Распределенная оптимизация в условиях Поляка-Лоясиевича*

И. О. Автор[1]***, И. О. Соавтор***[2]***, И. О. Фамилия***[1,2]
author@site.ru; co-author@site.ru; co-author@site.ru
[1]Организация, адрес; [2]Организация, адрес

В статье рассматривается новый метод децентрализованного решения больших систем нелинейных уравнений в условиях Поляка-Лоясиевича. Суть метода состоит в постановке эквивалентной задачи децентрализованной ограниченной оптимизации. Полученная задача сводится к задаче композитной оптимизации, но уже без ограничений. Предложенный метод сравнивается с методами градиентного спуска, ускоренного градиентного спуска, а также с последовательным и параллельным алгоритмом обратного распространения ошибки. Сравнение производится при обучении многослойной нейронной сети с нелинейной функцией активации нейрона.

Ключевые слова: *большие нелинейные системы; распределенная оптимизация; условия Поляка-Лоясиевича; многослойные нейронные сети*

DOI: 10.21469/22233792

1 Введение

С ростом числа параметров моделей как в машинном обучении, так и в других областях прикладной математики растет и популярность задач анализа больших данных. Примерами таких задач могут послужить задачи обработки непрерывно поступающих данных с измерительных устройств; изучения потоков сообщений в социальных сетях или метеорологических данных; анализа данных о местонахождении абонентов сетей и оптимальное распределение мощности между вышками сотовой связи. Суть этих задач заключается в решении огрромных систем нелинейных уравнений.

В этой статье рассматривается новый способ решения системы нелинейных уравнений:

$$\begin{cases} f_1(x_1,\ldots,x_n)=0 \\ f_2(x_1,\ldots,x_n)=0 \\ \ldots \\ f_m(x_1,\ldots,x_n)=0 \end{cases} \qquad f_i:\mathbb{R}^n\to\mathbb{R},\ i=1,\ldots,m. \tag{1}$$

Эту задачу можно представить в виде эквивалентной задачи оптимизации:

$$g(x):=\sum_{i=1}^{m} f_i^2(x)\to\min_{x\in\mathbb{R}^n}. \tag{2}$$

Речь идет о решении огромных систем, поэтому возникает мысль решать их децентрализованно. В этой статье мы рассмотрим децентрализованную оптимизацию. Схематически децентралицованную систему можно представить как несколько устройств или процессоров, которые связаны в сеть, при этом какие-то устройсвта связаны между собой, а

*Работа выполнена при финансовой поддержке РФФИ, проекты № №00-00-00000 и 00-00-00001.

какие-то нет, соотвественно информацией могут обмениваться только те, между которыми есть канал связи. Примерами таких систем могут служить архитектуры из нескольких видеокарт или сеть из нескольких компьютеров.

В [1] задача (2) представляется в децентрализованном виде:

$$\begin{aligned} \min_{x_i \in \mathbb{R}^n} \quad & \sum_{i=1}^{m} f_i^2(x_i), \\ \text{s.t.} \quad & x_1 = x_2 = \ldots = x_m, \end{aligned} \tag{3}$$

где x_i – это копия переменной x на каждом устройстве. Требуется, чтобы на каждом устройстве было одинаковое значение x – одинаковое решение, поэтому и вводится соответсвующее ограничение.

Таким образом, ставится задача ограниченной оптимизации, которую авторы статьи [1] решают методом прямодвойственного градиентного спуска [5]. Причем, если функция g удовлетворяет условиям Поляка-Лоясиевича с константой $\nu > 0$, то есть:

$$\frac{1}{2}||\nabla g(x)||^2 \geqslant \nu(g(x) - g^*), \ \forall x \in \mathbb{R}^n; \ g^* = \min_{x \in \mathbb{R}^n} g(x), \tag{4}$$

то метод будет иметь линейную скорость сходимости.

В этой статье задача ограниченной оптимизации сводится к задаче композитной оптимизации путем смягчения жестких условий на совпадение $x_i{}_{i=1}^m$ в задаче (3). Полученную задачу предлагается решать аналогами метода подобных треугольников или слайдинга [4].

Предложенный метод сравнивается с методами градиентного спуска и ускоренного градиентного спуска, описанными в [2] и [5], и с последовательным и параллельным вариантами алгоритма обратного распространения ошибки для обучения нейронных сетей с нелинейной функцией активации нейрона, предложенными в [3]. Сравнение производится в ходе вычислительного эксперимента при обучении нейронных сетей с различным числом слоев и сигмоидной функцией активации нейрона. Обучение производится на данных CIFAR, MNIST, IMAGNET.

