

## МЕТОД ПООЧЕРЕДНЫХ ПРОЕКЦИЙ ДЛЯ ПЕРЕСЕЧЕНИЯ ВЫПУКЛЫХ МНОЖЕСТВ, МНОГОАГЕНТНЫЕ АЛГОРИТМЫ КОНСЕНСУСА И УСРЕДНЯЮЩИЕ НЕРАВЕНСТВА<sup>1)</sup>

© 2024 г. А. В. Проскурников<sup>1,\*</sup>, И. С. Забарянская<sup>2</sup>

<sup>1</sup> 10129 Turin, Italy, Corso Duca degli Abruzzi, 24, Politecnico di Torino (Политехнический университет Турина), Италия

<sup>2</sup> 199178 Санкт-Петербург, 14-я линия В.О., 29, СПбГУ, Россия

\* e-mail: avp1982@gmail.com

\*\* e-mail: akshiira@yandex.ru

Поступила в редакцию 03.11.2023 г.

Переработанный вариант 11.11.2023 г.

Принята к публикации 20.11.2023 г.

История метода поочередных проекций для нахождения общей точки нескольких выпуклых множеств в евклидовом пространстве восходит к известному алгоритму Качмаржа для решения систем линейных уравнений, который появился в 1930-х годах и впоследствии нашел широкое применения в обработке изображений и компьютерной томографии. Важную роль в исследовании данного метода сыграли работы И.И. Ерёмкина, Л.М. Брэгмана и Б.Т. Поляка, появившиеся практически одновременно и содержащие весьма общие результаты о сходимости последовательных проекций к точке на пересечении множеств, если это пересечение непусто. В настоящей статье мы рассматриваем модификацию задачи о пересечении выпуклых множеств, относящуюся к теории многоагентных систем и называемую задачей о консенсусе с ограничениями. Каждое выпуклое множество в этой задаче связано со своим агентом и, вообще говоря, недоступно другим агентам. При этом группа агентов заинтересована в нахождении общей точки этих множеств: точки, удовлетворяющей ограничениям. Распределенные аналоги метода поочередных проекций, предложенные для решения этой задачи, приводят к достаточно сложной нелинейной системе уравнений, сходимость которой, обычно, доказывается с помощью специальных функций Ляпунова. В работе дается краткий обзор данных методов и показывается их связь с теоремой о консенсусе в системе усредняющих неравенств, недавно установленной в работах первого автора и развивающей результаты о сходимости обычного метода последовательных усреднений в задаче о консенсусе. Библ. 48. Фиг. 6.

**Ключевые слова:** метод поочередных проекций, выпуклое программирование, фейеровские отображения, распределенные алгоритмы, консенсус, многоагентные системы.

DOI: 10.31857/S0044466924040078, EDN: ZJPOEZ

### 1. ВВЕДЕНИЕ

Ряд задач численного анализа и оптимизации сводятся к нахождению точки в пересечении семейства замкнутых выпуклых множеств в конечномерном либо гильбертовом векторном пространстве (или установлению того, что данное пересечение пусто). Одним из специальных случаев является проверка совместности системы линейных уравнений с нахождением одного из возможных решений, а также проверка совместности ограничений в линейном программировании; взрывной рост интереса к быстрым алгоритмам для решения таких задач был связан, в частности, с изобретением компьютерной томографии и связанных с ней алгоритмов быстрого восстановления изображений (см. [1]). К нахождению точки в пересечении полупространств аффинного пространства сводится вопрос о разделении двух множеств точек в пространстве, важный для задач обучения (см. [2], [3]) и математической диагностики (см. [4]–[6]). К задачам более общего вида приводит выпуклая оптимизация: на начальной стадии прямого (primal) метода оптимизации требуется найти некоторую (произвольную) точку, удовлетворяющую системе выпуклых ограничений. Наконец, в теории управления пересечения выпуклых множеств возникают в связи с оцениванием параметров и траекторий систем при неизвестных помехах (см. [7]); примером являются системы рекуррентных целевых неравенств (см. [8], [9]).

<sup>1)</sup> Работа выполнена при финансовой поддержке РФФИ (код проекта 23-11-00229).

Большинство имеющихся в литературе алгоритмов для вычисления общей точки выпуклых множеств являются модификациями и обобщениями метода поочередных (alternating) проекций. В англоязычной литературе для данного метода также используется аббревиатура POCS (projection onto convex sets, т.е. проекции на выпуклые множества). История данного метода восходит к работам Качмаржа [10] и Чиммино [11] по релаксационным методам решения систем линейных уравнений, а впервые (в специальном случае двух линейных подпространств) он, по-видимому, упомянут в лекциях фон Неймана по геометрии гильбертова пространства, изданных в виде монографии [12]. Разработанная впоследствии теория метода поочередных проекций, во многом базируется на работах трех крупнейших специалистов в области оптимизации и численных методов: И. И. Ерёмину, Б. Т. Поляка и Л. М. Брэгмана, чьи пионерские работы [13]–[15] появились практически одновременно. Исследование последовательных итеративных проекций привело, с одной стороны, к развитию теории фейеровских процессов (см. [16]–[19]), а с другой стороны, послужило мотивацией для изучения  $D$ -расстояния (дивергенции) Брэгмана (см. [20]), в настоящее время играющего исключительно важную роль в задачах машинного обучения, оптимизации и вычислительной геометрии.

В настоящее время существуют достаточно подробные обзоры по задаче о пересечении выпуклых множеств и ее обобщениям (в частности, сходимости фейеровских приближений), среди которых особо следует отметить монографии [21], [22] и более раннюю работу [23], а также более специальный обзор по теории оценивания [7]. С некоторыми обобщениями задачи на невыпуклый случай (проектирование на многообразиях) можно ознакомиться в работе [24], содержащей также достаточно полный обзор литературы по данной теме. Эти обзоры, однако, не покрывают *многоагентные* версии задачи о пересечении выпуклых множеств, исследуемые в последние годы под различными названиями, — например задача о консенсусе с ограничениями (см. [25]) или “оптимальном консенсусе” (см. [26]). В этих задачах каждое выпуклое множество является собственностью одного из агентов и недоступно другим агентам (это может быть вызвано как требованиями приватности, так и сложностью описания самих множеств, информацию о которых по каким-то причинам может быть затруднительно передавать через сеть). Целью группы агентов, как и в исходной задаче, является нахождение точки в пересечении всех множеств, при этом обмениваться информацией каждый из агентов может лишь с некоторым набором “соседей”. Отношения соседства описываются коммуникационным графом, который может меняться и быть неполностью известен. Распределенные децентрализованные алгоритмы для решения многоагентной задачи о пересечении множеств изучались, главным образом, в журналах по теории управления и малоизвестны специалистам из других областей, а имеющиеся обзоры по теме в основном ограничиваются специальными случаями, такими как алгоритмы решения линейных уравнений (см. [27]).

Настоящая статья восполняет указанный пробел и дает краткий обзор распределенных алгоритмов, которые были предложены для решения многоагентной задачи о пересечении выпуклых множеств. Мы покажем, что

1) в данных алгоритмах (как и в централизованном случае) операторы проектирования могут быть заменены произвольными фейеровскими (парасжимающими) отображениями,

2) сходимость к консенсусу в предложенных алгоритмах может быть доказана единым способом в рамках теории *усредняющих неравенств* на графах. Данная теория была разработана в диссертации первого автора (см. [28]), а также работах [29], [30]. Для удобства читателя, мы приводим также доказательства основных результатов, прежде не публиковавшихся на русском языке.

Оставшаяся часть статьи организована следующим образом. В разд. 2 приводятся постановка задачи о пересечении выпуклых множеств, алгоритмы поочередных проекций и их обобщения (фейеровские процессы), а также краткий обзор основных результатов о сходимости этих алгоритмов, в частности, фундаментальные результаты Ерёмину, Брэгмана и Гурина–Поляка–Райка. Основной материал статьи, относящийся к многоагентной постановке задачи, приводится в разд. 3, где мы также приводим некоторые факты о сходимости классических усредняющих алгоритмов консенсуса на переменных графах, а также ассоциированных с ними систем усредняющих неравенств. Раздел 4 завершает статью. Доказательство основных результатов о сходимости многоагентных алгоритмов приводится в приложении.

## 2. ЗАДАЧА О ПЕРЕСЕЧЕНИИ МНОЖЕСТВ И ЕЕ ОБОБЩЕНИЯ

В настоящей статье мы будем рассматривать задачу о пересечении выпуклых множеств в конечномерном евклидовом пространстве, однако ряд упоминаемых ниже результатов может быть распространен также на гильбертовы пространства (с некоторыми уточнениями, связанными с типом сходимости последовательных приближений). Проекционные методы изучались также на многообразиях, допускающих эффективное вычисление проекционного оператора (см. [24]); данное обобщение выходит за рамки настоящей статьи.

2.1. Постановка задачи. Оператор проектирования

Классическая задача о пересечении выпуклых множеств (или о совместности выпуклых ограничений), как правило формулируется следующим образом.

**Задача А.** Пусть есть набор выпуклых множеств  $\{\Xi_i\}_{i \in \mathcal{V}}$  в  $\mathbb{R}^d$ , где  $\mathcal{V}$  – некоторое конечное множество индексов, пересечение которых непусто:

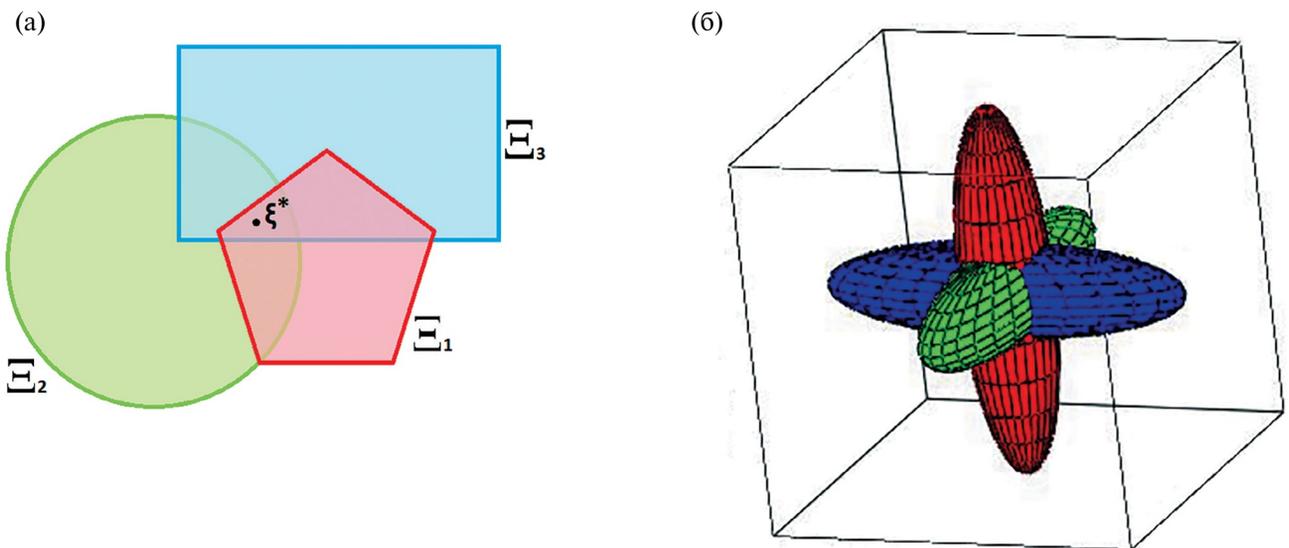
$$\Xi_* = \bigcap_{i \in \mathcal{V}} \Xi_i \neq \emptyset. \tag{1}$$

Требуется найти хотя бы одну точку  $\xi_* \in \Xi_*$ .

Более сложная задача — дополнительно распознать ситуацию, когда  $\Xi_* = \emptyset$ .

**Задача В.** Для заданного набора выпуклых множеств  $\{\Xi_i\}_{i \in \mathcal{V}}$  требуется проверить выполнение условия (1) и, если оно выполнено, найти хотя бы одну точку  $\xi_* \in \Xi_*$ .

Отметим, что в случае одномерного пространства  $d = 1$  задачи А и В решаются просто, поскольку единственно возможное выпуклое множество в  $\mathbb{R}$  – промежуток (открытый, полуоткрытый или замкнутый). Пересечение конечного семейства промежутков тривиально находится путем сортировки их левых и правых концов (фиг. 1). В этом случае мы можем определить множество  $\Xi_*$ . В случае  $d = 2$  множество  $\Xi_*$  обычно легко визуализировать с помощью компьютера, однако описать его структуру полностью или хотя бы найти одну из его точек  $\xi_*$  аналитически — достаточно трудная задача. Уже в случае трехмерного пространства  $d = 3$  даже визуализация множества  $\Xi_*$  весьма затруднительна.



Фиг. 1. Примеры пересечений выпуклых множеств: (а) — на плоскости, (б) — в пространстве.

Алгоритмы решения задач А и В, исходно предложенные в литературе, основывались на последовательных проекциях. Напомним фундаментальный результат о проектировании на выпуклое множество и определение проекционного оператора. Для любого замкнутого выпуклого множества  $\Omega \subset \mathbb{R}^d$  и  $x \in \mathbb{R}^d$  оператор проектирования  $P_\Omega : \mathbb{R}^d \rightarrow \Omega$  сопоставляет точке ближайший элемент из  $\Omega$ , т.е.  $|x - P_\Omega(x)| = \min_{y \in \Omega} |x - y|$ . Минимум расстояния всегда достигается, а ближайшая точка единственна. Это утверждение справедливо в произвольном гильбертовом пространстве (см. [31], теорема 1.4.1).

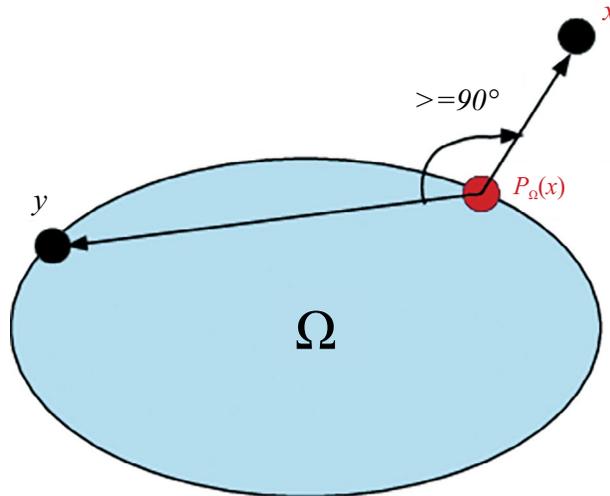
Можно показать, что  $\angle(y - P_\Omega(x), x - P_\Omega(x)) \geq \pi / 2 \forall y \in \Omega$  (фиг. 2)

и

$$|x - y|^2 \geq |x - P_\Omega(x)|^2 + |y - P_\Omega(x)|^2 \quad \forall y \in \Omega. \tag{2}$$

<sup>2</sup> Символ  $|\cdot|$  всюду ниже обозначает обычную евклидову норму в  $\mathbb{R}^d : |x|^2 = \sum_i x_i^2 = x^T x$ .

