

УДК 004.8

МИНИМАКСНАЯ ОПТИМИЗАЦИЯ НА МЕДЛЕННО МЕНЯЮЩИХСЯ ГРАФАХ

© 2023 г. Н. Ч. Нгуен^{1,*}, А. Рогозин^{1,**}, Д. Метелев^{1,***}, А. Гасников^{1,2,3,4,****}

Представлено академиком РАН А.А. Шананиным

Поступило 03.09.2023 г.

После доработки 08.09.2023 г.

Принято к публикации 15.10.2023 г.

Распределенная оптимизация — важное направление исследований в современной теории оптимизации. Ее приложения включают машинное обучение на больших данных, распределенную обработку сигналов и другие области. В статье исследуется децентрализованная оптимизация для седловых задач. Седловые задачи возникают при обучении генеративно-состязательных сетей, а также в робастном машинном обучении. Основное внимание в работе уделяется оптимизации на медленно меняющихся сетях. Топология сети меняется время от времени, и скорость этих изменений ограничена. В работе показано, что достаточно изменять два ребра на каждой итерации для того, чтобы замедлить сходимость метода оптимизации до случая, когда граф меняется произвольным образом. Одновременно с этим исследуются несколько классов меняющихся сетей, для которых коммуникационная сложность может быть уменьшена.

Ключевые слова: седловая задача, децентрализованная оптимизация, меняющийся граф, экстраградиентный метод

DOI: 10.31857/S2686954323601781, **EDN:** CUFWWB

1. ВВЕДЕНИЕ

В данной статье изучаются задачи минимаксной оптимизации типа

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y) := \frac{1}{M} \sum_{m=1}^M f_m(x, y), \quad (1)$$

где функции $f_m(x, y)$ выпуклы по x и вогнуты по y , а $\mathcal{X}$, $\mathcal{Y}$ — замкнутые выпуклые множества. Каждая функция $f_m(x, y)$ хранится локально на некотором компьютере. Пусть эти компьютеры соеди-

нены друг с другом через децентрализованную коммуникационную сеть. Каждый компьютер может выполнять локальные вычисления и обмениваться информацией со своими непосредственными соседями в сети. Кроме того, сеть может меняться со временем. Из-за некоторых сбоев или помех связи в сети могут время от времени выходить из строя или появляться вновь. Этот тип сетей называется *меняющимися графами*.

У оптимизации на меняющихся графах существует множество приложений [13, 15]. К ним относятся распределенное машинное обучение [5, 14, 16], распределенное управление энергетическими системами [6, 17], управление группой автономных транспортных средств [18], управление распределенными системами сенсоров [1].

Методы децентрализованной оптимизации и минимаксной оптимизации первого порядка используют два типа шагов: локальные градиентные шаги и коммуникации между вычислительными машинами. Мы рассматриваем случай, когда коммуникация осуществляется синхронизированными раундами. Поэтому сложность метода измеряется двумя величинами: количеством коммуникационных раундов и количеством вызовов локального оракула. Эти величины зависят от характеристик задачи, к которым относятся число обусловленности сети χ , число обусловленности

¹Московский физико-технический институт (государственный университет), Долгопрудный, Россия

²Институт проблем передачи информации имени А.А. Харкевича Российской академии наук, Москва, Россия

³Кавказский математический центр Адыгейского государственного университета, Майкоп, Республика Адыгея, Россия

⁴Институт системного программирования им. В.П. Иванникова Российской академии наук (ИСП РАН), Москва, Россия

*E-mail: ngnhtrg@phystech.edu

**E-mail: aleksandr.rogozin@phystech.edu

***E-mail: metelev.ds@phystech.edu

****E-mail: gasnikov@yandex.ru

Таблица 1. Нижние оценки оптимизации

	стат.	меняющ.	мед. мен.
комм.	$\sqrt{\chi} \sqrt{\frac{L}{\mu}} \log \frac{1}{\varepsilon}$	$\chi \sqrt{\frac{L}{\mu}} \log \frac{1}{\varepsilon}$	$\chi \sqrt{\frac{L}{\mu}} \log \frac{1}{\varepsilon}$
оракул	$\sqrt{\frac{L}{\mu}} \log \frac{1}{\varepsilon}$	$\sqrt{\frac{L}{\mu}} \log \frac{1}{\varepsilon}$	$\sqrt{\frac{L}{\mu}} \log \frac{1}{\varepsilon}$
статья	[19]	[8]	[11]

Таблица 2. Нижние оценки для задач седловой точки

	стат.	меняющ.	мед. мен.
комм.	$\sqrt{\chi} \frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\chi \frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\chi \frac{L}{\mu} \log \frac{1}{\varepsilon}$
оракул	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$
статья	[4]	[2]	Эта статья

функции L/μ и желаемая точность ε . Здесь L — константа Липшица градиента целевой функции, а μ — константа сильной выпуклости.

Данная статья посвящена *медленно меняющимся графам*. Это означает, что после каждого шага обмена данными может меняться только ограниченное количество ребер. Мы приводим нижние оценки сложности для рассматриваемого класса задач. Также мы предлагаем методы минимаксной оптимизации с улучшенной коммуникационной сложностью для двух конкретных классов медленно меняющихся сетей.

Для оптимизации по сетям (не минимаксной) известны нижние оценки, а также соответствующие оптимальные алгоритмы. Для постоянных графов нижние оценки были получены в [19], и в той же статье был предложен оптимальный двойственный (т.е. с использованием двойственного оракула) алгоритм. Оптимальные децентрализованные методы с использованием прямого оракула были предложены в [9]. Для меняющихся графов (с произвольными изменениями на каждой итерации) в [8] были предложены нижние оценки сложности. Оптимальный алгоритм с прямым оракулом был предложен в той же статье [8], а оптимальный двойственный метод впервые появился в [10]. После этого в [12] были изучены нижние оценки для медленно меняющихся графов с разными скоростями изменений сети. В [11] было показано, что достаточно менять только два ребра на каждой итерации, чтобы сделать сложность коммуникации равной случаю произвольной меняющейся сети. Обзор нижних оценок для децентрализованной оптимизации представлен в табл. 1 (обозначение $\Omega(\cdot)$ опущено).

Результаты для децентрализованных седловых задач аналогичны стандартной децентрализованной оптимизации. Нижние оценки для минимаксной оптимизации по статическим графам приведены в [4]. В той же статье [4] предложены алгоритмы, оптимальные с точностью до логарифмического члена. Случай (произвольных) меняющихся графов изучался в [2] вместе с методами, оптимальными с точностью до логарифмического множителя. Оптимальные алгоритмы для вариационных неравенств типа суммы (обобщение седловых задач) были предложены в [7]. Наконец, в этой статье изучаются нижние оценки для седловых задач на медленно меняющихся графах (только два ребра меняются за итерацию). Соответствующие результаты представлены в табл. 2. Стоит отметить, что нижние границы сложности такие же, как и для оптимизации (табл. 1), за исключением замены $\sqrt{L/\mu}$ на L/μ .

