

УДК 519.85

АЛГОРИТМ НАХОЖДЕНИЯ НЕОТРИЦАТЕЛЬНОГО РЕШЕНИЯ СИСТЕМЫ ЛИНЕЙНЫХ УРАВНЕНИЙ

Е. Н. Севостьянов^{1,a}, М. Г. Лепчинский^{2,b}

¹ ЗАО «Диджитал Айрон Пайп», Челябинск, Россия

² Челябинский государственный университет, Челябинск, Россия

^a y384@mail.ru; ^b myth@csu.ru

В статье рассматривается новый алгоритм для отыскания точного решения задачи линейного программирования. Метод основан на сведении задачи линейного программирования к задаче решения системы линейных уравнений неполного ранга в неотрицательных числах, которая решается через последовательные приближения в специально построенных подпространствах. Доказана сходимость метода за конечное число шагов. Приводятся результаты тестирования алгоритма на данных задачи Кли — Минти и на группе случайных тестов.

Ключевые слова: алгоритм линейного программирования, неотрицательное решение системы линейных уравнений, линейное неравенство.

Введение

Первоначально исследования в области численных методов линейного программирования (ЛП) концентрировались в основном на симплекс-методе. Далее разрабатывались разнообразные итерационные методы, а после опубликования статьи [1] внимание многих исследователей переключилось на методы внутренних точек.

В связи с тем, что не найден сильно полиномиальный алгоритм для задачи ЛП [2, с. 194–205], интерес вызывает изучение иных подходов к построению алгоритмов ЛП. Данная работа посвящена описанию и изучению алгоритма линейного программирования, отличного от алгоритмов метода внутренних точек и симплекс-метода. Отличительной особенностью алгоритма является то, что задача линейного программирования в стандартной форме с помощью комбинирования прямой и двойственной задачи сводится к задаче нахождения неотрицательного решения системы линейных уравнений, которая, в свою очередь, решается через последовательные приближения по направлениям вдоль гиперплоскостей, ограничивающих неотрицательную область пространства \mathbb{R}^n . Доказана сходимость алгоритма к точному решению не более чем за $2^n n$ шагов. Предложенный алгоритм можно считать обобщением и развитием идей, высказанных в работе [3].

1. Обозначения и начальный вид решаемой задачи

Договоримся обозначать множество матриц размером m на n над полем \mathbb{R} как $\mathbb{R}^{m,n}$. Единичную матрицу будем обозначать как E . Под вектором (точкой) условимся понимать вектор-строку, чертой над вектором будем делать различие в обозначении между вектором и скаляром. Векторы из нулей и одной единицы на позиции i будем обозначать как \bar{e}_i . Линейную оболочку множества X будем обозначать как $\mathcal{L}(X)$.

Задача линейного программирования в стандартной форме имеет следующий вид:

$$\begin{cases} \bar{c}\bar{x}^t \rightarrow \min, \\ A\bar{x}^t \leq \bar{b}^t, \text{ где } A \in \mathbb{R}^{m,n}, \\ x_i \geq 0. \end{cases} \quad (1)$$

Покажем, как задачу (1) свести к задаче более простого вида:

$$\begin{cases} A\bar{x}^t = \bar{b}^t, \text{ где } A \in \mathbb{R}^{m,n} \text{ и } m < n, \\ \text{требуется найти } \bar{x}_0 \text{ такое, что } A\bar{x}_0^t = \bar{b}^t \text{ и } x_i \geq 0. \end{cases} \quad (2)$$

По сути, в (2) надо решить систему линейных уравнений неполного ранга в неотрицательных числах.

Шаг 1. Комбинация прямой и двойственной задачи. Рассмотрим задачу

$$\begin{cases} \bar{b}\bar{u}^t - \bar{c}\bar{x}^t = 0, \\ A\bar{x}^t \leq \bar{b}^t, \\ -A^t\bar{u}^t \leq -\bar{c}^t, \\ -x_i \leq 0, \quad -u_i \leq 0. \end{cases} \quad (3)$$

Из теории двойственности известно, что (3) имеет допустимое решение тогда и только тогда, когда в (1) найдется конечное оптимальное решение [4, с. 385].

Шаг 2. Добавление дополнительных переменных. Введём дополнительные неотрицательные переменные, чтобы в задаче (3) неравенства стали равенствами:

$$\begin{cases} \bar{b}\bar{u}^t - \bar{c}\bar{x}^t = 0, \\ A\bar{x}^t + \bar{y}^t = \bar{b}^t, \\ -A^t\bar{u}^t + \bar{v}^t = -\bar{c}^t, \\ -x_i \leq 0, \quad -u_i \leq 0, \quad -y_i \leq 0, \quad -v_i \leq 0. \end{cases} \quad (4)$$

После соответствующих переобозначений задача (4) сведётся к задаче (2), а любое её неотрицательное решение будет решением задачи (1) с точностью до переобозначений.

