# Исследование параллельного алгоритма для решения задач линейного программирования на основе фейеровских отображений*

А.С. Шелудько, Л.Б. Соколинский

Работа посвящена вопросам применения фейеровских отображений для решения задач линейного программирования (ЛП) большой размерности на многопроцессорных системах с массовым параллелизмом. Предлагается новый параллельный итерационный алгоритм решения задачи ЛП, в основе которого лежит S-технология. Исследуется реализация предложенного алгоритма. Рассматривается поведение итерационного процесса в зависимости от выбора базовых параметров алгоритма. Обсуждается проблема выбора оптимальных значений параметров, обеспечивающих эффективную работу алгоритма.

## 1. Введение

В практике экономико-математического моделирования часто встречаются задачи линейного программирования большой размерности, требующие значительных вычислительных ресурсов для их решения. Ситуация может усугубляться тем, что система исходных данных задачи может эволюционировать с течением времени. В такой ситуации оказываются перспективными итерационные методы, базирующиеся на фейеровских отображениях [1]. Однако подобные алгоритмы характеризуются большой временной трудоемкостью. В соответствие с этим является актуальной задача построения параллельного алгоритма решения задач линейного программирования на основе фейеровских отображений, ориентированного на многопроцессорные системы с массовым параллелизмом.

## 2. Описание параллельного алгоритма

Предлагаемый параллельный алгоритм базируется на S-технологии [2]. Запишем задачу ЛП:

$$L: \quad \max\{(\boldsymbol{c},\boldsymbol{x}) \mid A\boldsymbol{x} \le \boldsymbol{b},\ \boldsymbol{x} \ge \boldsymbol{0}\}.$$

Здесь

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix},\ \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix},\ \boldsymbol{b} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{pmatrix},\ \boldsymbol{c} = \begin{pmatrix} c_1 \\ c_2 \\ \dots \\ c_n \end{pmatrix}.$$

Обозначим $i$-ю строку матрицы $A$ как $\boldsymbol{a}_i$, а $j$-й столбец как $\boldsymbol{a}^j$. Решение исходной задачи ЛП получается в результате выполнения итерационного процесса, в ходе которого циклически вычисляются значения следующих трех фейеровских отображений:

$$\tilde{\boldsymbol{x}} = \varphi_1(\boldsymbol{x}) = \boldsymbol{x} - (\lambda_1/\delta_1)\sum_{i=1}^{m} l_i^+(\boldsymbol{x})\boldsymbol{a}_i,\ \text{где}\ \delta_1 = \sum_{i=1}^{m}\|\boldsymbol{a}_i\|^2\ \text{и}\ l_i^+(\boldsymbol{x}) = \max\{(\boldsymbol{a}_i,\boldsymbol{x}) - b_i, 0\} \tag{1}$$

$$\tilde{\boldsymbol{u}} = \varphi_2(\boldsymbol{u}) = \boldsymbol{u} + (\lambda_2/\delta_2)\sum_{j=1}^{n} h_j^+(\boldsymbol{u})\boldsymbol{a}^j,\ \text{где}\ \delta_2 = \sum_{j=1}^{n}\|\boldsymbol{a}^j\|^2\ \text{и}\ h_j^+(\boldsymbol{u}) = \max\{c_j - (\boldsymbol{a}^j,\boldsymbol{u}), 0\} \tag{2}$$

* Работа поддержана грантами РФФИ 06-01-00380 и ФАНИ НШ-5595.2006.1.

$$\varphi_3(\tilde{\boldsymbol{x}},\tilde{\boldsymbol{u}}) = [\tilde{\boldsymbol{x}},\tilde{\boldsymbol{u}}] - \frac{(\boldsymbol{c},\tilde{\boldsymbol{x}}) - (\boldsymbol{b},\tilde{\boldsymbol{u}})}{\|\boldsymbol{c}\|^2 + \|\boldsymbol{b}\|^2}[\boldsymbol{c},-\boldsymbol{b}] \tag{3}$$

Параметры $\lambda_1, \lambda_2 \in (0,2)$ задают коэффициенты релаксации.