Данная статья организована следующим образом. В разделе 2 мы вводим необходимые предположения и обозначения. После этого в разделе 3 мы покажем, как добиться ускорения по коммуникациям, используя дополнительные предположения о изменяющейся сети. В разделе 4 мы приводим нижние оценки для медленно меняющихся сетей.

2. ОБОЗНАЧЕНИЯ И ПРЕДПОЛОЖЕНИЯ

Гладкость и сильная выпуклость

Мы рассматриваем задачу (1), где множества $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ и $\mathcal{Y} \subseteq \mathbb{R}^{n_y}$ являются замкнутыми и выпуклыми. Кроме того, мы вводим множество $\mathcal{Z} = \mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^{n_z}$, $z = (x, y)$, $n_z = n_x + n_y$, и оператор F :

$$F_m(z) = F_m(x, y) = \begin{pmatrix} \nabla_x f_m(x, y) \\ -\nabla_y f_m(x, y) \end{pmatrix},$$

$$F(z) = \frac{1}{M} \sum_{m=1}^M F_m(z).$$

Предположение 1. Пусть функции $f(x, y)$ и $f_m(x, y)$ удовлетворяют следующим условиям:

1. Функция $f(x, y)$ является L -гладкой, т.е. для всех $z_1, z_2 \in \mathcal{Z}$ выполнено

$$\|F(z_1) - F(z_2)\| \leq L\|z_1 - z_2\|.$$

2. Для любого m , $f_m(x, y)$ является $L_{\max}$ -гладкой, т.е. для всех $z_1, z_2 \in \mathcal{Z}$ выполнено

$$\|F_m(z_1) - F_m(z_2)\| \leq L_{\max}\|z_1 - z_2\|.$$

3. Функция $f(x, y)$ является сильно выпуклой-сильно вогнутой с константой μ , т.е. для всех $z_1, z_2 \in \mathcal{Z}$ выполнено

Таблица 3. Верхние оценки для седловых задач над произвольными и медленно меняющимися графами

	произв. меняющ.	медленно меняющ.	связный остов	Марковский
комм.	$\chi \frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\chi \frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\sqrt{\chi} \log \chi \frac{L}{\mu} \log^2 \frac{1}{\varepsilon}$	$\tau \left(\sqrt{\chi} + \frac{\rho^2}{(\lambda_{\min}^+)^2} \right) \frac{L}{\mu} \log^2 \frac{1}{\varepsilon}$
оракул	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$	$\frac{L}{\mu} \log \frac{1}{\varepsilon}$
статья	[7]	[7]	Эта статья	Эта статья

$$\langle F(z_1) - F(z_2), z_1 - z_2 \rangle \geq \mu \|z_1 - z_2\|^2.$$

Децентрализованная коммуникация

На каждом шаге обмена данными мы используем граф для представления соединения между вычислительными машинами. Обозначим коммуникационную сеть, меняющуюся во времени, последовательностью графов $\{\mathcal{G}^k\}_{k=0}^\infty = \{(\mathcal{V}, \mathcal{E}^k)\}_{k=0}^\infty$, где $\mathcal{V} = \{1, \dots, M\}$ — множество узлов (каждый узел соответствует одному компьютеру), а $\mathcal{E}^k$ — множество доступных соединений на k -м коммуникационном раунде. Для каждого узла $m \in \mathcal{V}$, мы используем $\mathcal{N}_m^k = \{i \in \mathcal{V} \mid (i, m) \in \mathcal{E}^k\}$ для обозначения множества его соседей на k -м раунде, и в это время он может взаимодействовать только с узлами в $\mathcal{N}_m^k$.

Матрицы коммуникации. Каждый узел m содержит свой локальный вектор $z_m = (x_m, y_m)$. Требуется удовлетворить ограничения консенсуса $z_1 = \dots = z_M$. Для этой цели мы используем *матрицы коммуникации*.

Предположение 2. *Каждый граф в коммуникационной сети соответствует матрице $W^k \in \mathbb{R}^{M \times M}$, которая удовлетворяет следующим свойствам.*

1. W^k является положительно полуопределенной,
2. $W_{i,j}^k = 0$ если $i \neq j$ и $(i, j) \notin \mathcal{E}^k$,
3. $\ker W^k = \text{span}(\mathbf{1})$, где $\mathbf{1} = (1, \dots, 1) \in \mathbb{R}^M$.

Число $\chi(W) = \frac{\lambda_{\max}(W)}{\lambda_{\min}^+(W)}$ называется *числом обусловленности* матрицы W , где $\lambda_{\max}(W)$ и $\lambda_{\min}^+(W)$ обозначают соответственно наибольшее и наименьшее положительные собственные значения матрицы коммуникации W . Для меняющейся сети $\{\mathcal{G}^k\}_{k=1}^\infty$, число обусловленности имеет вид $\chi =$

$$= \sup_{k \in \mathbb{N} \cup \{0\}} \frac{\lambda_{\max}(W^k)}{\lambda_{\min}^+(W^k)}.$$

В этой статье мы рассматриваем матрицы Лапласа $L(\mathcal{G}^k)$, которые являются типичным примером матрицы коммуникации.

Мы вводим *консенсусное подпространство* $\mathcal{L} \subseteq \mathbb{R}^{Mn_z}$, определенное как

$$\mathcal{L} = \{z = (z_1^T, \dots, z_M^T)^T \in \mathbb{R}^{Mn_z} : z_1 = \dots = z_M\}. \quad (2)$$

Рассмотрим также пространство $\mathcal{L}^\perp \subseteq \mathbb{R}^{Mn_z}$ которое является ортогональным дополнением к пространству $\mathcal{L}$, определенное как

$$\mathcal{L}^\perp = \left\{ z = (z_1^T, \dots, z_M^T)^T \in \mathbb{R}^{Mn_z} : \sum_{m=1}^M z_m = 0 \right\}. \quad (3)$$

3. ВЕРХНИЕ ОЦЕНКИ

В этом разделе мы рассмотрим два класса меняющихся во времени сетей: сети со связным остовом и сети с небольшими случайными марковскими изменениями. Для обоих сценариев мы предлагаем децентрализованный алгоритм оптимизации, использующий вспомогательную процедуру консенсуса. Показано, что для рассмотренных классов задач зависимость сложности коммуникации от фактора χ можно свести от χ до $\sqrt{\chi}$ с дополнительными слагаемыми. Обзор результатов представлен в табл. 3 (обозначение $O(\cdot)$ опущено).

Наши алгоритмы основаны на экстраградиентном методе с консенсусной процедурой. После каждого экстраградиентного шага они совершают несколько раундов общения. После достаточного количества сообщений достигается консенсус с желаемой точностью. Такое приближенное усреднение делает траектории вычислительных машин практически синхронизированными. Для каждого класса меняющихся во времени сетей мы используем соответствующую процедуру консенсуса и включаем ее в экстраградиентный метод, чтобы получить децентрализованный алгоритм оптимизации.