2. Геометрическая интерпретация алгоритма

На неотрицательную область пространства \mathbb{R}^n нам будет удобнее смотреть как на выпуклый конус $\mathcal{K} = \{\bar{x} \mid \bar{x} = a_1\bar{e}_1 + \dots + a_n\bar{e}_n, \text{ где } a_i \geq 0\}$, ограниченный гиперплоскостями $\mathcal{E}_i = \mathfrak{L}(\{\bar{e}_1, \dots, \bar{e}_n\} \setminus \{\bar{e}_i\})$. В дальнейшем нас будут интересовать линейные подпространства данных гиперплоскостей, порождённые векторами из $\{\bar{e}_i\}$, которые мы будем называть граничными К-плоскостями. Для каждой граничной К-плоскости введём квадратную диагональную матрицу $Z_i \in \mathbb{R}^{n,n}$, составленную из единичных и нулевых векторов-строк, причём на главной диагонали матрицы Z_i будет стоять единица, если данная единичная вектор-строка принадлежит граничной К-плоскости, и ноль в противном случае. Следовательно, ранг матрицы Z_i совпадает с размерностью граничной К-плоскости и равен К. Будем обозначать граничную К-плоскость, порождённую матрицей Z_i , как $\mathfrak{L}(Z_i)$, подразумевая, что граничная К-плоскость является линейной оболочкой строк матрицы Z_i . Множество решений системы линейных уравнений $A\bar{x}^t = \bar{b}^t$ обозначим как Ω .

Алгоритм решения системы линейных уравнений неполного ранга $A\bar{x}^t = \bar{b}^t$ в неотрицательных числах будет представлять из себя движение точки внутри конуса \mathcal{K} к неотрицательному решению системы. Для этого будем последовательно уменьшать расстояние между множеством решений системы линейных уравнений Ω и рассматриваемой на данном этапе точкой \bar{x}_{i-1} , не выходя за пределы неотрицательной области \mathcal{K} . Нам потребуется известная формула [5, с. 92] ортогональной проекции точки \bar{x}_0^t на множество решений системы линейных уравнений

$$\bar{x}_p^t = \bar{x}_0^t - A^t(AA^t)^{-1}A\bar{x}_0^t + A^t(AA^t)^{-1}\bar{b}^t. \quad (5)$$

Подойдя к какой-либо граничной К-плоскости $\mathfrak{L}(Z_i)$ положительной области \mathcal{K} , будем изменять направление движения вдоль $\mathfrak{L}(Z_i)$ в сторону уменьшения расстояния между $\mathfrak{L}(Z_i)$ и Ω . Ниже покажем, что для сходимости алгоритма достаточно брать в качестве направления движения ближайшую к \bar{x}_{i-1} точку рассматриваемой граничной К-плоскости $\mathfrak{L}(Z_i)$, расстояние от которой до Ω минимально на $\mathfrak{L}(Z_i)$. При достижении минимума расстояния от какой-нибудь граничной К-плоскости до Ω будем искать новое приближение к решению, уходя от данной граничной К-плоскости.

Соответственно, квадрат расстояния от точки $\bar{x} \in \mathfrak{L}(Z_i)$ до множества решений системы линейных уравнений $A\bar{x}^t = \bar{b}^t$ будет определяться формулой

$$\begin{aligned} \rho^2(Z_i\bar{x}^t, \Omega) &= \rho^2(Z_i\bar{x}^t, Z_i\bar{x}^t - A^t(AA^t)^{-1}AZ_i\bar{x}^t + A^t(AA^t)^{-1}\bar{b}^t) = \\ &= (A^t(AA^t)^{-1}AZ_i\bar{x}^t - A^t(AA^t)^{-1}\bar{b}^t)^2. \end{aligned} \quad (6)$$

Очевидно, что квадратичная форма (6) является неотрицательной полуопределённой, а значит, выпуклой. Таким образом, для любых двух точек $\bar{x}, \bar{y} \in \mathbb{R}^n$ и скаляра $\alpha \in [0, 1]$ выполняется неравенство Йенсена

$$\rho^2(\alpha\bar{x} + (1 - \alpha)\bar{y}, \Omega) \leq \alpha\rho^2(\bar{x}, \Omega) + (1 - \alpha)\rho^2(\bar{y}, \Omega).$$

Градиент же квадратичной формы (6) будет обращаться в ноль тогда и только тогда, когда расстояние от граничной К-плоскости $\mathfrak{L}(Z_i)$ до Ω будет минимальным. Несложно убедиться, что градиент рассматриваемой квадратичной формы имеет вид

$$\nabla\rho^2(Z_i\bar{x}_i^t, \Omega) = 2Z_iA^t(AA^t)^{-1}AZ_i\bar{x}_i^t - 2Z_iA^t(AA^t)^{-1}\bar{b}^t.$$

Приравняв градиент к нулю, получим систему линейных уравнений

$$Z_iA^t(AA^t)^{-1}AZ_i\bar{x}_i^t = Z_iA^t(AA^t)^{-1}\bar{b}^t.$$

3. Описание алгоритма

3.1. Начальные условия

Матрице Z_1 присвоим значение единичной матрицы. Присвоим начальное значение переменной $\bar{x}_0 = \bar{0}$.

