УДК 519.6

Параллельный алгоритм для решения задачи линейного программирования в условиях неполных данных*

И.М. Соколинская, Л.Б. Соколинский

Parallel algorithm for solving linear programming problem under conditions of incomplete data

I.M. Sokolinskaya, L.B. Sokolinsky

Аннотация

В работе предлагается подход к решению задачи линейного программирования при наличии неформализованных ограничений, ориентированный на многопроцессорные вычислительные системы с массовым параллелизмом. Описывается параллельный алгоритм, базирующийся на синтезе методов линейного программирования и дискриминантного анализа. Приводятся результаты численных экспериментов, подтверждающие эффективность предложенного подхода.

An approach to solving the linear optimization problem with unformalized restrictions is proposed. This approach is oriented to distributed memory multiprocessor systems. A parallel algorithm, based on combination of linear programming and discriminant analysis methods, is described. The results of computing experiments confirming the efficiency of our approach are presented.

1. Введение

В практике экономико-математического моделирования часто встречаются задачи, при решении которых приходится учитывать большое число взаимосвязанных факторов, включая плохо формализуемые [1]. Процесс решения плохо формализуемой задачи включает в себя преобразование ее формулировки путем уточнений и упрощений. В результате этого процесса мы получаем формализованную задачу, имеющую некоторое отношение к исходной постановке задачи. Во многих случаях удается построить формализованную задачу в виде задачи линейного программирования [2]. Однако решение формализованной задачи может сколь угодно сильно отличаться от решения исходной задачи в силу наличия неформализованных ограничений.

Помимо формальных методов и моделей для решения плохо формализуемой задачи используются средства неформального анализа, включающие в себя суждения экспертов для учета так называемых «внемодельных»

* Работа поддержана грантом РФФИ 06-01-00380 и грантом поддержки ведущих научных школ НШ-5595.2006.1.

факторов [3]. При этом человеческий компонент в процессе экспертизы может постепенно, с помощью обучения, заменяться машинным компонентом путем использования экспертных систем или нейросетевых программ. Однако при решении задач линейного программирования большой размерности эти подходы могут оказаться неэффективными, если их применять для решения всей задачи в целом. Это объясняется тем, что трудно адекватно настроить (обучить) экспертную систему или нейронную сеть в случае, когда система ограничений содержит тысячи линейных неравенств и десятки тысяч переменных. В таких ситуациях перспективным является подход, основанный на сочетании методов линейной оптимизации с процедурами экспертного оценивания. Алгоритм решения задачи в этом случае строится как итерационный процесс, в ходе которого шаг за шагом происходит уточнение ограничений, относящихся к неформализованной части задачи. Приближенные решения исходной задачи, получаемые в ходе итерационного процесса, формируют множество образцов, квалифицируемых экспертом. Применяя к этому множеству образцов процедуру дискриминантного анализа [4], мы получаем разделяющую поверхность, аппроксимирующую неформализованные ограничения. В соответствии с этим *актуальной* является задача разработки, анализа и реализации на ЭВМ алгоритмов решения задач линейного программирования с неформализованными ограничениями путем синтеза методов дискриминантного анализа и линейной оптимизации.

В работах [5, 6] был описан общий метод для решения задач с одним неформализованным ограничением, базирующийся на дискриминантном анализе и экспертизе новых образцов, получаемых в результате решения задачи линейного программирования. Каждый новый образец получается как решение аппроксимационной задачи линейного программирования, где вместо неформализованного ограничения фигурирует неравенство на основе разделяющей функции. В случае, когда решение аппроксимационной задачи уже присутствует в наборе образцов, применяется процедура рандомизации. Для описанного метода была доказана теорема сходимости и рассмотрены вопросы его практического применения в виде программы для ЭВМ. В качестве реализации общего метода был предложен итерационный алгоритм *СМ-ЛК*, использующий сочетание симплекс-метода и метода линейной коррекции.