<sup>3</sup> Символ  $\angle$  обозначает угол между векторами (в интервале  $[0, \pi]$ ).



Фиг. 2. Проекция точки  $x$  на замкнутое выпуклое множество  $\Omega$ .

В силу свойства (2), если  $y \in \Omega$  (иными словами,  $y$  – неподвижная точка  $P_\Omega$ ), то

$$|P_\Omega(x) - y| \leq |x - y|,$$

причем неравенство строгое тогда и только тогда, когда  $|x - P_\Omega(x)| > 0$ , т.е.  $x \notin \Omega$ . Непрерывное отображение с таким свойством называется *фейеровским* или (в англоязычной литературе) парасжимающим; ниже будет дано формальное определение.

2.2. Первые работы: системы линейных уравнений и неравенств

Простейшим (и первым в историческом плане) примером задачи о пересечении выпуклых множеств является решение системы линейных уравнений<sup>4</sup>

$$a_i^T \xi = b_i \quad \forall i \in \mathcal{V},$$

где  $\xi \in \mathbb{R}^d$  – неизвестный вектор, а  $a_i \in \mathbb{R}^d, b_i \in \mathbb{R}$ , определяют коэффициенты  $i$ -го уравнения. В данном случае  $\Xi_i = \{\xi : a_i^T \xi = b_i\}$  – аффинная гиперплоскость размерности  $d - 1$ , а поиск решения системы уравнений (более точно, одного из решений) может быть рассмотрен как пересечение этих выпуклых множеств. Подчеркнем, что система уравнений может быть как переопределенной ( $d < |\mathcal{V}|$ ), так и недоопределенной ( $d > |\mathcal{V}|$ ), уравнения могут быть линейно зависимы, а количество уравнений и переменных может быть очень велико. Такие системы уравнений, например, возникают в компьютерной томографии при восстановлении изображений по проекциям (см. [1]), леонтьевских моделях отраслевого баланса (см. [32]), вычислении вектора PageRank (см. [33], [34]) и аналогичных характеристик в задачах ранжирования и поиска, нахождении собственных векторов. Два простых проекционных алгоритма для решения линейных уравнений были практически одновременно предложены в 1930-е годы в работах Качмаржа и Чиммино.

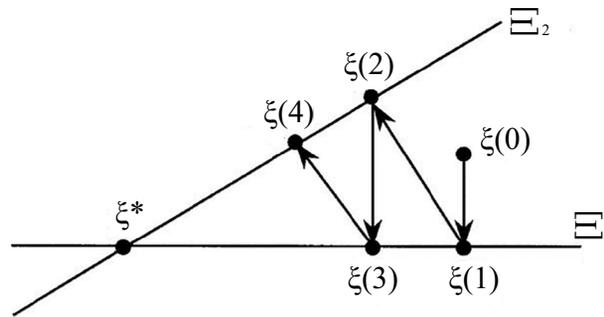
**2.2.1. Алгоритм Качмаржа.** Алгоритм Качмаржа<sup>5</sup> (см. [10], [35]) заключается в последовательном проектировании точки на гиперплоскости в циклическом порядке (для случая двух уравнений  $|\mathcal{V}| = 2$  и двух переменных  $d = 2$  процедура проиллюстрирована на фиг. 3).

Нетрудно видеть, что оператор проектирования на множество  $\Xi = \{\xi : a^T \xi = b\}$  (где  $|a| \neq 0$ ) является аффинным отображением:

$$P_\Xi : x \mapsto x + \frac{b - a^T x}{|a|^2} a. \tag{3}$$

<sup>4</sup> Здесь и далее элементы  $\mathbb{R}^d$  понимаются как вектор-столбцы, соответственно  $a^T \xi$  есть скалярное произведение двух векторов  $a$  и  $\xi$ .

<sup>5</sup> Также этот алгоритм называется методом построчного действия (row action method), поскольку на каждом шаге используется одна строка матрицы ограничений.



Фиг. 3. Итеративный метод Качмаржа.

Предположим, что уравнения некоторым образом пронумерованы от 1 до  $n$ , так что  $\mathcal{V} = \{1, \dots, n\}$ , и выбрано некоторое произвольное приближение к решению  $\xi(0)$ . На итерации с номером  $t = 0, 1, \dots$  приближение  $\xi(t)$  заменяется на

$$\xi(t + 1) = P_{\Xi_i}(\xi(t)) = \xi(t) + \frac{b_i - a_i^T \xi(t)}{|a_i|^2} a_i, \text{ где } t + 1 \equiv i \pmod n.$$

В случае двух множеств данный алгоритм впервые появляется в лекциях фон Неймана (1933), изданных впоследствии в виде монографии [12], при этом рассматривались линейные замкнутые подпространства гильбертова пространства  $\Xi_1, \Xi_2$ . Фон Нейманом было доказано (см. [12], теорема 10.7), что последовательность  $\xi(t)$  сходится к проекции начальной точки  $\xi(0)$  на пространство  $\Xi_* = \Xi_1 \cap \Xi_2$  по норме. Данная сходимость экспоненциальна, а ее скорость определяется квадратом косинуса угла между подпространствами  $\Xi_1$  и  $\Xi_2$  (см. [36]). Указанный результат — экспоненциальная сходимость  $\xi(t)$  к проекции  $P_{\Xi_*}(\xi(0))$  — остается справедливым и для  $n > 2$  подпространств, однако в этом случае известны лишь верхние оценки для скорости сходимости (см. [22], [23], [36]), которая, вообще говоря, не определяется только углами между всевозможными парами подпространств  $(\Xi_i, \Xi_j)$ . Более подробно с историей алгоритма Качмаржа и его обобщениями, полученными в последние годы, можно ознакомиться в обзоре [35].

**2.2.2. Алгоритм Чиммино.** Итеративный алгоритм Чиммино, предложенный в [11], в определенном смысле является прообразом распределенных многоагентных алгоритмов, рассматриваемых в разд. 3. Его отличием от алгоритма Качмаржа является то, что на итерации  $\xi(t)$  проектируется одновременно на все гиперплоскости. Затем вычисляется центр масс полученных проекций  $\xi_b(t)$ , который может быть выбран в качестве следующего приближения  $\xi(t + 1)$ . Более общо, следующее приближение может быть получено как

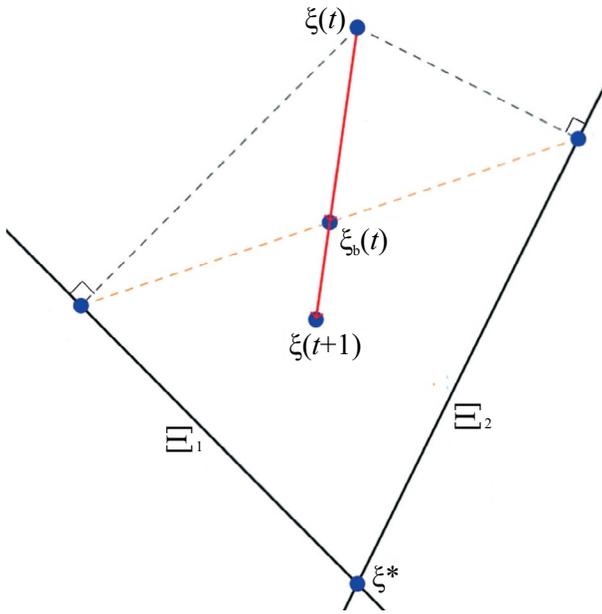
$$\xi(t + 1) = \xi(t) + \alpha(\xi_b(t) - \xi(t)),$$

где  $\alpha \in (0, 2)$  — некоторый фиксированный шаг (фиг. 4).

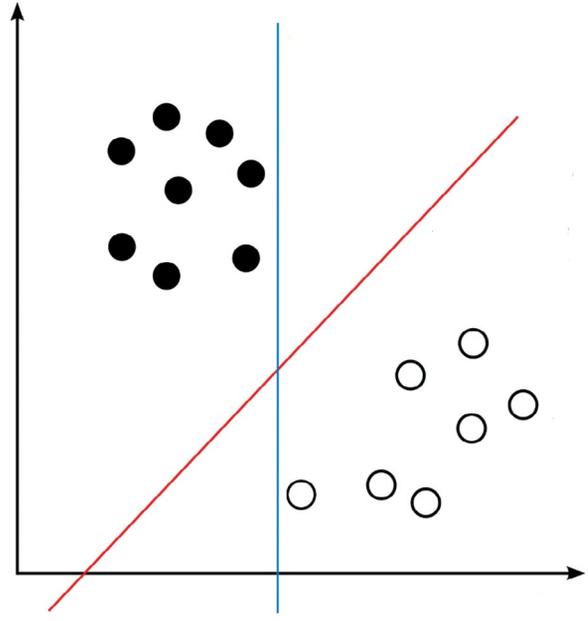
Большим преимуществом алгоритма Чиммино является возможность параллелизации операций проектирования, которые могут выполняться несколькими процессорами. Вместе с тем, число итераций данного алгоритма, требуемое для нахождения решения с заданной точностью, может быть достаточно велико. Существуют ускоренные версии алгоритма, в частности, блочный алгоритм Чиммино (см. [37]) для разреженных систем уравнений; их рассмотрение выходит за рамки настоящей статьи.

**2.2.3. Алгоритм Агмона–Мотцкина для пересечения полупространств.** Важным этапом в исследовании задачи о пересечении выпуклых множеств стали вышедшие одновременно работы Агмона [38] и Мотцкина и Шёнберга [39]; идея изучаемого в них алгоритма, как упомянуто в [38], принадлежит Мотцкину. В отличие от работ Качмаржа и Чиммино, рассматривается задача о решении системы линейных неравенств, т.е. нахождении точки в пересечении полупространств

$$\Xi_i = \left\{ \xi \in \mathbb{R}^d : a_i^T \xi \leq b_i \right\}.$$



Фиг. 4. Метод Чиммино (иллюстрация для двумерного случая).



Фиг. 5. Аффинные гиперплоскости, разделяющие два множества точек в пространстве.

Проектор на полупространство  $\Xi = \{ \xi : a^T \xi \leq b \}$  нетрудно вычислить. Отметим, что он уже не является аффинным отображением:

$$P_{\Xi} : x \mapsto x' = \begin{cases} x + \frac{b - a^T x}{|a|^2} a, & a^T x > b, \\ x, & a^T x \leq b. \end{cases} \quad (4)$$

Равенство  $a_i^T \xi = b_i$ , очевидно, может быть записано как пара неравенств, таким образом, метод из [38], [39] позволяет находить решения произвольной системы линейных уравнений и неравенств (в предположении, что такое решение существует). Отметим, что к решению неравенств сводятся многие “линейные” задачи обучения (например, алгоритмы настройки весов перцептрона), классификации и математической диагностики (см. [2], [4]–[6]). В этих задачах требуется найти гиперплоскость, разделяющую две группы точек в многомерном пространстве (фиг. 5). Задавая гиперплоскость линейным уравнением, условие разделения является системой неравенств на коэффициенты этого уравнения.

Метод из работ [38], [39] по реализации напоминает алгоритм Качмаржа (где проектор вместо (3) приобретает вид (4)), однако имеется одно существенное отличие: на каждой итерации  $t = 0, 1, \dots$  проектирование осуществляется на самое *дальнее* множество. Формально, находится индекс<sup>6</sup>

$$m(t) = \arg \max_{i \in \mathcal{V}} d(\xi(t), \Xi_i), \quad (5)$$

где  $d(\xi, \Xi) = \min_{x \in \Xi} \{ |\xi - x| \}$ . Проектирование осуществляется на полупространство с индексом  $m(t)$ , таким образом, следующее приближение имеет вид

$$\xi(t+1) = P_{\Xi_{m(t)}}(\xi(t)).$$

В работах [38], [39] была доказана (в предположении непустоты пересечения множеств) экспоненциальная сходимость этого алгоритма, а также его “релаксированного” варианта:

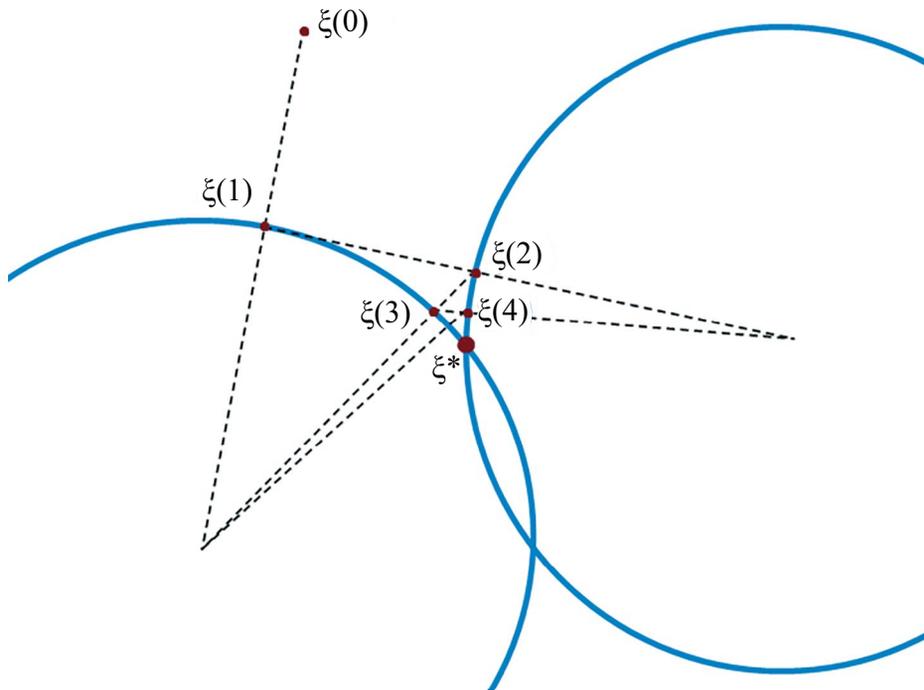
$$\xi(t+1) = (1 - \alpha)\xi(t) + \alpha P_{\Xi_{m(t)}}(\xi(t)), \quad \alpha \in (0, 2). \quad (6)$$

<sup>6</sup> В случае необщего положения, когда точка равноудалена от нескольких множеств и  $\arg \max$  определен неоднозначно, можно выбрать произвольное значение; обычно, выбирается наименьший индекс (см. [13]).

Отметим, что скорость сходимости может быть определена явно (см. [38]) и зависит от матрицы  $A$ , хотя ее вычисление нетривиально. По сложности итерация алгоритма Агмона–Мотцкина эквивалентна итерации алгоритма Чиммино, поскольку требуется вычислить расстояния до всех полупространств, прежде чем выполнить проектирование.

### 2.3. Метод попеременных проекций для произвольных множеств

Методы попеременных проекций для отыскания общей точки произвольных замкнутых выпуклых множеств чаще всего используют идею алгоритмов Качмаржа (поочередные проекции на множества в циклическом порядке, см. фиг. 6) и Агмона–Мотцкина (в этом случае на каждом шаге выбирается самое дальнее множество).



Фиг. 6. Проекции в циклическом порядке (иллюстрация для двух множеств).