2 Постановка задачи

2.1 Определения и обозначения

Скалярное произведение двух векторов $x, y \in \mathbb{R}^n$ обозначается $\langle x, y \rangle := \sum_{i=1}^n x_i y_i$. Скалярное произведение порождает второую норму, ℓ_2-норма, в $\mathbb{R}^n$ в следующем виде $\|x\|_2 := = \sqrt{\langle x, x \rangle}$. Определим произвольную норму ℓ_p как $\|x\|_p := (\sum_{i=1}^n |x_i|^p)^{1p}$ для $p \in (1, \infty)$, а для $p = \infty$ используется $\|x\|_\infty := \max\limits_{1 \leqslant i \leqslant n} |x_i|$. Для максимального и минимального собственного значения положительно определенной матрицы $A \in \mathbb{R}^{n \times n}$ вводятся следующие обозначения $\lambda_{\max}(A)$ и $\lambda^+_{\min}(A)$ соотвественно и под $\chi(A) := \lambda_{\max}(A)\lambda^+_{\min}(A)$ понимается число обусловленностей матрицы A. Для обозначения произведения Кронекера двух матриц $A \in \mathbb{R}^{m \times m}$ и $B \in \mathbb{R}^{n \times n}$ используется $A \otimes B \in \mathbb{R}^{nm \times nm}$. Единичная матрица размера $n \times n$ обозначается I_n. Для обозначения неориентированного графа на множестве вершин V с ребрами E используется $G(V, E)$.

Определение 1 (матрица Кирхгофа). L – матрица Кирхгофа графа $G(V, E)$, если

$$L_{ij} = \begin{cases} -1, & \text{if } (i, j) \in E, \\ \deg(i), & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Введем так же матрицу связи агентов в децентрализованной системе $W = L \otimes I_n$.

Рассмотрим задачу (3) и для удобства введем обозначения:

Определим X, как столбец, составленный из векторов аргументов функций $\{f_i\}_{i=1}^m$, т.е.

$$X = \mathrm{col}(x_1, \ldots, x_m). \tag{6}$$

Обозначим целевую функцию задачи (3):

$$F(X) = \sum_{i=1}^{m} f_i^2(x_i). \tag{7}$$

2.2 Сведение к задаче композитной оптимизации

Поставленная задача решения большой системы нелинейных уравнений (1), была переписана в виде задачи децентрализованной оптимизации (3). Теперь будем минимизировать каждую из функций $f_i(x_i)$ на отдельном процессоре децентрализованной системы, связи между процессорами которой обеспечивают равенство решений $\{x_i\}_{i=1}^m$.

Таким образом, задача (3) представляется в виде эквивалентной задачи условной оптимизации [1]:

$$\begin{aligned} \min_{X \in \mathbb{R}^{mn}} \quad & F(X), \\ \text{s.t.} \quad & W^{1/2}X = 0. \end{aligned} \tag{8}$$

Здесь условие $W^{1/2}X = 0$ эквивалентно условию $WX = 0$, которое, в свою очередь, гарантирует совпадение решений на различных процессорах. Такая замена производится авторами статьи [1] для доказательства линейной скорости сходимости прямодвойственного градиентного спуска для этой задачи в условиях Поляка-Лоясиевича.

В этой статье предлагается убрать жесткие условия и свести задачу (8) к задаче композитной оптимизации:

$$\min_{X \in \mathbb{R}^{m \times n}} \quad F(X) + R||W^{1/2}X||. \tag{9}$$

Здесь R – это некоторая правильно подобранная положительная константа.

3 Численный эксперимент

3.1 Скорость сходимости

В первом эксперименте мы решаем линейную систему уравнений:

$$Ax = b. \tag{10}$$

В виде задачи оптимизации она представляется следующим образом:

$$\min_x \|Ax - b\|^2. \tag{11}$$

Рассматриваются случаи, когда матрица A – симметричная положительно определенная, семмитричная положительно полуопределенная и произвольная прямоугольная. Для генерации случайных симметричных положительно определенных или полупределнных матриц мы используем формулу $A = Q^T D Q$, соотвественно мы создаем диагональную матрицу D с нужными нам собственными числами, а с помощью QR-разложения получаем ортогональную матрицу Q.

В данном эксперименте на каждом из вычислителей используется градиентный спуск. График сходимости для матрицы размера 10×10 представлен на Рис. 1.