В настоящей статье мы рассматриваем теоретические и реализационные аспекты *параллельной версии* алгоритма СМ-ЛК. Статья организована следующим образом. В разделе 2 выполнен теоретический анализ случая нескольких неформализованных ограничений. В разделе 3 обсуждается параллельная реализация алгоритма СМ-ЛК. В разделе 4 приводится описание задач, использованных для проверки эффективности предложенных подходов, и обсуждаются результаты проведенных экспериментов на вычислительном кластере. В заключении суммируются основные результаты, полученные в данной работе.

2. Теоретический анализ

Пусть модель экономической ситуации описывается в виде задачи линейной оптимизации, в которой нам не удалось полностью формализовать несколько ограничений. Такую модель мы можем представить в виде следующей *задачи линейного программирования с несколькими неформализованными ограничениями* (ЛПНО) в пространстве $\mathbf{R}^n$:

$$\bar{x} \in \operatorname{Arg}\max\left\{(c,x) \middle| Ax \le b; \bar{\varphi}_i(x) \le 0; i = 1,\ldots,q\right\}. \quad (1)$$

Здесь неравенства $\bar{\varphi}_i(x) \le 0$ $(i = 1,\ldots,q)$ играют роль неформализованных ограничений в том смысле, что нам не известен вид функций $\bar{\varphi}_i (i = 1,\ldots,q)$. Будем предполагать, что функции $\bar{\varphi}_i$ принадлежат к классу аффинных функций, и что задача (1) имеет единственное решение.

Пусть $D = \left\{x \middle| Ax \le b\right\}$ – многогранник, определяемый формализованной частью задачи ЛПНО. Будем предполагать, что D – телесное множество (то есть D содержит внутренние точки), и что D ограничен.

Обозначим как *ЛПФ задачу линейного программирования с формализованной частью*:

$$\max\left\{(c,x) \middle| Ax \le b\right\}. \quad (2)$$

Будем исходить из предположения, что задача (2) также имеет единственное решение $\hat{x} = \arg\max\left\{(c,x) \middle| Ax \le b\right\}$, причем $\hat{x} \ne \bar{x}$. Пусть $\bar{P}_i = \left\{x \middle| \bar{\varphi}_i(x) \le 0\right\}$ – множество значений, удовлетворяющих неформализованному ограничению $\bar{\varphi}_i(x) \le 0$. Так как $\bar{\varphi}_i$ является аффинной функцией, то $\bar{P}_i$ представляет собой замкнутое полупространство. Пусть

$$\bar{P}^* = \bigcap_{i=1}^{q} \bar{P}_i .$$

Тогда допустимое множество задачи ЛПНО будет иметь вид $D \cap \bar{P}^*$. Будем предполагать, что $D \cap \bar{P}^*$ содержит внутренние точки.

Имеет место следующая теорема устойчивости для случая нескольких неформализованных ограничений.

Теорема 1. Задача ЛПНО (1) является устойчивой по $\bar{\varphi}_1,\ldots,\bar{\varphi}_q$.

Доказательство аналогично доказательству теоремы для случая одного неформализованного ограничения, приведенному в [6].

Покажем теперь, как решение задачи ЛПНО может быть сведено к решению нескольких задач с одним неформализованным ограничением.

Рассмотрим следующую последовательность *независимых* задач ЛПНО, каждая из которых имеет только одно неформализованное ограничение:

$$\text{ЛПНО}_1 : \bar{x}_1 = \arg\max\{(c,x) | Ax \le b, \bar{\varphi}_1(x) \le 0\};$$
$$\dots$$
$$\text{ЛПНО}_q : \bar{x}_q = \arg\max\{(c,x) | Ax \le b, \bar{\varphi}_q(x) \le 0\},$$

(предполагается, что каждая из этих задач имеет единственное решение). Используя алгоритм $\mathfrak{L}$ порождения образцов из [6], мы можем построить для каждой задачи ЛПНО$_i$ $(i = 1,\dots,q)$ последовательность аффинных *разделяющих* функций $\tilde{\varphi}_i^1,\dots,\tilde{\varphi}_i^k$. Сконструируем следующую задачу линейного программирования ЛП^k:

$$\tilde{x}^k \in \operatorname{Arg}\max\{(c,x) \mid Ax \le b; \tilde{\varphi}_i^k \le 0; i = 1,\dots,q\}. \ (3)$$