Ускоренный консенсус для графов со связным остовом и невозстанавливаемыми ребрами

Рассмотрим тип графов со связным остовом. Мы предполагаем, что все графы в последовательности имеют общий связный подграф, который мы называем *связным остовом*. Ребра по-прежнему могут появляться и исчезать, но каждая машина запоминает, какие инцидентные ей ребра хотя бы раз вышли из строя, и прекращает связь по этим направлениям. Эту стратегию мы называем *невозстанавливаемыми ребрами*. Фактически сеть связи только теряет ребра на каждой ите-

рации, но остается связной. Другими словами, интересующую последовательность графов можно назвать “монотонно убывающей”.

Предположение 3 Последовательность графов $\{\mathcal{G}^k = (\mathcal{V}, \mathcal{E}^k)\}_{k=0}^{\infty}$ имеет связный остов: существует связный граф $\hat{\mathcal{G}} = (\mathcal{V}, \hat{\mathcal{E}})$ такой, что для всех $k \in \mathbb{N} \cup \{0\}$ имеем $\hat{\mathcal{E}} \subset \mathcal{E}^k$, $\lambda_{\max}(\mathbf{L}(\mathcal{G}^k)) \leq \lambda_{\max}$ и $\lambda_{\min}^+ \leq \lambda_{\min}^+(\hat{\mathcal{G}})$.

Учитывая эти свойства сети, мы вводим следующий алгоритм консенсуса.

Algorithm 1. Ускоренный консенсус с невозстанавливаемыми ребрами (AccGossipNonRecoverable)

Вход: Векторы $z_1, \dots, z_M$, количество итераций H , текущий раунд связи k_0 . Размеры шага $\eta, \beta > 0$

- 1: Построить вектор-столбец $\mathbf{z} = (z_1^T \dots z_M^T)^T$
 - 2: Пусть $\mathbf{u}^0 = \mathbf{z}^0 = \mathbf{z}$
 - 3: Каждая машина $i = 1, 2, \dots, M$ инициализирует набор соседей $\mathcal{N}_i = \mathcal{N}_i^{k_0}$
 - 4: **for** $k = 0, 1, \dots, H - 1$ **do**
 - 5: **for** $i = 1, 2, \dots, M$ **do**
 - 6: Обновить набор машин, с которыми общается машина: $\mathcal{N}_i = \mathcal{N}_i \cap \mathcal{N}_i^{k_0+k}$
 - 7: $u_i^{k+1} = z_i^k - \eta \left(|\mathcal{N}_i| z_i^k - \sum_{j \in \mathcal{N}_i} z_j^k \right)$
 - 8: $z_i^{k+1} = (1 + \beta) u_i^{k+1} - \beta u_i^k$
 - 9: **end for**
 - 10: **end for**
 - 11: **return** $z_1^H, \dots, z_M^H$
-

Лемма 1 (Из доказательства теоремы 4.3 в [12]).

Пусть предположение 3 выполнено и $\{\hat{z}_m\}_{m=1}^M$ будет выходом алгоритма 1 с входом $\{z_m\}_{m=1}^M$ и размеры шага $\eta = 1/\lambda_{\max}$, $\beta = (\sqrt{\chi} - 1)/(\sqrt{\chi} + 1)$, где $\chi = \lambda_{\max}/\lambda_{\min}^+$. Тогда после H итераций имеем следующее неравенство

$$\frac{1}{M} \sum_{m=1}^M \|\hat{z}_m - \bar{z}\|^2 \leq \frac{2\chi}{M} \left(1 - \frac{1}{\sqrt{\chi}}\right)^H \sum_{m=1}^M \|z_m - \bar{z}\|^2, \quad (4)$$

где $\bar{z} = \frac{1}{M} \sum_{m=1}^M z_m$.

На основе алгоритма 1 можно разработать децентрализованный алгоритм решения седловой задачи 1 с последовательностью графов, имеющих связный остов и невозстанавливаемые ребра.

Теорема 1. Пусть выполнены предположения 1 и 3. Пусть задача (1) решается алгоритмом 2 с

$\gamma \leq \frac{1}{4L_{\max}}$. Тогда, чтобы достигать консенсуса с точностью ϵ_0 на каждой итерации, требуется

$$H = \left(\sqrt{\chi} \log \left[\chi \left(4 + \frac{\frac{1}{2} \|z^0 - z^*\|^2 + \frac{Q^2}{2L_{\max}^2}}{\epsilon_0^2} \right) \right] \right)_{\text{коммуникаций}},$$

где $Q^2 = \frac{1}{M} \sum_{m=1}^M \|F_m(z^*)\|^2$ и z^* является решением (1).

Доказательство. Пусть после k итераций достигнута точность консенсуса ϵ_0 , т.е.

$$\frac{1}{M} \sum_{m=1}^M \|z_m^k - z^k\|^2 \leq \epsilon_0^2.$$

Algorithm 2. Экстраградиентный метод на переменных сетях с невозстанавливаемыми ссылками

Вход: Размер шага $\gamma > 0$; количество AccGossipNonRecoverable шагов H , количество раундов коммуникации K , количество итераций N

- 1: Выбрать $(x^0, y^0) = z^0 \in \mathcal{Z}$, $z_m^0 = z^0$
- 2: **for** $k = 0, 1, 2, \dots, N-1$ **do**
- 3: Каждая машина m вычисляет $\hat{z}_m^{k+1/2} = z_m^k - \gamma \cdot F_m(z_m^k)$
- 4: Коммуникация: $\hat{z}_1^{k+1/2}, \dots, \hat{z}_M^{k+1/2} = \text{AccGossipNonRecoverable}(\hat{z}_1^{k+1/2}, \dots, \hat{z}_M^{k+1/2}, H)$
- 5: Каждая машина m вычисляет $z_m^{k+1/2} = \text{proj}_{\mathcal{Z}}(\hat{z}_m^{k+1/2})$
- 6: Каждая машина m вычисляет $\hat{z}_m^{k+1} = z_m^{k+1/2} - \gamma \cdot F_m(z_m^{k+1/2})$
- 7: Коммуникация: $\hat{z}_1^{k+1}, \dots, \hat{z}_M^{k+1} = \text{AccGossipNonRecoverable}(\hat{z}_1^{k+1}, \dots, \hat{z}_M^{k+1}, H)$
- 8: Каждая машина m вычисляет $z_m^{k+1} = \text{proj}_{\mathcal{Z}}(\hat{z}_m^{k+1})$
- 9: **end for**

