Современная математика. Фундаментальные направления. Том 64, № 4 (2018). С. 616–636

DOI: 10.22363/2413-3639-2018-64-4-616-636 УДК 517.98+519.65

# ИТЕРАЦИОННАЯ ПРОЕКЦИОННАЯ СХЕМА В РЕАЛЬНОМ ВРЕМЕНИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ОБ ОБЩЕЙ НЕПОДВИЖНОЙ ТОЧКЕ И ЕЕ ПРИЛОЖЕНИЯ

© 2018 г. **А. ГИБАЛИ, Д. ТЕЛЛЕР**

Аннотация. В этой работе мы рассматриваем задачу об общей неподвижной точке (CFPP) с деми-сжимающими операторами и ее частный случай, выпуклую задачу о допустимости (CFP) в вещественных гильбертовых пространствах. Руководствуясь недавними результатами, полученными Ордонесом и др. в работе [35] и в области алгоритмов в реальном времени в общем, например, в [20, 21, 30], где с самого начала нам недоступны целые наборы операторов/множеств, которые затем получаются постепенно, мы предлагаем итерационную схему в реальном времени для решения задач об общей неподвижной точке (CFPP) и выпуклых задач о допустимости (CFP), в которой участвующие операторы/множества появляются со временем. Такая схема способна работать с любыми блоками данных и для любого конечного числа итераций с последовательным переходом к следующему блоку.

Схема основана на недавнем результате, описанном в работе Райха и Заласа [37] и известном как процедура модулярного строкового усреднения (MSA). Сходимость схемы следует из [37] и других классических результатов в теории неподвижных точек и области вариационных неравенств, например, [34].

Также в работе представлены вычислительные эксперименты для линейных и нелинейных задач о допустимости в приложении к восстановлению изображений. Они демонстрируют справедливость и потенциальную применимость нашей схемы, например, в условиях реального времени.

## ОГЛАВЛЕНИЕ



## 1. Введение

В этой работе мы имеем дело с *задачей об общей неподвижной точке* (CFPP) и ее частным случаем, *выпуклой задачей о допустимости* (CFP) в вещественных гильбертовых пространствах $\mathcal{H}$. Даны операторы $U_i : \mathcal{H} \to \mathcal{H}$ при $i \in I := \{1, 2, \ldots, m\}$ с непустыми множествами неподвижных точек. Задача об общей неподвижной точке состоит в нахождении точки $x^* \in \mathcal{H}$, такой, что

$$x^* \in \bigcap_{i=1}^{m} \operatorname{Fix}(U_i). \tag{1.1}$$

Выпуклая задача о допустимости является частным случаем задачи об общей неподвижной точке. В этом случае имеем $m$ непустых, замкнутых и выпуклых множеств $C_i \subseteq \mathcal{H}$ при $i \in I$. Теперь задача заключается в нахождении такой точки $x^* \in \mathcal{H}$, что

$$x^* \in \bigcap_{i=1}^{m} C_i \neq \varnothing. \tag{1.2}$$

Очевидно, что если мы выберем

$$U_i = P_{C_i}$$

для всех $i \in I$, где $P_{C_i}$ обозначает ортогональную проекцию на $i$-е множество $C_i$ (что будет пояснено далее) в CFPP (1.1), то мы получим CFP (1.2).

CFPP и CFP служат основными средствами моделирования при построении многих важных реальных задач, например, при формировании изображений, в сенсорных сетях, при составлении плана лечения лучевой терапией, повышении разрешающей способности и многих других; см., например, [5, 15]. Одна из самых ранних итерационных процедур для решения задач CFPP, см., например, [34], имеет следующий общий вид: выбираем произвольную начальную точку $x^0 \in \mathcal{H}$; получив текущую итерацию $x^k$, рассчитываем следующую итерацию $x^{k+1}$ таким образом:

$$x^{k+1} = T(x^k), \tag{1.3}$$

где оператор $T : \mathcal{H} \to \mathcal{H}$ фиксирован и зависит от семейства операторов

$$\{U_i : \mathcal{H} \to \mathcal{H} \mid i \in I\}.$$

Более общая схема для неподвижной точки позволяет включить семейство операторов

$$\{T_k : \mathcal{H} \to \mathcal{H}\}_{k=0}^{\infty};$$

см., например, обобщенный метод Опиала [11, п. 3.6]. Итерационная процедура формулируется следующим образом: выбираем произвольную начальную точку $x^0 \in \mathcal{H}$; получив текущую итерацию $x^k$, рассчитываем следующую итерацию $x^{k+1}$ так:

$$x^{k+1} = T_k(x^k), \tag{1.4}$$

где семейство операторов $\{T_k\}_{k=0}^{\infty}$ зависит от $\{U_i\}_{i\in I}$ и может иметь различные алгоритмические структуры, например:

1. *циклическую* (с релаксацией): $\alpha_k \in [\varepsilon, 2-\varepsilon]$ при $\varepsilon > 0$:
$$T_k = U_{i(k)}, \quad \text{где } i(k) = (k \bmod m) + 1;$$
2. *совместную*:
$$T_k := \frac{1}{m}\sum_{i=1}^{m} U_i;$$
3. *композиционную*:
$$T_k := \prod_{i=1}^{m} U_i;$$
4. «*жадную*» (наиболее удаленную):
$$T_k := U_{i_k}, \quad \text{где } i_k = \operatorname*{argmax}_{i\in I} \operatorname{dist}(\cdot, \operatorname{Fix}(U_i)),$$
здесь $\operatorname{dist}(\cdot,\cdot)$ — это *функция расстояния* между точкой и множеством.

Возвращаясь к выпуклой задаче о допустимости (CFP), хотелось бы обратить внимание на класс проекционных методов. В 1930-х годах Стефан Качмаж [29] и Джанфранко Чиммино [18] представили итерационные проекционные методы для решения систем линейных неравенств

$$Ax \leqslant b,$$

где $A \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^n$ и $b \in \mathbb{R}^m$. Как оказалось, данная задача может быть легко сведена к эквивалентной ей CFP следующим образом: обозначим через $A^i$ и $b_i$ $i$-е строку и элемент $A$ и $b$, соответственно. Определим понятие *полупространства*:

$$H_i^- := \{z \in \mathbb{R}^n \mid \langle A^i, z\rangle \leqslant b_i\} \tag{1.5}$$

и затем получим:

$$Ax \leqslant b \Leftrightarrow x \in \bigcap_{i=1}^{m} H_i^-. \tag{1.6}$$

Методы Качмажа [29] и Чиммино [18] используют ортогональные проекции (отражения) на полупространства $H_i^-$ последовательно и совместно, соответственно. Этим, собственно, и характеризуется класс проекционных методов, которые, будучи итерационными процедурами, используют проекции различных типов на множества с учетом того факта, что проекция на пересечение множеств являет собой довольно сложную вычислительную задачу, в то время как проекции на отдельные множества намного проще осуществимы. Именно поэтому эти методы успешно применяются во многих реальных приложениях, и даже были названы «Swiss Army knives», см. [6]. В последние десятилетия класс проекционных методов активно развивался, эти методы оказались способными решать общие выпуклые задачи о допустимости (1.2). Также они включают в себя различные алгоритмические структуры, такие как последовательную, совместную, блочно-итерационную, усреднение по строкам и т. д., см. [15], а также [10, 11, 17, 22, 23]).

Ордонес и др. в работе [35] изучают разреженную и переопределенную систему линейных уравнений

$$Ax = b \quad (A \in \mathbb{R}^{m\times n} \text{ с } m \sim 10^3 \text{ и } n \sim 10^9)$$

большой размерности, возникающую в области протонной вычислительной томографии (pCT). Исследуя эту задачу, авторы вводят два проекционных метода в режиме реального времени, метод диагонально ослабленных ортогональных проекций (DROP) [2] и метод координатно-усредненных строковых проекций (CARP) [25]. Они экспериментально демонстрируют, что DROP и CARP в режиме реального времени работают гораздо «лучше и быстрее», чем другие проекционные методы, см., например, [36].

Итак, исходя из [35] и результатов, полученных в области алгоритмов в реальном времени, например [20, 21, 30], нашей задачей в этой работе является введение новой итерационной схемы для неподвижной точки типа (1.3) или (1.4) для решения задач об общей неподвижной точке и выпуклых задач о допустимости. Сосредоточим наше внимание на том случае, когда с самого начала нам недоступны целые наборы операторов/множеств, и мы получаем их постепенно. Следовательно, нам необходимо придумать такую итерационную схему в реальном времени, которая будет способна работать с сегментами набора и включать в себя новый набор, когда таковой появится. В CFPP главная идея состоит в том, что операторы $U_i$ при $i \in I$ представлены в виде блоков

$$I = I_1 \cup I_2 \cup \ldots \cup I_M, \quad 1 \leqslant M \leqslant m,$$

и последовательны по времени. Возьмем процедуру Райха и Заласа [37], *модулярное строковое усреднение* (MSA), и покажем, как она может быть применена в вышеописанном случае. Некоторые численные эксперименты демонстрируют потенциальную применимость и преимущества предлагаемого метода для линейных и нелинейных задач о допустимости в режиме реального времени.

Краткое содержание данной работы таково: в разделе 2 представлены определения и понятия, необходимые в дальнейшем; в разделе 3 вводятся и анализируются новые итерационные схемы в реальном времени для задач о неподвижных точках и задач о допустимости; далее в разделе 4 будет представлено несколько численных примеров для линейных и нелинейных задач о допустимости и будут обоснованы справедливость и потенциальная применимость новой схемы, которая может быть использована, например, для задач в реальном времени; и, наконец, в разделе 5 представлены выводы и направления для дальнейших исследований.

## 2. Предварительные сведения

Всюду далее в этой работе $\mathcal{H}$ — это вещественное гильбертово пространство со скалярным произведением $\langle\cdot,\cdot\rangle$ и индуцированной нормой $\|\cdot\|$. Будем использовать записи $x^k \rightharpoonup x$ и $x^k \to x$ для обозначения слабой и сильной сходимостей последовательности $\left\{x^k\right\}_{k=0}^{\infty}$ к $x$, соответственно.

Вспомним теперь несколько определений и свойств некоторых классов операторов. Их, наряду со многими другими, можно найти, например, в блестящей работе Цегельского [11].