Теорема 2. Пусть задача ЛПНО (1) имеет единственное решение $\bar{x}$. Пусть каждая задача ЛПНО$_i$ $(1 \le i \le q)$ также имеет единственное решение $\bar{x}_i$, причем $\bar{x}_i \ne \hat{x}$, где $\hat{x} = \arg\max\{(c,x) | Ax \le b\}$ – единственное решение задачи ЛПФ. Пусть $\tilde{x}^k$ – решение задачи ЛП^k. Тогда

$$\{\tilde{x}^k\}_{k=0}^{\infty} \to \bar{x},$$

то есть при достаточно большом k в качестве приближенного решения задачи ЛПНО (1) можно взять решение задачи ЛП^k (3).

Доказательство. Пусть $\tilde{\varphi}_i^1,\dots,\tilde{\varphi}_i^k$ – последовательность разделяющих функций, получаемых в ходе выполнения алгоритма $\mathfrak{L}$ применительно к задаче ЛПНО$_i$ $(1 \le i \le q)$. Пусть $\tilde{x}_i^0, \tilde{x}_i^1,\dots,\tilde{x}_i^k$ – порождаемая ими последовательность приближенных решений задачи ЛПНО$_i$. В соответствии с теоремой 2 из [6] имеем

$$\{\tilde{x}_i^k\}_{k=0}^{\infty} \to \bar{x}_i. \ (4)$$

Так как $\bar{x}_i \ne \hat{x}$ и $\bar{x}_i$ – единственное решение задачи ЛПНО$_i$, то $\bar{x}_i \in \bar{H}_i$, где $\bar{H}_i = \{x \mid \bar{\varphi}_i(x) = 0\}$. С учетом этого в силу условия нормализации (см. [6]) из (4) следует, что последовательность функций $\tilde{\varphi}_i^1,\dots,\tilde{\varphi}_i^k$ сходится к $\bar{\varphi}_i$ в том смысле, что

$$\lim_{k\to\infty} \operatorname{dist}(\tilde{H}_i^k, \bar{H}_i) = 0, \quad (5)$$

где $\tilde{H}_i^k = \{x \mid \tilde{\varphi}_i^k(x) = 0\}$.

Поскольку $\bar{\varphi}_i$ принадлежит к классу аффинных функций, мы можем представить уравнение $\bar{\varphi}_i(x) = 0$ в виде

$$(\bar{d}_i, x) = \alpha, \quad \alpha \in \{0,1\}.(6)$$

Тогда, в силу (5) существует k_i' такой, что для всех $k > k_i'$ уравнение $\tilde{\varphi}_i^k(x) = 0$ можно представить в виде $(\tilde{d}_i^k, x) = \alpha$, причем $\left\| \bar{d}_i - \tilde{d}_i^k \right\| < \varepsilon / \sqrt{q}$ для любого наперед заданного числа $\varepsilon > 0$. Положим $k' = \max\{k_1', ..., k_q'\}$. Тогда для всех $k > k'$ имеем

$$\left\| [\bar{d}_1, ..., \bar{d}_q] - [\tilde{d}_1^k, ..., \tilde{d}_q^k] \right\| =$$

$$= \sqrt{\left\| \bar{d}_1 - \tilde{d}_1^k \right\|^2 + ... + \left\| \bar{d}_q - \tilde{d}_q^k \right\|^2} < \sqrt{q(\varepsilon / \sqrt{q})^2} = \varepsilon.$$

Отсюда по теореме 1 получаем $\lim\limits_{k \to \infty} \left| (c, \bar{x}) - (c, \tilde{x}^k) \right| = 0$. Так как $\bar{x}$ – единственное решение задачи ЛПНО, это равносильно $\{\tilde{x}^k\}_{k=0}^{\infty} \to \bar{x}$,что и требовалось доказать.

В формулировке теоремы 2 мы исходили из предположения, что *все* задачи ЛПНО$_i$ имеют решение $\bar{x}_i$, не совпадающее с решением $\hat{x}$ задачи ЛПФ (2). На практике это требование мы можем выполнить следующим образом.