В геометрической интерпретации точка \bar{x}_i является текущим приближением к неотрицательному решению, а точка \bar{y}_i будет ближайшей к \bar{x}_{i-1} точкой граничной К-плоскости $\mathfrak{L}(Z_i)$, расстояние от которой до Ω минимально. Присвоим i значение 1.

3.2. i -я итерация алгоритма

(а) Найдём ближайшее к точке \bar{x}_{i-1} решение системы $Z_i A^t (A A^t)^{-1} A Z_i \bar{x}^t = Z_i A^t (A A^t)^{-1} \bar{b}^t$ и обозначим его как \bar{y}_i .

(б) Переменной \bar{x}_i присвоим такую точку отрезка $(1 - \alpha_i)\bar{x}_{i-1} + \alpha_i \bar{y}_i$, при которой значение α_i является наиболее близким к единице, а точка \bar{x}_i остаётся неотрицательной.

(в) Если точка \bar{x}_i удовлетворяет системе $A^t (A A^t)^{-1} A \bar{x}^t = A^t (A A^t)^{-1} \bar{b}^t$, то выходим, так как мы нашли искомую неотрицательную точку.

(г) Если $\alpha_i = 1$, то мы нашли точку минимума расстояния от граничной K -плоскости $\mathfrak{L}(Z_i)$ до Ω , тогда матрице Z_{i+1} присвоим значение единичной матрицы, увеличим i на единицу и перейдём к шагу (а).

(д) Присвоим Z_{i+1} значение Z_i . На диагональной матрице Z_{i+1} обнулیم те координаты главной диагонали, при индексах которых координаты вектора \bar{y}_i меньше или равны нулю, а координаты вектора \bar{x}_i равны нулю. Тем самым мы не будем пересекать граничную K -плоскость $\mathfrak{L}(Z_i)$, а продолжим движение вдоль граничной K -плоскости $\mathfrak{L}(Z_{i+1})$ меньшей размерности.

(е) Увеличим i на единицу и перейдём к шагу (а).

4. Обоснование алгоритма

Рассмотрим поведение работы алгоритма более подробно. Для начала поймём, что происходит, если исходная задача не имеет решений (допустимое множество пусто или целевая функция неограниченно убывает). В этом случае Ω не будет иметь общих точек с выпуклым конусом \mathcal{K} , а значит, в ходе работы алгоритма мы придём в такую точку $\bar{x}_k \in \mathcal{K}$, расстояние от которой до Ω будет минимальным среди точек \mathcal{K} . Так как мы пытаемся двигаться в сторону уменьшения расстояния, то алгоритм «заиклится», что будет свидетельствовать об отсутствии решения.

Для сходимости алгоритма необходимо показать, что не произойдёт «заикливания» при наличии допустимого решения и мы гарантированно сделаем шаг в сторону уменьшения расстояния. Интуитивно этот факт можно понять из соображений, что из точки \bar{x}_{i-1} мы пытаемся совершить движение в сторону ближайшей точки K -плоскости $\mathfrak{L}(Z_i)$, на которой достигается минимум расстояния между $\mathfrak{L}(Z_i)$ и Ω . Поэтому если движение нам преграждает какая-нибудь граничная K -плоскость, то при наличии общей точки $\mathfrak{L}(Z_i)$ и Ω должна остаться возможность для движения хотя бы в сторону проекции на пересечение данной граничной K -плоскости и $\mathfrak{L}(Z_i)$. Строгое доказательство того, что возможность движения не только существует, но и алгоритм гарантированно выполняет нужный для предотвращения заикливания шаг, приведено в теореме 1.

Среди итераций алгоритма можно выделить множество таких, на которых достигается минимум расстояния от какой-нибудь K -плоскости $\mathfrak{L}(Z_i)$ до Ω . Очевидно, что такие итерации будут встречаться не реже чем через каждые n итераций, а покинув K -плоскость $\mathfrak{L}(Z_i)$, отвечающую подобной итерации, мы уже не вернёмся на неё. Так как всего Z_i матриц у нас 2^n , то не более чем за $2^n n$ шагов алгоритм гарантированно обойдёт все множества $\mathfrak{L}(Z_i)$, а значит, дойдёт до точного решения задачи 2.