**Определение 2.1.** Пусть $U : \mathcal{H} \to \mathcal{H}$ — некоторый оператор.

- *Множество неподвижных точек* оператора $U$, обозначаемое через $\mathrm{Fix}(U)$, определяется так:

$$\mathrm{Fix}(U) := \{x \in \mathcal{H} \mid U(x) = x\}. \tag{2.1}$$

- Оператор $U$ называется *срезающим*, если для любого $x \in \mathcal{H}$ и любого $z \in \mathrm{Fix}(U)$
$$\langle z - U(x), x - U(x)\rangle \leqslant 0. \tag{2.2}$$
- Оператор $U$ называется *нерастягивающим* (NE), если для любых $x, y \in \mathcal{H}$
$$\|U(x) - U(y)\| \leqslant \|x - y\|. \tag{2.3}$$
- Оператор $U$ называется *квази-нерастягивающим* (QNE), если для всех $(x, q) \in \mathcal{H} \times \mathrm{Fix}(U)$
$$\|U(x) - q\| \leqslant \|x - q\|. \tag{2.4}$$
- Оператор $U$ с $\mathrm{Fix}(U) \neq \varnothing$ называется *$\rho$-деми-сжимающим* (см., например, [32]) при $\rho \in [-1, 0)$, если для всех $(x, z) \in \mathcal{H} \times \mathrm{Fix}(U)$
$$\|U(x) - z\|^2 \leqslant \|x - z\|^2 - \rho\|U(x) - x\|^2. \tag{2.5}$$
- Оператор $U$ называется *$\rho$-сильно квази-нерастягивающим* при $\rho \geqslant 0$, если для всех $(x, z) \in \mathcal{H} \times \mathrm{Fix}(U)$
$$\|U(x) - z\|^2 \leqslant \|x - z\|^2 - \rho\|U(x) - x\|^2. \tag{2.6}$$
Если $\rho > 0$, то оператор $U$ называется *сильно квази-нерастягивающим*.
- При $\alpha \in [0, \infty]$ оператор
$$U_\alpha := Id + \alpha(U - Id)$$
называется *$\alpha$-релаксацией оператора* $U$, а $\alpha$ называется *параметром релаксации*. Легко видеть, что для каждого $\alpha \neq 0$
$$\mathrm{Fix}(U) = \mathrm{Fix}(U_\alpha). \tag{2.7}$$
- Оператор $U$ называется *усредняющим* [3] (см. также [9]), если существуют нерастягивающий оператор $N : \mathcal{H} \to \mathcal{H}$ и число $c \in (0, 1)$, такие, что
$$U = (1 - c)Id + cN, \tag{2.8}$$
где $Id$ — единичный оператор в $\mathcal{H}$.
- Квази-нерастягивающий оператор $U$ называется *деми-замкнутым* в точке $y \in \mathcal{H}$, если для любой последовательности $\{x^k\}_{k=0}^{\infty} \subset \mathcal{H}$ имеем
$$\left.\begin{array}{l} x^k \rightharpoonup \overline{x} \\ U(x^k) \to y \end{array}\right\} \Longrightarrow U(\overline{x}) = y. \tag{2.9}$$
- Квази-нерастягивающий оператор $U$ называется *приближенно стягивающим*, если для любой ограниченной последовательности $\{x^k\}_{k=0}^{\infty} \subseteq \mathcal{H}$ имеем
$$\lim_{k\to\infty} \|U(x^k) - x^k\| = 0 \quad \Longrightarrow \quad \lim_{k\to\infty} \mathrm{dist}(x^k, \mathrm{Fix}(U)) = 0. \tag{2.10}$$
Подробнее об этом классе операторов можно прочесть, например, в [13].

Следующая теорема является частью нашего анализа и показывает связь между двумя классами операторов, описанных выше.

**Теорема 2.1.** *Пусть $U : \mathcal{H} \to \mathcal{H}$ — оператор с неподвижной точкой, и пусть $\alpha \in (0, 2]$. Тогда $U$ — это срезающий оператор, если и только если его $U_\alpha$ является $(2-\alpha)/\alpha$-сильно квази-нерастягивающим.*

*Доказательство.* См., например, либо [19, Proposition 2.3(ii)], либо [11, Theorem 2.1.39]. □

Следующий принцип известен как принцип деми-замкнутости [8].

**Принцип деми-замкнутости.** *Пусть $\mathcal{H}$ — это вещественное гильбертово пространство, $C \subseteq \mathcal{H}$ — замкнутое и выпуклое множество, и пусть $S : C \to \mathcal{H}$ — нерастягивающее отображение. Тогда $Id - S$ деми-замкнут в $y \in \mathcal{H}$.*

Другим важным результатом, необходимым для нашего анализа, является следующее утверждение (см. [14, утверждение 4.1]).

**Предложение 2.1.** *Пусть $U : \mathcal{H} \to \mathcal{H}$ — это квази-нерастягивающий оператор. Тогда верны следующие утверждения*:

1. *Если $U$ является приближенно стягивающим, то $U - Id$ деми-замкнут в* $0$;
2. *Если* $\dim(\mathcal{H}) < \infty$ *($\mathcal{H}$ конечномерно) и $U - Id$ деми-замкнут в $0$, то $U$ является приближенно стягивающим.*

**Определение 2.2.** Пусть $C_i \subseteq \mathcal{H}$ при $i \in I$ — это замкнутые и выпуклые множества с непустым пересечением

$$\mathcal{C} := \bigcap_{i \in I} C_i \neq \varnothing.$$

Семейство множеств $\mathfrak{C} := \{C_i \mid i \in I\}$ называется *ограниченно регулярным*, если для любой ограниченной последовательности $\{x^k\}_{k=0}^{\infty} \subseteq \mathcal{H}$ имеем

$$\lim_{k \to \infty} \max_{i \in I} \operatorname{dist}(x^k, C_i) = 0 \quad \Longrightarrow \quad \lim_{k \to \infty} \operatorname{dist}(x^k, \mathcal{C}) = 0. \tag{2.11}$$

Следующее утверждение (см. [4, утверждение 5.4 (iii), следствие 5.14, следствие 5.22]) дает условия, гарантирующие ограниченную регулярность семейства множеств.

**Предложение 2.2.** *Пусть $C_i \subseteq \mathcal{H}$ при $i \in I$ — замкнутые и выпуклые множества с непустым пересечением*:

$$\mathcal{C} := \bigcap_{i=1}^{m} C_i.$$

*Если выполнено одно из следующих условий*:

1. $\dim(\mathcal{H}) < \infty$,
2. $\operatorname{int}(\mathfrak{C}) \neq \varnothing$,
3. *всякое $C_i$ является полупространством,*

*то семейство множеств $\mathfrak{C} := \{C_i \mid i \in I\}$ ограниченно регулярно.*

Теперь вспомним метрические проекции на замкнутые и выпуклые множества. Пусть $C \subset \mathcal{H}$. Для каждой точки $x \in \mathcal{H}$ существует единственная ближайшая точка из $C$, обозначенная $P_C(x)$, и такая, что

$$\|x - P_C(x)\| \leqslant \|x - y\| \ \text{ для всех } y \in C. \tag{2.12}$$

Отображение $P_C : \mathcal{H} \to C$ называется *метрической проекцией* пространства $\mathcal{H}$ на $C$ и является нерастягивающим отображением пространства $\mathcal{H}$ на $C$ (на самом деле, FNE (следовательно, срезающим), см. [11, теорема 2.2.21]). Отображение $P_C$ характеризуется следующими двумя свойствами (см. [24, п. 3]):

$$P_C(x) \in C \tag{2.13}$$

(следовательно, $\operatorname{Fix}(P_C) = C$) и

$$\langle x - P_C(x), P_C(x) - y \rangle \geqslant 0 \text{ для всех } x \in \mathcal{H},\ y \in C, \tag{2.14}$$

и если $C$ — гиперплоскость, то (2.14) превращается в равенство.

Другим важным типом проекций является субградиентная проекция (также срезающий оператор, см., например, [11, лемма 4.2.5] и [5]). Такие проекции крайне важны в случаях, когда выпуклое множество $C$ представляется в виде подуровневых множеств выпуклой функции $g : \mathcal{H} \to \mathbb{R}$, т. е.

$$C := \{x \in \mathcal{H} \mid g(x) \leqslant 0\}.$$

Теперь, когда мы дали определения различным типам проекций, а также различным классам операторов, необходимо вспомнить два частных типа проекционных методов: последовательный (предложенный Качмажем, также известный как *метод последовательных ортогональных проекций* (SOP) [1], *метод проекции на выпуклые пространства* (POCS) [4] и *алгебраический способ перестраивания* (ART) [26] в линейных случаях) и блочно-итерационные методы (полностью совместные, если есть только один блок, в линейном случае сводятся к методу Чиммино [18]). С этой целью рассмотрим выпуклую задачу о допустимости с непустыми, замкнутыми и выпуклыми множествами $C_i \subseteq \mathcal{H}$ при $i \in I$. Следующие определения управляющих последовательностей, $\{i(\nu)\}_{\nu=0}^{\infty}$, обуславливают порядок, в котором участвуют ортогональные проекции на множества $C_i$, $i \in I$ и, следовательно, указывают структуру алгоритма.

**Определение 2.3.**

1. Последовательность $\{i(\nu)\}_{\nu=0}^{\infty}$ называется *циклично управляющей*, если
$$i(\nu) = (\nu \bmod m) + 1,$$
где $m$ — это количество множеств в (1.2).
2. Последовательность $\{i(\nu)\}_{\nu=0}^{\infty}$ называется *почти циклично управляющей* на множестве $I = \{1, 2, \ldots, m\}$ если
$$i(\nu) \in I \text{ для всех } \nu \geqslant 0,$$
и существует целое $Q \geqslant m$ (называется *почти циклической константой*), такое, что
$$I \subseteq \{i(\nu+1), i(\nu+2), \ldots, i(\nu+Q)\} \text{ для всех } \nu \geqslant 0. \tag{2.15}$$
3. Последовательность $\{i(\nu)\}_{\nu=0}^{\infty}$ называется *удаленно управляющей*, если она получается указанием $i(\nu)$, такого что
$$\operatorname{dist}(x^\nu, C_{i(\nu)}) = \max\{\operatorname{dist}(x^\nu, C_i) \mid i \in I\}. \tag{2.16}$$
4. Последовательность $\{i(\nu)\}_{\nu=0}^{\infty}$ называется *произвольно управляющей*, если $i(\nu) \in I$ выбирается произвольно и определяется независимо в соответствии с заданным распределением вероятности $\{p_i\}$.

Теперь представим последовательный и совместный проекционные методы для решения выпуклых задач о допустимости.