Введем обозначения:

$$g_m^k = F_m(z_m^k), \quad g_m^{k+1/2} = F_m(z_m^{k+1/2}),$$

и

$$z^k = \frac{1}{M} \sum_{m=1}^M z_m^k, \quad z^{k+1/2} = \frac{1}{M} \sum_{m=1}^M z_m^{k+1/2},$$

$$g^k = \frac{1}{M} \sum_{m=1}^M g_m^k, \quad g^{k+1/2} = \frac{1}{M} \sum_{m=1}^M g_m^{k+1/2},$$

$$\hat{z}^k = \frac{1}{M} \sum_{m=1}^M \hat{z}_m^k, \quad \hat{z}^{k+1/2} = \frac{1}{M} \sum_{m=1}^M \hat{z}_m^{k+1/2},$$

$$\tilde{z}^k = \frac{1}{M} \sum_{m=1}^M \tilde{z}_m^k, \quad \tilde{z}^{k+1/2} = \frac{1}{M} \sum_{m=1}^M \tilde{z}_m^{k+1/2}.$$

Имеем

$$\frac{1}{M} \sum_{m=1}^M \|g_m^k - g^k\|^2 \leq \frac{1}{M} \sum_{m=1}^M \|g_m^k\|^2.$$

Пусть $T = 2\chi \left(1 - \frac{1}{\sqrt{\chi}}\right)^H$. Тогда

$$\begin{aligned} \frac{1}{M} \sum_{m=1}^M \|z_m^{k+1/2} - \tilde{z}^{k+1/2}\|^2 &\leq \frac{T}{M} \sum_{m=1}^M \|\hat{z}_m^{k+1/2} - \hat{z}^{k+1/2}\|^2 = \\ &= \frac{T}{M} \sum_{m=1}^M \|z_m^k - \gamma g_m^k - z^k - \gamma g^k\|^2 \leq \end{aligned}$$

$$\begin{aligned} &\leq \frac{2T}{M} \sum_{m=1}^M \|z_m^k - z^k\|^2 + \frac{2T\gamma^2}{M} \sum_{m=1}^M \|g_m^k - g^k\|^2 \leq \\ &\leq \frac{2T}{M} \sum_{m=1}^M \|z_m^k - z^k\|^2 + \frac{2T\gamma^2}{M} \sum_{m=1}^M \|g_m^k\|^2 = \\ &= 2T\epsilon_0^2 + \frac{2T\gamma^2}{M} \sum_{m=1}^m \|g_m^k\|^2. \end{aligned}$$

С другой стороны,

$$\begin{aligned} \frac{1}{M} \sum_{m=1}^M \|g_m^k\|^2 &= \frac{1}{M} \sum_{m=1}^M \|F_m(z_m^k)\|^2 \leq \\ &\leq \frac{2}{M} \sum_{m=1}^M \|F_m(z_m^k) - F_m(z^*)\|^2 + \frac{2}{M} \sum_{m=1}^M \|F_m(z^*)\|^2 \leq \\ &\leq 2L_{\max}^2 \|z_m^k - z^*\|^2 + \frac{2}{M} \sum_{m=1}^M \|F_m(z^*)\|^2 \leq \\ &\leq 2L_{\max}^2 \|z_m^0 - z^*\|^2 + \frac{2}{M} \sum_{m=1}^M \|F_m(z^*)\|^2. \end{aligned}$$

Пусть $Q^2 = \frac{1}{M} \sum_{m=1}^M \|F_m(z^*)\|^2$, тогда имеем

$$\begin{aligned} \frac{1}{M} \sum_{m=1}^M \|z_m^{k+1/2} - z^{k+1/2}\|^2 &= \\ &= \frac{1}{M} \sum_{m=1}^M \|\text{proj}_{\mathcal{Z}} \hat{z}_m^{k+1/2} - \text{proj}_{\mathcal{Z}} \hat{z}^{k+1/2}\|^2 \leq \\ &\leq \frac{1}{M} \sum_{m=1}^M \|\hat{z}_m^{k+1/2} - \hat{z}^{k+1/2}\|^2 \leq \\ &\leq 2T(\epsilon_0^2 + 2\gamma^2(L_{\max}^2 \|z_m^0 - z^*\|^2 + Q^2)) \leq \\ &\leq 2T \left(\epsilon_0^2 + \frac{1}{8} \|z_m^0 - z^*\|^2 + \frac{Q^2}{8L_{\max}^2} \right) = \\ &= \chi \left(1 - \frac{1}{\sqrt{\chi}}\right)^H \left(4\epsilon_0^2 + \frac{1}{2} \|z^0 - z^*\|^2 + \frac{Q^2}{2L_{\max}^2} \right). \end{aligned}$$

Если положить

$$H \geq \sqrt{\chi} \log \left[\chi \left(4 + \frac{\frac{1}{2} \|z^0 - z^*\|^2 + \frac{Q^2}{2L_{\max}^2}}{\epsilon_0^2} \right) \right],$$

то

$$\frac{1}{M} \sum_{m=1}^M \|z_m^{k+1/2} - z^{k+1/2}\|^2 \leq \epsilon_0^2.$$

Аналогично получается точно такая же оценка для H , дающая

$$\frac{1}{M} \sum_{m=1}^M \|z_m^{k+1} - z^{k+1}\|^2 \leq \varepsilon_0^2.$$

Следовательно, для достижения точности ε_0 , необходимо выполнить

$$H = \mathbb{O} \left(\sqrt{\chi} \log \left[\chi \left(4 + \frac{\frac{1}{2} \|z^0 - z^*\|^2 + \frac{Q^2}{2L_{\max}^2}}{\varepsilon_0^2} \right) \right] \right)$$

коммуникаций.

Теорема 2 (Теорема 6 в [3]). Пусть $\{z_m^k\}_{k \geq 0}$ – итерации алгоритма 2 для решения задачи (1). Пусть выполняются предположения 1, 3. Тогда, если $\gamma \leq \frac{1}{4L_{\max}}$, имеем следующие оценки

$$\|z^N - z^*\| = \mathbb{O} \left(\|z^0 - z^*\|^2 \exp \left(-\frac{\mu K}{8L \cdot H} \right) \right).$$

Следствие 2.1. В условиях теоремы 1 и теоремы 2, если $H = \mathbb{O} \left(\sqrt{\chi} \log \chi \log(1/\varepsilon) \right)$, тогда количество коммуникационных раундов, требующихся алгоритму 2 для достижения точности ε , ограничено сверху следующим выражением:

$$\mathbb{O} \left(\sqrt{\chi} \log \chi \frac{L}{\mu} \log^2 \frac{1}{\varepsilon} \right),$$

и количество локальных вычислений на каждом устройстве не превосходит

$$\mathbb{O} \left(\frac{L}{\mu} \log \frac{1}{\varepsilon} \right).$$

Консенсус в сетях с марковскими изменениями

Данный раздел посвящен медленно меняющимся графам со случайными изменениями. На каждой итерации несколько случайно выбранных ребер появляются или исчезают. Выбор ребер за-

висит только от текущей топологии графа, поэтому последовательность графов является марковским процессом.

Введем предположения на переменный граф с марковскими изменениями.

Предположение 4. Коммуникационная сеть удовлетворяет следующим условиям.

1. $\{W^k\}_{k=0}^\infty$ является стационарной марковской цепью на (W_G, W_σ) , где W_G – набор всевозможных матриц коммуникации, W_σ – σ -поле над W_G и цепь $\{W^k\}_{k=0}^\infty$ имеет марковское ядро Q и единственное стационарное распределение π .