Построим параллельный алгоритм [3]. Зададим в пространстве $R^n$ канонический базис $E = \{\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_n\}$. Пусть $E_1, E_2, ..., E_r$ – подбазисы базиса $E$, обладающие свойствами $\bigcup_{i=1}^{r} E_i = E$ и $E_i \bigcap E_j = \varnothing$ для всех $i \neq j$. Определим $L_i = \langle E_i \rangle$ – линейная оболочка подбазиса $E_i$ $(i = 1, \ldots, r)$. Тогда $\forall \boldsymbol{x} \in R^n \ \exists\, \boldsymbol{x}_1 \in L_1, \ldots, \boldsymbol{x}_r \in L_r : \boldsymbol{x} = \boldsymbol{x}_1 + \ldots + \boldsymbol{x}_r$.

Обозначим через $\pi_i$ проекцию $R^n$ на подпространство $L_i$ $(i = 1, \ldots, r)$, так что $\forall \boldsymbol{x} \in R^n$ имеем $\pi_i(\boldsymbol{x}) = \boldsymbol{x}_i \in L_i$. Определим совокупность отображений $\varphi_{1i}(\boldsymbol{x}) : R^n \to R^n$ $(i = 1, \ldots, r)$ следующим образом:

$$\varphi_{1i}(\mathbf{x}) = \pi_{1i}(\varphi_1(\mathbf{x})) + \sum_{l \neq i} \pi_{1l}(\mathbf{x}). \tag{4}$$

Выполнив аналогичные построения для двойственной переменной $\boldsymbol{u}$ в пространстве $R^m$ в предположении, что канонический базис этого пространства разбит на $s$ подбазисов, мы можем определить совокупность отображений $\varphi_{2j}(\boldsymbol{u}) : R^m \to R^m$ $(j = 1, ..., s)$ следующим образом:

$$\varphi_{2j}(\boldsymbol{u}) = \pi_j(\varphi_2(\boldsymbol{u})) + \sum_{i \neq j} \pi_i(\boldsymbol{x}). \tag{5}$$

Пусть задано начальное приближение $[\boldsymbol{x}_0, \boldsymbol{u}_0]$. Построим в пространстве $R^{n+m}$ последовательность точек $[\boldsymbol{x}_0, \boldsymbol{u}_0], \ldots, [\boldsymbol{x}_l, \boldsymbol{u}_l], [\boldsymbol{x}_{l+1}, \boldsymbol{u}_{l+1}], \ldots$. Эта последовательность вычисляется в результате циклического повторения следующих шагов:

1) $\boldsymbol{x}'_{li} = \varphi_{1i}^{k_1}(\boldsymbol{x}_l)$ для $i = 1, \ldots, r$;

2) $\boldsymbol{u}'_{lj} = \varphi_{2j}^{k_2}(\boldsymbol{u}_l)$ для $j = 1, \ldots, s$;

3) $\boldsymbol{x}''_l = \sum_{i=1}^{r} \pi_i(\boldsymbol{x}'_{li})$;

4) $\boldsymbol{u}''_l = \sum_{j=1}^{s} \pi_j(\boldsymbol{u}'_{lj})$;

5) $[\boldsymbol{x}_{l+1}, \boldsymbol{u}_{l+1}] = \varphi_3(\boldsymbol{x}''_l, \boldsymbol{u}''_l)$.

Отличительной особенностью описанного алгоритма является то, что $r$ точек $\boldsymbol{x}'_{l1}, \ldots, \boldsymbol{x}'_{lr}$ на шаге 1 могут вычисляться независимо друг от друга. То же самое мы можем сказать о $s$ точках $\boldsymbol{u}'_{l1}, \ldots, \boldsymbol{u}'_{ls}$, которые вычисляются на шаге 2. При этом шаги 1 и 2 в целом не зависят друг от друга. Это позволяет при выполнении шагов 1-2 организовать $r + s$ независимых процессов, не требующих между собой обменов данными. По построению алгоритма $r \leq n$ и $s \leq m$, следовательно, мы вправе ожидать, что для больших значений $n$ и $m$ описанный алгоритм может быть эффективно реализован на многопроцессорных системах кластерного типа.