Теорема 1. *Если положительное решение существует, то алгоритм сходится за конечное число шагов.*

Доказательство. Среди всех итераций выделим упорядоченное подмножество Λ , для которого $Z_{i \in \Lambda} = E$. Для удобства определим Z_0 как нулевую матрицу, тогда точка x_{i-1} , где $i \in \Lambda$, является точкой минимума функции (6) на линейном пространстве $\mathfrak{L}(Z_{i-1})$. Рассмотрим два соседних индекса итераций $i, k \in \Lambda$, причём $k > i$. Если мы покажем, что $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{x}_{k-1}, \Omega)$, то мы покажем, что в ходе работы алгоритма происходит постепенное приближение к Ω . Если для итерации $j \in \{i, i+1, \dots, k-1\}$ точка $\bar{x}_j \neq \bar{x}_{i-1}$, то мы совершили движение в сторону уменьшения расстояния, а значит, $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{x}_{k-1}, \Omega)$. Следовательно, нам надо показать, что существует $j \in \{i, i+1, \dots, k-1\}$, для которого $\bar{x}_{i-1} \neq \bar{x}_j$.

В лемме 2 покажем, что для индекса $j \in \{i, i+1, \dots, k-1\}$, при котором $\bar{x}_{j-1} = \bar{x}_{i-1}$, вектор $\bar{y}_i - \bar{x}_{i-1}$ не ортогонален пространству $\mathfrak{L}(Z_{j+1})$. Отсюда следует, что если мы обозначим проекцию точки \bar{y}_i на множество $\mathfrak{L}(Z_{j+1})$ как \bar{h} , то мы получим невырожденный прямоугольный треугольник $\Delta \bar{y}_i \bar{h} \bar{x}_{i-1}$, а значит, $\rho(\bar{x}_{i-1}, \Omega) \neq \min_{\bar{z} \in \mathfrak{L}(Z_{j+1})} \rho(\bar{z}, \Omega)$. Так как $\rho(\bar{x}_{i-1}, \Omega) = \min_{\bar{z} \in \mathfrak{L}(Z_{i-1})} \rho(\bar{z}, \Omega)$, то $\mathfrak{L}(Z_{j+1}) \not\subseteq \mathfrak{L}(Z_{i-1})$. Отсюда следует, что существует $j \in \{i, i+1, \dots, k-1\}$, такое, что $\bar{x}_{i-1} \neq \bar{x}_j$, а значит, $\bar{x}_{i-1} \neq \bar{x}_{k-1}$.

Покажем теперь, что алгоритм закончит работу за конечное число итераций. Действительно, для любого $i \in \Lambda$ выполняется соотношение $\rho(\bar{x}_{i-1}, \Omega) = \min_{\bar{z} \in \mathfrak{L}(Z_{i-1})} \rho(\bar{z}, \Omega)$. Но так как мы постепенно уменьшаем расстояние до Ω , то за конечное число шагов мы найдём минимум на всех K -плоскостях, а значит, доберёмся и до положительного решения за конечное число шагов. \square

Лемма 1. Пусть на итерации i матрица $Z_i = E$. Если неотрицательное решение существует, то $\mathfrak{L}(Z_{i+1})$ не ортогональна вектору $\vec{h} = \bar{y}_i - \bar{x}_{i-1}$.

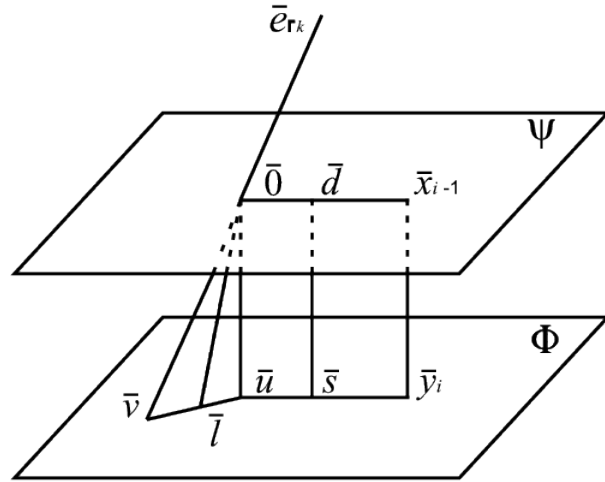
Доказательство. Предположим, что \vec{h} ортогонален $\mathfrak{L}(Z_{i+1})$. Через Ω проведём гиперплоскость Φ , ортогональную вектору \vec{h} . Через $\mathfrak{L}(Z_{i+1})$ проведём гиперплоскость Ψ , ортогональную вектору \vec{h} . В силу построений $\Phi \cap \Psi = \emptyset$. Рассмотрим конус неотрицательной области $\mathcal{K} = \{\bar{x} \mid \bar{x} = a_1 \bar{e}_1 + \dots + a_n \bar{e}_n, \text{ где } a_i \geq 0\}$. В дальнейшем покажем, что Φ и \mathcal{K} лежат по разные стороны от гиперплоскости Ψ (причём \mathcal{K} и Ψ могут иметь общие точки). Отсюда будет следовать, что $\Omega \cap \mathcal{K} = \emptyset$. Тем самым мы покажем, что если существует неотрицательное решение, принадлежащее конусу \mathcal{K} , то $\mathfrak{L}(Z_{i+1})$ не ортогональна вектору \vec{h} . Геометрические построения при доказательстве леммы 1 можно отследить по рисунку на с. 73.

