нейронов, значение на выходе которых равно значению на выходе целой архитектуры TPM. Это может быть описаны следующими соотношениями:

$$\omega_{kj}^{A/B} = \omega_{kj}^{A/B} + O^{A/B} x_{kj},$$

если $O^A = O^B \cup O^{A/B} = y_k^{A/B}$, и

$\omega_{kj}^{A/B} = \omega_{kj}^{A/B}$ – в противном случае.

Архитектура TPM налагает ограничение на значения весовых коэффициентов следующим образом:

$\omega_{kj}^{A/B} = \omega_{kj}^{A/B}$, если $|\omega_{kj}^{A/B}| > L$ и

$\omega_{kj}^{A/B} = \text{sign}\ (\omega_{kj}^{A/B}) L$ – в противном случае.

Процесс взаимного обучения сетей начинается с инициализации векторов весов. Потом на каждом шаге такого обучения на основе случайно выбранных входных векторов вычисляются значения O^A и O^B.

За время, обозначенное как t_{sync}, достигается состояние синхронизации (время обучения). Это означает, что векторы весов обеих архитектур равны друг другу. Затем обе TPM генерируют значения выходного параметра.

Главным достоинством представленного процесса взаимного обучения является его безопасность. Если оппонент C, «наблюдающий» за обменом информацией между сетями A и B, будет подстраивать свой вектор весов согласно с данными выше правилами, то этот вектор через время t_{learn}, достигнет состояния синхронизации. Однако время обучения t_{learn} значительно больше, чем время синхронизации t_{sync} ($t_{\text{learn}} > t_{\text{sync}}$) [3]. Поэтому, если процесс взаимного обучения заканчивается относительно быстро, то оппонент не успеет закончить синхронизацию собственного вектора весов с сетями A и B.

3. Архитектура TPM, основанная на использовании комплексных чисел.

Такую архитектуру условно назовем TPCM (tree parity complex machine).

Архитектура TPCM состоит из двух уровней. Элементами первого уровня являются персептроны, имеющие также N-элементные векторы весов ($[w_{k,1}, w_{k,2}, \ldots, w_{k,N}]$, где $1 \le k \le K$), коэффициентами которых являются, в том числе, комплексные величины, ограниченные интервалом $[-L, L]\times[-L, L]$. Входы персептронов отождествляются также часто с одним $N*K$-элементным вектором $[x_1, x_2, \ldots, x_{kN}]$) комплексных чисел из массивов $\{(1, 1), (-1, 1), (-1, -1), (1, -1)\}$. Выходы же нейронов – это комплексные числа, принадлежащие массивам $\{(1, 0), (0, 1), (-1, 0), (0, -1)\}$, обозначенным далее через $y_1, y_2, \ldots, y_K$. Выход O архитектуры TPCM вычисляется в основном аналогично способу, описанному выше. Различие состоит в модификации функции знака σ:

$\sigma(\alpha_k^{A/B}) = (1,0)$,

если $7\pi/4 < \arg(\sigma_{kj}^{A/B}) \le \pi/4$;

при $\pi/4 < \arg(\sigma_{kj}^{A/B}) \le 3\pi/4$ функция принимает значение (0,1);

в случае $3\pi/4 < \arg(\sigma_k^{A/B}) \le 5\pi/4$ – (–1,0)

и при $5\pi/4 < \arg(\sigma_k^{A/B}) \le 7\pi/4$ – (0,–1).

Модифицируется то же правило активизации – с целью ограничения величин векторов весов (отдельно для каждого элемента комплексного числа):

$$\text{Re}(\omega_{kj}^{A/B}) = \text{sign}\ (\text{Re}(\omega_{kj}^{A/B}))\ L,$$

если $|\omega_{kj}^{A/B}| > L$ и $\text{Re}(\omega_{kj}^{A/B}) = \text{Re}(\omega_{kj}^{A/B})$ – в противном случае; вместе с тем $\text{Im}(\omega_{kj}^{A/B}) = \text{sign}\ (\text{Im}(\omega_{kj}^{A/B}))\ L$, если $|\omega_{kj}^{A/B}| > L$ и $\text{Im}(\omega_{kj}^{A/B}) = \text{Im}(\omega_{kj}^{A/B})$ – в противном случае.

Процесс обучения происходит по тому же алгоритму, что и для архитектуры TPM.