Поскольку исследованию данного класса методов посвящена обширная литература, в частности, обзоры [22], [23], перечислим лишь основные работы, положившие начало целому ряду исследований по выпуклой оптимизации с ограничениями.

Исторически первой была статья И. И. Ерёмкина [40] (написанная и представленная в редакцию в 1962 г.), в которой рассматривался метод Агмона–Мотцкина (5), (6) для произвольных замкнутых выпуклых множеств  $\Xi_i$ . Было показано, что если пересечение множеств непусто, то метод сходится, и наоборот, если последовательность  $\xi(t)$  имеет предел, то ее предел автоматически принадлежит пересечению всех множеств  $\Xi_*$ .

В работе Л. М. Брэгмана [14] исследовался вопрос сходимости алгоритмов при последовательном проектировании в циклическом порядке (как в алгоритме Качмаржа) и при проекции на самое дальнее выпуклое множество, как в алгоритме Агмона–Мотцкина. Сходимость исследовалась в гильбертовом пространстве, однако понималась лишь в слабом смысле (в случае конечномерного пространства она эквивалентна сходимости по норме) и была доказана в предположении, что множества имеют непустое пересечение. Более того, была исследована сходимость модифицированного алгоритма Агмона–Мотцкина в случае полупространств  $\Xi_i = \{\xi : a_i^T \xi \leq b_i\}$ . В этом модифицированном методе проекция осуществляется не на самое дальнее полупространство  $\Xi_i$ , а на множество, для которого ограничение  $a_i^T \xi \leq b_i$  на шаге  $t$  “наиболее нарушено”, иными словами, (5) нужно заменить условием

$$m(t) = \arg \max_{i \in \mathcal{V}} [a_i^T \xi(t) - b_i],$$

в то время как правило проектирования (6) остается без изменения.

Дальнейшее обобщение результатов Ерёмкина и Брэгмана было получено в работе Л. Г. Гурина<sup>7</sup>, Б. Т. Поляка и Э. В. Райка [15]. Как и в работе [14], исследовались два типа алгоритма попеременных проекций (в циклическом порядке и на самое дальнее множество) для замкнутых выпуклых множеств в произвольном гильбертовом пространстве. Были получены условия экспоненциальной (в норме гильбертова пространства) сходимости обоих алгоритмов в случае, если каждое из множеств  $\Xi_i$  имеет непустое пересечение с внутренностями остальных множеств:

$$\Xi_i \cap \bigcap_{j \neq i} \overset{\circ}{\Xi}_j \neq \emptyset \quad \forall i \in \mathcal{V}.$$

Последнее условие можно снять, если  $\Xi_i$  являются полупространствами (таким образом, результат [38] был обобщен на бесконечномерный случай). Приведено также некоторое условие сходимости по норме (“равномерная выпуклость” множеств  $\Xi_i$ ). Во всех указанных результатах точный оператор проектирования можно заменить его “релаксированной” версией, как в алгоритме (6) (более того, шаг  $\alpha$  может на каждой итерации алгоритма быть разным). Показано, что алгоритм циклических проекций при пустом пересечении множеств  $\Xi_i$  экспоненциально быстро сходится к периодической траектории. Был предложен также ускоренный вариант метода Агмона–Мотцкина (проекция на самое дальнее множество). В статье [15] приводится ряд интересных приложений метода последовательных проекций, в частности, к задаче аппроксимации функции полиномами на отрезке и специальной задаче оптимального управления.

**2.3.1. Дальнейшее развитие: фейеровские операторы и оптимизация.** Отметим, что алгоритмы для вычисления общей точки выпуклых множеств, рассматривавшиеся до сих пор, основаны на неявном предположении, что оператор проектирования на каждое из множеств может быть эффективно вычислен. Если в случае линейных гиперплоскостей и полупространств он допускает просто аналитическое представление, то в случае более сложных множеств (например, заданных некоторым нелинейным выпуклым неравенством  $f(\xi) \leq 0$ ) вычисление проекции возможно только численно и, вообще говоря, сводится к решению задачи выпуклой оптимизации. Вместе с тем, уже в первых работах (см. [10], [13], [38], [39]) было замечено, что оператор проекции может быть заменен “релаксированной” проекцией (6), и, более того, основное свойство, используемое в доказательстве — то, что он обладает некоторым слабым свойством сжатия и приближает любую точку за пределами выбранного множества  $\Xi$  к этому множеству. Данное наблюдение привело к развитию теории *фейеровских процессов*, основы которой были заложены Ерёмкиным в пионерских работах [13], [16], [42]–[44]. Наиболее полное изложение данной теории доступно в монографии [21]. Как будет показано в разд. 3, есть возможность обобщить результаты о сходимости итеративных процедур с фейеровскими операторами на многоагентный случай (см. теорему 1). В следующем пункте мы подробнее рассмотрим фейеровские отображения и приведем некоторые примеры.

Помимо того, часто интерес представляет нахождение не произвольной точки в пересечении выпуклых множеств, а точки, доставляющей оптимум некоторому функционалу (возможно, негладкому). Возможность решения задач линейного программирования с помощью метода последовательных проектирований была замечена Брэгманом (см. [14]), в последующих работах которого (см. [20], [45]) были рассмотрены итерационные методы “проектирования” в смысле некоторой “псевдометрики” на линейном пространстве, которая была названа  $D$ -расстоянием. Наиболее часто используемый тип  $D$ -расстояний называется *дивергенцией* Брэгмана и задается формулой

$$D_\varphi(x, y) = \varphi(x) - \varphi(y) - \nabla\varphi(y)(x - y)$$

для некоторой выпуклой дифференцируемой функции  $\varphi$ . Брэгманом было показано (см. [20]), что методы проекции в смысле подходящим образом построенного  $D$ -расстояния позволяют решать задачи выпуклого программирования с ограничениями в виде равенств и неравенств. Альтернативный подход, предложенный Ерёмкиным (см. [43]), приводит к изучению фейеровских отображений со сдвигом. Отметим, что в многоагентном случае алгоритмы оптимизации такого рода малоизучены, как правило, используются альтернативные алгоритмы, полученные децентрализацией методов проекции градиента (см. [25], [46], [47]); зачастую ограничения также заменяются барьерными функциями. Изучение алгоритмов оптимизации с ограничениями выходит за рамки настоящей статьи, достаточно полное изложение современных методов доступно, например, в монографии [48].

<sup>7</sup> К сожалению, при оцифровке переводной версии работы [41] фамилия первого автора была ошибочно написана как Gubin; такое написание до сих пор сохраняется в электронной базе ScienceDirect и попало в большинство обзоров по алгоритмам пересечения выпуклых множеств.

2.4. Фейеровские (паражсжимающие) отображения

Всюду в данном пункте мы предполагаем заданными некоторый оператор  $P : \mathbb{R}^d \rightarrow \mathbb{R}^d$  (вообще говоря, нелинейный) и некоторую норму  $\| \cdot \|$  на  $\mathbb{R}^d$  (не обязательно евклидову). Напомним, что на конечномерном пространстве все нормы эквивалентны и порождают одну и ту же топологию, соответственно понятия открытости, замкнутости и непрерывности в различных нормах также эквивалентны.

**Определение 1.** Оператор  $P$  называется *фейеровским*<sup>8</sup> относительно множества  $M \subset \mathbb{R}^d$ , или *M-фейеровским*, если выполнены три условия: [(а)]

1.  $P$  – непрерывное отображение на всем пространстве;
2. множество  $M$  неподвижно относительно  $P$ ;
3. оператор удовлетворяет условию сжатия относительно  $M$ , т.е.

$$\| P(\xi) - \xi_0 \| = \| P(\xi) - P(\xi_0) \| < \| \xi - \xi_0 \| \quad \forall \xi \notin M, \xi_0 \in M. \tag{7}$$

Очевидно, что из (7) следует, что  $P$  обладает свойством

$$\| P(\xi) - P(\xi_0) \| \leq \| \xi - \xi_0 \| \quad \forall \xi \in \mathbb{R}^d, \xi_0 \in M. \tag{8}$$

В англоязычной литературе фейеровские операторы называются часто парасжимающими (paracontracting) (см. [50]), встречаются также и иные термины, например, “притягивающий” (attracting) к множеству  $M$  оператор (см. [23]). Очевидно, что если  $\| x - y \| < \| P(x) - P(y) \|$ , то  $x \neq P(x)$ ; таким образом,  $M$  в определении 1 совпадает со множеством всех неподвижных точек  $M$ -фейеровского оператора  $\mathcal{F}(P) \triangleq \{ \zeta : P(\zeta) = \zeta \}$ ; поэтому, если не требуется явное уточнение данного множества, мы для краткости будем просто называть оператор фейеровским. Легко видеть также, что множество  $M$  автоматически является замкнутым и выпуклым. Первое свойство вытекает из непрерывности оператора  $P$ : если последовательность  $\{x_n\}$  элементов  $M$  сходится к некоторому пределу  $x \in \mathbb{R}^d$ , то, очевидно, для произвольного элемента  $a \in M$  имеем  $\| x - a \| = \lim_{n \rightarrow \infty} \| x_n - a \| = \lim_{n \rightarrow \infty} \| P(x_n) - a \| = \| P(x) - a \|$ , что возможно лишь в случае, когда  $x \in M$ . Второе свойство также легко доказать: если  $z$  – произвольный элемент отрезка, соединяющего точки  $x, y \in M$ , то  $\| x - y \| = \| x - z \| + \| z - y \| \geq \| x - P(z) \| + \| P(z) - y \| \geq \| x - y \|$ . Данное неравенство может быть выполнено только в случае, когда  $z \in M$  (в противном случае,  $\| x - P(z) \| < \| x - z \|$  и  $\| y - P(z) \| < \| y - z \|$ ). Следовательно, вместе с двумя точками  $M$  содержит и отрезок, их соединяющий.

Фейеровское отображение является естественным обобщением сжимающего (по Банаху) отображения  $P$ , для которого  $\| P(x) - P(y) \| \leq q \| x - y \| \quad \forall x, y \in \mathbb{R}^d$  при некоторой постоянной  $q \in (0, 1)$ . Принципиальным отличием сжимающих и фейеровских отображений является то, что у сжимающих отображений, в силу известной теоремы Банаха, существует ровно одна неподвижная точка, а множество неподвижных точек фейеровского отображения может быть любым замкнутым выпуклым множеством. В следующем пункте приводятся примеры фейеровских операторов.

2.4.1. Примеры фейеровских операторов.

**Пример 1.** Как мы видели, проектор  $P_\Omega$  на замкнутое выпуклое множество является фейеровским (в стандартной евклидовой норме). Более общо, можно показать, что при любом коэффициенте  $a \in (0, 2)$  оператор  $(1 - a)\text{Id} + aP_\Omega$ , где  $\text{Id}$  – тождественное отображение, также является фейеровским в евклидовой норме (см., например, лемму 2 в [40] и следствие 2.5 в [23]).

**Пример 2.** Выпуклая комбинация нескольких фейеровских отображений является фейеровским отображением. Пусть  $P_1, \dots, P_n$  – фейеровские операторы с множествами неподвижных точек  $M_1, \dots, M_n$ , и  $a_1, \dots, a_n > 0$  – коэффициенты, такие что  $\sum_{i=1}^n a_i = 1$ . Тогда легко проверить, что оператор  $P = \sum_{i=1}^n a_i P_i$  является фейеровским относительно  $M = M_1 \cap \dots \cap M_n$  (в предположении, что оно непусто). Объединяя данный пример с примером 1, можно показать, что в алгоритме Чиммино, описанном в п. 2.1, приближение  $\xi(t + 1)$  получается из  $\xi(t)$  путем применения фейеровского оператора.

**Пример 3.** Аналогично, легко проверить, что композиция фейеровских операторов  $P_n \circ \dots \circ P_1$  также является фейеровским относительно пересечения множеств неподвижных точек  $M = M_1 \cap \dots \cap M_n$  (в предположении, что оно непусто). Таким образом, последовательные  $n$  операций (где  $n$  – число множеств) в алгоритме Качмаржа эквивалентны применению фейеровского оператора.

<sup>8</sup> Термин “ $M$ -фейеровский оператор”, введенный в работах И.И. Ерёмкина, мотивирован понятием монотонной по Фейеру (относительно  $M$ ) последовательности, введенным в [39]. Так называется последовательность  $\{x_n\} \subset \mathbb{R}^d \setminus M$ , такая что  $x_{n+1} \neq x_n$  при всех  $n$  и  $\| x_n - y \| \geq \| x_{n+1} - y \|$  при всех  $n$  и  $y \in M$ . Это понятие, в свою очередь, восходит к работе Фейера [49] о расположении корней полиномов.

**Пример 4.** Пусть  $P$  в некоторой строго выпуклой норме<sup>9</sup>  $\|\cdot\|$  является нерасширяющим ( $\|P(x) - P(y)\| \leq 1$  при всех  $x, y \in \mathbb{R}^d$ ) с непустым множеством неподвижных точек  $M = \mathcal{F}(P)$ . Тогда при  $a \in (0, 1)$  оператор  $(1 - a)\text{Id} + aP$  также является нерасширяющим, а также  $M$ -фейеровским. Первое свойство проверяется непосредственно, кроме того, если  $y \in M$  и  $x \notin M$ , то  $P(x) - y \neq x - y$ , следовательно, либо  $\|P(x) - y\| < \|x - y\|$ , либо векторы  $P(x) - y$  и  $x - y$  неколлинеарны, либо  $P(x) - y = y - x$ . Во всех случаях

$$\|(1 - a)x + aP(x) - y\| = \|(1 - a)(x - y) + a(P(x) - y)\| < \|x - y\|.$$

В силу сделанного выше замечания получаем, в частности, известный факт: множество неподвижных точек нерасширяющего (относительно строго выпуклой нормы) отображения всегда является выпуклым (см. [23]).

Примеры 2—4 интересны тем, что явное вычисление проекции на множество  $M$  может быть затруднительно, тем не менее, может быть известен фейеровский оператор, приближающий точку не из  $M$  к этому множеству. На самом деле, таких примеров достаточно много. Следующий пример относится к распространенному случаю, когда выпуклое множество задается некоторым скалярным выпуклым неравенством.

**Пример 5.** Рассмотрим непрерывно дифференцируемую выпуклую функцию  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  и предположим, что множество  $M_f \triangleq \{x : f(x) \leq 0\}$  непусто (очевидно, это множество выпукло и замкнуто). Выбрав постоянную  $\alpha \in (0, 2)$ , рассмотрим отображение

$$P_{\alpha, f} : x \mapsto x' = \begin{cases} x - \alpha f(x) \frac{\nabla f(x)}{|\nabla f(x)|^2}, & f(x) > 0, \\ x, & f(x) \leq 0. \end{cases}$$

Отметим, что градиент  $\nabla f$ , в силу выпуклости, может обращаться в нуль лишь в точках (глобального) минимума  $f$ , а все точки минимума (если они существуют) должны принадлежать  $M_f$ . В силу этого, выражение  $P_{\alpha, f}(x)$  корректно определено при  $f(x) > 0$ . Нетрудно также показать, что  $P_{\alpha, f}$  — непрерывное (но, вообще говоря, недифференцируемое) отображение с множеством неподвижных точек  $M_f$ . Важное наблюдение, сделанное в [13], состоит в том, что оператор  $P_{\alpha, f}$  является фейеровским относительно евклидовой нормы. Конструкция тривиально распространяется на недифференцируемые выпуклые отображения, для которых можно выбрать непрерывную ветвь субдифференциала. Более точно, если существует непрерывное отображение  $g : \mathbb{R}^d \rightarrow \mathbb{R}^d$ , такое что  $g(x) \in \partial f(x)$  при всех  $x$ , то в определении  $P_{\alpha, f}$  градиент  $\nabla f(x)$  можно заменить на  $g(x)$  (см. [13]).