На первом этапе находим приближенные значения коэффициентов и свободных членов для всех неформализованных ограничений $\bar{\varphi}_i(x) \le 0$, удовлетворяющих условию $\bar{x}_i \ne \hat{x}$ (каждое ограничение вычисляется *независимо* от других). Заметим, что если ни одно неформализованное ограничение не удовлетворяет условию $\bar{x}_i \ne \hat{x}$, то решением задачи ЛПНО будет $\hat{x}$.

На втором этапе добавляем к системе ограничений задачи ЛПФ найденные на первом этапе *приближения* неформализованных ограничений и полагаем $\hat{x}$ равным решению расширенной задачи ЛПФ. Из оставшихся неформализованных ограничений снова выбираем все, удовлетворяющие условию $\bar{x}_i \ne \hat{x}$, и находим их приближенный вид.

Процесс заканчивается, когда либо все неформализованные ограничения будут найдены, либо останутся несколько ограничений, для которых $\bar{x}_i = \hat{x}$ (эти ограничения можно отбросить). Последняя расширенная задача ЛПФ даст приближенное решение исходной задачи ЛПНО.

3. Методика параллельной реализации алгоритма

При решении практических задач экономико-математического моделирования часто встречаются задачи с большим количеством переменных и ограничений. Размер типичной средней задачи может составлять 20 000 переменных и 5 000 ограничений [7]. В отдельных случаях количество переменных может превышать 100 000, а количество ограничений – 20 000.

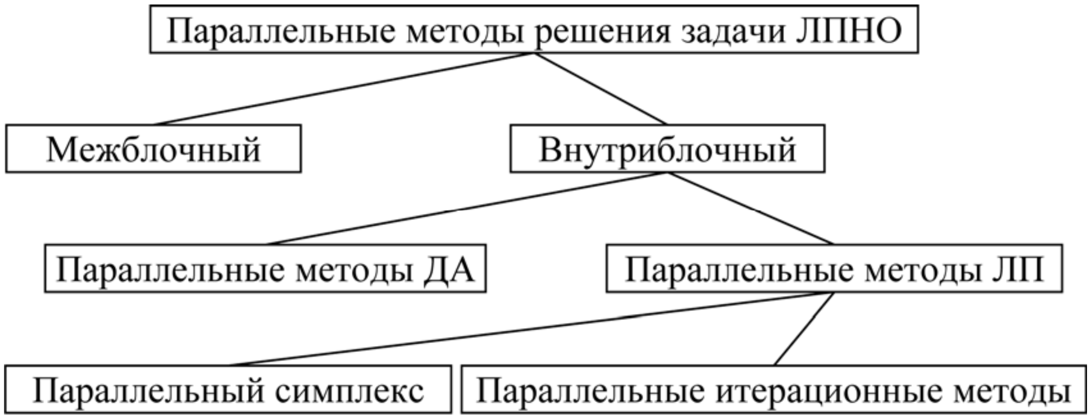

Рис. 1. Классификация параллельных методов решения задачи ЛПНО.

Подобные задачи линейного программирования при решении требуют значительных вычислительных мощностей. В случае наличия неформализованных ограничений объем требуемых вычислений может возрастать многократно. В связи с этим актуальной является задача разработки параллельных алгоритмов решения задачи ЛПНО.

3.1. Классификация параллельных методов решения задачи ЛПНО

При анализе структуры алгоритма СМ-ЛК мы можем выделить следующие классы методов параллельного решения задачи ЛПНО (Рис. 1).

Межблочный метод предполагает разбиение задачи ЛПНО на блоки, каждый из которых параллельно вычисляет одно неформализованное ограничение, после чего симплекс-методом вычисляется приближенное решение исходной задачи ЛПНО. Детально межблочный метод описывается ниже в разделе 3.2.

Внутриблочный метод предполагает параллельное выполнение отдельных блоков алгоритма СМ-ЛК [5]. В соответствии с этим мы можем рассматривать параллельные методы дискриминантного анализа (блок дискриминантного анализа) и параллельные методы линейного программирования (блок оптимизации).