Входные величины, используемые в процессе обучения, это векторы, состоящие из комплексных чисел, расположенных на углах квадрата (рис. 1).

Коэффициенты векторов весов – это комплексные числа, ограниченные квадратом: [–L, L] × [–L, L] (рис. 2).

Выходные параметры архитектуры TPCM как произведение выходных параметров персептронов принадлежат также к области значений, представленной на рис. 3.

Модифицированная функция знака σ может быть представлена графически: выходные величины могут обозначаться точками, находящимися в соответствующих четвертях, как это показано на рис. 4.

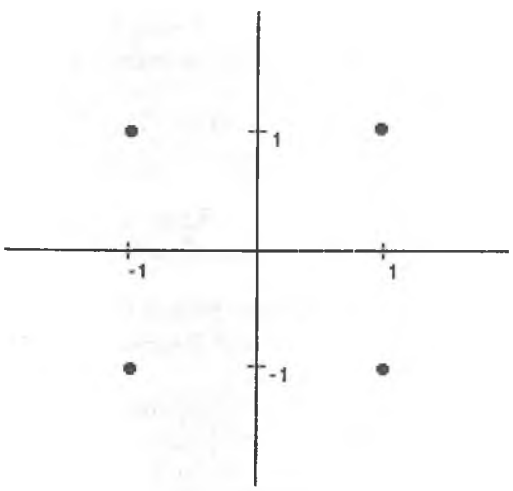

Рис. 1. Входные величины, используемые в процессе обучения сетей TPCM

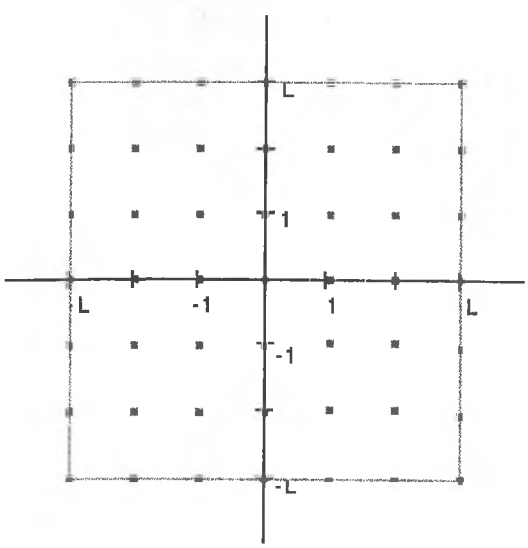

Рис. 2. Точки, которые могут быть использованы в качестве коэффициентов векторов весов

Выходные величины – это коллективные числа. расположенные на углах квадрата (рис. 3).

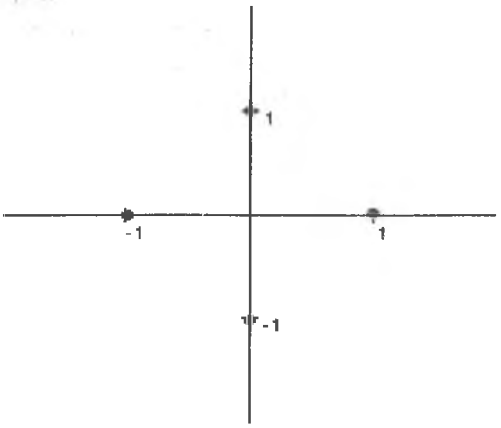

Рис. 3. Точки, которые используются как выходные величины персептронов

4. Анализ архитектуры TPCM в контексте безопасного обмена ключами.

Главным элементом, обеспечивающим безопасность процесса синхронизации сетей, базирующихся на архитектуре TPM, есть тот факт, что выходная величина не определяется однозначно через выходные величины отдельных персептронов. Например, в классической модели архитектуры TPM с параметром $K = 3$ для каждой выходной величины существуют четыре возможности внутренних выходных величин персептронов, при которых $O = 1$: (1, 1, 1), (–1, –1, 1), (1, –1, –1), (–1, 1, –1)).