Другими примерами фейеровских отображений являются проксимальные операторы и операторы градиентного спуска, соответствующие строго выпуклым функциям (см. [50]).

**2.4.2. Итерации фейеровских операторов.** Следующая простая лемма показывает, что итерации фейеровского оператора так же, как и сжимающего (в смысле Банаха) отображения, сходятся к неподвижной точке.

**Лемма 1.** Пусть оператор  $P : \mathbb{R}^d \rightarrow \mathbb{R}^d$  является фейеровским с непустым множеством неподвижных точек  $M = \mathcal{F}(P)$ . Тогда последовательность  $(\xi(t))_{t=0}^\infty$ , задаваемая рекуррентным соотношением

$$\xi(t + 1) = P(\xi(t)), \quad t = 0, 1, \dots,$$

при любой начальной точке  $\xi(0)$  сходится к некоторому элементу множества  $M$ .

Доказательство леммы 1 будет дано в приложении. В силу примеров 2 и 3, из леммы 1 нетрудно вывести сходимость алгоритма Качмаржа и алгоритма Чиммино (для проекторов на произвольные множества не обязательно линейной структуры). Понятие фейеровского оператора и лемму 1 можно обобщить и на многозначные отображения (возникающие, например, в алгоритме Агмона–Мотцкина, если разрешить индексу  $m(t)$  в (5) принимать несколько значений). Наиболее общие результаты о сходимости и расхождении фейеровских процессов могут быть найдены в [17], [21], [23], [51].

<sup>9</sup> Норма строго выпукла, если выполнено строгое неравенство треугольника:  $\|x + y\| < \|x\| + \|y\|$  для любых линейно независимых векторов  $x, y$ .

### 3. МНОГОАГЕНТНЫЕ АЛГОРИТМЫ ДЛЯ ПЕРЕСЕЧЕНИЯ МНОЖЕСТВ

Обратимся теперь к многоагентной<sup>10</sup> постановке задачи о пересечении выпуклых множеств. В этой задаче каждое замкнутое выпуклое множество  $\Xi_i \subseteq \mathbb{R}^d$  связано со своим агентом (это может быть программный модуль, робот или живое существо, обладающее некоторой автономностью в принятии решений). Часто это множество трактуется как ограничение, выдвигаемое агентом  $i$ . Мы считаем, что агент  $i$  располагает достаточной информацией о своем множестве  $\Xi_i \subseteq \mathbb{R}^d$ , в частности, может вычислять проекцию  $P_{\Xi_i}(\xi)$  произвольной точки  $\xi \in \mathbb{R}^d$  на множество  $\Xi_i$  либо, более общо, значение какого-либо иного фейеровского оператора  $P_i(\xi)$  с множеством неподвижных точек  $\Xi_i$ . При этом агенту  $i$  недоступна информация о множествах других агентов  $\Xi_j$ ,  $j \neq i$ . Это может быть связано как со сложной структурой множеств, информацию о которых затруднительно передать через сеть, так и с требованиями приватности (множества могут зависеть от некоторой конфиденциальной информации). Целью агентов является найти некоторую точку  $\xi_* \in \Xi_* \triangleq \bigcap_{i \in \mathcal{I}} \Xi_i$ , которая удовлетворяет всем ограничениям.

В предположении неограниченной коммуникации между агентами общая точка может, теоретически, быть найдена с помощью обычного метода попеременных проекций (обобщенного алгоритма Качмаржа). Для этого требуется некоторым образом занумеровать агентов и обеспечить после каждой операции проектирования передачу полученного приближения к точке  $\xi(t)$  следующему (в циклическом порядке) агенту. Агент 1 выполняет проектирование начальной точки  $\xi(0)$  на множество  $\Xi_1$  либо, более общо, вычисление некоторого фейеровского оператора  $P_1(\xi(0))$  и передает результат агенту 2, применяющему оператор  $P_2$  и так далее, агент  $n$  применяет оператор  $P_n$  и передает результат агенту 1. Такая процедура, однако, неудовлетворительна по ряду причин. Во-первых, будучи формально независимыми, агенты лишены возможности выполнять операции параллельно: каждый из них простаивает  $n - 1$  из последовательных  $n$  итераций. Во-вторых, с точки зрения парадигмы многоагентных систем, агенты в общем случае должны быть равнозначны и не иметь каких-либо отличительных признаков в виде уникального индекса, соответственно, для получения некоторого упорядочения от 1 до  $n$  агентам требуется осуществить специальную процедуру. Наконец, как уже было сказано, на шаге с номером  $t$  должна быть возможность коммуникации между агентами с номерами  $1 + (t \bmod n)$  и  $1 + (t + 1 \bmod n)$ . На практике это гарантировано не всегда: коммуникационный граф может быть неполным, меняться с течением времени и быть заранее неизвестным (например, ряд коммуникационных линий может отключаться вследствие отказов или необходимости передать более важную информацию).

Более удобны с точки зрения многоагентной реализации обобщенные методы Чиммино (проектирование на все множества и вычисление барицентра) и Агмона–Мотцкина (нахождение самого дальнего множества), однако они предполагают, что агенты могут обмениваться информацией с некоторым центральным узлом, который собирает у них значения  $P_i(\xi(t))$ , а затем вычисляет их барицентр (либо самую дальнюю проекцию) для нахождения  $\xi(t + 1)$ . Наличие такого центрального узла в сети также не всегда возможно, кроме того, сложность такой централизованной системы растет с ростом числа агентов.

В силу сказанного, для решения задачи о пересечении множеств, принадлежащих агентам, были предложены альтернативные алгоритмы, которые близки по идее алгоритму Чиммино, однако основываются на идее консенсуса, достигаемого путем последовательных усреднений (короткое введение в консенсусные алгоритмы дано в следующем пункте). В отличие от обычных проекционных алгоритмов, каждый агент  $i$  имеет свое собственное состояние  $\xi^i(t) \in \mathbb{R}^d$ , которое на каждом шаге становится доступным соседним (в смысле коммуникационного графа) агентам, обновляется в зависимости от состояний соседей и в конечном итоге сходится к точке из пересечения множеств, при этом данное установившееся состояние одинаково для всех агентов.

Оставшаяся часть данного раздела организована следующим образом. В п. 3.1 приведено краткое изложение консенсусных алгоритмов и условие достижения консенсуса в сети агентов с переменным направленным графом. Нам данный результат понадобится в несколько более сильном виде (робастность консенсуса к исчезающе малым возмущениям). Другой важный результат — лемма о консенсусе, гарантируемом усредняющими неравенствами, которому посвящен п. 3.2. В п. 3.3 мы приводим несколько классов алгоритмов, предложенных в литературе для решения многоагентной задачи о пересечении множеств и формулируем основной результат статьи, который будет доказан в приложении.

<sup>10</sup> Отметим, что в настоящем обзоре мы не рассматриваем общую философию теории многоагентных (мультиагентных) систем и не даем точного определения агентов, которое различается в теории управления и компьютерных науках. С историей вопроса и основными задачами теории многоагентных систем (также называемых сетевыми) можно ознакомиться, например, в статьях и монографиях [52]–[58].

### 3.1. Классические алгоритмы консенсуса. Сведения из теории графов

Алгоритмы усреднения имеют длинную историю, восходящую, с одной стороны, к динамике марковских процессов (см. [59]), а с другой стороны, к агентным (микроскопическим) моделям динамики мнений (см. [60], [61]). В данном пункте мы ограничимся лишь кратким описанием линейных алгоритмов усреднения и приведем основное (на сегодняшний день) условие достижения консенсуса.

**3.1.1. Динамика последовательного усреднения. Консенсус.** Рассмотрим множество агентов  $\mathcal{V}$ , каждый из которых описывается скалярной переменной  $x_i(t)$ ,  $i \in \mathcal{V}$ , называемой *состоянием* и меняющейся в дискретном времени  $t = 0, 1, \dots$ . На каждом шаге  $t$  каждый из агентов вычисляет *взвешенное среднее* своего собственного состояния и состояний других агентов. Агенту, однако, доступна информация не обо всех состояниях, а лишь об агентах из некоторого множества  $\mathcal{V}_i(t) \subseteq \mathcal{V}$ , которое мы будем называть множеством *соседей* агента  $i$  в момент  $t$ . Мы всегда считаем, что  $i \in \mathcal{V}_i(t)$ . В каждый момент времени  $t \in \mathbb{N}$ , агент  $i$  присваивает *веса усреднения*  $a_{ij}(t)$  всем соседям. Веса каждого агента  $i$  удовлетворяют ограничениям

$$a_{ij}(t) \geq 0 \quad \forall j \in \mathcal{V}_i(t), \quad \sum_{j \in \mathcal{V}_i(t)} a_{ij}(t) = 1. \quad (9)$$

Затем, агент обновляет свое значение следующим образом:

$$x_i(t+1) = \sum_{j \in \mathcal{V}_i(t)} a_{ij}(t) x_j(t), \quad t = 0, 1, \dots, \quad (10)$$

Доопределяя  $a_{ij}(t) \stackrel{\Delta}{=} 0$  при  $j \notin \mathcal{V}_i(t)$ , мы получаем *стохастическую* матрицу усредняющих весов  $A(t) = (a_{ij})_{i,j \in \mathcal{V}}$ . Удобно переписать (10) в компактной матричной форме:

$$x(t+1) = A(t)x(t), \quad t = 0, 1, \dots, \quad (11)$$

где  $x(t)$  обозначает вектор-столбец, составленный из состояний агентов  $x_i(t)$ .

В случае постоянной матрицы  $A$  поведение системы (11), очевидно, определяется структурой матрицы  $A$ , хорошо изученной в теории марковских цепей. В приложениях, однако, принципиально важно иметь возможность работать с переменной матрицей, поскольку каналы коммуникации между агентами могут открываться и закрываться, что приводит к изменению отношений соседства (множеств  $\mathcal{V}_i$ ) и, как следствие, необходимости перераспределять веса. Поведение систем с переменной матрицей, или бесконечными матричными произведениями, в общем случае остается мало изученным. Основные результаты можно найти в монографии [62] и обзоре [59], а также недавних работах [63]–[65]. Мы приведем ниже лишь один результат, который будет использован в дальнейшем.

Поскольку на каждом шаге агент приближает свое состояние к состояниям соседей, то можно ожидать, что в итоге все состояния окажутся одинаковыми, иными словами, будет достигнут консенсус:

$$\bar{x}_1 = \dots = \bar{x}_n, \text{ где } \bar{x}_i \stackrel{\Delta}{=} \lim_{t \rightarrow \infty} x_i(t) \quad \forall i \in \mathcal{V}. \quad (12)$$

Иногда консенсус определяется в формально более слабом смысле, например, как асимптотическая синхронизация значений агентов:

$$\lim_{t \rightarrow \infty} \max_{i,j \in \mathcal{V}} |x_j(t) - x_i(t)| = 0. \quad (13)$$

Отметим, что в нашем случае для алгоритма (11) последнее требование эквивалентно “сильному” консенсусу. Этот факт известен из литературы по теории цепей Маркова (см. [62], теорема 4.17) (синхронизацией в этом случае является слабая эргодичность обратного матричного произведения, что эквивалентно сильной эргодичности, влекущей за собой консенсус). Связано это с тем, что максимальный (соответственно, минимальный) элемент вектора  $x(t)$  в силу уравнений (10) не возрастает (соответственно, не убывает) с ростом  $t$ , сходясь к конечному пределу. Поэтому нетрудно видеть, что

$$\lim_{t \rightarrow \infty} \max_{i,j \in \mathcal{V}} |x_j(t) - x_i(t)| = \lim_{t \rightarrow \infty} \max_i x_i(t) - \lim_{t \rightarrow \infty} \min_i x_i(t),$$

а следовательно, (13) эквивалентно тому, что  $\lim_{t \rightarrow \infty} \max_i x_i(t) = \lim_{t \rightarrow \infty} \min_i x_i(t)$ , т.е. все элементы  $x(t)$  сходятся к одному и тому же конечному значению.

Всюду далее мы называем (10) (или (11)) *алгоритмом итеративного усреднения*, заданного последовательностью стохастических матриц  $A(t)$ . Подчеркнем, что хотя матрицы  $A(t)$  “скрывают” в себе структуру исходных множеств  $\mathcal{V}_i(t)$  (отношений “соседства”), в действительности именно эти множества являются первичными, поскольку они определяют информационную структуру многоагентного алгоритма (кто и кому пересылает свое состояние), в то время как веса усреднения играют существенно меньшую роль в рассматриваемой ниже задаче консенсуса, хотя и не могут быть выбраны полностью произвольно. Вместо матрицы часто говорят о (нагруженном) графе влияния между агентами. В следующем пункте мы приводим определения из теории графов.

### 3.1.2. Графы влияния и связанные понятия.

**Определение 2.** *Граф* (формально, направленный или ориентированный граф)  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  – это пара двух конечных множеств  $\mathcal{V}$  (могут быть произвольными) и  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . Элементы  $\mathcal{V}$  называются вершинами графа, а элементы  $\mathcal{E}$  – ребрами. Обычно предполагается, что вершины индексируются от 1 до  $n$ , так что  $\mathcal{V} = [1 : n]$ . Ребро  $(i, j)$  соединяет вершину  $i$  с вершиной  $j$  (и также обозначается как  $i \rightarrow j$ ). Во всех графах, рассматриваемых ниже, в каждой вершине имеется *петля*, т.е. ребро  $(i, i)$ .

Следующие два понятия относятся к типам связности (направленных) графов.

**Определение 3.** Последовательность ребер  $v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_s$  называется маршрутом, соединяющим вершину  $v_0$  с вершиной  $v_s$ , а число ребер в маршруте  $s$  называется его *длиной*. Граф называется *сильно связным*, если каждые две разные вершины соединены маршрутом. Граф называется *квазисильно связным*, если существуют маршруты из некоторой вершины (называемой *корнем*) во все остальные вершины.

Отметим, что квазисильно связный граф имеет исходящее *остовное дерево* (spanning tree) (см. [66]), и наоборот, наличие такового в графе влечет квазисильную связность. Поэтому часто в критериях консенсуса квазисильная связность формулируется как существование остовного дерева в графе (см. [67], [68]).