Рассмотрим множество Γ' тех индексов, для координат которых мы обнуляли строки матрицы Z_{i+1} на шаге (д) алгоритма. Исключим из Γ' все индексы, единичные векторы которых лежат в Ψ , и обозначим данное множество как Γ . Фиксируем какой-нибудь индекс $\Gamma_k \in \Gamma$. Пусть $\bar{v} = \Phi \cap \lambda \bar{e}_{\Gamma_k}$, где $\lambda \in \mathbb{R}$. Заметим, что \bar{v} существует, так как иначе $\bar{e}_{\Gamma_k} \in \Psi$, а мы исключили этот случай.

Произведём перенос точки \bar{x}_{i-1} в точку $\bar{0} \in \Psi$. Точка $\bar{y}_i \in \Phi$ тогда перейдёт в новую точку $\bar{u} \in \Phi$, которая является проекцией точки $\bar{0}$ на Φ . Так как $\Psi \parallel \Phi$, то для любой точки \bar{g} из Ψ её проекция на Φ будет вычисляться как $\bar{g} + (\bar{y}_i - \bar{x}_{i-1})$. Покажем, что знак Γ_k -й координаты точки \bar{y}_i и знак Γ_k -й координаты точки \bar{u} совпадают. Если это не так, то существует точка $\bar{s} \in [\bar{y}_i, \bar{u}] \subseteq \Phi$, у которой Γ_k -я координата равна нулю. Пусть \bar{d} — это проекция \bar{s} на $\mathfrak{L}(Z_{i+1})$. Точка \bar{d} принадлежит области $(\bar{x}_{i-1}, \bar{0})$ в силу того, что $\bar{0} \in \mathfrak{L}(Z_{i+1}) \subseteq \Psi$ и $\bar{x}_{i-1} \in \mathfrak{L}(Z_{i+1}) \subseteq \Psi$. Заметим, что Γ_k -я координата вектора $(\bar{s} - \bar{d})$ нулевая, а Γ_k -я координата вектора $(\bar{y}_i - \bar{x}_{i-1})$ отрицательная, но этого не может быть, так как $\Psi \parallel \Phi$. Следовательно, знак Γ_k -й координаты точки \bar{u} совпадает со знаком Γ_k -й координаты точки \bar{y}_i и отрицателен.

Покажем, что и Γ_k -я координата точки $\bar{v} = \Phi \cap \lambda \bar{e}_{\Gamma_k}$ также отрицательна. В силу того, что $\Psi \cap \Phi = \emptyset$, точка $\bar{v} \neq \bar{0}$. Пусть Γ_k -я координата точки \bar{v} положительна, тогда среди внутренних точек отрезка $[\bar{u}, \bar{v}]$ существует точка \bar{l} с нулевой Γ_k -й координатой. Рассмотрим треугольники $\Delta \bar{0}\bar{u}\bar{v}$, $\Delta \bar{0}\bar{u}\bar{l}$ и $\Delta \bar{0}\bar{l}\bar{v}$. Угол $\bar{0}\bar{u}\bar{l}$ прямой, следовательно, угол $\bar{0}\bar{l}\bar{v}$ тупой. Так как Γ_k -я координата точки \bar{l} нулевая, то $(\bar{0} - \bar{l}, \bar{e}_{\Gamma_k}) = 0$. В силу этого угол $\bar{v}\bar{0}\bar{l}$ прямой. Отсюда следует, что сумма углов треугольника $\bar{0}\bar{l}\bar{v}$ больше 180° . Пришли к противоречию.

Так как Γ_k -я координата точки \bar{v} отрицательна, то луч $\lambda \bar{e}_{\Gamma_k}$, где $\lambda \geq 0$, лежит на противоположной Ω стороне относительно гиперплоскости Ψ (возможны общие точки Ψ и $\lambda \bar{e}_{\Gamma_k}$, где $\lambda \geq 0$). Данное утверждение справедливо для всех лучей, образованных единичными векторами с индексами из Γ . Обозначим выпуклый конус, порождённый единичными векторами с индексами из Γ' , как \mathfrak{G} . Очевидно, что $\mathcal{K} \subseteq \{\bar{x} \mid \bar{x} = \bar{l} + \bar{g}, \bar{l} \in \mathfrak{L}(Z_{i+1}), \bar{g} \in \mathfrak{G}\}$. Отсюда следует, что $\mathcal{K} \cap \Omega = \emptyset$, а значит, если существует неотрицательное решение, принадлежащее конусу \mathcal{K} , то $\mathfrak{L}(Z_{i+1})$ не ортогонально вектору \bar{h} .



