

Р. В. Таранов, А. В. Маликов

ПРОГНОЗИРОВАНИЕ ЭНЕРГОПОТРЕБЛЕНИЯ ПРИ ПОМОЩИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИИ CUDA

Проведен анализ существующих методов прогнозирования энергопотребления. Обоснован выбор метода прогнозирования с использованием искусственной нейронной сети. Описано применение параллельных вычислений на GPU. Предложен новый алгоритм прогнозирования энергопотребления на основе теории искусственных нейронных сетей с применением технологии CUDA для оптимизации процесса прогнозирования потребления электроэнергии на промышленном предприятии. Для практического использования нового алгоритма разработана компьютерная программа для операционных систем Windows, написанная на языке программирования C++. Поставлен эксперимент, на основе которого проводится анализ работы программы и расчет потребляемых ресурсов. По результатам экспериментов разработанный параллельный алгоритм достиг необходимой точности прогноза за значительно меньший промежуток времени. Применение алгоритма позволит предприятиям получить более точный прогноз и сократить издержки, связанные с оплатой электроэнергии.

Ключевые слова: методы прогноза, прогнозирование энергопотребления, нейронная сеть, параллельный алгоритм, технология CUDA.

Введение

У крупных промышленных предприятий в современных условиях потребления электроэнергии возникает задача составления заявки в энергоснабжающую организацию на объем электроэнергии, необходимый в определенный период времени. Если реальные показания потребления будут отличаться от заявленных сумм больше чем на 5 %, на предприятие будут наложены штрафы [1].

Известные методы прогнозирования электрической нагрузки плохо «работают» с «зашумленными» и неполными данными. Наиболее распространен метод прогноза, основанный на удельных нормах расхода электроэнергии с экспертной корректировкой (корреляционно-регрессионный анализ). В качестве модели для аппроксимации графика нагрузки и его прогноза могут успешно использоваться также полиномиальные разложения [2]. Если необходим прогноз на длительный интервал времени, то в таких случаях могут использоваться авторегрессионные методы [3].

Все эти методы обладают существенными недостатками:

- их применение требует непосредственного участия аналитиков;
- отличаются сложностью вычислений;
- характеризуются недостаточной точностью прогноза;
- могут применяться только к определенному типу прогноза;
- отличаются чувствительностью к входным данным.

В настоящее время разрабатываются и внедряются новые подходы к прогнозированию электрической нагрузки. В частности, изучаются проблемы применения нейронных сетей для решения задач прогнозирования энергопотребления [4–12].

Задача нашего исследования – рассмотреть классический перцептрон – искусственную нейронную сеть (ИНС).

Описание модели нейронной сети

Классический перцептрон – одна из самых распространенных моделей нейронной сети благодаря способности отражать сложные нелинейные отношения между входными и выходными параметрами [8].

Перцептрон состоит из множества сенсорных элементов, которые образуют входной слой; одного или нескольких скрытых слоев вычислительных нейронов и одного выходного слоя нейронов. Входной сигнал распространяется по сети в прямом направлении, от слоя к слою [13]. Каждый нейрон сети имеет нелинейную функцию активации:

$$OUT = F(\xi) = \frac{1}{1 + \exp(-\xi)}, \quad (1)$$

где $\xi = \sum_{k=1}^L x_k \omega_k$; x_k – входы нейрона; ω_k – синаптические веса входов; L – количество входов нейрона.

Обучение методом обратного распространения ошибки предполагает два прохода по всем слоям сети: прямой и обратный. Принцип функционирования алгоритма обратного распространения ошибки заключается в использовании метода градиентного спуска и корректировки весов для минимизации ошибки нейронной сети [13]. Множество весовых коэффициентов инициализируется произвольными значениями в диапазоне от 0 до 0,1. Формула коррекции весов для выходного слоя имеет следующий вид:

$$\omega_{p-k}(i+1) = \omega_{p-k}(i) + \eta \delta_k OUT_p,$$

где i – номер итерации обучения; ω_{p-k} – синаптический вес, соединяющий нейрон p скрытого слоя с нейроном k выходного слоя; $\delta_k = OUT_k(1 - OUT_k)(T_k - OUT_k)$; η – коэффициент скорости обучения; OUT_p – выход нейрона; T_k – целевое значение выхода нейрона. Формула для коррекции весов скрытого слоя записывается в виде

$$\omega_{p-q}(i+1) = \omega_{p-q}(i) + \eta \delta_q OUT_p,$$

где ω_{p-q} – синаптический вес, соединяющий нейрон p предыдущего слоя с нейроном q скрытого слоя; OUT_p – выход нейрона; $\delta_q = OUT_q(1 - OUT_q) \sum_{k=1}^N \delta_k \omega_{q-k}$; N – количество нейронов следующего слоя.