Следуя традиции теории многоагентных систем, мы будем рассматривать графы, вершины которых ассоциируются с агентами, а ребро  $j \rightarrow i$  ассоциируется с влиянием агента  $j$  на агента  $i$  в заданный момент времени. Применительно к алгоритму (10), влияние означает, что состояние агента  $j$  участвует в формировании следующего состояния агента  $i$ , т.е.  $a_{ij}(t) \neq 0$ . Соответственно, вес  $a_{ij}(t)$  можно трактовать как силу влияния или вес соответствующего ребра в графе. Это мотивирует следующее определение.

**Определение 4.** Для заданной квадратной матрицы  $A = (a_{ij})_{i,j \in \mathcal{V}}$  определим ассоциированный граф  $(\mathcal{V}, \mathcal{E}[A])$ , вершины которого находятся в взаимно однозначном соответствии с индексным множеством  $\mathcal{V}$  и  $\mathcal{E}[A] = (j, i) : a_{ij} \neq 0$ . Тройка  $\mathcal{G}[A] \triangleq (\mathcal{V}, \mathcal{E}[A], A)$  в дальнейшем называется *нагруженным графом*, определяемым матрицей  $A$ .

**3.1.3. Предположения о весах влияния.** Несмотря на то что выбор матрицы весов, на самом деле, имеет существенно меньшую роль, чем структура графа влияния, консенсус не может быть достигнут при произвольном выборе весов. Например, циклический алгоритм

$$x_1(t+1) = x_2(t), \quad x_2(t+1) = x_3(t), \quad \dots, \quad x_n(t+1) = x_1(t)$$

не только не приводит к консенсусу, но и вообще не сходится в случае попарно различных начальных состояний  $x_i(0)$ , поскольку вектор  $x(t+1)$  просто получается из  $x(t)$  циклической перестановкой. Для исключения подобного поведения требуются некоторые условия, аналогичные аperiodичности марковских цепей. Наиболее распространенное условие, накладываемое на матрицы в большинстве работ — это свойство равномерной положительности диагональных элементов стохастической матрицы:

$$a_{ii}(t) \geq \eta > 0 \quad \forall i \in \mathcal{V} \quad \forall t. \quad (14)$$

Помимо этого ясно, что для достижения консенсуса связь между агентами не должна полностью прерываться, более того, веса влияния не могут становиться слишком малыми с течением времени. Это легко проиллюстрировать на примере двух агентов и матрицы  $a_{11}(t) = a_{22}(t) = 1 - a(t)$ ,  $a_{12}(t) = a_{21}(t) = a(t)$ . Нетрудно проверить, что

$$x_1(t+1) - x_2(t+1) = (1 - 2a(t))(x_1(t) - x_2(t)) = \dots = (x_1(0) - x_2(0)) \prod_{s=0}^t (1 - 2a(s)).$$

Если  $|a(t)| \neq 1/2$  при всех  $t$  и  $\sum_{t \geq 0} a(t) < \infty$ , то произведение в правой части сходится к ненулевому значению, соответственно, консенсус не достигается.

В связи с этим обычно вводятся некоторые требования “усиленной” связности графа влияний, которая должна сохраняться при отбрасывании слишком слабых связей между агентами. Мы будем рассматривать лишь наиболее распространенное свойство равномерной (или повторяющейся) связности.

**Определение 5.** Если дана неотрицательная матрица  $A \in \mathbb{R}^{\mathcal{V} \times \mathcal{V}}$ , то ее  $\varepsilon$ -скелет — это матрица  $A^\varepsilon$ , полученная обнулением всех элементов, которые меньше  $\varepsilon$ , т.е.

$$a_{ij}^\varepsilon \triangleq \begin{cases} a_{ij}, & a_{ij} \geq \varepsilon, \\ 0 & \text{в противном случае.} \end{cases}$$

Нагруженный граф  $\mathcal{G}[A^\varepsilon]$  называется  $\varepsilon$ -скелетом нагруженного графа  $\mathcal{G}[A]$ . Мы говорим, что нагруженный граф (квази)сильно  $\varepsilon$ -связен, если его  $\varepsilon$ -скелет является (квази)сильно связным<sup>11</sup>.

**Определение 6.** Имея последовательность матриц  $A(t) \in \mathbb{R}^{\mathcal{V} \times \mathcal{V}}$ , где  $t \in [t_0 : t_1]$ , мы определяем объединение соответствующих графов над  $[t_0 : t_1]$  как граф

$$\bigcup_{t=t_0}^{t_1} \mathcal{G}[A(t)] \triangleq \mathcal{G}\left[\sum_{t=t_0}^{t_1} A(t)\right].$$

Графы  $\mathcal{G}[A(t)]$ ,  $t \in \mathcal{T}$ , где  $\mathcal{T} = [t_0 : t_1]$ , называются совместно (квази)сильно  $\varepsilon$ -связными на интервале  $\mathcal{T}$ , если их объединение на  $\mathcal{T}$  является (квази)сильно  $\varepsilon$ -связным.

**Определение 7.** Меняющийся во времени граф  $\mathcal{G}[A(\cdot)]$  является равномерно (квази)сильно связным, если существуют такие константы  $\varepsilon > 0$  и  $T > 0$ , что графы  $\mathcal{G}[A(t)]$  совместно (квази)сильно  $\varepsilon$ -связны в каждый момент времени  $[t_0 : t_0 + T]$ ,  $t_0 \geq 0$ .

Отметим, что часто предполагается дополнительное ограничение на веса усреднения, которое на практике обычно выполняется. Именно, считается, что ненулевые элементы матрицы весов не могут быть слишком малы:

$$a_{ij}(t) \in \{0\} \cup [\varepsilon, 1] \quad \forall t = 0, 1, \dots \quad (15)$$

При этом понятие равномерной связности просто означает, что объединение графов за достаточно длинный период  $T$  является связным (в сильном или квазисильном смысле).

**3.1.4. Достаточное условие робастного консенсуса.** При изучении многоагентных алгоритмов пересечения множеств мы будем использовать следующее условие консенсуса, которое на самом деле гарантирует робастность консенсуса к исчезающим на бесконечности возмущениям. Наряду с (11), рассмотрим возмущенную систему

$$x(t+1) = A(t)x(t) + f(t), \quad t = 0, 1, \dots \quad (16)$$

Справедлива следующая

**Лемма 2.** Пусть  $A(t)$  — стохастические матрицы с равномерно положительными диагональными элементами (14). Тогда алгоритм (11) устанавливает консенсус (12), если переменный граф  $\mathcal{G}[A(\cdot)]$  обладает равномерной квазисильной связностью. Более того, если возмущение  $f(\cdot)$  исчезает на бесконечности  $f(t) \xrightarrow{t \rightarrow \infty} 0$ , то

$$\max x(t) - \min x(t) \xrightarrow{t \rightarrow \infty} 0 \quad (17)$$

для любого решения (16).

Лемма 2 обычно доказывается при дополнительном предположении (15), в таком виде она доступна в [46]. Как показано в диссертации [28], это условие можно снять, однако доказательство становится существенно более сложным. Можно также распространить рассуждения из работы [69], доказывающей непрерывный аналог леммы 2 (наряду с более общими критериями робастного консенсуса), на случай систем с дискретным временем. Все доказательства из указанных источников, однако, достаточно сложны технически, и мы их здесь не приводим.

<sup>11</sup> Иными словами, граф является (квази)сильно  $\varepsilon$ -связным, если он остается (квази)сильно связным после удаления всех дуг, которые имеют вес меньше, чем  $\varepsilon$ .

### 3.2. Консенсус в усредняющих неравенствах

В этом пункте мы введем важный инструмент для доказательства главного результата настоящей статьи. Именно, наряду с уравнениями (11), мы будем рассматривать *рекуррентные усредняющие неравенства* следующего вида:

$$x(t + 1) \leq A(t)x(t), \quad t \in \mathbb{N}. \tag{18}$$

Неравенство между векторами здесь понимается покомпонентно.

Следует отметить, что рекуррентное неравенство не может рассматриваться как алгоритм управления для группы агентов, поскольку его решение не является однозначно определенным. Однако такие неравенства представляются удобным инструментом в анализе многоагентных алгоритмов (см. [30], [61], [70]). Может показаться удивительным, что столь слабое ограничение на вектор состояния системы агентов в принципе позволяет установить какие-то свойства, однако, как будет показано, в предположении равномерной сильной связности графа может, как и в случае уравнений, быть доказан консенсус (12). В отличие от уравнений (11), решения которых ограничены, решения неравенств только полуограничены сверху. В частности, некоторые компоненты могут сходиться к  $-\infty$  при  $t \rightarrow \infty$ . Так консенсус в случае неограниченных решений означает, что  $\bar{x}_i = -\infty \forall i$ . В практических приложениях, однако, мы часто знаем априорную нижнюю границу для решения, таким образом, ситуация консенсуса в бесконечности не реализуется.

Как мы увидим, при изучении решений неравенств часто важно знать поведение вектора невязок между правой и левой частью:

$$\Delta_i(t) \triangleq \sum_j a_{ij}(t)x_j(t) - x_i(t + 1) \geq 0, \tag{19}$$

в частности, сходимость этих невязок к нулю:

$$\Delta_i(t) \xrightarrow{t \rightarrow \infty} 0. \tag{20}$$

Следующая ключевая лемма гарантирует как достижение консенсуса, так и сходимость к нулю невязок для любого ограниченного решения (18).

**Лемма 3.** Пусть последовательность стохастических матриц  $A(t)$  имеет равномерно положительные диагональные элементы  $a_{ii}(t) \geq \eta > 0$ , а переменный граф  $\mathcal{G}[A(\cdot)]$  равномерно сильно связан. Тогда для любого решения неравенства (18) имеет место консенсус (12), а если решение ограничено снизу, то выполнено (20) при любом  $i \in \mathcal{V}$ .

Доказательство леммы 3 дано в приложении; специальные случаи этой леммы (предполагающие равномерную положительность всех ненулевых элементов матрицы, а не только диагональных) были доказаны в работах первого автора [30], [70]. Отметим, что требование равномерной сильной связности может быть существенно ослаблено, наиболее общий результат приведен в диссертации [28].

### 3.3. Консенсусные алгоритмы для пересечения выпуклых множеств

Рассмотрим несколько классов алгоритмов, предложенных в литературе для решения многоагентной задачи о пересечении выпуклых множеств. Как будет показано, техника усредняющих неравенств позволяет достаточно просто доказать сходимость состояния всех агентов в каждом из указанных алгоритмов к некоторой консенсусной точке, лежащей в пересечении множеств.

Именно, предположим, что выпуклое замкнутое множество  $\Xi_i \subset \mathbb{R}^d$ , связанное с агентом  $i \in \mathcal{V}$ , является множеством неподвижных точек фейеровского относительно некоторой нормы  $\|\cdot\|$  отображения  $P_i$  (в наиболее распространенном случае,  $P_i$  – проектор на множество  $\Xi_i$ , а норма евклидова). Предположим, что агент  $i$  способен вычислять знач.  $P_i(\xi)$  в произвольной заданной точке  $\xi$ .

**3.3.1. Классы консенсусных алгоритмов.** Мы покажем, что при выполнении некоторых ограничений на граф, точка, принадлежащая  $\Xi_*$ , может быть вычислена одним из следующих алгоритмов, основанных на последовательном усреднении с одновременным применением фейеровских операторов:

$$\xi^i(t + 1) = P_i \left[ \sum_{j \in \mathcal{V}} a_{ij}(t) \xi^j(t) \right], \quad i \in \mathcal{V}, \tag{21}$$

$$\xi^i(t + 1) = P_i \left[ \sum_{j \in \mathcal{V}} a_{ij}(t) P_j(\xi^j(t)) \right], \quad i \in \mathcal{V}, \tag{22}$$

$$\xi^i(t+1) = \sum_{j \in \mathcal{V}} a_{ij}(t) P_j(\xi^j(t)), \quad i \in \mathcal{V}, \quad (23)$$

$$\xi^i(t+1) = a_{ii}(t) P_i(\xi^i(t)) + \sum_{j \neq i} a_{ij}(t) \xi^j(t), \quad i \in \mathcal{V}. \quad (24)$$

Во всех указанных алгоритмах, как и прежде,  $A(t) = (a_{ij}(t))$  – стохастические матрицы. Отметим, что в вырожденном случае, когда у агентов нет ограничений ( $\Xi_i = \mathbb{R}^d$  и  $P_i = \text{Id}$ ) алгоритмы (21)–(24) превращаются в стандартную процедуру последовательного усреднения, применяемую для достижения консенсуса (см. [54], [59], [71]) (с единственной разницей, что состояние агента является многомерным вектором, а не скаляром – это, очевидно, никоим образом не меняет условий сходимости алгоритма). Вместе с тем, в общем случае условие сходимости существенно более ограничительно: требуется равномерная сильная связность, а не квазисильная, как в лемме 2. Это ограничение на граф накладывается во всех известных авторам работах. Метод доказательства сходимости, изложенный в настоящей статье, сводит каждый из алгоритмов к системе усредняющих неравенств, сильная связность требуется для применения леммы 3.

Происхождение и особенности алгоритмов (21)–(24). Алгоритм (21) был предложен в основополагающей статье [25] для случая проекционных операторов  $P_i$ , посвященной проблемам распределенной оптимизации, а затем развит в [46], где были ослаблены некоторые предположения (в частности, требование симметричности матрицы  $a_{ij}$ ) и исследована робастность алгоритма [25] к запаздываниям<sup>12</sup>. В более поздней работе [50] алгоритм (21) был предложен для поиска общей точки фейеровских<sup>13</sup> отображений  $P_i$ . Существенной особенностью данного алгоритма является то, что агент сначала получает состояния соседей, а затем, вычисляет  $P_i$  от усредненного значения. Отметим, что если  $P_i$  – проектирование на  $\Xi_i$ , то состояние агента  $i$  не покидает множества  $\Xi_i$  при  $t \geq 0$ . В этом случае агент стремится сблизить свое состояние с состояниями соседей, не нарушая собственное ограничение.

Алгоритм (24) был изначально предложен [72] для решения систем линейных уравнений, однако работает и для более общего случая. В этом случае  $\Xi_i$  – линейные гиперплоскости, а операторы проектирования  $P_i$  можно вычислить аналитически. Как и в алгоритме (21), агенты обмениваются своими состояниями. Однако имеются два принципиальных отличия: во-первых, агент может вычислять значение  $P_i$ , не дожидаясь получения соседних измерений (на практике, это может привести к ускорению вычислений), а во-вторых, состояние агента  $i$ , как правило, не принадлежит  $\Xi_i$  ни на одном шаге.

Алгоритм (23) аналогичен по структуре алгоритму Чиммино, с той разницей, что каждый агент, во-первых, имеет свое приближение к точке из пересечения множеств  $\xi^i$ , а во-вторых, вместо барицентра всех проекций вычисляется некоторая выпуклая комбинация проекций соседних агентов. Близкие по структуре распределенные алгоритмы изучались в [17], однако в этой работе не вводились явно агенты, и не рассматривался случай переменного графа. Отметим, что в случае этого алгоритма агенты не только не раскрывают соседям информацию о своем множестве, но даже не обмениваются состояниями (это может быть существенно для защиты информации).