2. Q является равномерно геометрически эргодичным со временем перемешивания $\tau \in \mathbb{N}$, т.е. для каждого $m \in \mathbb{N}$,

$$\Delta(Q^m) = \sup_{W, W' \in W_G} (1/2) \|Q^m(W, \cdot) - Q^m(W', \cdot)\|_{TV} \leq (1/4)^{\lfloor m/\tau \rfloor}.$$

3. Для всех $k \in \mathbb{N} \cup \{0\}$ выполняется $\mathbb{E}_\pi[W(q)] = \bar{W}$ и $\bar{W}$ удовлетворяет предположению 2.

Обозначим $\lambda_{\max} = \lambda_{\max}(\bar{W})$, $\lambda_{\min}^+ = \lambda_{\min}^+(\bar{W})$, $\chi = \frac{\lambda_{\max}}{\lambda_{\min}^+}$.

4. Для всякого графа $\mathcal{G}$ в последовательности сетей выполняется:

$$\|W(\mathcal{G}) - \bar{W}\| \leq \rho.$$

Рассмотрим следующую задачу поиска консенсуса

$$\begin{aligned} \min_{z \in \mathbb{R}^{Mn_z}} [r(z) &= \|(\sqrt{\bar{W}} \otimes \mathbf{I}_{n_z})z\|^2] \\ \text{s.t. } \sum_{m=1}^M z_m &= \sum_{m=1}^M z_m^0, \end{aligned} \quad (5)$$

где $z = (z_1^T, \dots, z_M^T)^T$.

Algorithm 3. Ускоренный консенсус для графов с марковскими изменениями (ACOGWMC)

Вход: Размер шага $\gamma > 0$, коэффициенты инерции $\theta, \eta, \beta, \rho$, количество итераций N , предельный размер батча S

- 1: Выбрать $z_f^0 = z^0$, $T^0 = 0$, одинаково инициализировать генератор случайных чисел $\{J_k\}$ на всех устройствах
- 2: **for** $k = 0, 1, \dots, N-1$ **do**
- 3: $z_g^k = \theta z_f^k + (1 - \theta) z^k$
- 4: Сгенерировать $J_k \sim \text{Geom}(1/2)$
- 5: Разослать z_g^k соседям в сетях $\{\mathcal{G}^{T^k+i}\}_{i=1}^{J_k B}$

6: Вычислим $g^k = g_0^k + \begin{cases} 2^{J_k}(g_{J_k}^k - g_{J_k-1}^k), & \text{если } 2^{J_k} \leq S \\ 0, & \text{иначе} \end{cases}$

$$c g_j^k = 2^{-j} B^{-1} \sum_{i=1}^{2^j B} W^{T^{k+i}} z_g^k$$

7: $z_f^{k+1} = z_g^k - p \gamma g^k$

8: $z^{k+1} = \eta z_f^{k+1} + (p - \eta) z_f^k + (1 - p)(1 - \beta) z^k + (1 - p) \beta z_g^k$

9: $T^{k+1} = T^k + 2^{J_k} B$

10: end for

Теорема 3 (Теорема 1 из [11]). Пусть выполнены предположения 4. Пусть задача (5) решается алгоритмом 3. Тогда для любого $b \in \mathbb{N}$,

$$\gamma \in \left(0; \min \left\{ \frac{3}{4\lambda_{\max}}; \frac{\lambda_{\min}^3}{[1800\rho^2(\tau b^{-1} + \tau^2 b^{-2})]^2} \right\} \right),$$

и $\beta, \theta, \eta, p, S, B$, удовлетворяющих

$$p = \frac{1}{4}, \quad \beta = \sqrt{\frac{4\rho^2\mu\gamma}{3}},$$

$$\eta = \frac{3\beta}{\rho\mu\gamma} = \frac{\sqrt{12}}{\sqrt{\mu\gamma}}, \quad \theta = \frac{p\eta^{-1} - 1}{\beta p\eta^{-1} - 1},$$

$$S = \max \left\{ 2; \sqrt{\frac{1}{4} \left(1 + \frac{2}{\beta} \right)} \right\}, \quad B = \lceil b \log_2 S \rceil,$$

выполняется

$$\mathbb{E} \left[\|z^N - z^*\|^2 + \frac{24}{\lambda_{\min}} (r(z_f^N) - r(z^*)) \right] =$$

$$= \mathbb{O} \left(\exp \left(-N \sqrt{\frac{\rho^2 \lambda_{\min} \gamma}{3}} \right) \times \right.$$

$$\left. = \left[\|z^0 - z^*\|^2 + \frac{24}{\lambda_{\min}} (r(z^0) - r(z^*)) \right] \right),$$

где $z_M^* = \frac{1}{M} \sum_{i=1}^M z_i$ для $m = 1, \dots, M$.

Следствие 3.1. В предположениях теоремы 3, если $b = \tau$ и $\gamma \simeq \min \left\{ \frac{1}{\lambda_{\max}}; \frac{\lambda_{\min}^3}{\rho^4} \right\}$, тогда для достижения решения с точностью ε (в терминах $\mathbb{E}[\|z - z^*\|^2] \leq \varepsilon$) требуется

$$\tilde{\mathcal{O}} \left(\tau \left[\sqrt{\chi} + \frac{\rho^2}{\lambda_{\min}^2} \log \frac{1}{\varepsilon} \right] \right)$$

коммуникаций.

Algorithm 4. Экстраградиентный метод на переменных сетях с марковскими изменениями

Вход: Размер шага $\gamma \leq \frac{1}{4L}$, количество ACOGWMС шагов H , количество итераций N

1: Выбрать $(x^0, y^0) = z^0 \in \mathcal{Z}$, $z_m^0 = z^0$

2: for $k = 0, 1, 2, \dots, N$ do

3: Каждая машина m вычисляет $\hat{z}_m^{k+1/2} = z_m^k - \gamma \cdot F_m(z_m^k)$

4: Коммуникация: $\tilde{z}_1^{k+1/2}, \dots, \tilde{z}_M^{k+1/2} = \text{ACOGWMC}(\hat{z}_1^{k+1/2}, \dots, \hat{z}_M^{k+1/2}, H)$

5: Каждая машина m вычисляет $z_m^{k+1/2} = \text{proj}_{\mathcal{Z}}(\tilde{z}_m^{k+1/2})$

6: Каждая машина m вычисляет $\hat{z}_m^{k+1} = z_m^{k+1/2} - \gamma \cdot F_m(z_m^{k+1/2})$

7: Коммуникация: $\tilde{z}_1^{k+1}, \dots, \tilde{z}_M^{k+1} = \text{ACOGWMC}(\hat{z}_1^{k+1}, \dots, \hat{z}_M^{k+1}, H)$

8: Каждая машина m вычисляет $z_m^{k+1} = \text{proj}_{\mathcal{Z}}(\tilde{z}_m^{k+1})$

9: end for

Теорема 4. Пусть выполнены предположения 1, 4. Пусть проблема (1) решается алгоритмом 4. Тогда, если $\gamma \leq \frac{1}{4L_{\max}}$ и $H = \mathcal{O}\left(\tau \left[\sqrt{\chi} + \frac{\rho^2}{\lambda_{\min}^2} \log \frac{1}{\varepsilon}\right]\right)$, то для достижения ε -решения (такого что $\mathbb{E}[f(z) - f(z^*)] \leq \varepsilon$) требуется

$$\tilde{\mathcal{O}}\left(\tau \left[\sqrt{\chi} + \frac{\rho^2}{\lambda_{\min}^2}\right] \frac{L}{\mu} \log^2 \frac{1}{\varepsilon}\right)$$

коммуникаций и

$$\mathcal{O}\left(\frac{L}{\mu} \log \frac{1}{\varepsilon}\right) \text{ локальных вызовов}$$

оракула на каждом узле.