Существует много различных алгоритмов, которые могут быть использованы для обучения ИНС. Так как в данной работе моделируется алгоритм прогнозирования потребления электроэнергии, в качестве исторических данных мы можем использовать, как обучающие выборки, значения расхода электроэнергии за предыдущие периоды. Под обучающей выборкой понимается набор данных, состоящий из входных параметров и соответствующего им известного значения прогнозируемого параметра, поэтому нами рассматривается метод обратного распространения ошибки, относящийся к парадигме контролируемого обучения.

Основной недостаток алгоритма обратного распространения ошибки – медленная сходимость, поэтому процесс обучения ИНС часто занимает очень много времени. В сложных задачах, к которым относится прогнозирование электропотребления, для обучения сети могут потребоваться дни или даже недели [14].

Модификация модели

В настоящее время не существует общего алгоритма определения структуры ИНС, подходящего для каждой рассматриваемой проблемы. Часто такую структуру выбирают методом проб и ошибок, который отнимает у исследователей много времени [15]. Поскольку для каждой структуры необходимо подбирать значения весовых коэффициентов, то сокращение времени этого подбора должно ускорить процесс обучения ИНС.

Нами предлагается модель, основанная на параллельном прохождении нейронов как при прямом распространении сигналов, так и в обратном направлении при корректировке весов. Тот факт, что нейроны, находящиеся в одном слое, не зависят друг от друга, позволяет распараллелить вычислительный процесс.

На рис. 1 представлена блок-схема модифицированного алгоритма. Блоки алгоритма, в которых предполагаются параллельные вычисления, отмечены штрихпунктирной линией. Для реализации распараллеливания нами была применена технология CUDA.

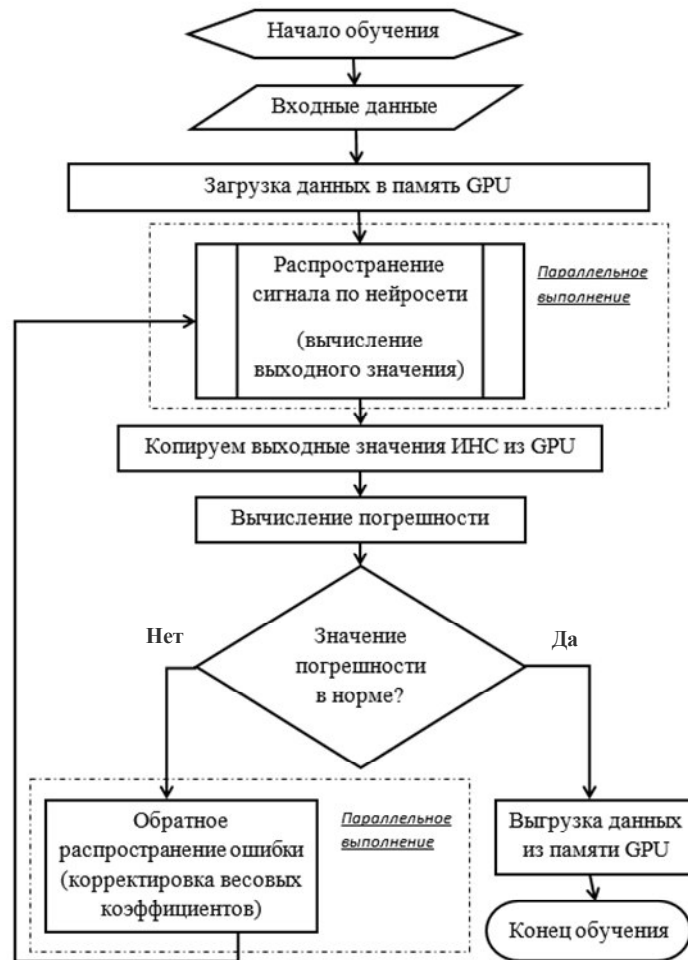


Рис. 1. Алгоритм обучения ИНС с распараллеливанием вычислений по технологии CUDA: GRU – графический процессор

CUDA – это архитектура параллельных вычислений от NVIDIA, позволяющая существенно увеличить вычислительную производительность благодаря использованию GPU. В настоящее время разработчики программного обеспечения, ученые и исследователи широко используют CUDA в различных областях, включая обработку видео и изображений, вычислительную биологию и химию, моделирование динамики жидкостей, восстановление изображений, полученных путем компьютерной томографии, сейсмический анализ, трассировку лучей и многое другое [16].