Наконец, алгоритм (22) был предложен в работе [73], также посвященной алгоритмам решения алгебраических уравнений. Как и в алгоритме (23), агенты обмениваются значениями  $P_i$ , однако затем агент  $i$  повторно вычисляет значение  $P_i$ . В случае проекций, как уже обсуждалось, это гарантирует, что состояние агента  $i$  не покидает желаемого множества. Как будет показано ниже, это также позволяет избежать некоторых патологических точек равновесия, возникающих в системе (23) (при пустом пересечении множеств  $\Xi_* \neq \emptyset$ ).

**3.3.2. Теорема о консенсусе с ограничениями.** Предположим, что  $\Xi_* = \bigcap \Xi_i = \bigcap \mathcal{F}(P_i)$  непусто. Будем говорить, что установлен консенсус с ограничениями (см. [25]), если при произвольных начальных состояниях агентов существуют и равны между собой следующие пределы:

$$\lim_{t \rightarrow \infty} \xi^1(t) = \dots = \lim_{t \rightarrow \infty} \xi^n(t) \in \Xi_*. \quad (25)$$

<sup>12</sup> Как вытекает из результатов [28], [30], на самом деле к коммуникационным запаздываниям робастны все алгоритмы (21)–(24), поскольку такие запаздывания не нарушают свойств усредняющих алгоритмов и неравенств. В настоящем обзоре мы не рассматриваем вопросы робастности, которые в основном изучаются в рамках теории управления многоагентными системами и представляют интерес, если агенты ограничены в возможности быстро передавать информацию (например, связаны через Интернет).

<sup>13</sup> Отметим, что [50] накладывает ряд ограничений, в частности, свойство Фейера должно выполняться относительно  $\ell_p$  нормы на  $\mathbb{R}^d$ ; как мы увидим ниже, это ограничение можно снять.

Анализ алгоритмов (21)–(24) опирается на свойства сходимости усредняющих неравенств (18) и лемму 2 о робастности консенсуса. Сформулируем основной результат, который будет доказан в приложении.

**Теорема 1.** *Предположим, что отображения  $P_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  являются фейеровскими относительно некоторой общей нормы  $\|\cdot\|$  и имеют по крайней мере одну общую неподвижную точку, т.е.  $\Xi_* = \bigcap_{i \in \mathcal{V}} \mathcal{F}(P_i) \neq \emptyset$ . Пусть последовательность стохастических матриц  $A(t)$  имеет равномерно положительные диагональные элементы  $a_{ii}(t) \geq \eta > 0$ , и переменный граф  $G[A(\cdot)]$ , соответствующий  $A(t)$ , равномерно сильно связан. Тогда каждый из алгоритмов (21)–(24) находит общую неподвижную точку  $\{P_i\}$ , т.е. состояния агентов  $\xi^i$  сходятся к консенсусу (25).*

Отметим что теорема 1 в ее полной общности публикуется впервые. Сходимость алгоритмов (21), (22) и (24) с помощью рекуррентных усредняющих неравенств впервые была доказана в [30] при дополнительном предположении (15), которое впоследствии было снято в [28]. Исследование алгоритма (23), однако, проведено не было.

**3.3.3. Замечания к теореме 1.** Отметим, что в условиях отсутствия общих точек у множеств  $\Xi_i$  сходимость векторов  $\xi^i(t)$  к общему значению  $\xi_*$  при  $t \rightarrow \infty$  для алгоритмов (21), (22) и (24), очевидно, невозможна. Для алгоритмов (21) и (22) это так, поскольку  $\xi^i(t) \in \Xi_i$  при любом  $i \in \mathcal{V}$ , таким образом, общий предел автоматический является общей точкой всех  $\Xi_i$ . Для (24) это следует из непрерывности  $P_i$ : предполагая, что  $\xi^i(t) \xrightarrow{t \rightarrow \infty} \xi_*$ , мы имели бы

$$a_{ii}(t) \left[ P_i(\xi^i(t)) - \xi^i(t) \right] = \xi^i(t+1) - \sum_{j \in \mathcal{V}} a_{ij}(t) \xi^j(t) \xrightarrow{t \rightarrow \infty} 0,$$

что в силу равномерной положительности  $a_{ii}$  вновь означает, что  $\xi_* = P_i(\xi_*)$  при всех  $i$ , т.е.  $\xi_*$  – общая неподвижная точка семейства операторов. Алгоритм (23), однако, является исключением: консенсус при этом алгоритме возможен, теоретически, и при пустом пересечении множеств  $\Xi_i$ . Рассмотрим, к примеру, систему из двух агентов с множествами ограничений  $\Xi_1 = \{-1\}$ ,  $\Xi_2 = \{1\}$  и весовыми коэффициентами  $a_{11} = a_{12} = a_{21} = a_{22} = 1/2$ . Очевидно, что  $\xi^1(t) \equiv \xi^2(t) \equiv 0$  – точка равновесия системы (23), не принадлежащая ни одному из множеств  $\Xi_i$ .

Вместе с тем, в случае, когда консенсус не достигается, вопрос о поведении траекторий системы является, насколько известно авторам, открытым. Неизвестно в частности, даже при постоянной матрице весов, всегда ли в этом случае имеет место сходимость к периодической или квазипериодической траектории. Более того, даже ограниченность решений, как видно из доказательства в приложении, опирается на непустоту  $\Xi_* = \bigcap_i \Xi_i$  и, за исключением специальных случаев (например,  $P_i$  – проектор на ограниченное множество  $\Xi_i$ ), вообще говоря, не является очевидным свойством.

Следует отметить также, что скорость сходимости алгоритмов (21)–(24) достаточно трудно оценить явно даже в случае постоянного графа. В случае же переменного графа, даже точная скорость сходимости обычных консенсусных алгоритмов ( $P_i = \text{Id}$ ) до конца не исследована; ряд оценок на скорость сходимости может быть найден в обзоре [74]. Как показано в недавней работе [75], оценка скорости сходимости алгоритмов консенсуса сводится к вычислению наибольшего коэффициента эргодичности для некоторого компактного множества стохастических матриц. Существуют “ускоренные” модификации алгоритма (21) для решения систем линейных уравнений (см. [27], [73]), для которых доказана экспоненциальная сходимость к консенсусу. Данные алгоритмы не были исследованы в случае произвольных фейеровских операторов, а их анализ требует техники, выходящей за рамки данного обзора.

В настоящей статье мы не рассматриваем стохастические варианты алгоритма пересечения выпуклых множеств, в частности, рандомизированную версию алгоритма (24) из работы [76], а также рандомизированные модификации алгоритма Качмаржа (см., например, обзор литературы в [77]). Следует отметить, что для некоторых из рассмотренных выше алгоритмов существуют аналоги с непрерывным временем, например, в работе [26] приведен следующий аналог алгоритма (24):

$$\dot{\xi}^i(t) = \sum_{j \in \mathcal{V}} a_{ij}(t) (\xi^j(t) - \xi^i(t)) + P_i(\xi^i(t)) - \xi^i(t). \tag{26}$$

Исследование данного алгоритма возможно с помощью теории усредняющих неравенств в непрерывном времени (см. [28], [29]); мы не приводим соответствующие результаты, поскольку они полностью аналогичны критериям сходимости в дискретном времени.

## 4. ЗАКЛЮЧЕНИЕ

Авторы благодарны редакторам специального выпуска, посвященного Б. Т. Поляку, за возможность представить данный обзор, который представляет расширенный материал лекции, прочитанной первым автором на Традиционной летней школе по управлению и оптимизации им. Б. Т. Поляка летом 2023 г. В обзоре представлена история задачи о пересечении выпуклых множеств в пространстве, а также недавние результаты первого автора, относящиеся к многоагентной постановке этой задачи и децентрализованным алгоритмам для ее решения. Борис Теодорович Поляк считал статью [15], в которой был получен ряд важных результатов об экспоненциальной сходимости проекционных алгоритмов, одной из важнейших своих работ. Об этой статье он, в частности, рассказывал на семинаре ИПУ РАН, организованном в честь присуждения ему премии им. Хачияна<sup>14</sup>. Согласно базе данных Google Scholar, английская версия данной статьи к настоящему моменту процитирована более 1000 раз.

Следует отметить, что хотя с доказательства первой теоремы фон Неймана, устанавливающей сходимость метода последовательных проекций для двух линейных подпространств, прошло уже 90 лет, теория проекционных алгоритмов далека от своего завершения. Это относится как к классической (централизованной) версии задачи, так и многоагентной (децентрализованной) постановке. Как уже было сказано, достаточно много вопросов связано со скоростью сходимости алгоритмов, в частности, точная оценка скорости неизвестна даже для алгоритма Качмаржа. При замене проекционных операторов фейеровскими отображениями, ассоциированными с множествами, оценки скорости сходимости удается получить лишь в специальных случаях (см. [44]). В случае децентрализованного алгоритма, даже для консенсуса без ограничений (когда все множества совпадают с пространством) оценка скорости сходимости состояний агента к консенсусному значению является нетривиальной задачей (см. [74]).

Другим открытым вопросом является нахождения минимальных (необходимых и достаточных) условий связности графа. Как показано в диссертации [28], равномерная связность в леммах 2 и 3 может быть существенно ослаблена, однако приведенное условие все равно является только достаточным, а не необходимым, кроме того, его проверка для произвольного переменного графа затруднительна.

Перспективным направлением исследований представляется также развитие проекционных методов для многообразий (см. [24]) на многоагентный случай. Данная задача связана с вопросом о сходимости консенсусных алгоритмов на многообразиях (см. [78]).

## ПРИЛОЖЕНИЕ

**1. Доказательство леммы 3.** Предположим, что все условия леммы 3 выполнены, в частности,  $a_{ii}(t) \geq \eta$ . Рассмотрим сначала ситуацию, когда условие равномерной связности выполнено с  $T = 1$ , т.е. все графы  $\mathcal{G}[A(t)]$  являются сильно  $\varepsilon$ -связными. Обозначим число агентов  $n = |\mathcal{V}|$  и, при каждом  $t = 0, 1, \dots$  занумеруем элементы  $\mathcal{V}$  в порядке возрастания компонент  $x(t)$ :

$$\mathcal{V} = \{\sigma_1(t), \dots, \sigma_N(t)\},$$

$$y_1(t) \triangleq x_{\sigma_1(t)}(t) \leq y_2(t) \triangleq x_{\sigma_2(t)}(t) \leq \dots \leq y_n(t) \triangleq x_{\sigma_n(t)}(t).$$

Мы докажем следующее вспомогательное неравенство:

$$y_{k+1}(t+1) \leq (1 - \varepsilon_0)y_n(t) + \varepsilon_0 y_k(t) \quad \forall k = 0, 1, \dots, n-1, \quad (27)$$

где  $\varepsilon_0 = \min(\varepsilon, \eta)$ . Действительно, обозначим  $I \triangleq \{\sigma_{k+1}(t), \dots, \sigma_n(t)\}$ . Так как  $\mathcal{G}[A(t)]$  – сильно  $\varepsilon$ -связный граф, то существует  $i \in I$  и  $m \in J \triangleq \mathcal{V} \setminus I$  такие, что  $a_{im}(t) \geq \varepsilon \geq \varepsilon_0$ . Следовательно,

$$x_i(t+1) \leq a_{im}(t) \underbrace{x_m(t)}_{\leq y_k(t)} + \sum_{j \in \mathcal{V} \setminus \{m\}} a_{ij}(t) \underbrace{x_j(t)}_{\leq y_n(t)} \leq (1 - \varepsilon_0)y_n(t) + \varepsilon_0 y_k(t).$$

С другой стороны, используя неравенство  $a_{ii}(t) \geq \eta \geq \varepsilon_0$ , для каждого  $j \in J$  мы имеем

$$x_j(t+1) \leq a_{jj}(t) \underbrace{x_j(t)}_{\leq y_k(t)} + \sum_{\ell \in \mathcal{V} \setminus \{j\}} a_{j\ell}(t) \underbrace{x_\ell(t)}_{\leq y_n(t)} \leq (1 - \varepsilon_0)y_n(t) + \varepsilon_0 y_k(t).$$

Следовательно, все компоненты  $x$  с индексами из множества  $J \cup \{i\}$  не больше правой части из (27). Для доказательства (27) остается заметить, что мощность последнего множества равна  $k + 1$ .

<sup>14</sup> Видеозапись семинара доступна по ссылке [https://www.youtube.com/watch?v=\\_6p5qQ15fpw](https://www.youtube.com/watch?v=_6p5qQ15fpw)

Поскольку  $y_n(t) = \max x(t)$  не возрастает, существует предел  $y_* = \lim_{t \rightarrow \infty} y_n(t) \geq -\infty$ . Если  $y_* = -\infty$ , то, очевидно,  $x_i(t) \xrightarrow{t \rightarrow \infty} -\infty$ . Иначе, мы можем использовать (27) для доказательства того, что  $y_k(t) \xrightarrow{t \rightarrow \infty} y_* > -\infty$ , используя обратную индукцию по  $k = n, n-1, \dots, 1$ , иными словами, достигается консенсус (12).

Случай произвольного периода  $T > 1$  сводится к уже рассмотренному случаю: достаточно заметить, что в силу равномерной положительности диагональных элементов графы матриц

$$\tilde{A}(k) = A(kT)A(kT+1)\dots A(kT+T-1)$$

являются сильно  $\tilde{\varepsilon}$ -связными при некотором  $\tilde{\varepsilon} = \tilde{\varepsilon}(\varepsilon, \eta) > 0$ . Поэтому сходимость к консенсусу имеет место для подпоследовательности  $\tilde{x}(k) = x(kT)$ , удовлетворяющей рекуррентному неравенству

$$\tilde{x}(k+1) \leq \tilde{A}(k)\tilde{x}(k).$$

Поскольку для любого  $\ell = 1, 2, \dots, T-1$  справедливы неравенства

$$\begin{aligned} \tilde{x}(k+1) &= x(kT+T) \leq A(kT+T-1)\dots A(kT+\ell)x(kT+\ell), \\ x(kT+\ell) &= A(kT+\ell-1)\dots A(kT)\tilde{x}(k), \end{aligned}$$

то из сходимости  $\tilde{x}(k) \xrightarrow{k \rightarrow \infty} \tilde{c}1_n$  (где  $\tilde{c} \in \mathbb{R}$  и  $1_n$  обозначает вектор-столбец из единиц размерности  $n$ ) и стохастичности матриц  $A(t)$  вытекает сходимость  $x(kT+\ell) \xrightarrow{k \rightarrow \infty} \tilde{c}1_n$  при любом  $\ell = 1, 2, \dots, T-1$ , следовательно, достигается консенсус между агентами.

Наконец, свойство (20) для ограниченных решений также очевидно из того, что матрицы  $A(t)$  стохастические (и, в частности, равномерно ограничены): если  $x(t) \xrightarrow{t \rightarrow \infty} c1_n$  (при некотором  $c \in \mathbb{R}$ ), то  $A(t)1_n = 1_n$ , а следовательно,

$$A(t)x(t) = A(t)(x(t) - c1_n) + c1_n \xrightarrow{t \rightarrow \infty} c1_n.$$

## 2. Доказательство леммы 1 и теоремы 1.

### Лемма о фейеровском отображении.

Доказательства теоремы и леммы используют следующий технический результат.