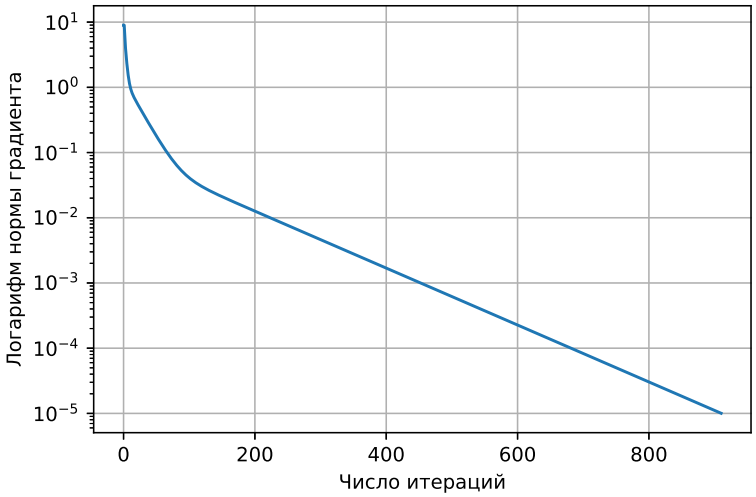

Рис. 1 Сходимость градиетного спуска для распределенного децентрализованного решения задачи (11).

3.2 Анализ ошибки

Для анализа ошибок рассматривается поведение члена $R||W^{1/2}X||$ в ходе оптимизации, который отвечает за синхронизацию решений на каждом устрйостве с другими. При хорошей работе метода ошибка синхронизации должна быть маленькой (см. Рис. 2).

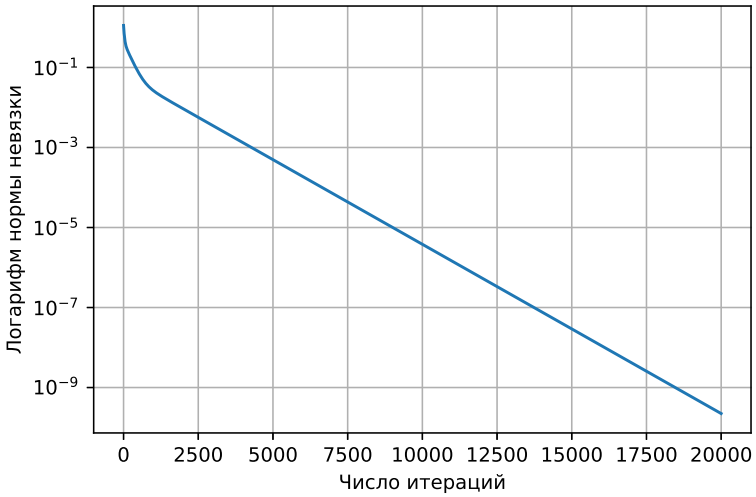

Рис. 2 .

Также анализируется поведение метода при разных R (см. Рис.2).

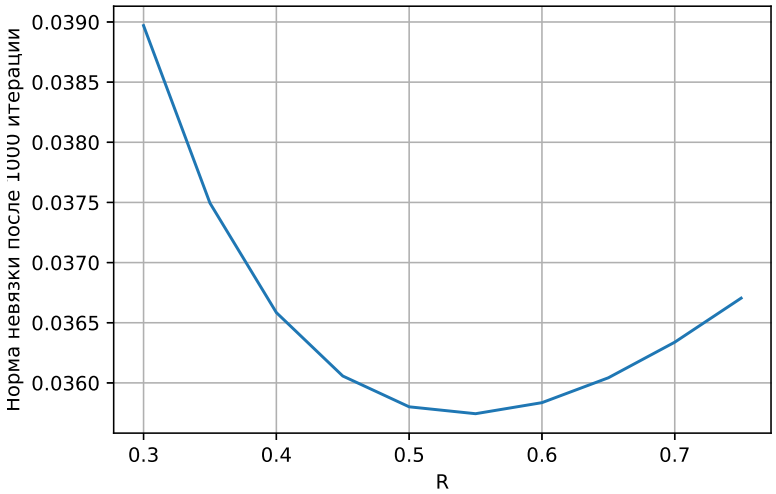

Рис. 3 .

На графике показано качество решения через 1000 итераций в зависимости от R. Видно, что при маленьких R решение плохое, это связано с плохой синхронизацией, так как член за нее отвечающий слишком мал. При больших R также наблюдается ухудшение решения, потому что член ответственный за синхронизацию начинает превалировать над основной задачей.

Литература

[1] Hamed Karimi, Julie Nutini, and Mark W. Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. *CoRR*, abs/1608.04636, 2016.

[2] Hamed Karimi, Julie Nutini, and Mark W. Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. *CoRR*, abs/1608.04636, 2016.

[3] G. Sandhya Prafulla. Speaker independent vowel recognition using backpropagation neural network on master-slave architecture jv.s. srinivas,, October 02 2013.

[4] А. В. Гасников. Универсальный метод для задач стохастической композитной оптимизации. 2016.

[5] А. В. Гасников. Современные численные методы оптимизации, метод универсального градиентного спуска. 2018.

Поступила в редакцию 01.01.2017