Графическое построение леммы 1

□

Лемма 2. Пусть на итерации i матрица $Z_i = E$. Тогда если неотрицательное решение существует, то для итерации $j \geq i$, на которой $\bar{x}_j = \bar{x}_{i-1}$, граничная K -плоскость $\mathfrak{L}(Z_{j+1})$ не ортогональна вектору $\bar{h} = \bar{y}_i - \bar{x}_{i-1}$.

Доказательство. Доказательство проведём по индукции. Базой индукции послужит случай $j = i$, доказанный в лемме 1. Пусть верно для $j = k - 1$. Покажем для $j = k$. Отметим, что точки \bar{y}_i и \bar{y}_j не совпадают по построению, точка $\bar{y}_i \in \Omega$.

Обозначим проекцию точки \bar{y}_i на $\mathfrak{L}(Z_j)$ как \bar{y}_i^\perp . Из формулы (5) следует, что такая проекция строится обнулением тех координат, которые обнулены в матрице Z_j . Из предположения индукции следует, что $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{y}_i^\perp, \Omega)$ как отношение длины катета и гипотенузы. Следовательно, $\bar{x}_{i-1} \neq \bar{y}_i^\perp$. Заметим, что отрезок $[\bar{y}_j, \bar{y}_i^\perp]$ лежит в $\mathfrak{L}(Z_j)$, а для функции (6) на линейном пространстве \mathbb{R}^n выполняются следующие неравенства $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{y}_j, \Omega)$.

Далее покажем, что на K -плоскости $\mathfrak{L}(Z_{j+1})$ лежит точка, расстояние от которой до Ω меньше, чем $\rho(\bar{x}_{i-1}, \Omega)$. Отсюда сразу же будет следовать, что граничная K -плоскость $\mathfrak{L}(Z_{j+1})$ не ортогональна вектору $\bar{h} = \bar{y}_i - \bar{x}_{i-1}$.

Пусть множество Θ — это множество тех индексов единичных векторов \bar{e}_i , которые входят в строки матрицы Z_j , но отсутствуют среди строк матрицы Z_{j+1} . Заметим, что координаты с индексами из Θ точки \bar{y}_j неположительны по условиям работы алгоритма. Векторное подпространство $\mathfrak{L}(Z_j) \subseteq \mathfrak{L}(Z_{i+1})$, проекция вектора \bar{y}_i на $\mathfrak{L}(Z_{i+1})$ строится обнулением соответствующих координат, а координаты с индексами из Θ проекции \bar{y}_i на $\mathfrak{L}(Z_{i+1})$ неотрицательны в силу построений леммы 1, следовательно, координаты с индексами из Θ точки \bar{y}_i^\perp неотрицательны.

Так как $Z_j \neq Z_{j+1}$, то мощность Θ больше нуля. Проведём процедуру, уменьшающую мощность Θ так, чтобы построить точку из $\mathfrak{L}(Z_{j+1})$, в которой значение

функции (6) меньше, чем в точке \bar{x}_{i-1} . Для этого введём следующие обозначения: $\bar{v}_1 = \bar{y}_j$, $\bar{u}_1 = \bar{y}_i^\perp$, $Z'_1 = Z_j$.

После переобозначений получим следующие необходимые для проведения процедуры уменьшения мощности Θ условия: отрезок $[\bar{v}_1, \bar{u}_1] \subseteq \mathfrak{L}(Z'_1)$, расстояние $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{v}_1, \Omega)$, расстояние $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{u}_1, \Omega)$, координаты с индексами из Θ точки \bar{v}_1 неположительны, а координаты с индексами из Θ точки \bar{u}_1 неотрицательны.

Процедура, уменьшающая мощность Θ : найдём точку \bar{v}_2 отрезка $[\bar{v}_1, \bar{u}_1]$, которая первой при движении от точки \bar{v}_1 в сторону точки \bar{u}_1 обнуляет хотя бы одну координату, соответствующую индексам из Θ . Удалим из множества Θ индексы, которые обнулились в точке \bar{v}_1 . Обозначим матрицу Z'_2 как матрицу, совпадающую с матрицей Z_{j+1} . Добавим Z'_2 единичные векторы-строки, индексы которых присутствуют в Θ . Далее обозначим проекцию точки \bar{y}_i на $\mathfrak{L}(Z'_2)$ как \bar{u}_2 .

Покажем, что у нас выполнены все условия, чтобы повторить процедуру для отрезка $[\bar{v}_2, \bar{u}_2] \subseteq \mathfrak{L}(Z'_2)$. Действительно, так как выполняется неравенство Йенсена, то $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{v}_2, \Omega)$. Отсюда следует, что на $\mathfrak{L}(Z'_2)$ есть точка минимума функции (6), не совпадающая с \bar{x}_{i-1} , следовательно, $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{u}_2, \Omega)$, а значит, $\bar{x}_{i-1} \neq \bar{u}_2$. Координаты с индексами из Θ точки \bar{v}_2 неположительны по построению. Векторное подпространство $\mathfrak{L}(Z'_2) \subseteq \mathfrak{L}(Z_{i+1})$, проекция вектора \bar{y}_i на $\mathfrak{L}(Z_{i+1})$ строится обнулением соответствующих координат, а координаты с индексами из Θ проекции \bar{y}_i на $\mathfrak{L}(Z_{i+1})$ неотрицательны в силу построений леммы 1, следовательно, координаты с индексами из Θ точки \bar{u}_2 неотрицательны.