4. НИЖНИЕ ОЦЕНКИ

Предыдущий раздел был посвящен применению новых алгоритмов консенсуса для конкретных типов меняющихся сетей: сетей со связным остовом и графам, меняющимся по марковскому закону. В этом разделе мы покажем, что без конкретных ограничений на сети, таких как связный остов или марковские изменения, ускорение не достигается на медленно меняющихся графах общего вида, даже если ограничения на скорость изменений очень высоки. То есть при постоянных сетевых ограничениях, когда за итерацию изменяется не более постоянного числа ребер в сети, наихудшая зависимость от χ не может быть улучшена по сравнению с произвольно меняющимися сетями. В частности, мы показываем, что за итерацию достаточно изменить только два ребра.

Мы начнем с определения класса алгоритмов в модели черного ящика, на основе которого мы будем строить нижнюю оценку.

Определение 1. Алгоритм с T локальными итерациями и K коммуникационных раундами, удовлетворяющими следующим свойствам, работающий в модели черного ящика, обозначается $\mathbf{BBP}(T, K)$.

Каждый узел m поддерживает локальную память с $\mathcal{M}_m^x$ и $\mathcal{M}_m^y$ для x - и y -переменных, которые инициализируются как $\mathcal{M}_m^x = \mathcal{M}_m^y = \{0\}$. $\mathcal{M}_m^x$ и $\mathcal{M}_m^y$ обновляются следующим образом:

- **Локальные вычисления:** Каждый узел m вычисляет и добавляет к своим $\mathcal{M}_m^x$ и $\mathcal{M}_m^y$ конечное число точек x , y , каждая из которых удовлетворяет

$$x \in \text{span}\{x', \nabla_x f_m(x'', y'')\},$$

$$y \in \text{span}\{y', \nabla_y f_m(x'', y'')\},$$

для данных $x', x'' \in \mathcal{M}_m^x$ и $y', y'' \in \mathcal{M}_m^y$.

- **Коммуникация:** $\mathcal{M}_m^x$ и $\mathcal{M}_m^y$ обновляются следующим образом

$$\mathcal{M}_m^x := \text{span}\left\{\bigcup_{(i,m) \in \mathcal{E}^k} \mathcal{M}_i^x\right\}, \quad \mathcal{M}_m^y := \text{span}\left\{\bigcup_{(i,m) \in \mathcal{E}^k} \mathcal{M}_i^y\right\},$$

где $\mathcal{G}^k = (\mathcal{V}, \mathcal{E}^k)$ — текущее состояние сети.

- **Выход:** Окончательный глобальный результат в текущий момент времени рассчитывается как:

$$\hat{x} \in \text{span}\left\{\bigcup_{m=1}^M \mathcal{M}_m^x\right\}, \quad \hat{y} \in \text{span}\left\{\bigcup_{m=1}^M \mathcal{M}_m^y\right\}.$$

Чтобы оценить нижнюю оценку распределенной минимаксной проблемы (1), нам нужно предоставить “плохую функцию” и “плохую последовательность графов”, такие, что никакой алгоритм не может решить задачу, используя меньше данного количества шагов. Используя меняющуюся сеть из [11] и целевую функцию из [20], мы можем доказать следующую теорему.

Теорема 5. Для любых $L > \mu > 0$ и любого $\chi \geq 1$, существует децентрализованная распределенная седловая задача на $\mathcal{X} \times \mathcal{Y} = \mathbb{R}^n \times \mathbb{R}^n$ (где n достаточно велико) с $x^*, y^* \neq 0$, последовательностью графов $\{\mathcal{G}^k = (\mathcal{V}, \mathcal{E}^k)\}_{k=0}^\infty$, где последовательные графы отличаются не более чем двумя ребрами, и последовательность соответствующих матриц коммуникации $\{W^k\}_{k=0}^\infty$ с числом обусловленности χ , такие, что для любого выхода $\hat{x}, \hat{y}$ после K коммуникационных раундов любой процедуры $\mathbf{BBP}(T, K)$ имеется следующая оценка:

$$\|\hat{x} - x^*\|^2 + \|\hat{y} - y^*\|^2 \geq \Omega\left(\exp\left(-\frac{32\mu}{L-\mu} \cdot \frac{K}{\chi}\right) \|y_0 - y^*\|^2\right).$$

Доказательство. Представим граф $T_{a,b}$ из [11]. Этот граф содержит два разбиения: левое и правое, эти разбиения состоят из a и b вершин соответственно. Каждая вершина в разбиении соединена со своим корнем. Эти корни соединены с другой вершиной, называемой центральным корнем.

Рассмотрим сеть с $|\mathcal{V}| = 2d + 3$ узлов ($d \geq 2$). Мы выбираем узел в качестве центрального корня и два других узла в качестве левого и правого корней. Центральный корень может со временем меняться, но корни разбиения фиксированы.

Также для каждого разбиения выбираем $\left\lceil \frac{d}{2} \right\rceil$ фиксированных вершин и обозначаем их $\mathcal{V}_1$ (левое разбиение) и $\mathcal{V}_2$ (правое разбиение). На каждом

коммуникационном раунде k , граф $\mathcal{G}^k$ имеет вид $T_{a,b}$, где $a + b = 2d$ и $a, b \geq \left\lceil \frac{d}{2} \right\rceil$.

Коммуникационная сеть меняется поочередно в две фазы. Первая фаза начинается с графа $T_{2d-\left\lceil \frac{d}{2} \right\rceil, \left\lceil \frac{d}{2} \right\rceil}$ и заканчивается графом $T_{\left\lceil \frac{d}{2} \right\rceil, 2d-\left\lceil \frac{d}{2} \right\rceil}$.

На каждой итерации центральный корень перемещается в правое разбиение, центральным корнем становится одна вершина из левого разбиения, но не в $\mathcal{V}_1$. Вторая фаза проходит по тому же принципу, но справа налево.