**Алгоритм 2.1** (Метод SOP).
***Дано***. *Пусть $x^0 \in \mathcal{H}$ — произвольная начальная точка.*
***Итерационный шаг***. *Имея текущую итерацию $x^k$, рассчитаем следующую итерацию таким образом*:
$$x^{k+1} = x^k + \lambda_k \left(P_{C_{i(\nu)}}(x^k) - x^k\right), \tag{2.17}$$
*где $P_{C_{i(\nu)}}$ обозначает ортогональную проекцию на множество $C_{i(\nu)}$, $\lambda_k \in [\varepsilon_1, 2-\varepsilon_2]$ для всех $k \geqslant 0$ и $\varepsilon_1, \varepsilon_2 > 0$. Управляющая последовательность $\{i(\nu)\}_{\nu=0}^{\infty}$ циклична на $I$.*

Для следующего алгоритма необходимо определить некоторые термины. Вектор
$$\omega = (\omega(i))_{i \in I}$$
называется *весовым вектором*, если
$$\omega(i) \geqslant 0 \text{ для всех } i \in I$$
и
$$\sum_{i \in I} \omega(i) = 1.$$
Имея весовой вектор $\omega$, можем дать определение *выпуклой комбинации*:
$$P_\omega(x) := \sum_{i \in I} \omega(i) P_{C_i}.$$
Последовательность весовых векторов $\{\omega^k\}_{k=0}^{\infty}$ называется *правильной*, если для любого $i \in I$ существует бесконечно много значений $k$, для которых $\omega^k(i) > 0$.

**Алгоритм 2.2** (Метод блочного типа).
***Дано***. *Пусть $x^0 \in \mathcal{H}$ — произвольная начальная точка.*
***Итерационный шаг***. *Имея текущую итерацию $x^k$, рассчитаем следующую итерацию таким образом*:
$$x^{k+1} = x^k + \lambda_k \left(P_{\omega^k}(x^k) - x^k\right), \tag{2.18}$$
*где $\{\omega^k\}_{k=0}^{\infty}$ — это правильная последовательность весовых векторов, а $\lambda_k \in [\varepsilon_1, 2-\varepsilon_2]$ для всех $k \geqslant 0$ и $\varepsilon_1, \varepsilon_2 > 0$.*

Чтобы проиллюстрировать несколько типов проекционных методов для решения выпуклой задачи о допустимости, ограничимся лишь линейной задачей о допустимости, которая является системой линейных уравнений

$$Ax = b,$$

в которой $A \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^n$ и $b \in \mathbb{R}^m$. Обозначим через $A^i$ и $b_i$ $i$-е строку и элемент $A$ и $b$, соответственно, и зададим $i$-ю гиперплоскость

$$H_i = \{z \in \mathbb{R}^n \mid \langle A^i, z\rangle = b_i\}.$$

Иллюстрации этих и других методов представлены на рис. 1, взятом из [16].

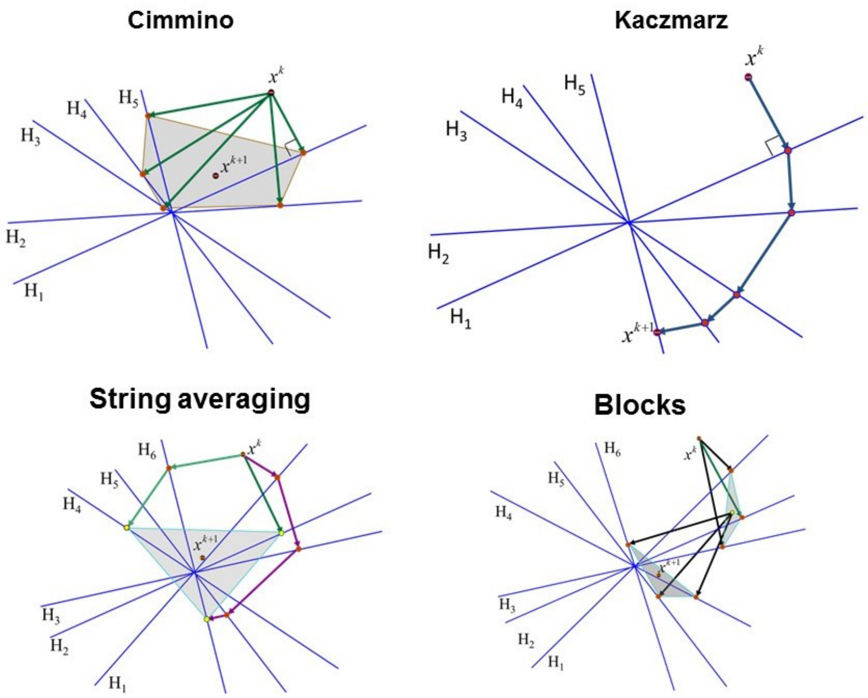


Рис. 1. Различные проекционные методы для линейного случая (рисунок из [16]).

**Замечание 2.1.** Заметим, что итерации для неподвижной точки (1.3) и (1.4) включают в себя вышеописанные методы. Например, если мы рассмотрим задачу об общей неподвижной точке с $U_i = P_{C_i}$, то мы получим выпуклую задачу о допустимости. Более того, если $T_k = U_{i(k)}$, где $i(k) = (k \bmod m)+1$, то мы получим метод последовательных ортогональных проекций (SOP) (2.1), а если возьмем только один блок $I$ размера $m$, то, приняв $T = \dfrac{1}{m}\sum_{i=1}^{m} U_i$, мы получим метод Чиммино в линейном случае и блочный метод в общем случае.

Далее вспомним две теоремы о неподвижной точке, классическую теорему Опиала [34] и ее обобщение [11, п. 3.6].

**Теорема 2.2.** *Пусть $\mathcal{H}$ — вещественное гильбертово пространство, и пусть $C \subset \mathcal{H}$ — замкнутое и выпуклое множество. Если $T : C \to C$ — усредненный оператор с $\mathrm{Fix}(T) \neq \varnothing$, то для любого $x^0 \in C$ последовательность $\{x^k\}_{k=0}^{\infty}$, порожденная $x^{k+1} = T(x^k)$, сходится слабо к точке $x^* \in \mathrm{Fix}(T)$.*

Приведем обобщенную теорему Опиала, см., например, [11, п. 3.6], позволяющую работать с семейством операторов $\{T_k : \mathcal{H} \to \mathcal{H}\}_{k=1}^{\infty}$.

**Теорема 2.3.** *Пусть $C \subseteq \mathcal{H}$ — непустое, замкнутое и выпуклое множество, $S : C \to \mathcal{H}$ — оператор со множеством неподвижных точек, такой что $S - Id$ — деми-замкнут в 0. Пусть*

$\{T_k : \mathcal{H} \to \mathcal{H}\}_{k=1}^{\infty}$ — *асимптотически регулярная последовательность квази-нерастягивающих операторов, таких что*

$$\operatorname{Fix}(S) \subseteq \left( \bigcap_{k=1}^{\infty} \operatorname{Fix}(T_k) \right).$$

*Пусть* $\left\{x^k\right\}_{k=0}^{\infty}$ — *последовательность, порожденная*

$$x^{k+1} = T_k(x^k),$$

*с произвольным* $x^0 \in \mathcal{H}$.

1. *Если последовательность операторов* $\{T_k\}_{k=1}^{\infty}$ *имеет свойство*

$$\lim_{k\to\infty} \|T_k(x^k) - x^k\| = 0 \quad \Longrightarrow \quad \lim_{k\to\infty} \|S(x^k) - x^k\| = 0, \tag{2.19}$$

*то* $\left\{x^k\right\}_{k=0}^{\infty}$ *сходится слабо к точке* $\operatorname{Fix}(S)$.

2. *Если* $\mathcal{H}$ *конечномерно и последовательность операторов* $\{T_k\}_{k=1}^{\infty}$ *имеет свойство*

$$\lim_{k\to\infty} \|T_k(x^k) - x^k\| = 0 \quad \Longrightarrow \quad \lim_{k\to\infty} \inf \|S(x^k) - x^k\| = 0, \tag{2.20}$$

*то* $\left\{x^k\right\}_{k=0}^{\infty}$ *сходится к точке* $\operatorname{Fix}(S)$.

## 3. Алгоритм

В этом разделе мы сосредоточимся на задаче об общей неподвижной точке (CFPP) (1.1) с семейством деми-сжимающих операторов $\{U_i\}_{i\in I}$, таких что

$$\bigcap_{i\in I} \operatorname{Fix}(U_i) \neq \varnothing.$$

Ситуация заключается в том, что множество индексов $I$ разбивается на $M$ блоков

$$I = I_1 \cup \ldots \cup I_M$$

выбором $\{m_t\}_{t=0}^{M} \subset \mathbb{Z}$ (где $\mathbb{Z}$ — это множество целых чисел), таких что

$$0 = m_0 < m_1 < \ldots < m_M = m,$$

и каждому $1 \leqslant t \leqslant M$ соответствует подмножество

$$I_t := \{m_{t-1} + 1, m_{t-1} + 2, \ldots, m_t\}.$$

Это, очевидно, делит семейство операторов $\{U_i\}_{i\in I}$ на соответствующие группы операторов.

Так как наша задача состоит в построении итерационной схемы в реальном времени, сосредоточимся на том случае, когда блоки и соответствующие операторы находятся в нашем распоряжении не с самого начала, а становятся известны постепенно. В недавней работе Ордонеса и др. [35] два метода в реальном времени ((DROP) [2] и (CARP) [25]) для решения систем линейных уравнений

$$Ax = b, \quad \text{где } A \in \mathbb{R}^{m\times n} \text{ с } m \sim 10^3 \text{ и } n \sim 10^9,$$

возникают в области протонной вычислительной томографии (pCT). Недавно Райх и Залас [37] представили процедуру *модулярного строкового усреднения* (MSA) для решения задачи об общей неподвижной точке в вещественных гильбертовых пространствах. Данная схема является довольно гибкой и позволяет строить вспомогательные операторы $T_k$, названные *модулями*, которые могут быть использованы во внутреннем цикле расширенного алгоритма с конечным количеством итераций $N_k$.

Для представления алгоритма введем несколько структур операторов $T_k$, построенных с помощью семейства операторов $\{U_i\}_{i\in I}$ с учетом $M$ блоков $I = I_1 \cup \ldots \cup I_M$ и использующихся во вспомогательном цикле нашего алгоритма. Эти структуры представлены как частные случаи для модулярного строкового усреднения [37]; более подробную информацию (в том числе и исторической справки) см. в работе [37] и имеющейся там библиографии.