CUDA не использует графические API и свободна от ограничений, свойственных им. Основным преимуществом является её простота – все программы пишутся на «расширенном» языке C++. Дополнительные плюсы – наличие хорошей документации, набор готовых инструментов и библиотек, кроссплатформенность. Для решения задач CUDA использует очень большое количество параллельно выполняемых нитей, при этом каждой нити соответствует обычно один элемент вычисляемых данных [17].

CUDA разбивает задачу в сети блоков, каждый блок содержит несколько потоков. Блоки могут работать в любом порядке. Блок должен выполнить от начала до завершения всю задачу, и может быть запущен на одном из N процессоров. Каждый процессор получает равное распределение задач [18].

Постановка задачи

Рассмотрим пример месячного прогнозирования электропотребления предприятием «Ставропольский электротехнический завод «Энергомера» (СЭТЗ «Энергомера») с применением ИНС. Входными параметрами являются следующие величины:

- средняя месячная температура;
- количество выходных в месяце;
- продолжительность дня;
- плановые объемы производства на месяц;
- потребление электроэнергии за предыдущий месяц;
- максимальное и минимальное месячные значения потребления за предыдущий год.

Искусственная нейронная сеть будет состоять из входного слоя, одного скрытого слоя нейронов и выходного слоя. Входной слой будет состоять из 11 нейронов: по одному нейрону на среднемесячную температуру, количество выходных в месяце, продолжительность дня, потребление за предыдущий месяц, значения максимального и минимального месячного расхода за год, а также 5 нейронов, характеризующие плановые объемы производства для 5 разных типов выпускаемой продукции (характерно для конкретного предприятия). Выходной слой будет представлен одним нейроном, т. к. надо спрогнозировать только один параметр – месячный расход электроэнергии. Необходимо определить оптимальное количество нейронов в скрытом слое практическим путем. Для этого надо поставить несколько экспериментов с разной конфигурацией сети по количеству нейронов в скрытом слое и сравнить результаты обучения.

Для обучения сети будет использован алгоритм обратного распространения ошибки и его модификация с распараллеливанием вычислений. Для реализации модифицированного алгоритма с распараллеливанием прохождения сигнала по сети необходимо разработать программу, которая будет производить вычисления на графическом процессоре с использованием технологии CUDA.

Так как задача состоит в прогнозировании месячного значения, то минимальное количество обучающих выборок 12 – значения потребления электроэнергии для каждого месяца прошедшего года, используемые в качестве обучающих выборок при обучении сети. Выходное значение – это потребление электроэнергии за прошедший месяц из выбранного периода для обучения, а входные параметры – характеристики этого же месяца, которые могут повлиять на расход электроэнергии. Все входные параметры необходимо нормализовать на отрезке значений от 0 до 1. Нормализация достигается за счет деления каждой компоненты входного вектора на длину самого вектора. Длина вектора находится извлечением квадратного корня из суммы квадратов всех компонент вектора:

$$X_i^x = \frac{X_i}{\sqrt{x_1^2 + x_2^2 + \dots + x_n^2}}.$$

Цель процесса обучения заключается в нахождении оптимальных значений весовых коэффициентов, при которых нейронная сеть выдаёт значение в пределах заданной погрешности для всех обучающих выборок.

Реализация искусственной нейронной сети на языке программирования C++

Нейронная сеть реализована на языке программирования C++. Для реализации параллельного алгоритма использовались библиотеки CUDA.