Параллельные методы дискриминантного анализа (ДА) в подавляющем большинстве случаев сводятся к использованию параллельных нейросетевых алгоритмов [8, 9]. Метод линейной коррекции, используемый в блоке дискриминантного анализа, не поддается эффективному распараллеливанию. Более перспективными в этом отношении являются фейеровские методы сильной отделимости выпуклых полиэдральных множеств [10].

Параллельные методы линейного программирования (ЛП) можно разбить на две основные группы: параллельный симплекс-метод и параллельные итерационные методы. *Симплекс-метод* в общем случае допускает эффективное распараллеливание в том случае, когда число процессоров не превышает 30 [11]. Однако, в случае, когда матрица A имеет специальную блочно-диагональную структуру, эффективного распараллеливания можно

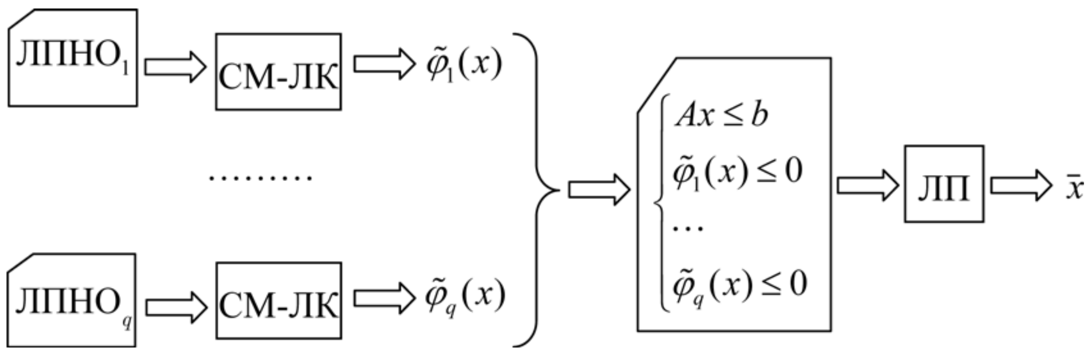

Рис. 2. Схема параллельного алгоритма.

добиться и на большем количестве процессоров [12]. В основе параллельного алгоритма в этом случае лежит принцип декомпозиции Данцига-Вульфа [13, 14].

Параллельные итерационные методы решения задачи линейного программирования базируются главным образом на S-технологии [10, 15] в основе которой лежит использование фейеровских отображений. Здесь наметились два подхода. Первый подход ориентирован на матрицы со специальной блочной структурой [2, 16, 17]. Преимуществом этого подхода является отсутствие массовых пересылок данных между процессорами. Главным недостатком является то, что степень параллелизма при использовании данного подхода ограничена количеством блоков, на которые разбивается матрица A. Второй подход [19] основан на разбиении симметрической задачи на подзадачи путем проектирования допустимого множества на гиперплоскости, задаваемые координатными осями. Итерационный процесс выполняется параллельно для каждой проекции. Через определенное число шагов происходит периодическое *связывание* независимых итерационных процессов путем подстановки данных в общую симметрическую задачу. Однако этот подход нуждается в дальнейших исследованиях.

3.2. Параллельная версия алгоритма СМ-ЛК

Схема параллельного алгоритма решения задачи ЛПНО, основанного на межблочном методе, приведена на Рис. 2. На первом шаге задача ЛПНО, имеющая q неформализованных ограничений, разбивается на q задач ЛПНО$_1$,…,ЛПНО$_q$. Каждая из этих задач независимо решается методом СМ-ЛК. Из вычисленных приближений неформализованных ограничений и формализованной части задачи ЛПНО строится полная система ограничений. На втором шаге полученная таким образом задача решается одним из методов линейного программирования. В результате с некоторым приближением мы получаем $\bar{x}$ – решение исходной задачи ЛПНО.