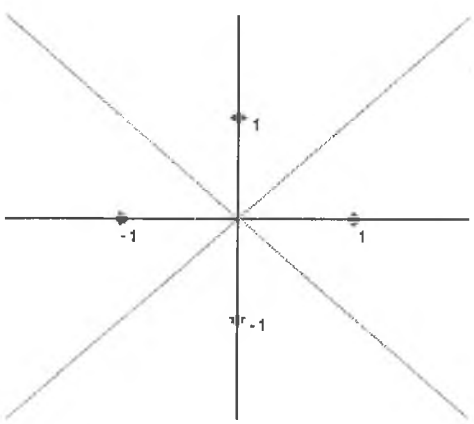

Рис. 4. Упрощенный чертеж функции знака σ

В архитектуру же TPCM количество комбинаций внутренних выходных величин персептронов, обеспечивающих одинаковое значение выходной величины, возрастает квадратично. Например, для $K = 3$ существует 16 комбинаций, при которых $O = 1$: (1, 1, 1), (–1, –1, 1), (1, –1, –1), (–1, 1, –1), (*i*, *i*, –1),

$(i, -1, i)$, $(-1, i, i)$, $(-i, i, 1)$, $(-i, 1, i)$, $(i, 1, -i)$, $(i, -i, 1)$, $(1, i, -i)$, $(1, -i, i)$, $(-i, -i, -1)$, $(-i, -1, -i)$, $(-1, -i, -i)$). Поэтому можно предположить, что системы обмена ключами, основанные на архитектуре TPCM, будут характеризоваться большим уровнем безопасности, чем архитектуры TPM.

Полагаем, что сети A, B и C (интруз) характеризуются архитектурой TPM; сети cA, cB і cC – архитектурой TPCM. Параметры L и K принимаем одинаковыми в обоих случаях: $L = 3$, $K = 3$. Выбор параметра N зависит от архитектуры: в сетях TPM $N = 20$, а в TPCM – $N = 10$. Эта разница возникает в связи с тем, что комплексное число определяется двумя целыми числами.

Итак, целые числа из общего входного вектора X составляют входы для TPCM. Нижеследующая таблица содержит усредненные результаты, характеризующие время обучения сетей A и B – $t_{sync}(A, B)$, а также сетей cA и cB – $t_{sync}(cA,cB)$, время синхронизации сети C (cC) с соответствующей сетью (например, с сетью A и сетью cA соответственно). Кроме того, в табл. дано отношение времени обучения двух сетей и времени наступления синхронизации одной из них с интрузом – $(t_{sync}(A, B)/t_{sync}(A, C)$; здесь мы с целью понимания сути процессов разделили понятия: время обучения означает по сути время синхронизации между взаимообучаемыми сетями.

Таблица

Архитектура	Время обучения, кол-во шагов	Время синхр., кол-во шагов	$t_{sync}(A, B)/t_{sync}A,C)$
TPM	214,4	1035,8	0,207
TPCM	14811,9	239391,7	0,0602

5. Обсуждение результатов.

Приведенные результаты свидетельствуют о том, что использование архитектуры TPCM позволяет значительно повысить уровень безопасности процесса передачи зашифрованной с использованием криптографических алгоритмов информации ключами. Этот уровень (измеренный отношением времени обучения ко времени наступления синхронизации) в три раза увеличивается при использовании предлагаемого метода. Однако видно, что при этом значительно возрастает и время обучения. С учетом того, что этот параметр в значительной степени зависит от метода обучения и структуры сети [4–6], можно надеяться, что этот недостаток может быть значительно нейтрализован.

Литература

1. Kanter I., Kinzel W.,Kanter E. Secure exchange of information by synchronization of neural networks// Europhys. Lett. 57. P. 141–147 (2002).

2. M. Volkmer, S. Wallner. Tree Parity Machine Rekeying Architectures// Cryptology ePrint Archive: REport 2004/216.

3. M. Rosen-Zivi, I. Kanter, W. Kinzel. Cryptography based on neural network – analitical results.

4. W. Kinzel, I. Kanter. Neural Cryptography// 9th Intern. Conf. On Neural Information Processing, Singapore, Nov. 2002.

5. A. Klimov, A. Mityaguine, A. Shamir. «Analysis of Neural Cryptography»// Proc. of AsiaCrypt 2002, volume 2501 of LNCS. – P. 288–298. Springer Verlag, 2002.

6. Plonkowski M. Trainig neural network for pattern recognition// Труды БГТУ. Сер. VI. Физико-математические науки и информатика. Вып. XII. – 2004. – С. 149–156.