Мы модифицируем целевую функцию из раздела В.1 в [3] на основе нашего типа графов:

$$f_m(x, y) = \begin{cases} f_1(x, y) = \frac{M}{2|\mathcal{V}_2|} \cdot \frac{L}{2} x^T A_1 y + \frac{\mu}{2} \|x\|^2 - \frac{\mu}{2} \|y\|^2 + \frac{M}{2|\mathcal{V}_2|} \cdot \frac{L^2}{2\mu} e_1^T y, & m \in \mathcal{V}_2 \\ f_2(x, y) = \frac{M}{2|\mathcal{V}_1|} \cdot \frac{L}{2} x^T A_2 y + \frac{\mu}{2} \|x\|^2 - \frac{\mu}{2} \|y\|^2, & m \in \mathcal{V}_1 \\ f_3(x, y) = \frac{\mu}{2} \|x\|^2 - \frac{\mu}{2} \|y\|^2, & \text{иначе} \end{cases} \quad (6)$$

где $e_1 = (1, 0, \dots, 0)$ и

$$A_1 = \begin{pmatrix} 1 & 0 & & & & & \\ & 1 & -2 & & & & \\ & & 1 & 0 & & & \\ & & & 1 & -2 & & \\ & & & & \dots & \dots & \\ & & & & & 1 & -2 \\ & & & & & & 1 & 0 \\ & & & & & & & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & -2 & & & & & \\ & 1 & 0 & & & & \\ & & 1 & -2 & & & \\ & & & 1 & 0 & & \\ & & & & \dots & \dots & \\ & & & & & 1 & 0 \\ & & & & & & 1 & -2 \\ & & & & & & & 1 \end{pmatrix}.$$

Рассмотрим задачу с глобальной целевой функцией:

$$\begin{aligned} f(x, y) &:= \frac{1}{M} \sum_{m=1}^M f_m(x, y) = \frac{1}{M} (|\mathcal{V}_2| \cdot f_1(x, y) + \\ &+ |\mathcal{V}_1| \cdot f_2(x, y) + (M - |\mathcal{V}_1| - |\mathcal{V}_2|) \cdot f_3(x, y)) = \\ &= \frac{L}{2} x^T A y + \frac{\mu}{2} \|x\|^2 - \frac{\mu}{2} \|y\|^2 + \frac{L^2}{4\mu} e_1^T y, \end{aligned} \quad (7)$$

$$\text{где } A = \frac{1}{2} (A_1 + A_2).$$

Мы оцениваем количество коммуникационных раундов, необходимых для получения нового ненулевого элемента в локальной памяти, используя следующую лемму.

Лемма 2. Пусть задача (1) решается некоторым алгоритмом **ВВР**(T, K). Тогда после K коммуникаций не более $\left\lceil \frac{K}{d} \right\rceil$ первых координат выхода алгоритма могут быть ненулевыми, в то время как оставшиеся $n - \left\lceil \frac{K}{d} \right\rceil$ координат строго равны нулю.

Доказательство. Заметим, что новая ненулевая координата появляется, только когда информация доходит от вершин $\mathcal{V}_1$ до вершин $\mathcal{V}_2$ и происходит градиентный шаг на одной из вершин $\mathcal{V}_2$ (или в обратном направлении). После каждого такого распространения информации добавляется не более одной координаты.

Между двумя последовательными коммуникационными раундами только узлы из $\mathcal{V}_1$ и $\mathcal{V}_2$ могут добавлять в свою локальную память новую ненулевую координату, но в этом интервале два узла из $\mathcal{V}_1$ и $\mathcal{V}_2$ не могут выполнять одновременно. Более того, между двумя раундами связи можно добавить не более одной новой ненулевой координаты (подробнее см. [3]).

Следовательно, нам постоянно приходится передавать информацию из множества узлов $\mathcal{V}_1$ в $\mathcal{V}_2$ и обратно, чтобы получить новые ненулевые координаты. Для каждой новой ненулевой координаты нам нужно хотя бы одно локальное вычисление. Таким образом, требуется $T > K$. Из [11] мы знаем, что каждая передача требуется как минимум d коммуникационных раундов. Таким

образом, после K коммуникационных раундов не более первых $\left\lfloor \frac{K}{d} \right\rfloor$ координат выхода алгоритма могут быть ненулевыми.

Следующие вспомогательные леммы позволяют получить нижние оценки сложности.

Лемма 3 (Лемма 4 из [3].) Пусть $\alpha = \frac{4\mu^2}{L^2}$ и $q = \frac{1}{2}(2 + \alpha - \sqrt{\alpha^2 + 4\alpha}) \in (0; 1)$ — меньший корень уравнения $q^2 - (2 + \alpha)q + 1 = 0$. Также введем $\bar{y}^*$

$$\bar{y}_i^* = \frac{q^i}{1 - q}.$$

Тогда ошибка между аппроксимацией и реальным решением задачи (7) может быть ограничена как

$$\|\bar{y}^* - y^*\| \leq \frac{q^{n+1}}{\alpha(1 - q)}.$$

Лемма 4 (Лемма 5 из [3]). Рассмотрим распределенную седловую задачу вида (6), (7) с последовательностью графов $\{\mathcal{G}_k = (\mathcal{V}, \mathcal{E}_k)\}_{k=0}^\infty$ и соответствующей последовательностью матриц коммуникации $\{W(\mathcal{G}_k)\}_{k=0}^\infty$. Для всех пар $T, K (T > K)$ можно показать, что размер задачи $n \geq \max \left\{ 2 \log_q \left(\frac{\alpha}{4\sqrt{2}} \right), 2K \right\}$, где $\alpha = \frac{4\mu^2}{L^2}$ и $q = \frac{1}{2}(2 + \alpha - \sqrt{\alpha^2 + 4\alpha}) \in (0; 1)$. Тогда для всякой пары $\hat{x}, \hat{y}$, выданной некоторым алгоритмом **ВВР**(T, K) после K коммуникационных раундов и T локальных вычислений, выполняется

$$\|\hat{x} - x^*\|^2 + \|\hat{y} - y^*\|^2 \geq q^{\frac{2K}{d}} \frac{\|y_0 - y^*\|^2}{16}.$$

Используя результаты доказательства из утверждения 3.6 из [20], получаем

$$\ln(q) \geq \frac{q-1}{q} = \frac{2}{1 - \sqrt{\frac{L^2}{\mu^2} + 1}} \geq \frac{2}{1 - \frac{L}{\mu}} = \frac{-2\mu}{L - \mu}.$$

Каждому графу в нашей последовательности сопоставим взвешенный лапласиан из леммы 8 в [11], так что $\chi \leq 8d$. Имеем

$$\ln(q) \cdot \frac{2K}{d} \geq \frac{-4\mu}{L - \mu} \cdot \frac{K}{d} \geq \frac{-32\mu}{L - \mu} \cdot \frac{K}{\chi}.$$

Далее, получаем

$$q^{\frac{2K}{d}} = \exp \left(\ln(q) \cdot \frac{2K}{d} \right) \geq \exp \left(\frac{-32\mu}{L - \mu} \cdot \frac{K}{\chi} \right).$$