При реализации параллельной версии алгоритма СМ-ЛК нахождение каждого неформализованного ограничения оформлялось в виде независимого процесса. Вычисленные ограничения передавались некоторому выделенному процессору-координатору, который формировал полную задачу линейного программирования и решал ее каким-либо методом линейной оптимизации. Указанный подход позволяет без какой-либо модификации выполнять параллельную программу на различном количестве процессоров. Для организации обменов данными между процессами была использована система параллельного программирования MPI, основанная на передаче сообщений [20]. При реализации параллельной версии алгоритма СМ-ЛК мы использовали передачу сообщений на основе односторонних коммуникаций MPI-2 [21]. Разработанная параллельная версия алгоритма СМ-ЛК может использоваться без каких-либо изменений как на многопроцессорных системах с общей памятью, так и на системах с массовым параллелизмом, включая кластерные системы. Исходные тексты параллельной версии свободно доступны в Интернет по адресу: http://life.susu.ru/lpnoMPI/.

4. Вычислительные эксперименты

Для проведения вычислительных экспериментов мы использовали сконструированную нами *модельную задачу Mod-*n_q, которая имеет следующий вид:

$$\begin{cases} x_1 - 2x_2 \le 0 \\ \quad\vdots \\ x_1 - 2x_n \le 0 \\ x_1 + 2x_2 \le 8000 \\ \quad\vdots \\ x_1 + 2x_n \le 8000 \\ x_1 \ge 0, x_2 \ge 0, \cdots, x_n \ge 0 \end{cases}$$

$$Q_{\max} = x_1$$

$$\begin{cases} e_1(x) = \operatorname{sgn}(3x_1 - x_2 - 3000) \\ \quad\vdots \\ e_q(x) = \operatorname{sgn}(3x_1 - x_q - 3000) \end{cases}$$

Здесь $e_1, \ldots, e_q$ – экспертные функции, задающие неформализованные ограничения ($q \le n-1$). Таким образом, задача *Mod-*n_q представляет собой задачу ЛПНО в пространстве $\mathbf{R}^n$, имеющую q неформализованных ограничений.

Для проведения экспериментов был использован вычислительный кластер с характеристиками, представленными в Табл. 1. На указанном вычислительном кластере нами были исследованы ускорение и расширяемость, получаемые при использовании параллельного алгоритма СМ-ЛК на базе симплекс-метода и метода линейной коррекции.

Табл. 1. Характеристики вычислительного кластера.

Количество узлов	26	
Количество процессоров	52	
Конфигурация узла	Процессоры	2×Intel Xeon64 3,2 GHz
	RAM	2 Gb
	HDD	80 Gb
Тип системной сети	InfiniBand (PCI-Express 4X)	
Тип управляющей сети	Gigabit Ethernet	
Операционная система	SUSE Linux Enterprise Server 9.1	
MPI-2	MVAPICH2	
Компилятор	Intel C/C++ Compiler 9.0 EMT 64	

Результаты по исследованию ускорения представлены на Рис. 3. Эксперименты проводились для трех размерностей: $n = 60$, $n = 80$ и $n = 100$ при фиксированном количестве неформализованных ограничений $r = 48$. Ускорение вычислялось по формуле $\alpha = \alpha_0 + t_p / t_8$, где t_8 – время, затраченное на решение задачи Mod-n_{48} на 8 процессорах, t_p – время, затраченное на решение этой же задачи на p процессорах, $\alpha_0 = (n - 60)/60$. Для каждой размерности варьировалось количество процессоров, используемых для решения задачи. Во всех трех случаях было получено ускорение, близкое к линейному.

Результаты по исследованию расширяемости для параллельного алгоритма СМ-ЛК приведены на Рис. 4. Эксперименты проводились также для трех размерностей: $n = 60$, $n = 80$ и $n = 100$. Здесь t_{pr} обозначает время, затраченное на решение модельной задачи размерности n с q неформализованными ограничениями на p процессорах. При проведении эксперимента для каждой размерности варьировалось количество процессоров, используемых для решения задачи. При этом количество неформализованных ограничений всегда равнялось количеству процессоров.