Пусть на повторе процедуры под номером s мы пришли к ситуации, когда мощность Θ равна нулю. Очевидно, что s конечна, так как конечна мощность Θ и на каждом повторе процедуры гарантированно происходит уменьшение мощности Θ . Заметим, что на повторе s матрица $Z'_s = Z_{j+1}$, следовательно, $\bar{v}_{s+1} \in [\bar{v}_s, \bar{u}_s] \subseteq \mathfrak{L}(Z_{j+1})$. По построению $\rho(\bar{x}_{i-1}, \Omega) > \rho(\bar{v}_{s+1}, \Omega)$, поэтому граничная K -плоскость $\mathfrak{L}(Z_{j+1})$ не ортогональна вектору $\vec{h} = \bar{y}_i - \bar{x}_{i-1}$. \square

5. Численные эксперименты

Целью численных экспериментов ставились поиск задач, на которых алгоритм показал бы время работы, наиболее близкое к полученной оценке, и проверка алгоритма на устойчивость. Для этих целей алгоритм был проверен на задаче Кли — Минти [6], на данных которой симплекс-метод Данцига совершает обход по всем вершинам симплекса за 2^n шагов, а также и на группе случайных тестов.

Реализация алгоритма была осуществлена без потери точности в классе рациональных чисел библиотеки GMP [7]. Для проверки численной устойчивости в тестах отслеживались два параметра — β и γ , где β — это минимальное количество бит, необходимых для кодирования любой переменной матрицы входных данных $A^t(AA^t)^{-1}A$, а γ — это максимальное количество бит, которое потребовалось для кодирования любой промежуточной координаты вектора \bar{x}_i , вектора \bar{y}_i или коэффициента α из шага (б) алгоритма.

Результаты практической проверки для задачи Кли — Минти сведены в табл. 1. Первая строка n таблицы содержит размерность задачи Кли — Минти, во второй строке S приводится экспериментальное число шагов предложенного алгоритма для данной размерности. Данные третьей и четвёртой строк соответствуют экспериментальным данным по параметрам β и γ , а пятая строка является отношением γ/β , которое позволяет определить соотношение количества бит, необходимых для кодирования промежуточных и входных данных.

Таблица 1

n	9	10	11	12	13	14	15	16	17	18	19	20
S	54	63	72	82	92	103	114	126	138	151	164	178
β	120	134	158	170	200	220	236	262	276	303	330	348
γ	224	270	302	346	390	436	468	522	563	604	655	697
$\frac{\gamma}{\beta}$	1.86	2.01	1.91	2.03	1.95	1.98	1.98	1.99	2.03	1.99	1.98	2.00

Для инициализации случайных тестов квадратная матрица размерами n на n , отвечающая системе линейных неравенств, и вектор целевой функции заполнялись случайными знаковыми 32-битными целыми. Сгенерированная таблица симплекс-методом Данцига проверялась на наличие решения и в случае существования оптимума отдавалась на проверку предложенному в работе алгоритму. Для каждого рассматриваемого размера проводилось по тысяче случайных тестов. Результаты тестов сведены в табл. 2. Первая строка n таблицы содержит размерность случайной задачи. Во второй строке S приводится максимальное экспериментальное число шагов алгоритма для данной размерности, которое было обнаружено при тестировании. Данные третьей и четвёртой строк соответствуют экспериментальным данным по параметрам β и γ , которые отвечают максимальному обнаруженному значению дроби γ/β для данной размерности, значение которой занесено в пятую строку.

Таблица 2

n	7	8	9	10	11	12	13	14	15
S	45	50	54	63	61	72	77	82	89
β	768	912	1028	1136	1256	1368	1504	1624	1736
γ	2822	3227	3837	4027	4527	4418	4630	5424	5879
$\frac{\gamma}{\beta}$	3.67	3.53	3.73	3.54	3.60	3.22	3.07	3.34	3.38

Низкое значение дроби γ/β и отсутствие роста её значения при увеличении размерности задачи на случайных экспериментальных данных свидетельствуют о хорошей численной устойчивости предлагаемого алгоритма.