**Определение 3.1.**

1. *Циклическая* (с релаксацией): $\alpha_k \in [\varepsilon, 2-\varepsilon]$ при $\varepsilon > 0$:
$$T_k = U_{i(k)}, \text{ где } i(k) = (k \bmod m) + 1.$$
2. *Выпуклая комбинация*: Для весового вектора $\omega^k(i) \geqslant 0$ для любых $i \in I_k$, таких что
$$\sum_{i \in I_k} \omega^k(i) = 1,$$
положим
$$T_k = \sum_{i \in I_k} \omega^k(i) U_i.$$
3. *Композиционная*:
$$T_k = \prod_{i \in I_k} U_i.$$
4. *Блочная*: $\alpha_k \in [\varepsilon, 2-\varepsilon]$ при $\varepsilon > 0$:
$$T_k = Id + \alpha_k \left( \sum_{i \in I_k} \omega^k(i) U_i - Id \right).$$
5. «*Жадная*» (наиболее удаленная):
$$T_k := U_{i_k}, \text{ где } i_k = \operatorname*{argmax}_{i \in I_k} \operatorname{dist}(\cdot, \operatorname{Fix}(U_i)).$$

Другими, более частными, структурами, которые могут быть здесь использованы, являются *усреднение по строкам* и различные типы *операторов Дугласа—Рашфорда* (см., например, [7]), в случае, если вместо $U_i$, используется $2U_i - Id$.

**Алгоритм 3.1** (Блочно-итерационная схема в режиме реального времени).

***Дано***. *Пусть $x^0 \in \mathcal{H}$ — произвольная начальная точка, определено $N_0 \in \mathbb{N}$ (количество итераций), и даны первый блок $I_1$ и соответствующее подмножество операторов $\{U_i\}_{i \in I_1}$. Вычислим $x^1$ таким образом*:
$$x^1 = T_0(x^0), \tag{3.1}$$
*где оператор $T_0$ строится согласно определению* 3.1 *и может быть циклическим, совместным или композицией $\{U_i\}_{i \in I_1}$.*

***Итерационный шаг***. *Имея текущую итерацию $x^k$ и зная $N_k \in \mathbb{N}$ (количество итераций), вычислим следующую итерацию*:
$$x^{k+1} = T_k(x^k), \tag{3.2}$$
*где оператор $T_k$ может быть построен следующим образом.*

1. *Если $k < m$: заданы блоки $I_1, I_2, \ldots, I_k$ и, следовательно, операторы $\{U_i\}_{i \in I_1 \cup \ldots \cup I_k}$. Тогда $T_k$ могут быть построены с учетом каждого, некоторых или всех операторов $\{U_i\}_{i \in I_1 \cup \ldots \cup I_k}$ в циклической, совместной или композиционной формах, основанных на определении* 3.1.
2. *Если $k \geqslant m$: заданы сразу все блоки и, следовательно, операторы $\{U_i\}_{i \in I}$, и тогда $T_k$ могут быть построены на основании определения* 3.1 *с учетом всего семейства операторов $\{U_i\}_{i \in I}$.*

**3.1. Сходимость.** Для сходимости нашего алгоритма 3.1 будем считать, что выполнены следующие условия.

**Условие 3.1.** *Для всех $i \in I$ операторы $U_i$ — деми-сжимающие с $\operatorname{Fix}(U_i) \neq \varnothing$, и такие, что $U_{i,\alpha}$ ($\alpha$-релаксация $U_i$) являются $(2-\alpha)/\alpha$-сильно квази-нерастягивающими.*

**Условие 3.2.** *$I \subseteq I_k \cup I_{k+1} \cup \ldots \cup I_{k+s-1}$ для всех $k = 0, 1 \ldots$ и некоторого $s \geqslant m - 1$.*

**Условие 3.3.** *Последовательность $\{N_k\}_{k=0}^{\infty}$, представляющая количество итераций на каждый блок, является ограниченной.*

В журнале *Numerical Algorithms* Райх и Залас [37] предложили процедуру *модулярного строкового усреднения* (MSA) для решения задачи об общей неподвижной точке в вещественных гильбертовых пространствах. Они представили гибкий метод [37, процедура 1.1]) построения вспомогательных операторов $T_k$, названных *модулями*, которые могут быть использованы во внутреннем цикле расширенного алгоритма с конечным числом итераций $N_k$ для семейства операторов $\{U_i\}_{i\in I}$. Вследствие модульности их схемы, и с учетом условий 3.1–3.3, сходимость нашей итерационной схемы в реальном времени, алгоритма 3.1, вытекает непосредственно из доказательства теоремы 4.1 в работе Райха и Заласа [37], хотя термин «в реальном времени» в этой работе и не упоминается. Следующая теорема является модификацией теоремы [37, теорема 4.1], скорректированной для алгоритма 3.1.

**Теорема 3.1.** *Пусть* $\mathcal{H}$ — *вещественное гильбертово пространство, и даны операторы* $U_i : \mathcal{H} \to \mathcal{H}$ *при* $i \in I$, *такие что* $\operatorname{Fix}(U_i) \neq \varnothing$. *Предположим, что условия* 3.1–3.3 *выполнены, и пусть последовательность* $\left\{x^k\right\}_{k=0}^{\infty}$ *сгенерирована алгоритмом* 3.1.

1. *Если для каждого* $i \in I$ *оператор* $U_i$ *удовлетворяет принципу деми-замкнутости Опиала, то последовательность* $\{x^k\}_{k=0}^{\infty}$ *сходится слабо к некоторой точке из* $\mathcal{C} = \bigcap_{i=1}^{m} \operatorname{Fix}(U_i)$.
2. *Если для каждого* $i \in I$ *оператор* $U_i$ *является приближенно стягивающим, а семейство* $\mathfrak{C} := \{\operatorname{Fix}(U_i) \mid i \in I\}$ *ограниченно регулярно, то последовательность* $\{x^k\}_{k=0}^{\infty}$ *сходится сильно к некоторой точке из* $\mathcal{C}$.

Необходимо упомянуть, что в этой работе мы считаем, что для всех $i \in I$ операторы $U_i$ являются деми-сжимающим, и, следовательно, по теореме 2.1 для всех $i \in I$ и $\alpha \in (0,2]$ мы полагаем $U_{i,\alpha}$ ($\alpha$-релаксация $U_i$) $(2-\alpha)/\alpha$-сильно квази-нерастягивающими и, таким образом, срезающими; это, собственно, и используется в алгоритме 3.1.

**Замечание 3.1.**

1. Используя условие 3.1 и теорему 2.1, получаем, что операторы $U_i$ являются срезающими, а структура алгоритма 3.1 может быть произвольного типа согласно определению 3.1.

2. Условие 3.2 означает, что управляющая последовательность почти циклична (определение 2.3 (2)).

3. Условие 3.3 говорит о том, что количество итераций $N_k$ ограничено, что в свою очередь означает, что допустимо любое конечное число промежуточных шагов в каждом блоке.

3. В случае, если $U_i = P_{C_i}$ и $\mathcal{H} = \mathbb{R}^n$, все предположения относительно операторов, касающиеся непрерывности, будь то срезающие, деми-замкнутые или приближенно стягивающие операторы, справедливы.

4. Для задач о допустимости и, в частности, для линейных задач о допустимости произвольная управляющая последовательность оказывается эффективной управляющей последовательностью (это также известно как рандомизированный метод Качмажа, см., например, [31, 33]). Хотя доказательство сходимости не охватывает нашу ситуацию, представляется довольно интересным изучение поведения в теории; это, вероятно, будет исследовано в дальнейших работах. Несмотря на это, мы изучили этот случай в наших численных экспериментах. Другим относящимся к теме результатом, в котором произвольные управляющие последовательности также рассматриваются как класс стохастических алгоритмов (в частности, для задач о допустимости и для вариационных неравенств), является работа Иусема и соавторов [28].

5. В [12, теорема 4.3 и следствие 4.4] представлено новое обобщение теоремы Опиала с меньшим количеством ограничивающих предположений, чем в теореме 2.3. Поэтому будет интересно наблюдать, как данный результат будет изучен и приложен к результатам этой работы.

6. Ордонес и соавторы в работе [35] представили два проекционных метода в реальном времени, (DROP) [2] и (CARP) [25], для решения разреженных и переопределенных систем линейных уравнений $Ax = b$ большой размерности, в которых $A \in \mathbb{R}^{m\times n}$ с $m \sim 10^3$ и $n \sim 10^9$, возникающих в области протонной вычислительной томографии (pCT). Эта работа предлагает многообещающие результаты в экспериментальном поведении, но ей, к сожалению, не хватает достаточного математического обоснования. И хотя наш подход не применяется к их схемам, мы способны провести анализ данной гибкой схемы, которая может быть приложена не только к линейным задачам о допустимости.

## 4. Вычислительные эксперименты

В этом разделе мы будем сравнивать 4 схемы в реальном времени: Чиммино [18] (алгоритм 2.1), Качмажа [29] (алгоритм 2.2), рандомизированный метод Качмажа [31, 33] и «жадный» метод Качмажа для линейных и нелинейных (квадратичных) выпуклых задач о допустимости (CFP) в евклидовых пространствах $\mathbb{R}^d$. Все вычислительные эксперименты были проведены на обычном ноутбуке Lenovo с процессором Intel(R) Core(TM) i5-4200MQ CPU 1,6 ГГц с 8 Гб оперативной памяти. Программа реализована с помощью пакета MATLAB 2017b.

**Пример 4.1** (Линейные CFP)**.** В этом примере мы рассматриваем разрешение системы линейных уравнений

$$Ax = b,$$

в частности, восстановление тестового изображения $x \in [0,1]^n$ «Ленна» (https://en.wikipedia.org/wiki/Lenna), состоящего из ограниченного количества томографических проекций. Каждый пиксель обозначим через $x_i \in [0,1]$, а каждый элемент из $b \in \mathbb{R}^m$, названный *томографическим измерением*, или *единичной проекцией*, соответствует интегрированному уровню яркости $x$ вдоль единичного луча. Всякий элемент матрицы $a_{ij} \geqslant 0$ задает длину пересечения $i$-го луча с $j$-м пикселем. Если луч $i$ и пиксель $j$ не пересекаются, то $a_{ij} = 0$, см. рис. 2. Сложение всех уравнений для всех лучей вместе приводит к системе линейных уравнений $Ax = b$, а размеры таковы, что

$$A = (A_{\theta_1}^T \;\; A_{\theta_2}^T \;\; \ldots \;\; A_{\theta_{\mathtt{nA}}}^T)^T,$$

и каждая блочная матрица $A_{\theta_i}$ соответствует своему проекционному углу.