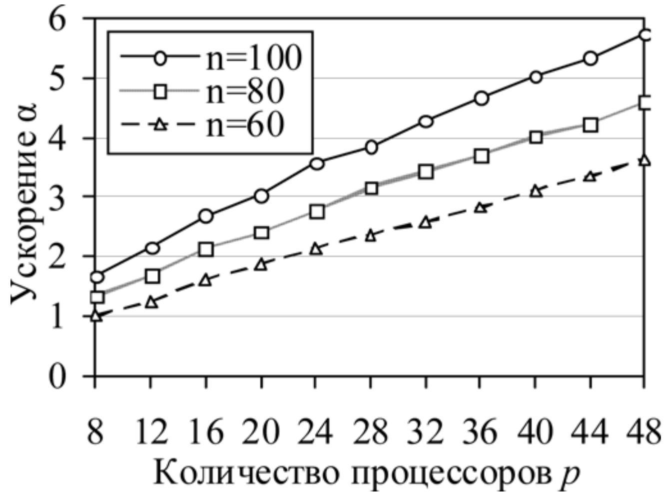

Рис. 3. Ускорение для Mod-n_{48}.

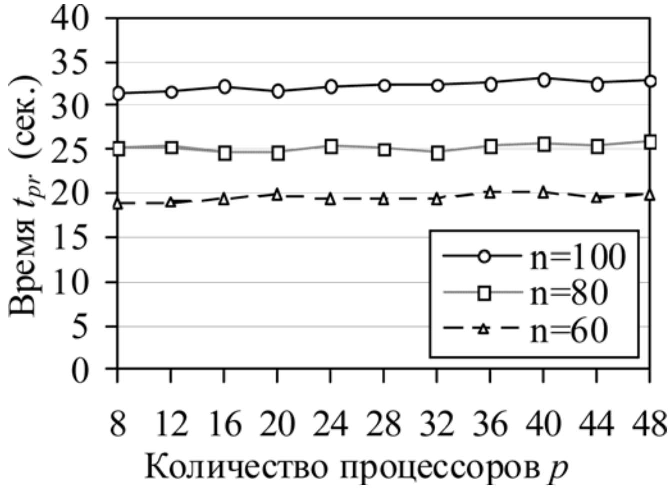

Рис. 4. Расширяемость для Mod-n_p.

Проведенные эксперименты показали, что во всех случаях параллельный алгоритм СМ-ЛК демонстрировал расширяемость, близкую к линейной. Таким образом, можно сделать вывод о том, что параллельный алгоритм СМ-ЛК показывает хорошую масштабируемость на многопроцессорных системах с массовым параллелизмом при решении задач ЛПНО с различным количеством неформализованных ограничений.

5. Заключение

В данной работе был рассмотрен итерационный алгоритм СМ-ЛК для решения задач линейного программирования при наличии неформализованных ограничений. Этот алгоритм базируется на синтезе алгоритмов линейного программирования и дискриминантного анализа. Показано и математически обосновано, как случай задачи ЛПНО с одним неформализованным ограничением может быть обобщен на случай задачи с несколькими неформализованными ограничениями. Предложена параллельная версия алгоритма СМ-ЛК, которая может быть использована без каких-либо изменений как на многопроцессорных системах с общей памятью, так и на системах с массовым параллелизмом, включая кластерные системы. Дана классификация параллельных методов решения задачи ЛПНО. Предложена схема параллельного алгоритма решения задачи ЛПНО, основанного на межблочном методе. Проведены вычислительные эксперименты на высокопроизводительном кластере по исследованию масштабируемости описанной параллельной версии алгоритма СМ-ЛК, подтверждающие эффективность предложенных подходов. Исходные тексты параллельной версии свободно доступны в Интернет по адресу: http://life.susu.ru/lpnoMPI/.

Список литературы

1. *Мазуров Вл.Д.* О задаче оптимизации с плохо формализуемой целью // Параметрическая оптимизация. –Свердловск: УНЦ АН ССР, 1985. -С. 51-53.