**Лемма 4.** Пусть  $P$  – фейеровский оператор в некоторой норме  $\|\cdot\|$ , и  $\xi^0$  – его неподвижная точка. Обозначим  $d(\xi) \triangleq \|\xi - \xi^0\| - \|P\xi - \xi^0\| \geq 0$  и рассмотрим ограниченную последовательность векторов  $\xi(t)$  такую, что  $d(\xi(t)) \xrightarrow{t \rightarrow \infty} 0$ . Тогда

$$\|P\xi(t) - \xi(t)\| \xrightarrow{t \rightarrow \infty} 0.$$

**Доказательство.** Предположим противное:  $\|P\xi(t_r) - \xi(t_r)\| \geq \varepsilon$  для любого  $r \in \mathbb{N}$ ,  $\varepsilon > 0$ , и последовательности  $t_r \rightarrow \infty$ . Переходя к меньшей подпоследовательности, можно предположить, не умаляя общности, что векторы  $\xi(t_r)$  сходятся к пределу  $\xi_* \in \mathbb{R}^d$ . Поскольку  $P$  – непрерывное отображение, имеем  $\|P\xi_* - \xi_*\| \geq \varepsilon$  при  $d(\xi_*) = 0$ . Пришли к противоречию с (7), так как  $\xi_* \notin \mathcal{F}(P)$ , в то время как  $\|P\xi_* - \xi_*\| = \|\xi_* - \xi_*^0\|$ . Лемма 4 доказана.

**Доказательство леммы 1.** Очевидно, что последовательность  $\xi(t)$  из леммы 1 ограничена в силу того, что расстояния от  $\xi(t)$  до любой точки из  $M$  не возрастают. Невозрастающая последовательность

$$a(t) \triangleq \|\xi(t) - \xi^0\| \geq 0$$

для произвольного элемента  $\xi^0 \in M$  имеет конечный предел при  $t \rightarrow \infty$ . В частности, поскольку  $P\xi(t) = \xi(t+1)$ , то

$$a(t) - a(t+1) = \|\xi(t) - \xi^0\| - \|P\xi(t) - \xi^0\| = d(\xi(t)) \xrightarrow{t \rightarrow \infty} 0.$$

В силу леммы 4, мы доказали, что  $\|P\xi(t) - \xi(t)\| \xrightarrow{t \rightarrow \infty} 0$ .

Отметим теперь, что в силу ограниченности, существует подпоследовательность  $\xi(t_k)$ , сходящаяся к некоторому элементу  $\xi_* \in \mathbb{R}^d$ . Очевидно, что

$$\|P(\xi_*) - \xi_*\| = \lim_{k \rightarrow \infty} \|P(\xi(t_k)) - \xi(t_k)\| = 0,$$

т.е.  $\xi_* \in M$ . Поскольку  $\xi^0$  в рассуждении выше произвольно, мы можем выбрать  $\xi^0 = \xi^*$ . Поскольку при  $t \geq t_k$ , мы имеем

$$0 \leq \|\xi(t) - \xi_*\| \leq \|\xi(t_k) - \xi_*\|,$$

где правая часть может быть сделана сколь угодно малой при большом  $k$ , то  $\|\xi(t) - \xi_*\| \xrightarrow{t \rightarrow \infty} 0$ . Лемма 1 доказана.

**Доказательство теоремы 1.** Сначала введем некоторые вспомогательные обозначения. Выберем произвольную точку  $\xi^0 \in \Xi_*$  и положим, по определению,  $\delta_i(\xi) \triangleq \|\xi - \xi^0\| - \|P_i \xi - \xi^0\| \geq 0$ . Пусть  $\zeta^i(t) = \sum_{j \in \mathcal{V}} a_{ij}(t) \xi^j(t)$ . Центральная идея доказательства состоит в исследовании свойств векторов  $x(t) = (x_i(t))_{i \in \mathcal{V}}$ , компоненты которых  $x_i(t) \triangleq \|\zeta^i(t) - \xi^0\|$  обозначают расстояния от состояний агентов  $\xi^i(t)$  до выбранной неподвижной точки.

**Шаг 1.** Покажем сначала, что для каждого из указанных в теореме алгоритмов векторы  $x(t)$  удовлетворяют усредняющему рекуррентному неравенству (18).

В случае алгоритма (21) имеем

$$\begin{aligned} x_i(t+1) &= \|P_i(\zeta^i(t)) - \xi^0\| \stackrel{(8)}{\leq} \|\zeta^i(t) - \xi^0\| = \\ &= \sum_j a_{ij}(t) (\xi^j(t) - \xi^0) \leq \sum_j a_{ij}(t) \|\xi^j(t) - \xi^0\| = \sum_j a_{ij}(t) x_j(t). \end{aligned} \tag{28}$$

Аналогично рассматривается случай (22). Обозначив  $\bar{\zeta}^i(t) \triangleq \sum_{j \in \mathcal{V}} a_{ij}(t) P_j(\xi^j(t))$ , имеем

$$\begin{aligned} x_i(t+1) &= \|P_i(\bar{\zeta}^i(t)) - \xi^0\| \stackrel{(8)}{\leq} \|\bar{\zeta}^i(t) - \xi^0\| = \\ &= \sum_j a_{ij}(t) (P_j(\xi^j(t)) - \xi^0) \leq \sum_j a_{ij}(t) \|P_j(\xi^j(t)) - \xi^0\| \stackrel{(8)}{\leq} \\ &\leq \sum_j a_{ij}(t) \|\xi^j(t) - \xi^0\| = \sum_j a_{ij}(t) x_j(t). \end{aligned} \tag{29}$$

В случае алгоритма (23) нетрудно видеть, что

$$\begin{aligned} x_i(t+1) &= \sum_j a_{ij}(t) (P_j(\xi^j(t)) - \xi^0) \leq \sum_j a_{ij}(t) \|P_j(\xi^j(t)) - \xi^0\| \stackrel{(8)}{\leq} \\ &\leq \sum_j a_{ij}(t) \|\xi^j(t) - \xi^0\| = \sum_j a_{ij}(t) x_j(t). \end{aligned} \tag{30}$$

В случае алгоритма (24) имеем

$$\begin{aligned} x_i(t+1) &= \left\| a_{ii}(t) (P_i(\xi^i(t)) - \xi^0) + \sum_{j \neq i} a_{ij}(t) (\xi^j(t) - \xi^0) \right\| \leq \\ &\leq a_{ii}(t) \|P_i(\xi^i(t)) - \xi^0\| + \sum_{j \neq i} a_{ij}(t) \|\xi^j(t) - \xi^0\| \stackrel{(8)}{\leq} \\ &\leq a_{ii}(t) \|\xi^i(t) - \xi^0\| + \sum_{j \neq i} a_{ij}(t) x_j(t) = \sum_{j \in \mathcal{V}} a_{ij}(t) x_j(t). \end{aligned} \tag{31}$$

Заметим, что  $x_i(t)$  неотрицательны в силу определения. В силу теоремы 3 усредняющее неравенство (18) устанавливает консенсус, т.е.  $x_i(t) \xrightarrow{t \rightarrow \infty} c \geq 0 \forall i$ , где  $c$  зависит от конкретного решения. Из теоремы 3 также следует, что

$$\Delta_i(t) \xrightarrow[t \rightarrow \infty]{} 0 \quad \forall i \in \mathcal{V}, \tag{32}$$

где невязки  $\Delta_i(t)$  определены формулой (19).

**Шаг 2.** Исследуя структуру невязок  $\Delta_i$ , покажем, что

$$\left\| \xi^i(t+1) - \zeta^i(t) \right\| \xrightarrow[t \rightarrow \infty]{} 0 \quad \forall i, \tag{33}$$

$$\left\| P_i(\xi^i(t)) - \xi^i(t) \right\| \xrightarrow[t \rightarrow \infty]{} 0 \quad \forall i. \tag{34}$$

Векторы  $\xi^i(t)$  (а значит, и  $\zeta^i(t)$ ) ограничены, так как расстояния  $x_i(t)$  от них до  $\xi^0$  ограничены сверху в силу (18). Для алгоритма (21) из неравенства (28) следует, что

$$\Delta_i(t) \geq \left\| \zeta^i(t) - \xi^0 \right\| - \left\| P_i(\zeta^i(t)) - \xi^0 \right\| = \delta_i(\zeta^i(t)). \tag{35}$$

Применяя лемму 4 к  $P = P_i$  из (32) получаем  $\left\| P_i(\zeta^i(t)) - \zeta^i(t) \right\| \xrightarrow[t \rightarrow \infty]{} 0$ , что эквивалентно (33), поскольку  $\xi^i(t+1) = P_i(\zeta^i(t))$ . Чтобы вывести (34), заметим, что

$$0 \leq \left\| P_i(\xi^i(t+1)) - \xi^i(t+1) \right\| \leq \left\| P_i(\zeta^i(t)) - \zeta^i(t) \right\| + \left\| \zeta^i(t) - \xi^i(t+1) \right\| \xrightarrow[t \rightarrow \infty]{} 0.$$

В случае алгоритма (24), заметим, что (31) дает

$$\Delta_i(t) \geq a_{ii}(t) \left( \left\| \xi^i(t) - \xi^0 \right\| - \left\| P_i(\xi^i(t)) - \xi^0 \right\| \right) \geq \eta \delta_i(\xi^i(t)), \tag{36}$$

где  $\eta > 0$  – константа из предположения теоремы. Применяя лемму 4, можно доказать (34), что влечет за собой также (33), поскольку

$$\xi^i(t+1) = \zeta^i(t) + a_{ii}(t) \left( P_i(\xi^i(t)) - \xi^i(t) \right).$$

Доказательство в случае алгоритма (22) объединяет две вышеупомянутые оценки. Во-первых, (29) приводит к неравенству

$$\Delta_i(t) \geq \left\| \bar{\zeta}^i(t) - \xi^0 \right\| - \left\| P_i(\bar{\zeta}^i(t)) - \xi^0 \right\| = \delta_i(\bar{\zeta}^i(t)), \tag{37}$$

которое аналогично (35) и, поскольку  $P_i(\bar{\zeta}^i(t)) = \xi^i(t+1)$ , влечет

$$\left\| \xi^i(t+1) - \bar{\zeta}^i(t) \right\| \xrightarrow[t \rightarrow \infty]{} 0. \tag{38}$$

Во-вторых, (29) также влечет (36), что, в свою очередь, влечет (34). Таким образом,

$$\left\| \bar{\zeta}^i(t) - \zeta^i(t) \right\| \xrightarrow[t \rightarrow \infty]{} 0. \tag{39}$$

Это доказывает (33), так как  $\left\| \xi^i(t+1) - \zeta^i(t) \right\| \leq \left\| \xi^i(t+1) - \bar{\zeta}^i(t) \right\| + \left\| \bar{\zeta}^i(t) - \zeta^i(t) \right\|$ .

Доказательство для алгоритма (23) аналогично доказательству для алгоритма (22). В этом случае (38) выполнено по определению  $\bar{\zeta}^i$  (левая часть равна 0), а неравенство (36) легко следует из (30), поскольку

$$a_{ij}(t) \left( \left\| \xi^j(t) - \xi^0 \right\| - \left\| P_j(\xi^j(t)) - \xi^0 \right\| \right) \geq 0 \quad \forall j \neq i.$$

По аналогии с алгоритмом (22) можно проверить выполнение уравнений (34) и (33).

Таким образом, мы доказали, что при всех алгоритмах, изучаемых в теореме, решения обладают тем свойством, что

$$f^i(t) \stackrel{\Delta}{=} \xi^i(t+1) - \zeta^i(t) = \xi^i(t+1) - \sum_{j \in \mathcal{V}} a_{ij}(t) \xi^j(t) \xrightarrow[t \rightarrow \infty]{} 0.$$

Отметим теперь, что состояния агентов  $\xi^i$ , в силу определения  $f^i(t)$ , могут рассматриваться как траектории усредняющей системы с возмущением:

$$\xi^i(t+1) = \sum_j a_{ij}(t) \xi^j(t) + f^i(t),$$

где  $f^i(t)$  — нечто, сходящееся к нулю. В силу леммы 2 наличие исчезающих на бесконечности возмущений<sup>15</sup> не нарушает синхронизации между агентами:  $\|\xi^i(t) - \xi^j(t)\| \xrightarrow{t \rightarrow \infty} 0$ .

**Шаг 3.** Рассмотрим теперь состояние агента с индексом  $i \in \mathcal{V}$ . Поскольку векторы  $\xi^i(t)$  ограничены, то существует последовательность  $t_r \rightarrow \infty$  такая, что  $\xi^i(t_r) \xrightarrow{r \rightarrow \infty} \xi_* \in \mathbb{R}^d$ . Ввиду синхронизации мы имеем  $\xi^j(t_r) \rightarrow \xi_*$  для каждого  $j \in \mathcal{V}$ . Из свойства (34) следует, что  $P_i(\xi_*) = \xi_* \forall i$ , откуда  $\xi_* \in \Xi_*$ . Остается показать, что  $\xi^i(t) \xrightarrow{t \rightarrow \infty} \xi_*$ . Заметим, что на шаге 1 мы не уточняли выбор  $\xi^0$ , которая может быть произвольной точкой в  $\Xi_*$ . Мы доказали, что для любой такой точки расстояния  $x_i(t) = \|\xi^i(t) - \xi^0\|$  сходятся к некоторому значению  $c$ , зависящему от  $\xi^0$  и начальных условий. В частности, подставляя  $\xi^0 = \xi_*$ , получаем, что существуют пределы

$$x_i(t) = \lim_{t \rightarrow \infty} \|\xi^i(t) - \xi_*\| = c_* \quad \forall i.$$

Вспоминая, что  $x_i(t_r) \xrightarrow{r \rightarrow \infty} 0$ , имеем  $c_* = 0$ , что доказывает достижение консенсуса (25), причем общий предел состояний агентов в (25) равен  $\xi_* \in \Xi_*$ .