На рис. 2 представлен фрагмент кода, реализующий функцию активации (в коде функция `ActiveSigm(int iter)`) по формуле (1). Функция `ProcNeuro(int iter)` выполняет последовательное суммирование произведений входных параметров на весовые коэффициенты нейронов. Функция вызывается циклически для всех нейронов слоя. Параметр функции `iter` – номер итерации прохождения слоя, т. е. номер нейрона в слое. Цикл `for(){}` внутри функции реализует последовательное прохождение всех весовых коэффициентов, относящихся к текущему нейрону. `INCOUT` – количество обучающих выборок, задается пользователем. При реализации функции активации для возведения основания натурального логарифма в степень используется функция `pow()` вместо `exp()`. Выбор обусловлен тем, что функция `exp()` в своем алгоритме использует функцию `pow()`, в результате чего появляется мелкая погрешность [19]. Но при многочисленном перемножении значений в длительном итерационном процессе обучения ИНС эта погрешность может многократно увеличиться.

```

//-----
void CNeuroNet::ProcNeuro(int iter)
{
    double Net = 0;    // NET - Взвешенная сумма входных сигналов
    for (int i = 0; i < INCOUNT; i++)
    {
        Net += m_Neuro_hidden[iter].x[i] * m_Neuro_hidden[iter].w[i];
    }
    m_Neuro_hidden[iter].sum = Net;
    m_Neuro_hidden[iter].output = ActiveSigm(iter);
}
// Функция активации
double CNeuroNet::ActiveSigm(int iter)
{
    double e = (double)E;
    double y = 1. / (1 + pow(e, -(m_Neuro_hidden[iter].sum)));
    return y;
}
//-----

```

Рис. 2. Листинг программы. Функция активации. Последовательная реализация

На рис. 3 представлен фрагмент кода, реализующий функцию активации для алгоритма параллельного прохождения сигнала по сети средствами расширений библиотек CUDA.

```

//-----
// *input - входные параметры для слоя
// *weights - весовые коэффициенты слоя
// *sum_vector - вспомогательный вектор для вычисления промежуточных сумм
// *output - выходные значения слоя
// Size - количество нейронов в слое
__global__ void MainKernel(float * input, float * weights, float *sum_vector, float
*output, int Size)
{
    float e = (float)E;
    int xIndex = blockIdx.x * blockDim.x + threadIdx.x;

    sum_vector[xIndex] = 0;
    output[xIndex] = 0;

    for (int k = 0; k < Size; k++)
    {
        sum_vector[xIndex] += weights[xIndex * Size + k] * input[k];
    }

    output[xIndex] = 1. / (1 + pow(e, -(sum_vector[xIndex])));
}
//-----

```

Рис. 3. Листинг программы. Функция активации. Параллельная реализация

Функция `MainKernel<<<>>>()` – функция-ядро, выполняется на графическом процессоре (GPU). Переменная `blockIdx` – индекс текущего блока в вычислении на GPU. Переменная `blockDim` – размерность текущего блока в вычислении на GPU. Переменная `threadIdx` – индекс текущей нити в вычислении на GPU. Цикл `for(){}` выполняет суммирование произведений весовых коэффициентов на входные параметры. Эта функция выполняется одновременно (параллельно) для всех нейронов слоя.

Эксперименты

Все вычисления производились на персональном компьютере:

- процессор (CPU): Intel(R) Core(TM) i5 760 2.8 GHz;
- ОЗУ: 8,00 Гб;
- тип системы: Windows 7 Профессиональная 64-разрядная.

Программы написаны в среде Microsoft Visual Studio 2013 и скомпилированы на встроенном компиляторе. При компиляции программы для параллельного алгоритма использовался компилятор, входящий в состав дополнительного набора инструментов CUDA Toolkit 6.5. Исполняемые файлы 64-разрядные.

Для выполнения программы с параллельной реализацией алгоритма обучения ИНС необходимо наличие графического процессора (видеокарты) NVIDIA. Нами использовалась видеокарта NVIDIA GeForce GT 240.

На рис. 4 представлен результат обучения сети в течение 10 секунд при проведении эксперимента, целью которого было сравнение полученной погрешности за равный интервал времени для алгоритмов с последовательной и параллельной реализацией вычислений. Очевидно, что уже с первых секунд процесса обучения сети результат алгоритма с параллельным прохождением сигнала по сети ближе к эталонному значению обучающей выборки. Это объясняется тем, что параллельный алгоритм успевает произвести больше эпох обучения, чем последовательный алгоритм и, следовательно, получает более точный результат. В обоих случаях использовались одни и те же данные и одна структура сети с идентичными параметрами.