Из табл. 1 видно, что на данных задачи Кли — Минти предложенный алгоритм показывает преимущество по количеству шагов над алгоритмом симплекс-метода Данцига. Из данных табл. 2 видно, что набору случайных тестов не удалось найти задачу, которая по числу итераций приближалась бы к значению $2^n n$. Однако результаты данных тестов нельзя интерпретировать в пользу полиномиальности предлагаемого алгоритма, так как и симплекс-метод Данцига, являющийся полиномиальным в среднем алгоритмом [2, с. 142–147], тоже не показал на этих тестах «плохих» результатов. Поэтому вопрос о наличии примеров, подтверждающих экспоненциальность предложенного алгоритма, остаётся открытым.

Список литературы

1. **Дикин, И. И.** Итеративное решение задач линейного и квадратичного программирования / И. И. Дикин // Докл. АН СССР. — 1967. — Т. 174, № 4. — С. 745–747.
2. **Schrijver, A.** Theory of Linear and Integer Programming / A. Schrijver. — Wiley-Interscience series in discrete mathematics. — Chichester : John Wiley & Sons, 1986. — xi+471 p.
3. **Моллаверди, Н.** Методы решения задач линейной оптимизации большой размерности : дис... канд. физ.-мат. наук / Н. Моллаверди. — М., 2005. — 96 с.

4. **Karmarkar, N.** A new polynomial-time algorithm for linear programming / N. Karmarkar // *Combinatorica*. — 1984. — № 4. — P. 373–395.
5. **Розенфельд, Б. А.** Многомерные пространства / Б. А. Розенфельд. — М. : Наука, 1966. — 648 с.
6. **Klee, V.** How good is the Simplex algorithm? / V. Klee, G. J. Minty // *Inequalities: III*, ed. O. Shisha. — New York : Academic Press, 1972. — P. 158–175.
7. **Nikolaevskaya, E.** Programming with Multiple Precision / E. Nikolaevskaya, A. Khimich, T. Chistyakova. — *Studies in Computational Intelligence*. — Vol. 397. — Richmond, USA : Springer, 2012. — 234 p.

Поступила в редакцию 08.04.2016

После переработки 21.05.2016

Сведения об авторах

Севостьянов Егор Николаевич, ведущий инженер-программист (руководитель группы математического моделирования), ЗАО «Диджитал Айрон Пайп», Челябинск, Россия; e-mail: y384@mail.ru.

Лепчинский Михаил Германович, кандидат физико-математических наук, доцент кафедры вычислительной математики, Челябинский государственный университет, Челябинск, Россия; e-mail: myth@csu.ru.

Chelyabinsk Physical and Mathematical Journal. 2016. Vol. 1, iss. 2. P. 68–77.

ALGORITHM FOR FINDING OF A NON-NEGATIVE SOLUTION FOR LINEAR SYSTEM OF EQUATIONS

E.N. Sevostyanov^{1,a}, M.G. Leptchinski^{2,b}

¹ZAO "Digital Iron Pipe", Chelyabinsk, Russia

²Chelyabinsk State University, Chelyabinsk, Russia

^ay384@mail.ru; ^bmyth@csu.ru

The new algorithm for finding of the exact solution for the linear programming problem is constructed. The method is based on the converting of the linear programming problem to the problem of solving in nonnegative numbers for a system of linear equations with an incomplete rank. This problem is solved by successive approximations in specially constructed subspaces. The convergence of the algorithm for a finite number of the steps is proved. The algorithm has been tested on the data of the Klee – Minty problem and on a set of random tests.

Keywords: *algorithm of linear programming, non-negative solution of linear system of equations, linear inequality.*

References

1. **Dikin I.I.** Iterativnoe reshenie zadach lineynogo i kvadraticnogo programmirovaniya [Iterative solving of linear and quadratic programming problems]. *Doklady Akademii Nauk SSSR* [USSR Academy of Science Reports], 1967, vol. 174, no. 4, pp. 745–747. (In Russ.).
2. **Schrijver A.** *Theory of Linear and Integer Programming*. Wiley-Interscience series in discrete mathematics. Chichester, John Wiley & Sons, 1986. xi+471 p.

3. **Mollaverdi N.** *Metody resheniya zadach lineynoy optimizatsii bol'shoy razmernosti* [Methods for solving of linear optimization problems with large dimension. Thesis]. Moscow, 2005. 96 p. (In Russ.).
4. **Karmarkar N.** A new polynomial-time algorithm for linear programming. *Combinatorica*, 1984, no. 4, pp. 373–395.
5. **Rozenfel'd B.A.** *Mnogomernye prostranstva* [Multidimensional spaces]. Moscow, Nauka Publ., 1966. 648 p. (In Russ.).
6. **Klee V, Minty G.J.** How good is the Simplex algorithm? *Inequalities: III*. New York, Academic Press, 1972. Pp. 158–175.
7. **Nikolaevskaya E., Khimich A., Chistyakova T.** *Programming with Multiple Precision*. Studies in Computational Intelligence, vol. 397. Richmond, USA, Springer, 2012. 234 p.

Accepted article received 08.04.2016

Corrections received 21.05.2016