### СПИСОК ЛИТЕРАТУРЫ

1. Gordon R., Bender R., Herman G. T. Algebraic Reconstruction Techniques (ART) for three-dimensional electron microscopy and X-ray photography // *Journal of Theoretical Biology*. 1970. Т. 29. No 3. С. 471—481.
2. Гелиг А. Х., Матвеев А. С. Введение в математическую теорию обучаемых распознающих систем и нейронных сетей. СПб.: Изд-во СПбГУ, 2014.
3. Якубович В. А. Некоторые общие теоретические принципы построения обучаемых опознающих систем // *Вычислительная техника и вопросы программирования*. Ленинград: Изд-во ЛГУ. 1965. С. 3—71.
4. Demyanov V. F. Mathematical diagnostics via nonsmooth analysis // *Optimization Methods and Software*. — 2005. Т. 20. No 2/3. С. 197—218.
5. Оптимизационные методы в задачах диагностики / К. И. Ананьев [и др.] // *Вестник СПбГУ. Прикл. матем. Информ. Прогр. упр.* 2011. Т. 10. No 3. С. 3—12.
6. Малоземов В. Н., Плоткин А. В. Строгое полиномиальное отделение двух множеств // *Вестник СПбГУ. Математика. Механика. Астрономия*. 2019. Т. 6, No 2. С. 232—240.
7. Combettes P. L. The Foundations of Set Theoretic Estimation // *Proceedings of IEEE*. 1993. Т. 81. No 2. С. 182—208.
8. Петров И. П., Тимофеев А. В. Конечно-сходящиеся рекуррентные алгоритмы решения целевых неравенств при наличии ограничений // *Ж. вычисл. матем. и матем. физ.* 1975. Т. 15. No 6. С. 1582—1588.
9. Фомин В. Н., Фрадков А. Л., Якубович В. А. Адаптивное управление динамическими объектами. — М.: Наука, 1981.
10. Kaczmarz S. Angenäherte Auflösung von Systemen linearer Gleichungen // *Bulletin International de l'Académie Polonaise des Sciences et des Lettres*. 1937. Т. 35. С. 355—357.
11. Cimmino G. Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari // *La Ricerca Scientifica*. 1938. Т. 2. No 9. С. 326—333.
12. Von Neumann J. Functional operators, Vol. II. The geometry of orthogonal spaces. — Princeton, NJ: Princeton University Press, 1950. — (Annals of Mathematical Studies). — Reprint of mimeographed lecture notes first distributed in 1933.
13. Еремин И. И. Релаксационный метод решения систем неравенств с выпуклыми функциями в левых частях // *Докл. АН СССР*. 1965. Т. 160. No 5. С. 994—996.
14. Брэгман Л. М. Нахождение общей точки выпуклых множеств методом последовательного проектирования // *Докл. АН СССР*. 1965. Т. 162. No 3. С. 487—490.

<sup>15</sup> Формально, лемма 2 была сформулирована для случая агентов со скалярными состояниями  $x_i \in \mathbb{R}$ , однако она, очевидно, сохраняет свою силу и для многомерного случая (достаточно рассмотреть проекции состояний агентов на каждую координатную ось).

15. Гурин Л. Г., Поляк Б. Т., Райк Э. В. Методы проекций для отыскания общей точки выпуклых множеств // Ж. вычисл. матем. и матем. физ. 1967. Т. 7. No 6. С. 1211—1228.
16. Еремин И. И., Мазуров В. Д. Итерационный метод решения задачи выпуклого программирования // Докл. АН СССР. 1966. Т. 170. No 1. С. 57—60.
17. Бердникова Е. А., Ерёмин И. И., Попов Л. Д. Распределенные фейеровские процессы для систем линейных неравенств и задач линейного программирования // Автомат. и телемех. 2004. No 2. С. 16—32.
18. Ерёмин И. И., Попов Л. Д. Замкнутые фейеровские циклы для несовместных систем выпуклых неравенств // Изв. вузов. Матем. 2008. No 1. С. 11—19.
19. Ерёмин И. И., Попов Л. Д. Фейеровские процессы в теории и практике: обзор последних результатов // Изв. вузов. Матем. 2009. No 1. — С. 44—65.
20. Брэгман Л. М. Релаксационный метод нахождения общей точки выпуклых множеств и его применение для решения задач выпуклого программирования // Ж. вычисл. матем. и матем. физ. 1967. Т. 7, No 3. С. 620—631.
21. Васин И. И., Еремин И. И. Операторы и итерационные процессы фейеровского типа: теория и приложения. — Ижевск: Регулярная и хаотическая динамика, 2005.
22. Escalante R., Raydan M. Alternating Projection Methods. — Society for Industrial, Applied Mathematics Philadelphia, 2011.
23. Bauschke H. H., Borwein J. M. On Projection Algorithms For Solving Convex Feasibility Problems // SIAM Review. 1996. Т. 38. No 3. С. 367—426.
24. Lewis A. S., Malick J. Alternating Projections on Manifolds // Mathematics of Operations Research. 2008. Т. 33. No 1. С. 216—234.
25. Nedic A., Ozdaglar A., Parrilo P. Constrained consensus and optimization in multi-agent networks // IEEE Trans. Autom. Control. 2010. Т. 55. No 4. С. 922—938.
26. Shi G., Johansson K., Hong Y. Reaching an optimal consensus: dynamical systems that compute intersections of convex sets // IEEE Trans. Autom. Control. 2013. Т. 58. No 3. С. 610—622.
27. Solving a system of linear equations: from centralized to distributed algorithms / P. Wang [и др.] // Ann. Rev. Control. 2019. Т. 47. С. 306—322.
28. Проскурников А. В. Усредняющие алгоритмы и неравенства в задачах многоагентного управления и моделирования. Диссертация д.ф.-м.н. — Санкт-Петербург, 2021.
29. Proskurnikov A., Cao M. Differential inequalities in multi-agent coordination and opinion dynamics modeling // Automatica. 2017. Т. 85. С. 202—210.
30. Proskurnikov A., Calafiore G., Cao M. Recurrent averaging inequalities in multi-agent control and social dynamics modeling // Ann. Rev. Control. 2020. Т. 49. С. 95—112.
31. Балакришнан А. В. Прикладной функциональный анализ. — М.: Наука, 1980.
32. Peterson B., Olinick M. Leontief models, Markov chains, substochastic matrices, and positive solutions of matrix equations // Mathematical Modelling. 1982. Т. 3. No 3. С. 221—239.
33. Polyak B., Tremba A. Regularization-based solution of the PageRank problem for large matrices // Autom. Remote Control. 2012. Т. 73. No 11. С. 1877—1894.
34. Гасников А. В., Дмитриев Д. Ю. Об эффективных рандомизированных алгоритмах поиска вектора PageRank // Ж. вычисл. матем. и матем. физ. 2015. Т. 55. No 3. С. 355—371.
35. Sznajder R. Kaczmarz Algorithm Revisited // Technical Transactions Fundamental Sciences. 2015. No 2. С. 248—254.
36. Deutsch F. The Angle Between Subspaces of a Hilbert Space // Approximation Theory, Wavelets and Applications / под ред. S. P. Singh. — Dordrecht: Springer, 1995. С. 107—130.
37. A block projection method for sparse matrices / M. Arioli [и др.] // SIAM Journal on Scientific and Statistical Computing. 1988. Т. 13. С. 326—333.
38. Agmon S. The Relaxation Method For Linear Inequalities // Canadian Journal of Mathematics. 1954. Т. 6. С. 382—392.
39. Motzkin T. S., Schoenberg I. J. The Relaxation Method For Linear Inequalities // Canadian Journal of Mathematics. 1954. Т. 6. С. 393—404.

40. Ерёмин И. И. Обобщение релаксационного метода Моцкина—Агмона // Успехи мат. наук. 1965. Т. 20, No 2. С. 183—187.
41. Gurin L., Polyak B., Raik E. The method of projections for finding the common point of convex sets // USSR Comp. Math. Math. Phys. 1967. Т. 7. No 6. С. 1—24.
42. Ерёмин И. И. О скорости сходимости в методе фейеровских приближений // Матем. заметки. 1968. Т. 4. No 1. С. 53—61.
43. Ерёмин И. И. Фейеровские отображения и задача выпуклого программирования // Сиб. матем. журн. 1969. Т. 10. No 5. С. 1034—1047.
44. Ерёмин И. И. Применение метода фейеровских приближений к решению задач выпуклого программирования с негладкими ограничениями // Ж. вычисл. матем. и матем. физ. 1969. Т. 9. No 5. С. 1153—1160.
45. Брэгман Л. М. Релаксационный метод нахождения общей точки выпуклых множеств и его применение для задач оптимизации // Докл. АН СССР. 1966. Т. 171. No 5. С. 1019—1022.
46. Lin P., Ren W. Constrained consensus in unbalanced networks with communication delays // IEEE Trans. Autom. Control. 2014. Т. 59. No 3. С. 775—781.
47. Notarstefano G., Notarnicola I., Camisa A. Distributed Optimization for Smart Cyber-Physical Networks // Foundations and Trends in Systems and Control. 2019. Т. 7. No 3. С. 253—383.
48. Гасников А. В. Современные численные методы оптимизации. Универсальный градиентный спуск. — М.: МФТИ, 2018.
49. Fejér L. Über die Lage der Nullstellen von Polynomen, die aus Minimumforderungen gewisser Art entspringen // Math. Ann. 1922. No 85. С. 41—48.
50. Fullmer D., Morse A. S. A distributed algorithm for computing a common fixed point of a finite family of parac-  
ontractions // IEEE Trans. Autom. Control. 2018. Т. 63. No 9. С. 2833—2843.
51. Vasin V. V., Eremin I. I. Operators and Iterative Processes of Fejér Type: Theory and Applications. — De Gruyter, 2009.
52. Rzevski G., Skobelev P. Managing complexity. — Southampton, UK: WIT Press, 2014.
53. Wooldridge M. An Introduction To Multiagent Systems. — Chichester, England: John Wiley & Sons Ltd., 2002.
54. Проскурников А. В., Фрадков А. Л. Задачи и методы сетевого управления // Автоматика и телемеханика. 2016. No 10. С. 3—39.
55. Граничин О. Н. Как действительно устроены сложные информационно-управляющие системы? // Сто-  
хастическая оптимизация в информатике. 2016. Т. 12. No 1. С. 3—19.
56. Proskurnikov A. V., Granichin O. N. Evolution of clusters in large-scale dynamics networks // Cybern. Phys. 2018. Т. 7. No 3. С. 102—129.
57. Проблемы сетевого управления / А. Л. Фрадков [и др.]. — Ижевск: ИКИ, 2015.
58. Marik V., Gorodetsky V., Skobelev P. Multi-agent technology for industrial applications: barriers and trends // Proc. of IEEE Int. Conf. Syst., Man, Cybern. 2020. С. 1980—1987.
59. Агаев Р. П., Чеботарев П. Ю. Сходимость и устойчивость в задачах согласования характеристик // Управ-  
ление большими системами. 2010. Т. 30.1. С. 470—505.
60. Proskurnikov A., Tempo R. A tutorial on modeling and analysis of dynamic social networks. Part I // Ann. Rev. Control. 2017. Т. 43. С. 65—79.
61. Proskurnikov A., Tempo R. A tutorial on modeling and analysis of dynamic social networks. Part II // Ann. Rev. Control. 2018. Т. 45. С. 166—190.
62. Seneta E. Non-negative matrices and Markov chains. — New York: Springer-Verlag, 1981.
63. Leizarowitz A. On infinite products of stochastic matrices // Linear Alg. Appl. 1992. Т. 168. С. 189—219.
64. Bolouki S., Malhame R. P. Consensus algorithms and the decomposition-separation theorem // IEEE Trans. Au-  
tom. Control. 2016. Т. 61. No 9. С. 2357—2369.
65. Touri B., Nedic A. On backward product of stochastic matrices // Automatica. 2012. Т. 48. No 8. С. 1477—1488.
66. Chebotarev P., Agaev R. Forest matrices around the Laplacian matrix // Linear Alg. Appl. 2002. Т. 356. С. 253—274.
67. Ren W., Beard R. Distributed consensus in multi-vehicle cooperative control: theory and applications. — London: Springer-Verlag, 2008.
68. Ren W., Cao Y. Distributed coordination of multi-agent networks. — Springer, 2011.

69. Shi G., Johansson K. Robust consensus for continuous-time multi-agent dynamics // *SIAM J. Control Optim.* 2013. Т. 51. No 5. С. 3673—3691.
70. Proskurnikov A., Cao M. Modulus consensus in discrete-time signed networks and properties of special recurrent inequalities // *Proc. of IEEE Conf. Decision and Control.* 2017. С. 2003—2008.
71. Proskurnikov A., Matveev A. Popov-type criterion for consensus in nonlinearly coupled networks // *IEEE Trans. Cybernetics.* 2015. Т. 45. No 8. С. 1537—1548.
72. You K., Song S., Tempo R. A networked parallel algorithm for solving linear algebraic equations // *Proc. IEEE Conf. Decision and Control.* 2016. С. 1727—1732.
73. Mou S., Liu J., Morse A. A distributed algorithm for solving a linear algebraic equation // *IEEE Trans. Autom. Control.* 2015. Т. 60. No 11. С. 2863—2878.
74. Olshevsky A., Tsitsiklis J. Convergence speed in distributed consensus and averaging // *SIAM Rev.* 2011. Т. 53. No 4. С. 747—772.
75. Proskurnikov A. V., Calafiore G. C. Delay Robustness of Consensus Algorithms: Continuous-Time Theory // *IEEE Trans. Autom. Control.* 2023. Т. 68. No 9. С. 5301—5316.
76. Shi G., Johansson K. Randomized optimal consensus of multi-agent systems // *Automatica.* 2012. Т. 48. Вып. 12. С. 3018—3030.
77. Steinerberger S. Randomized Kaczmarz Converges Along Small Singular Vectors // *SIAM Journal on Matrix Analysis and Applications.* 2021. Т. 42. No 2. С. 608— 615.
78. Sarlette A., Sepulchre R. Consensus Optimization on Manifolds // *SIAM Journal on Control and Optimization.* — 2009. Т. 48, No 1. С. 56—76.

# ALTERNATING PROJECTION METHOD FOR INTERSECTION OF CONVEX SETS, MULTI-AGENT CONSENSUS ALGORITHMS, AND AVERAGING INEQUALITIES

A. V. Proskurnikov<sup>a,\*</sup>, I. S. Zabarianska<sup>b,\*\*</sup>

<sup>a</sup>*Politecnico di Torino,*

*Corso Duca degli Abruzzi, 24, Turin, 10129 Italy*

<sup>b</sup>*St. Petersburg State University,*

*14th line Vasilyevskiy Island, 29, St. Petersburg, 199178 Russia*

<sup>\*</sup>*e-mail: avp1982@gmail.com*

<sup>\*\*</sup>*e-mail: akshiira@yandex.ru*

Received 03 November, 2023

Revised 11 November, 2023

Accepted 20 November, 2023

**Abstract.** The history of the alternating projection method for finding a common point of several convex sets in Euclidean space goes back to the well-known Kaczmarz algorithm for solving systems of linear equations, which was devised in the 1930s and later found wide applications in image processing and computed tomography. An important role in the study of this method was played by I.I. Eremin's, L.M. Bregman's, and B.T. Polyak's works, which appeared nearly simultaneously and contained general results concerning the convergence of alternating projections to a point in the intersection of sets, assuming that this intersection is nonempty. In this paper, we consider a modification of the convex set intersection problem that is related to the theory of multi-agent systems and is called the constrained consensus problem. Each convex set in this problem is associated with a certain agent and, generally speaking, is inaccessible to the other agents. A group of agents is interested in finding a common point of these sets, that is, a point satisfying all the constraints. Distributed analogues of the alternating projection method proposed for solving this problem lead to a rather complicated nonlinear system of equations, the convergence of which is usually proved using special Lyapunov functions. A brief survey of these methods is given, and their relation to the theorem ensuring consensus in a system of averaging inequalities recently proved by the first author is shown (this theorem develops convergence results for the usual method of iterative averaging as applied to the consensus problem).

**Keywords:** alternating projection method, convex programming, Fejér mappings, distributed algorithms, consensus, multi-agent systems.