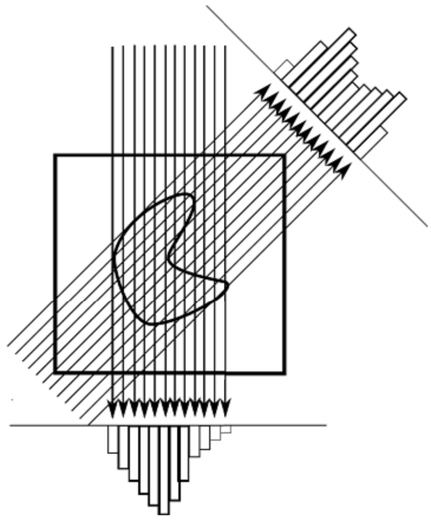

Рис. 2. Геометрическая структура параллельного пучка: множество параллельных лучей пропускается через объект по различным направлениям. Это обычно считается одной проекцией. Случай с двумя проекциями проиллюстрирован на рисунке: каждая проекция отвечает измерению вдоль одного луча и соответствует криволинейному интегралу от кусочно-постоянной функции.

В наших экспериментах мы используем в MATLAB шаблон `paralleltomo.m` из пакета AIR Tools [27], который обеспечивает построение томографической матрицы для заданного вектора углов. Размер шкалы яркости изображения Ленны составляет $N = 128$ (это означает $128 \times 128$ пикселей), и выберем для каждого угла число параллельных пучков $nA = 100$. Еще зададим параметр $p = \mathrm{round}(128\sqrt{2}) = 169$. Выбрав параметры таким образом, получаем переопределенную матрицу $A$ размера $(nA * p) \times (N^2) = 18100 \times 16384$. В этом случае нашими данными являются строки $A$ и соответствующие элементы из $b$. Далее разобьем систему $Ax = b$ на 10 подсистем $A_j x = b^j$ размера $1810 \times 16384$ для $j = 1, \ldots, 10$. Время прибытия для каждого блока фиксировано и должно составлять 1 минуту. Критерием остановки для всех алгоритмов является $\|x^{k+1} - x^k\| \leqslant 10^{-3}$, а начальная стартовая точка $x^0 = 0$.

Во всех алгоритмах, кроме метода Чиммино, мы полагаем параметры релаксации $\lambda_k \equiv 1$, а в методе Чиммино $\lambda_k \equiv 1{,}9$. Мы тестируем и сравниваем наши схемы на основе изображения «Ленна» (https://en.wikipedia.org/wiki/Lenna).

Напомним, что ортогональная проекция точки $x \in \mathbb{R}^d$ на гиперплоскость

$$H = \{z \in \mathbb{R}^d \mid \langle a, z\rangle \leqslant \beta\},$$

где $a \in \mathbb{R}^d$ (ненулевой) и $\beta \in \mathbb{R}$, рассчитывается следующим образом:

$$P_H(x) = \begin{cases} x & \text{при } \langle a, x\rangle = \beta, \\ x - \dfrac{\langle x, a\rangle - \beta}{\|a\|^2} a & \text{при } \langle a, x\rangle \neq \beta. \end{cases} \tag{4.1}$$

На рис. 3 представлено сравнение времени работы программы (в секундах) для метода Чиммино в реальном времени и регулярного метода Чиммино по всем данным. Далее на рис. 4–6 изображены графики (слева), позволяющие сравнить время выполнения программы, а также различия восстановленных изображений, полученных с помощью работы алгоритмов в реальном времени и их регулярных аналогов по всем данным, начиная с момента получения этих данных. На графиках ось $y$, означающая *погрешность*, определяется как $\|Ax^k - b\|$. И хотя разница в восстановленных разными методами изображениях едва заметна невооруженному глазу (изображение слева получено с помощью регулярных алгоритмов, в то время как изображение справа — с помощью аналогов в реальном времени), графики демонстрируют, что для получения необходимой аппроксимации лучше применять алгоритмы в реальном времени (порой даже до получения всех данных).

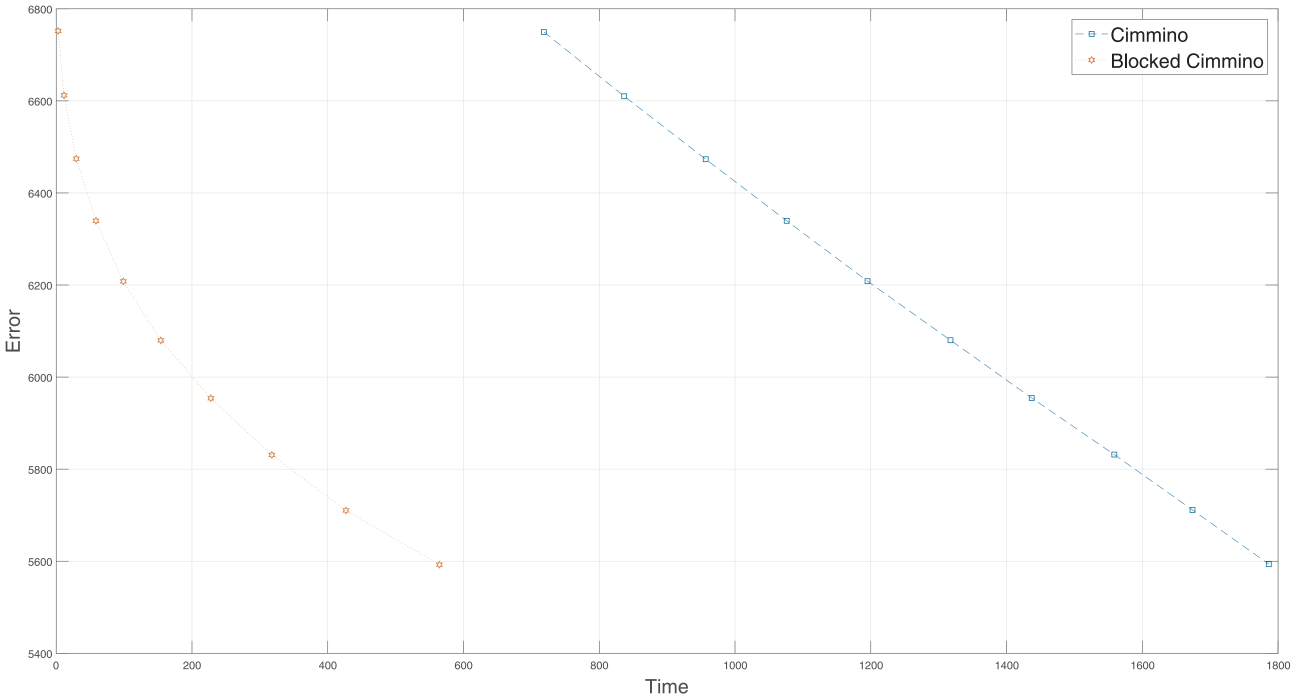


Рис. 3. Метод Чиммино с 10-ю свипами и 10-ю блоками.

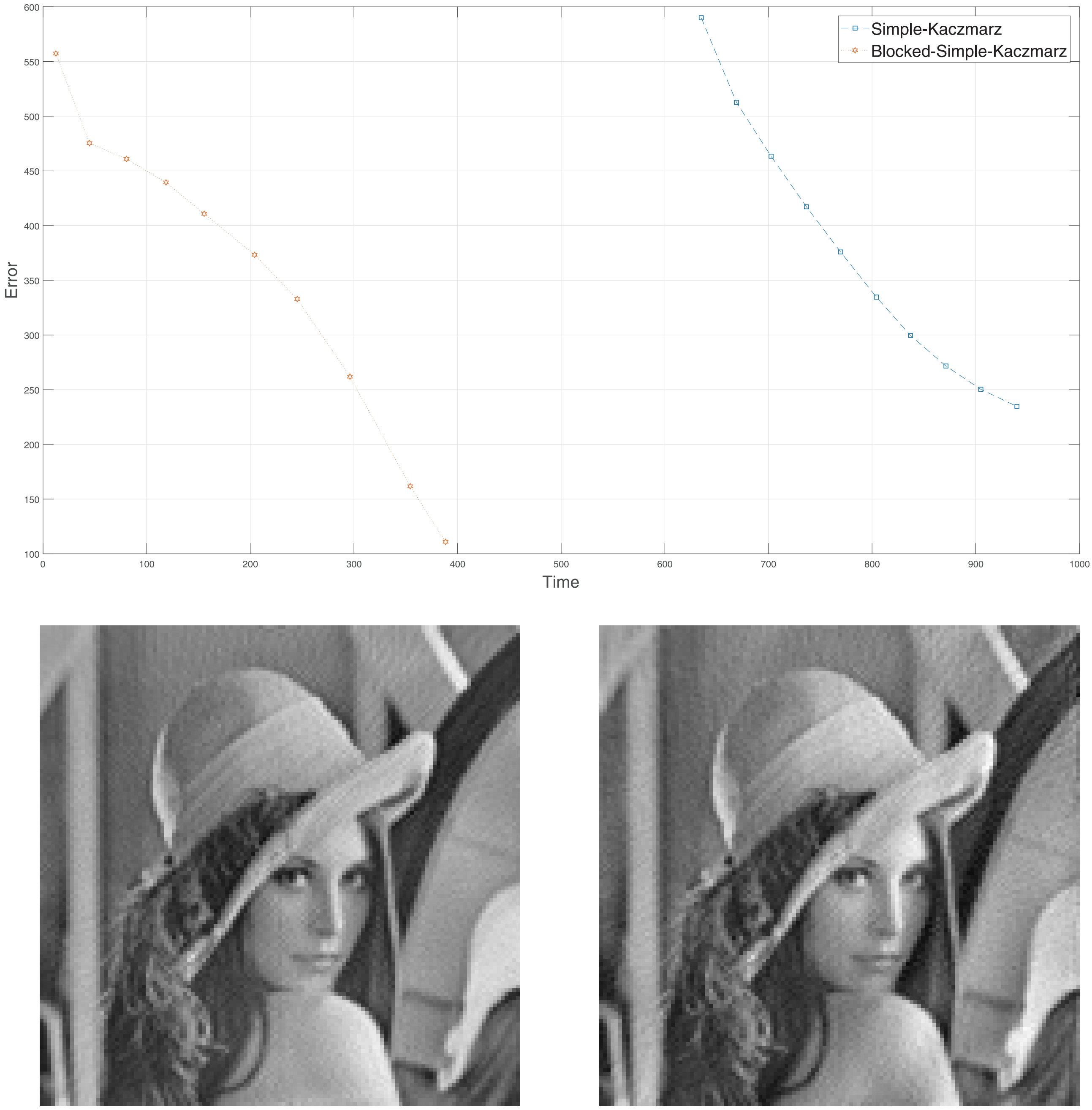


Рис. 4. Сравнение регулярного метода Качмажа и метода Качмажа в реальном времени. Сверху — график сравнения выполнения регулярного метода и метода Качмажа в реальном времени. Изображение слева получено регулярным методом, а изображение справа — его аналогом в режиме реального времени.