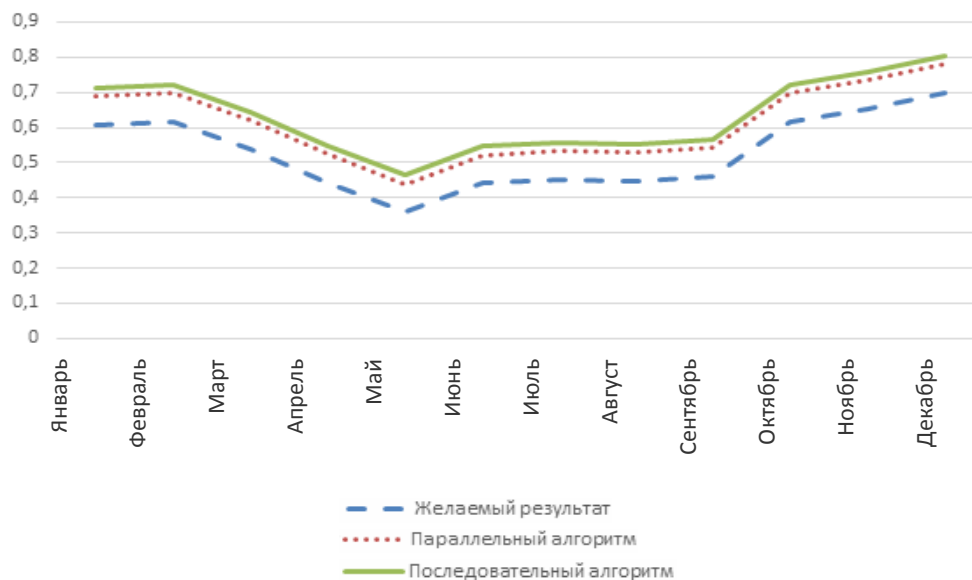


Рис. 4. Результаты обучения сети в течение первых 10 секунд (по вертикали – шкала выходных значений)

При реализации параллельного алгоритма незначительная часть мощности тратится на перенос данных в память GPU и обратно. Это схематично показано на рис. 5.

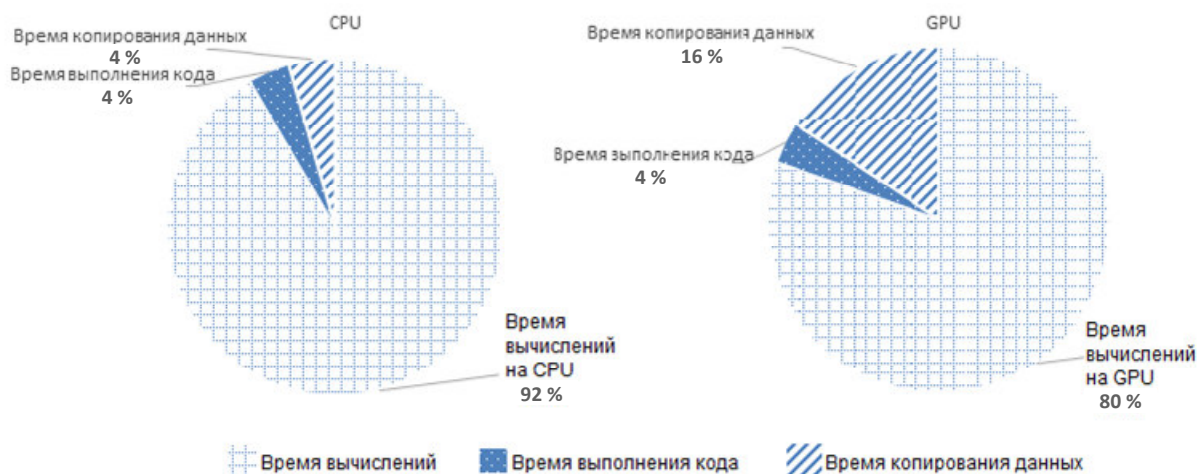


Рис. 5. Затраты вычислительной мощности

Время, затраченное на копирование данных при реализации параллельного обучения, будет значительно больше, чем при реализации последовательного алгоритма, но итоговое время обучения сети при параллельном алгоритме – во много раз меньше.

В таблице представлены результаты работы алгоритмов, полученные в ходе эксперимента, целью которого было достижения заданной точности (меньше 3 %).

Сравнение результатов выполнения последовательного и параллельного алгоритмов

Параметр	Алгоритм	
	последовательный	параллельный
Количество эпох обучения	1 млн	1 млн
Время выполнения, с	10365,5	1418,6
Абсолютная погрешность, %	2,9999	2,9999
Среднеквадратичное отклонение (MSE)	0,01689	0,01689

Согласно данным таблицы, очевидно значительное преимущество параллельного алгоритма обучения ИНС по времени выполнения.