Таким образом,

$$\begin{aligned} \|\hat{x} - x^*\|^2 + \|\hat{y} - y^*\|^2 &= \\ &= \Omega \left(\exp \left(-\frac{32\mu}{L - \mu} \cdot \frac{K}{\chi} \right) \|y_0 - y^*\|^2 \right). \end{aligned}$$

Следствие 5.1. Для достижения точности ε в условиях теоремы 5, количество коммуникационных раундов ограничено снизу величиной

$$\Omega \left(\chi \frac{L}{\mu} \cdot \log \left(\frac{\|y^*\|^2}{\varepsilon} \right) \right),$$

а количество локальных вызовов оракула ограничено снизу величиной

$$\Omega \left(\frac{L}{\mu} \cdot \log \left(\frac{\|y^*\|^2}{\varepsilon} \right) \right).$$

5. ЗАКЛЮЧЕНИЕ

В статье изучались децентрализованные минимаксные задачи на медленно меняющихся сетях. Мы показали, что если изменения графа враждебны и меняется константное число ребер на каждой итерации, то нижние оценки сложности совпадают с произвольными переменными сетями. Более того, мы показали, что для определенного класса переменных графов — сетей со связным остовом и сетей с марковскими изменениями — возможно ускорение коммуникационной процедуры. Были предложены соответствующие алгоритмы для седловых задач для данных двух классов сетей.

ИСТОЧНИК ФИНАНСИРОВАНИЯ

Исследование выполнено за счет гранта Российского научного фонда (проект № 23-11-00229), <https://rscf.ru/project/23-11-00229/>.

СПИСОК ЛИТЕРАТУРЫ

1. Bazerque J.A., Giannakis G.B. Distributed spectrum sensing for cognitive radio networks by exploiting sparsity. *IEEE Transactions on Signal Processing*. 2009. V. 58. № 3. P. 1847–1862.
2. Beznosikov A., Rogozin A., Kovalev D., Gasnikov A. Near-optimal decentralized algorithms for saddle point problems over time-varying networks. In *International Conference on Optimization and Applications*, pages 246–257. Springer, 2021.
3. Beznosikov A., Samokhin V., Gasnikov A. Distributed saddle-point problems: Lower bounds, near-optimal and robust algorithms. *arXiv preprint arXiv:2010.13112*, 2020.
4. Beznosikov A., Samokhin V., Gasnikov A. Distributed saddle-point problems: Lower bounds, optimal algorithms and federated gans. *arXiv preprint arXiv:2010.13112*, 2021.

5. *Forero P.A., Cano A., Giannakis G.B.* Consensus-based distributed support vector machines. *Journal of Machine Learning Research*. 2010. V. 11. № 5.
6. *Gan L., Topcu U., Low S.H.* Optimal decentralized protocol for electric vehicle charging. *IEEE Transactions on Power Systems*. 2012. V. 28. № 2. P. 940–951.
7. *Kovalev D., Beznosikov A., Sadiev A., Persianov M., Richt'arik P., Gasnikov A.* Optimal algorithms for decentralized stochastic variational inequalities. *Advances in Neural Information Processing Systems*. 2022. V. 35. P. 31073–31088.
8. *Kovalev D., Gasanov E., Gasnikov A., Richt'arik P.* Lower bounds and optimal algorithms for smooth and strongly convex decentralized optimization over time-varying networks. *Advances in Neural Information Processing Systems*. 2021. V. 34.
9. *Kovalev D., Salim A., Richt'arik P.* Optimal and practical algorithms for smooth and strongly convex decentralized optimization. *Advances in Neural Information Processing Systems*. 2020. V. 33.
10. *Kovalev D., Shulgin E., Richt'arik P., Rogozin A., Gasnikov A.* Adom: Accelerated decentralized optimization method for time-varying networks. *arXiv preprint arXiv:2102.09234*, 2021.
11. *Metelev D., Beznosikov A., Rogozin A., Gasnikov A., Proskurnikov A.* Decentralized optimization over slowly time-varying graphs: Algorithms and lower bounds. *arXiv preprint arXiv:2307.12562*, 2023.
12. *Metelev D., Rogozin A., Kovalev D., Gasnikov A.* Is consensus acceleration possible in decentralized optimization over slowly time-varying networks? In *International Conference on Machine Learning*. PMLR, 2023. P. 24532–24554.
13. *Nedic A.* Distributed gradient methods for convex machine learning problems in networks: Distributed optimization. *IEEE Signal Processing Magazine*. 2020. V. 37. № 3. P. 92–101.
14. *Nedić A., Olshevsky A., Uribe C.A.* Fast convergence rates for distributed non-bayesian learning. *IEEE Transactions on Automatic Control*. 2017. V. 62. № 11. P. 5538–5553.
15. *Nedic A., Ozdaglar A.* Cooperative distributed multi-agent optimization. *Convex optimization in signal processing and communications*. 2010. V. 340.
16. *Rabbat M., Nowak R.* Distributed optimization in sensor networks. In *Proceedings of the 3rd international symposium on Information processing in sensor networks*. 2004. P. 20–27.
17. *Ram S.S., Veeravalli V.V., Nedic A.* Distributed non-autonomous power control through distributed convex optimization. In *IEEE INFOCOM 2009, IEEE, 2009*. P. 3001–3005.
18. *Ren W., Beard R.W.* Distributed consensus in multi-vehicle cooperative control, V. 27. Springer, 2008.
19. *Scaman K., Bach F., Bubeck S., Lee Y.T., Massoulié L.* Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017*. P. 3027–3036.
20. *Zhang J., Hong M., Zhang S.* On lower iteration complexity bounds for the saddle point problems. *arXiv preprint arXiv:1912.07481*, 2019.

MIN-MAX OPTIMIZATION OVER SLOWLY TIME-VARYING GRAPHS

N. T. Nguyen^a, A. Rogozin^a, D. Metelev^a, and A. Gasnikov^{a,b,c,d}

^a*Moscow Institute of Physics and Technology, Moscow, Russia*

^b*Institute for Information Transportation Problems, Moscow, Russia*

^c*Caucasus Mathematic Center of Adygh State University, Moscow, Russia*

^d*Ivannikov Institute for System Programming of the Russian Academy of Sciences, Research Center for Trusted Artificial Intelligence, Moscow, Russia*

Presented by Academician of the RAS A.A. Shanin

Distributed optimization is an important direction of research in modern optimization theory. Its applications include large scale machine learning, distributed signal processing and many others. The paper studies decentralized min-max optimization for saddle point problems. Saddle point problems arise in training adversarial networks and in robust machine learning. The focus of the work is optimization over (slowly) time-varying networks. The topology of the network changes from time to time, and the velocity of changes is limited. We show that, analogically to decentralized optimization, it is sufficient to change only two edges per iteration in order to slow down convergence to the arbitrary time-varying case. At the same time, we investigate several classes of time-varying graphs for which the communication complexity can be reduced.

Keywords: saddle point problem, decentralized optimization, time-varying graph, extragradient method