**Пример 4.2** (Квадратичные CFP)**.** Здесь мы генерируем 10 квадратичных задач о допустимости в $\mathbb{R}^{1000}$, т. е. каждое множество представляет собой шар. В каждом из экспериментов мы увеличиваем количество множеств и сравниваем выполнение всех алгоритмов в режиме реального времени и их регулярных аналогов, ожидающих поступления данных. Каждый шар получается выбором центральной точки $c^i \in \mathbb{R}^{1000}$ с произвольными, равномерно сгенерированными координатами на отрезке $[-5,5]$. Выбор радиуса $r_i := \left\|c^i\right\| + \alpha_i$ определяется добавлением к расстоянию от центра до начала координат $\left\|c^i\right\|$ произвольного числа, равномерно выбранного из отрезка $[0,0{,}1]$. Что гарантирует, что шар содержит начало координат и, следовательно, приводит к состоятельной CFP. Векторы $x^0$ задаются произвольным выбором координат из отрезка $[-10,10]$. Число ограничений (шаров) варьируется от 200 до 20000. Как и в примере 4.1, для каждой CFP мы разбиваем

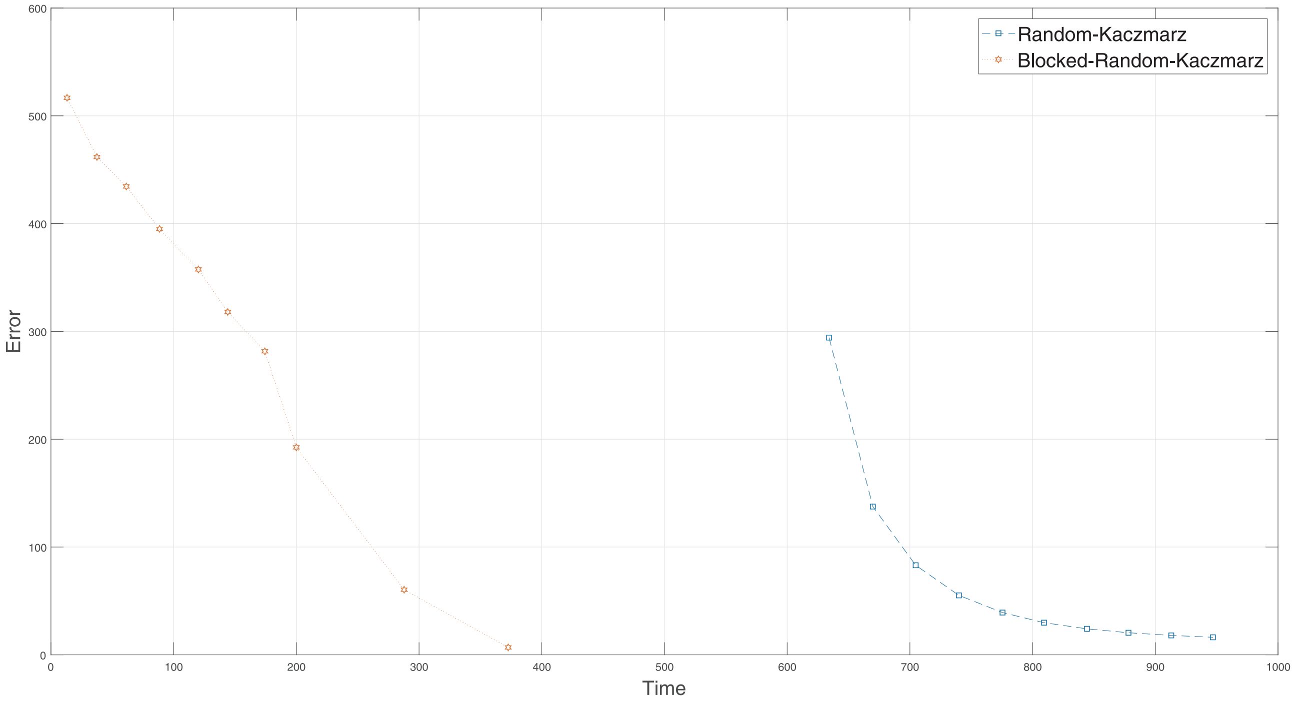


Рис. 5. Сравнение регулярного метода Качмажа и рандомизированного метода Качмажа в реальном времени. Сверху — график сравнения выполнения этих двух методов. Изображение слева получено регулярным методом, а изображение справа — его аналогом в режиме реального времени.

число ограничений (шаров) на 10 блоков и устанавливаем правило остановки:

$$\|x^{k+1} - x^k\| \leqslant \varepsilon = 10^{-7}.$$

Параметры релаксации $\lambda_k$ полагаем равными 1, а в методе Чиммино они были равны 1,9. В этих экспериментах мы можем наблюдать, что с увеличением количества ограничений возрастает и разница в выполнении схем в реальном времени и их регулярных аналогов. Это вновь подчеркивает потенциальную применимость данных методов к задачам в реальном времени. Результаты, представленные на рис. 7, демонстрируют, что с увеличением количества ограничений (шаров) алгоритмы в реальном времени сходятся быстрее, чем их регулярные аналоги.

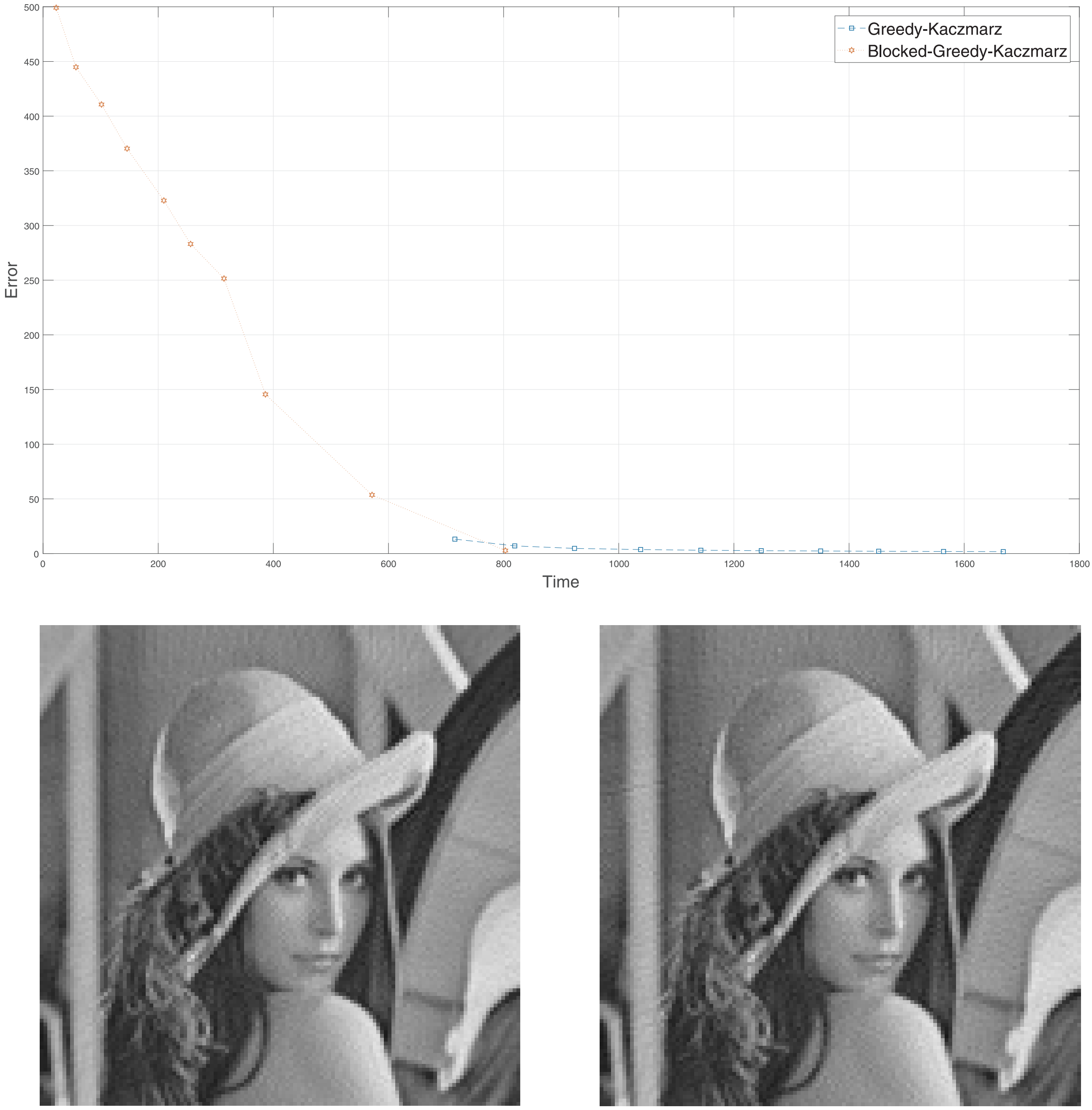


Рис. 6. Сравнение регулярного метода и «жадного» метода в реальном времени. Сверху — график сравнения выполнения регулярного метода и «жадного» рандомизированного метода в реальном времени. Изображение слева получено регулярным методом, а изображение справа — его аналогом в режиме реального времени.