Заключение

Таким образом, в ходе исследования были получены следующие результаты.

Обзор популярных методов прогнозирования расхода электроэнергии позволил привести доводы в пользу метода прогнозирования с использованием ИНС. Проведено исследование теории ИНС, построен алгоритм обучения нейронной сети. Полученный алгоритм был оптимизирован путем распараллеливания отдельных блоков. На языке программирования C++ разработаны программы, реализующие оба алгоритма. Программы были оптимизированы с точки зрения производительности, функциональности и надежности. Процесс распараллеливания осуществлен с использованием платформы параллельных вычислений CUDA, которая обеспечивает набор расширений для языков C и C++. Поставлен практический эксперимент с использованием реальных показателей СЭТЗ «Энергомера» и выполнено сравнение результатов работы программ.

В процессе исследования предложенной модели ИНС удалось добиться значения средней относительной погрешности прогноза на месяц меньше 3 % за значительно меньший интервал времени. Время обучения до заданной точности сокращается за счет параллельного выполнения вычислений алгоритма. Данная модель легко может быть перенастроена для любого предприятия. Внедрённая в модель технология распараллеливания вычислительного процесса CUDA позволила получить преимущество даже при обработке небольших объемов данных. Полученные результаты говорят о том, что в условиях ограниченности ресурсов разработанная модель параллельного обучения нейронной сети позволит получить более высокую точность прогнозирования при тех же ресурсах.

СПИСОК ЛИТЕРАТУРЫ

1. *Об электроэнергетике*: Федеральный закон РФ от 26.03.2003 № 35-ФЗ // Рос. газета. 2003. 1 апр. С. 1.
2. *Ахметьянов Р. Р., Делегодина Л. А., Копылова Н. П. и др.* Задачи прогнозирования энергопотребления в интегрированной АСКУЭ Новосибирского научного центра // Энергосбережение. 2007. № 1. С. 42–47.
3. *Ахметьянов Р. Р., Делегодина Л. А., Копылова Н. П., Луценко Б. Н., Собстель Г. М., Чейдо Г. П.* Мультипликативная модель сезонного энергопотребления предприятий // Автометрия. 2008. Т. 44, № 3. С. 106–118.
4. *Hong T., Shahidehpour M.* Load Forecasting Case Study. National Association of Regulatory Utility Commissioners, 2015. 171 p.
5. *Francisco V. Cipolla-Ficarra.* Handbook of Research on Interactive information Quality in Expanding Social Network Communications. Latin Association of Human-Computer Interaction, Spain & International Association of Interactive Communication, Italy, 2015. 449 p.
6. *Hyndman R. J., Athanasopoulos G.* Forecasting: principles and practice. Publisher: OTexts, 2013. 291 p.
7. *Zelinka I., Chen G., Rössler O. E., Snášel V., Abraham A.* Nostradamus 2013: Prediction, Modeling and Analysis of Complex Systems. Springer International Publishing, Switzerland, 2013. 550 p.
8. *Надтока И. И., Аль Зихери Б. М.* Байесовская нейросетевая модель для краткосрочного прогнозирования электропотребления региона // Студ. науч. весна: материалы регион. науч.-практ. конф. студ., аспирантов и молодых ученых вузов Ростов. обл. Новочеркасск: ЮРГПУ (НПИ), 2012. С. 136–138.
9. *Соловьева И. А., Дзюба А. П.* Прогнозирование электропотребления с учетом факторов технологической и рыночной среды // Научный диалог. 2013. № 7 (19). С. 97–111.
10. *Гофман А. В., Ведерников А. С., Ведерникова Е. С.* Повышение точности краткосрочного и оперативного прогнозирования электропотребления энергосистемы с применением искусственной нейронной сети // Электрические станции. 2012. № 7. С. 36–41.
11. *Алексеева И. Ю.* Краткосрочное прогнозирование электропотребления в электроэнергетических системах с использованием искусственных нейронных сетей: автореф. дис. ... канд. техн. наук. Иваново, 2014. 20 с.
12. *Shayeghi H.* STLF Based on Optimized Neural Network Using PSO // International Journal of Electrical and Computer Engineering. 2009. 4:10. P. 1190–1199.
13. *Хайкин С.* Нейронные сети: полный курс. М.: Изд. дом «Вильямс», 2006. 1104 с.

14. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. М.: Мир, 1992. 240 с.
15. Арзамасцев А. А., Крючин О. В., Азарова П. А., Зенкова Н. А. Универсальный программный комплекс для компьютерного моделирования на основе искусственной нейронной сети с самоорганизацией структуры // Вестн. Тамбов. ун-та. Сер.: Естественные и технические науки. 2006. Т. 11, вып. 5. С. 372–375.
16. Таранов Р. В. Вычисления на GPU. Применение библиотек ускоренных по технологии CUDA // Фундаментальные и прикладные исследования в современном мире: материалы IX Междунар. науч.-практ. конф. 2015. Т. 1. С. 100–109.
17. Боресков А. В., Харламов А. А. Основы работы с технологией CUDA. М.: ДМК Пресс, 2010. С. 26–44.
18. Shane Cook. CUDA Programming. A Developer's Guide to Parallel Computing with GPUs. Elsevier Inc., 2013. P. 69–104.
19. Microsoft Software Developer Network. URL: [https://msdn.microsoft.com/ru-ru/library/system.math.exp\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.math.exp(v=vs.110).aspx) (дата обращения: 09.04.2016).

Статья поступила в редакцию 30.03.2016

ИНФОРМАЦИЯ ОБ АВТОРАХ

Таранов Роман Владимирович – Россия, 355012, Ставрополь; Северо-Кавказский федеральный университет; аспирант кафедры прикладной информатики; Roman_boxh@mail.ru.

Маликов Андрей Валерьевич – Россия, 355012, Ставрополь; Северо-Кавказский федеральный университет; г-р техн. наук, профессор; профессор кафедры прикладной информатики; Amalikov@ncfu.ru.



R. V. Taranov, A. V. Malikov

FORECASTING OF ENERGY CONSUMPTION WITH ARTIFICIAL NEURAL NETWORKS USING CUDA TECHNOLOGY

Absrtact. The existing methods of forecasting energy consumption are analyzed. The choice of the method of forecasting with the help of artificial neural network is explained. The application of parallel calculations on GPU is described. The paper considers a new algorithm of forecasting energy consumption based on the theory of artificial neural networks using CUDA technology to optimize forecasting energy consumption at an industrial enterprise. For practical usage of the new algorithm a software for Windows is designed, it is programed by C++. The experiment is made; on its bases the analysis of software operation and calculation of the consumed resources are carried out. In accordance with the results of the experiments, the developed parallel algorithm has reached the required accuracy of forecasting for a short period of time. The application of the algorithm will allow the enterprises to get more accurate forecasting and reduce the costs concerning electricity payments.

Key words: methods of forecasting, energy consumption forecasting, neural network, parallel algorithm, CUDA technology.

REFERENCES

1. Ob elektroenergetike [On power engineering]. Federal'nyi zakon RF ot 26.03.2003 № 35-F3. *Rossiiskaia gazeta*, 2003, 1 apr., p. 1.
2. Akhmet'ianov R. R., Delegodina L. A., Kopylova N. P. i dr. Zadachi prognozirovaniia energopotrebleniia v integrirovannoi ASKUE Novosibirskogo nauchnogo tsentra [Tasks on forecasting energy consumption in the integrated ASKUE of Novosibirsk scientific center]. *Energoberezhenie*, 2007, no. 1, pp. 42–47.
3. Akhmet'ianov R. R., Delegodina L. A., Kopylova N. P., Lutsenko B. N., Sobstel' G. M., Cheido G. P. Mul'tiplikativnaia model' sezonnogo energopotrebleniia predpriatii [Multiplicative model of seasonal energy consumption at enterprises]. *Avtometriia*, 2008, vol. 44, no. 3, pp. 106–118.
4. Hong T., Shahidehpour M. *Load Forecasting Case Study*. National Association of Regulatory Utility Commissioners, 2015. 171 p.