## 3. Результаты вычислительных экспериментов

Для проведения численных экспериментов построим задачу ЛП, в которой ограничения образуют пирамиду относительно первой переменной:

$$A=\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & \dots & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}_{99\times 100},$$

$$b=\begin{pmatrix} 200 & 200 & \dots & 200 \end{pmatrix}_{1\times 99},$$

$$c=\begin{pmatrix} 0 & 0 & \dots & 0 & 100 & 100 & \dots & 100 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{pmatrix}_{1\times 100},$$

$$x_0=\begin{pmatrix} 200 & 200 & \dots & 200 \end{pmatrix}_{1\times 100},$$

$$u_0=\begin{pmatrix} 0 & 0 & \dots & 0 \end{pmatrix}_{1\times 99}.$$

При проведении вычислительных экспериментов были взяты следующие коэффициенты релаксации: $\lambda_1=0.5$, $\lambda_2=1.5$. На Рис. 1 представлены результаты вычислительного эксперимента, в котором исследовалось отношение суммарного числа операций, выполняемых всеми процессорами, к общему числу обменов в зависимости от значения параметра $k_2$. Эксперимент проводился для трех различных значений $s$. При этом количество используемых процессоров всегда совпадало с количеством подбазисов $s$.

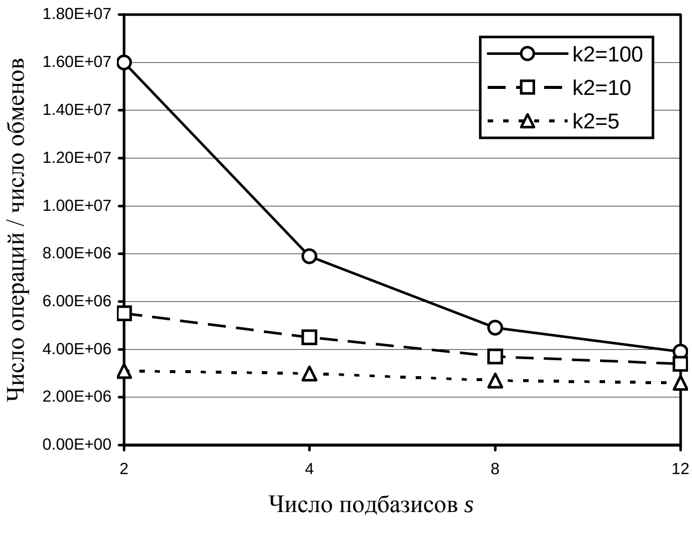


**Рис. 1.** Отношение числа операций к числу обменов

Проведенный эксперимент показал, что при правильном выборе значения параметра $k_2$ мы можем добиться сбалансированной загрузки процессоров, изменяя количество подбазисов.

## Литература

1. Еремин И.И. Применение метода фейеровских приближений к решению задач выпуклого программирования с негладкими ограничениями // Журнал вычислительной математики и математической физики. –Т. 9, № 5. -1969. -С. 1153-1160.
2. Васин В.В., Еремин И.И. Операторы и итерационные процессы фейеровского типа. Теория и приложения. – Екатеринбург: УрО РАН, 2005. -210 с.
3. Соколинский Л.Б., Цымблер Н.Ю. Параллельный алгоритм решения задач линейного программирования на базе фейеровских отображений. Тех. отчет РФФИ#06-01-00380/Y1N1. -Челябинск: ЮУрГУ, 2006. [http://life.susu.ru/reports/NZymbler-TR-06-01-00380-Y1N1.pdf]