Вспомним, как вычисляется ортогональная проекция точки $x \in \mathbb{R}^n$ на замкнутый шар $B(z, r) := \{x \in \mathbb{R}^n \mid \|x - z\| \leqslant r\}$, где $z \in \mathbb{R}^n$ и $r > 0$:

$$P_{B(z,r)}(x) = \begin{cases} x & \text{при } \|x - z\| \leqslant r, \\ z + \dfrac{r}{\|x - z\|}(x - z) & \text{при } \|x - z\| > r. \end{cases} \tag{4.2}$$

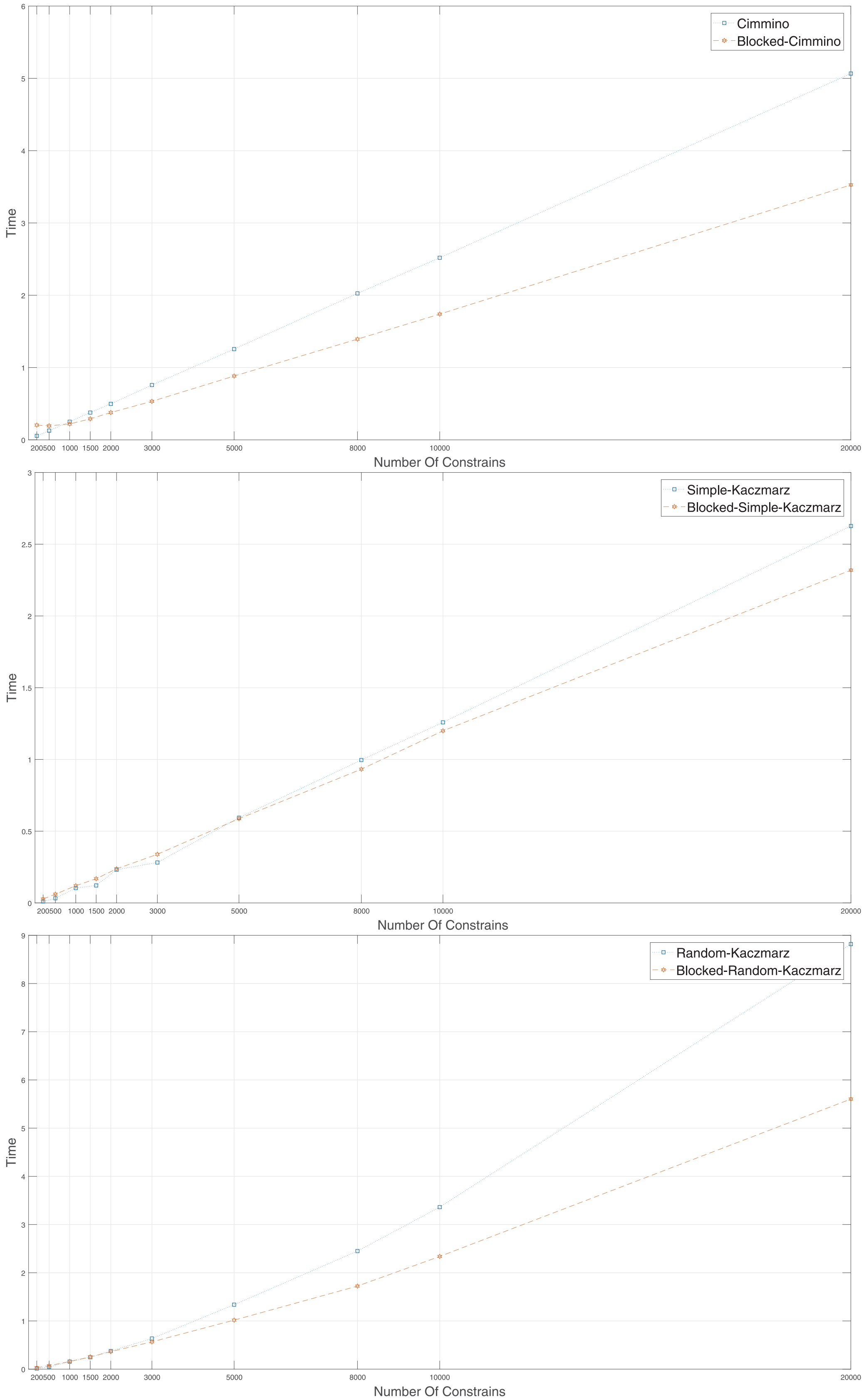


Рис. 7. Время выполнения программы (в секундах) для 10 свипов и 10 блоков, соответственно, для метода Чиммино, циклического (Качмаж) и произвольного методов для решения квадратичных задач о допустимости при увеличении количества шаров.

## 5. Заключение

В данной работе были рассмотрены *задача об общей неподвижной точке* (CFPP) и *выпуклая задача о допустимости* (CFP) в вещественных гильбертовых пространствах. Мы изучили ситуацию, в которой нам с самого начала недоступны целые наборы (операторов/множеств), и мы получаем их постепенно.

Руководствуясь недавней работой Ордонеса и соавторов [35], мы представили итерационную схему в режиме реального времени, способную работать с любыми блоками данных (операторов/множеств) для любого конечного числа итераций перед переходом к следующему блоку. Доказательство сходимости нашей схемы основано на недавнем результате Райха и Заласа [37], процедуре *модулярного строкового усреднения* (MSA). Мы также провели вычислительные эксперименты, демонстрирующие, что схема в реальном времени вычисляет решение быстрее в сравнении с тем случаем, когда все данные известны с самого начала. В то время как Ордонес и соавторы в работе [35] сосредоточены на исследовании только линейных систем уравнений без достаточного математического обоснования, мы проводим более общий анализ задач об общей неподвижной точке с полным теоретическим обоснованием.

И хотя структуры методов CARP и DROP не включают в себя алгоритмических структур $T_k$ в алгоритме 3.1, мы планируем продолжить изучение в этом направлении и, более того, получить оценки ошибок и скорости сходимости этой новой схемы.

## Благодарности

Мы высоко ценим конструктивные комментарии анонимных рецензентов, которые помогли нам улучшить нашу работу. Авторы выражают искреннюю признательность проф. Яиру Цензору и д-ру Рафалу Заласу за их ценные советы. Работа Авива Гибали поддержана программой EU FP7 IRSES STREVCOMS, грант номер PIRSES-GA-2013-612669.

## СПИСОК ЛИТЕРАТУРЫ

1. *Губин Л. Г., Поляк Б. Т., Райк Е. В.* Метод проекции для нахождения общей точки выпуклых множеств// Журн. выч. мат. и мат. физ. — 1967. — *7.* — С. 1–24.
2. *Aharoni R., Censor Y.* Block-iterative projection methods for parallel computation of solutions to convex feasibility problems// Linear Algebra Appl. — 1989. — *120.* — С. 165–175.
3. *Baillon J.-B., Bruck R. E., Reich S.* On the asymptotic behavior of nonexpansive mappings and semigroups in banach spaces// Houston J. Math. — 1978. — *4.* — С. 1–9.
4. *Bauschke H. H., Borwein J.* On projection algorithms for solving convex feasibility problems// SIAM Rev. — 1996. — *38.* — С. 367–426.
5. *Bauschke H. H., Combettes P. L.* Convex analysis and monotone operator theory in Hilbert spaces. — Berlin: Springer, 2011.
6. *Bauschke H. H., Koch V. R.* Projection methods: Swiss army knives for solving feasibility and best approximation problems with halfspaces// В сб.: «Infinite Products of Operators and Their Applications. A Research Workshop of the Israel Science Foundation, Haifa, Israel, May 21–24, 2012». — Providence: Am. Math. Soc., 2015. — С. 1–40.
7. *Borwein J. M., Tam M. K.* A cyclic Douglas—Rachford iteration scheme// J. Optim. Theory Appl. — 2014. — *160.* — С. 1–29.
8. *Browder F. E.* Fixed point theorems for noncompact mappings in Hilbert space// Proc. Natl. Acad. Sci. USA. — 1965. — *53.* — С. 1272–1276.
9. *Byrne C. L.* A unified treatment of some iterative algorithms in signal processing and image reconstruction// Inverse Problems. — 1999. — *20.* — С. 1295–1313.
10. *Byrne C. L.* Applied iterative methods. — Wellsely: AK Peters, 2008.
11. *Cegielski A.* Iterative methods for fixed point problems in Hilbert spaces. — Berlin—Heidelberg: Springer, 2012.
12. *Cegielski A., Reich S., Zalas R.* Regular sequences of quasi-nonexpansive operators and their applications// SIAM J. Optim. — 2018. — *28.* — С. 1508–1532.
13. *Cegielski A., Zalas R.* Methods for variational inequality problem over the intersection of fixed point sets of quasi-nonexpansive operators// Numer. Funct. Anal. Optim. — 2013. — *34.* — С. 255–283.
14. *Cegielski A., Zalas R.* Properties of a class of approximately shrinking operators and their applications// Fixed Point Theory. — 2014. — *15.* — С. 399–426.

15. *Censor Y., Chen W., Combettes P. L., Davidi R., Herman G. T.* On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints// Comput. Optim. Appl. — 2012. — *51.* — С. 1065–1088.
16. *Censor Y., Elfving T., Herman G. T.* Averaging strings of sequential iterations for convex feasibility problems// В сб.: «Infinite Products of Operators and Their Applications. A Research Workshop of the Israel Science Foundation, Haifa, Israel, March 13–16, 2000». — Amsterdam: North-Holland, 2001. — С. 101–113.
17. *Censor Y., Zenios S. A.* Parallel optimization: theory, algorithms, and applications. — New York: Oxford Univ. Press, 1997.
18. *Cimmino G.* Calcolo approssiomatto per le soluzioni dei sistemi di equazioni lineari// La Ricerca Sci. XVI. Ser. II. — 1938. — *1.* — С. 326–333.
19. *Combettes P. L.* Quasi-Fejérian analysis of some optimization algorithms// В сб.: «Infinite Products of Operators and Their Applications. A Research Workshop of the Israel Science Foundation, Haifa, Israel, March 13–16, 2000». — Amsterdam: North-Holland, 2001. — С. 115–152.
20. *Das I., Potra F. A.* Subsequent convergence of iterative methods with applications to real-time model-predictive control// J. Optim. Theory Appl. — 2003. — *119.* — С. 37–47.
21. *Diehl M.* Real-Time optimization for large scale nonlinear processes. — Heidelberg: Univ. Heidelberg, 2001.
22. *Escalante R., Raydan M.* Alternating projection methods. — Philadelphia: SIAM, 2011.
23. *Galántai A.* Projectors and projection methods. — Boston—Dordrecht—London: Kluwer Academic Publ., 2004.
24. *Goebel K., Reich S.* Uniform convexity, hyperbolic geometry, and nonexpansive mappings. — New York—Basel: Marcel Dekker, 1984.
25. *Gordon D., Gordon R.* Component-averaged row projections: A robust block-parallel scheme for sparse linear systems// SIAM J. Sci. Comput. — 2005. — *27.* — С. 1092–1117.
26. *Gordon R., Bender R., Herman G. T.* Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and X-ray photography// Bull. Am. Math. Soc. — 1970. — *29.* — С. 471–481.
27. *Hansen P. C., Saxild-Hansen M.* AIR Tools — a MATLAB package of algebraic iterative reconstruction methods// J. Comput. Appl. Math. — 2012. — *236*, № 8. — С. 2167–2178.
28. *Iusem A., Jofré A., Thompson P.* Incremental constraint projection methods for monotone stochastic variational inequalities// arXiv:1703.00272v2. — 2017.
29. *Kaczmarz S.* Angenäherte auflösung von systemen linearer gleichungen// Bull. Int. l'Acad. Polon. Sci. Lett. A. — 1937. — *35.* — С. 355–357.
30. *Karp R. M.* On-line algorithms versus off-line algorithms: How much is it worth to know the future?// В сб.: «Proceedings of the IFIP 12th World Computer Congress on Algorithms, Software, Architecture, Information Processing '92». — 1992. — *1.* — С. 416–429.
31. *Leventhal L., Lewis A. S.* Randomized methods for linear constraints: convergence rates and conditioning// Math. Oper. Res. — 2010. — *35.* — С. 641–654.
32. *Măruşter Şt., Popirlan C.* On the Mann-type iteration and the convex feasibility problem// J. Comput. Appl. Math. — 2008. — *212.* — С. 390–396.
33. *Needell D.* Randomized Kaczmarz solver for noisy linear systems// BIT Numer. Math. — 2010. — *50.* — С. 395–403.
34. *Opial Z.* Weak convergence of the sequence of successive approximations for nonexpansive mappings// Bull. Am. Math. Soc. — 1967. — *73.* — С. 591–597.
35. *Ordoñez C. E., Karonis N., Duffin K., Coutrakon G., Schulte R., Johnson R., Pankuch M.* A real-time image reconstruction system for particle treatment planning using proton computed tomography (PCT)// Phys. Proc. — 2017. — *90.* — С. 193–199.
36. *Penfold S., Censor Y., Schulte R. W., Bashkirov V., McAllister S., Schubert K. E., Rosenfeld A. B.* Block-iterative and string-averaging projection algorithms in proton computed tomography image reconstruction// В сб.: «Biomedical Mathematics: Promising Directions in Imaging, Therapy Planning and Inverse Problems». — Madison: Medical Phys. Publ., 2010. — С. 347–368.
37. *Reich S., Zalas R.* A modular string averaging procedure for solving the common fixed point problem for quasi-nonexpansive mappings in hilbert space// Numer. Algorithms. — 2016. — *72.* — С. 297–323.