2. *Еремин И.И.* Общая теория устойчивости в линейном программировании // Известия ВУЗов. Математика. -1999. -N 12. –С. 43-52.
3. *Мазуров Вл.Д.* Метод комитетов в задачах оптимизации и классификации. -М.: Наука, 1990. -248 с.
4. *McLachlan G.J.* Discriminant Analysis and Statistical Pattern Recognition. -New York: John Wiley and Sons, 1992. -526 p.
5. *Соколинская И.М.* Синтез симплекс-метода и метода линейной коррекции в задачах линейной оптимизации с неформализованными ограничениями // Вычислительные методы и программирование. -2005. -Том 6, №1. -С. 226-238.
6. *Mazurov Vl.D., Sokolinskaya I.M.* Discrimination analysis and randomization in linear optimization problems with not formalized restrictions // Pattern Recognition and Image Analysis. -2005. -Vol. 15, No. 4.
7. *Forrest J.J.H., Tomlin J.A.* Implementing the simplex method for the optimization subroutine library // IBM Systems Journal. -1992. -Vol. 31, No. 1. -P. 11-25.
8. *Ghosh J., Hwang K.* Critical issues in mapping neural networks on message-passing multicomputers // Proceedings of the 15th Annual International Symposium on Computer architecture. Honolulu, Hawaii, United States. -1988. -P. 3-11.
9. *Kennedy J.V., Austin J., Pack R., Cass B.* CNNAP – A Parallel Processing Architecture for Binary Neural Networks // Proceedings of the IEEE International Conference on Neural Networks (ICNN'95), Perth, Western Australia. -1995.
10. *Еремин И.И.* Фейеровские методы сильной отделимости выпуклых полиэдральных множеств // Известия вузов. Сер. математика. -2006. -№ 12. -С. 33-43.
11. *Stunke C.B.l, Reed D.A.* Hypercube implementation of the simplex algorithm // Proceedings of the third conference on Hypercube concurrent computers and applications - Volume 2. -New York, NY, USA: ACM Press. -1989. -P. 1473-1482.
12. *Lyu J.J., Luh H., Lee M.-C.* Solving Linear Programming Problems on the Parallel Virtual Machine Environment // American Journal of Applied Sciences. -2004. -Vol. 1, No. 2. -P. 90-94.
13. *Molina F.W.* A Survey of Resource Directive Decomposition in Mathematical Programming // ACM Computing Surveys. -1979. -Vol. 11, No. 2. -P. 95-104.
14. *Martinson R.K., Tind J.* An interior point method in Dantzig-Wolfe decomposition // Computers & Operations Research. -1999. -Vol. 26, No. 12. -P. 1195-1216.

15. *Еремин И.И.* Методы фейеровских приближений в выпуклом программировании // Мат. заметки. -1968. -Т. 3, вып. 2. -С. 217-234.
16. *Бердникова Е.А.* Параллельный алгоритм решения задачи линейного программирования на основе оператора проектирования на линейное многообразие // Высокопроизводительные вычисления и их приложения: Труды Всероссийск. науч. конф. (30 октября - 2 ноября 2000 г., г. Черноголовка). -М.: Изд-во МГУ. -2000. -С. 71-72.
17. *Бердникова Е.А., Ерёмин И.И., Попов Л.Д.* Распределенные фейеровские процессы для систем линейных неравенств и задач линейного программирования // Автоматика и телемеханика. -№ 2. -2004. -С. 16-32.
18. *Еремин И.И.* Теория линейной оптимизации. –Екатеринбург: Изд-во “Екатеринбург”, 1999. -312 с.
19. *Соколинский Л.Б.* Иерархический параллелизм: новая парадигма программирования // Информационный бюллетень Ассоциации математического программирования. -№ 11. -Екатеринбург: УрО РАН. -2007. -С. 76-77.
20. *Dongarra J.J., Otto S.W., Snir M., Walker D.* A message passing standard for MPP and workstations // Communications of the ACM. -1996. -Vol. 39, No. 7. -P. 84-90.
21. *Gropp W., Huss-Lederman S., Lumsdaine A., Lusk E., Nitzberg B., Saphir W., Snir M.* MPI - The Complete Reference: Volume 2, The MPI Extensions. -MIT Press, 1998.