5. Francisco V. Cipolla-Ficarra. *Handbook of Research on Interactive information Quality in Expanding Social Network Communications*. Latin Association of Human-Computer Interaction, Spain & International Association of Interactive Communication, Italy, 2015. 449 p.
6. Hyndman R. J., Athanasopoulos G. *Forecasting: principles and practice*. Publisher: OTexts, 2013. 291 p.
7. Zelinka I., Chen G., Rössler O. E., Snášel V., Abraham A. *Nostradamus 2013: Prediction, Modeling and Analysis of Complex Systems*. Springer International Publishing Switzerland, 2013. 550 p.
8. Nadtoka I. I., Al' Zikheri B. M. Baiesovskaia neurosetevaia model' dlia kratkosrochnogo prognozirovaniia elektropotrebleniia regiona [Bayes neural network model for short-term forecasting energy consumption in the region]. *Studencheskaia nauchnaia vesna. Materialy regional'noi nauchno-prakticheskoi konferentsii studentov, aspirantov i molodykh uchenykh vuzov Rostovskoi oblasti*. Novocherkassk, IuRGPU (NPI), 2012. P. 136–138.
9. Solov'eva I. A., Dziuba A. P. Prognozirovanie elektropotrebleniia s uchetom faktorov tekhnologicheskoi i rynochnoi sredy [Energy consumption forecasting with the factors of technological and market environment]. *Nauchnyi dialog*, 2013, no. 7 (19), pp. 97–111.
10. Gofman A. V., Vedernikov A. S., Vedernikova E. S. Povyshenie tochnosti kratkosrochnogo i operativnogo prognozirovaniia elektropotrebleniia energosistemy s primeneniem iskusstvennoi neuronnoi seti [Increase in accuracy of short-term and operative forecasting energy consumption of the energy system using artificial neural network]. *Elektricheskie stantsii*, 2012, no. 7, pp. 36–41.
11. Alekseeva I. Iu. *Kratkosrochnoe prognozirovanie elektropotrebleniia v elektroenergeticheskikh sistemakh s ispol'zovaniem iskusstvennykh neuronnykh setei. Avtoreferat dis. ... kand. tekhn. nauk* [Short-term forecasting of energy consumption in the electrical systems using artificial neural networks. Abstract of dis. cand. tech. sci.]. Ivanovo, 2014. P. 9–10.
12. Shayeghi H. STLF Based on Optimized Neural Network Using PSO. *International Journal of Electrical and Computer Engineering*, 2009, 4:10, pp. 1190–1199.
13. Haykin S. *Neural Networks. A Comprehensive Foundation*. Hamilton, Ontario, Canada, 1999. 823 p.
14. Wasserman F. *Neurocomputer Techniques: Theory and Practice*. Moscow, Mir Publ., 1992. 240 p. (in Russian).
15. Arzamastsev A. A., Kriuchin O. V., Azarova P. A., Zenkova N. A. Universal'nyi programmnyi kompleks dlia komp'iuternogo modelirovaniia na osnove iskusstvennoi neuronnoi seti s samoorganizatsiei struktury [Universal software complex for computer modeling based on artificial neural network with self-organizational structure]. *Vestnik Tambovskogo universiteta. Seriya: Estestvennye nauki i tekhnicheskie nauki*, 2006, vol. 11, iss. 5, pp. 372–375.
16. Taranov R. V. Vychisleniia na GPU. Primenenie bibliotek, uskorenykh po tekhnologii CUDA [GPU computing. Use of libraries accelerated by CUDA technology]. *Fundamental'nye i prikladnye issledovaniia v sovremennom mire. Materialy IX Mezhdunarodnoi nauchno-prakticheskoi konferentsii*, 2015, vol. 1, pp. 100–109.
17. Boretkov A. V., Kharlamov A. A. *Osnovy raboty s tekhnologiei CUDA* [Fundamentals of the operations with CUDA technology]. Moscow, DMK Press, 2010. P. 26–44.
18. Shane Cook. *CUDA Programming. A Developer's Guide to Parallel Computing with GPUs*. Elsevier Inc., 2013. P. 69–104.
19. *Microsoft Software Developer Network*. Available at: [https://msdn.microsoft.com/ru-ru/library/system.math.exp\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.math.exp(v=vs.110).aspx) (accessed: 09.04.2016).

The article submitted to the editors 30.03.2016

INFORMATION ABOUT THE AUTHORS

Taranov Roman Vladimirovich – Russia, 355012, Stavropol; North-Caucasus Federal University; Postgraduate Student of the Department of Applied Computer Science; Roman_boxx@mail.ru.

Malikov Andrey Valerievich – Russia, 355012, Stavropol; North-Caucasus Federal University; Doctor of Technical Sciences; Professor; Professor of the Department of Applied Computer Science; Amalikov@ncfu.ru.