Aviv Gibali
Department of Mathematics, Ort Braude College, Karmiel 2161002, Israel;
The Center for Mathematics and Scientific Computation, University of Haifa,
Mt. Carmel, Haifa 3498838, Israel
E-mail: avivg@braude.ac.il

Dimitri Teller
Department of Mathematics, Ort Braude College, Karmiel 2161002, Israel
E-mail: ktui619@gmail.com

DOI: 10.22363/2413-3639-2018-64-4-616-636 UDC 517.98+519.65

## A Real-Time Iterative Projection Scheme for Solving the Common Fixed Point Problem and Its Applications

 **A. Gibali, D. Teller**

**Abstract**. In this paper, we are concerned with the Common Fixed Point Problem (CFPP) with demi-contractive operators and its special instance, the Convex Feasibility Problem (CFP) in real Hilbert spaces. Motivated by the recent result of Ordoñez et al. [35] and in general, the field of online/real-time algorithms, e.g., [20, 21, 30], in which the entire input is not available from the beginning and given piece-by-piece, we propose an online/real-time iterative scheme for solving CFPPs and CFPs in which the involved operators/sets emerge along time. This scheme is capable of operating on any block, for any finite number of iterations, before moving, in a serial way, to the next block.

The scheme is based on the recent novel result of Reich and Zalas [37] known as the Modular String Averaging (MSA) procedure. The convergence of the scheme follows [37] and other classical results in the fields of fixed point theory and variational inequalities, such as [34].

Numerical experiments for linear and non-linear feasibility problems with applications to image recovery are presented and demonstrate the validity and potential applicability of our scheme, e.g., to online/real-time scenarios.

## REFERENCES

1. L. G. Gubin, B. T. Polyak, and E. V. Raik, “Metod proektsii dlya nakhozhdeniya obshchey tochki vypuklykh mnozhestv” [The method of projections for finding the common point of convex sets], *Zhurn. vych. mat. i mat. fiz.* [J. Comput. Math. Math. Phys.], 1967, **7**, 1–24 (in Russian).
2. R. Aharoni and Y. Censor, “Block-iterative projection methods for parallel computation of solutions to convex feasibility problems,” *Linear Algebra Appl.*, 1989, **120**, 165–175.
3. J.-B. Baillon, R. E. Bruck, and S. Reich, “On the asymptotic behavior of nonexpansive mappings and semigroups in banach spaces,” *Houston J. Math*, 1978, **4**, 1–9.
4. H. H. Bauschke and J. Borwein, “On projection algorithms for solving convex feasibility problems,” *SIAM Rev.*, 1996, **38**, 367–426.
5. H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, Berlin, 2011.
6. H. H. Bauschke and V. R. Koch, “Projection methods: Swiss army knives for solving feasibility and best approximation problems with halfspaces,” In: *Infinite Products of Operators and Their Applications. A Research Workshop of the Israel Science Foundation, Haifa, Israel, May* 21–24*,* 2012, Am. Math. Soc., Providence, 2015, 1–40.
7. J. M. Borwein and M. K. Tam, “A cyclic Douglas—Rachford iteration scheme,” *J. Optim. Theory Appl.*, 2014, **160**, 1–29.
8. F. E. Browder, “Fixed point theorems for noncompact mappings in Hilbert space,” *Proc. Natl. Acad. Sci. USA*, 1965, **53**, 1272–1276.
9. C. L. Byrne, “A unified treatment of some iterative algorithms in signal processing and image reconstruction,” *Inverse Problems*, 1999, **20**, 1295–1313.
10. C. L. Byrne, *Applied Iterative Methods*, AK Peters, Wellsely, 2008.
11. A. Cegielski, *Iterative Methods for Fixed Point Problems in Hilbert Spaces*, Springer, Berlin—Heidelberg, 2012.
12. A. Cegielski, S. Reich, and R. Zalas, “Regular sequences of quasi-nonexpansive operators and their applications,” *SIAM J. Optim.*, 2018, **28**, 1508–1532.

13. A. Cegielski and R. Zalas, "Methods for variational inequality problem over the intersection of fixed point sets of quasi-nonexpansive operators," *Numer. Funct. Anal. Optim.*, 2013, **34**, 255–283.
14. A. Cegielski and R. Zalas, "Properties of a class of approximately shrinking operators and their applications," *Fixed Point Theory*, 2014, **15**, 399–426.
15. Y. Censor, W. Chen, P. L. Combettes, R. Davidi, and G. T. Herman, "On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints," *Comput. Optim. Appl.*, 2012, **51**, 1065–1088.
16. Y. Censor, T. Elfving, and G. T. Herman, "Averaging strings of sequential iterations for convex feasibility problems," In: *Infinite Products of Operators and Their Applications. A Research Workshop of the Israel Science Foundation, Haifa, Israel, March* 13–16*,* 2000, North-Holland, Amsterdam, 2001, 101–113.
17. Y. Censor and S. A. Zenios, *Parallel Optimization*: *Theory, Algorithms, and Applications*, Oxford Univ. Press, New York, 1997.
18. G. Cimmino, "Calcolo approssiomatto per le soluzioni dei sistemi di equazioni lineari," *La Ricerca Scientifica XVI. Ser. II*, 1938, **1**, 326–333.
19. P. L. Combettes, "Quasi-Fejérian analysis of some optimization algorithms," In: *Infinite Products of Operators and Their Applications. A Research Workshop of the Israel Science Foundation, Haifa, Israel, March* 13–16*,* 2000, North-Holland, Amsterdam, 2001, 115–152.
20. I. Das and F. A. Potra, "Subsequent convergence of iterative methods with applications to real-time model-predictive control," *J. Optim. Theory Appl.*, 2003, **119**, 37–47.
21. M. Diehl, *Real-Time Optimization for Large Scale Nonlinear Processes*, Univ. Heidelberg, Heidelberg, 2001.
22. R. Escalante and M. Raydan, *Alternating Projection Methods*, SIAM, Philadelphia, 2011.
23. A. Galántai, *Projectors and Projection Methods*, Kluwer Academic Publ., Boston—Dordrecht—London, 2004.
24. K. Goebel and S. Reich, *Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings*, Marcel Dekker, New York—Basel, 1984.
25. D. Gordon and R. Gordon, "Component-averaged row projections: A robust block-parallel scheme for sparse linear systems," *SIAM J. Sci. Comput.*, 2005, **27**, 1092–1117.
26. R. Gordon, R. Bender, and G. T. Herman, "Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and X-ray photography," *Bull. Am. Math. Soc.*, 1970, **29**, 471–481.
27. P. C. Hansen and M. Saxild-Hansen, "AIR Tools — a MATLAB package of algebraic iterative reconstruction methods," *J. Comput. Appl. Math.*, 2012, **236**, No. 8, 2167–2178.
28. A. Iusem, A. Jofré, and P. Thompson, "Incremental constraint projection methods for monotone stochastic variational inequalities," *arXiv:1703.00272v2*, 2017.
29. S. Kaczmarz, "Angenäherte auflösung von systemen linearer gleichungen," *Bulletin International de l'Académie Polonaise des Sciences et des Lettres A*, 1937, **35**, 355–357.
30. R. M. Karp, "On-line algorithms versus off-line algorithms: How much is it worth to know the future?," In: *Proceedings of the IFIP 12th World Computer Congress on Algorithms, Software, Architecture, Information Processing* '92, 1992, **1**, 416–429.
31. L. Leventhal and A. S. Lewis, "Randomized methods for linear constraints: convergence rates and conditioning," *Math. Oper. Res.*, 2010, **35**, 641–654.
32. Şt. Măruşter and C. Popirlan, "On the Mann-type iteration and the convex feasibility problem," *J. Comput. Appl. Math.*, 2008, **212**, 390–396.
33. D. Needell, "Randomized Kaczmarz solver for noisy linear systems," *BIT Numer. Math.*, 2010, **50**, 395–403.
34. Z. Opial, "Weak convergence of the sequence of successive approximations for nonexpansive mappings," *Bull. Am. Math. Soc.*, 1967, **73**, 591–597.
35. C. E. Ordoñez, N. Karonis, K. Duffin, G. Coutrakon, R. Schulte, R. Johnson, and M. Pankuch, "A real-time image reconstruction system for particle treatment planning using proton computed tomography (PCT)," *Physics Procedia*, 2017, **90**, 193–199.
36. S. Penfold, Y. Censor, R. W. Schulte, V. Bashkirov, S. McAllister, K. E. Schubert, and A. B. Rosenfeld, "Block-iterative and string-averaging projection algorithms in proton computed tomography image reconstruction," In: *Biomedical Mathematics*: *Promising Directions in Imaging, Therapy Planning and Inverse Problems*, Medical Phys. Publ., Madison, 2010, 347–368.
37. S. Reich and R. Zalas, "A modular string averaging procedure for solving the common fixed point problem for quasi-nonexpansive mappings in hilbert space," *Numer. Algorithms*, 2016, **72**, 297–323.

Aviv Gibali
Department of Mathematics, Ort Braude College, Karmiel, Israel;
The Center for Mathematics and Scientific Computation, University of Haifa, Haifa, Israel
E-mail: avivg@braude.ac.il

Dimitri Teller
Department of Mathematics, Ort Braude College, Karmiel, Israel
E-mail: ktui619@gmail.com