УДК 519.8

О.П. ЛИХОВИД

ПАРАЛЕЛЬНИЙ АЛГОРИТМ РОЗВ'ЯЗУВАННЯ ДВОЕТАПНОЇ ЗАДАЧІ СТОХАСТИЧНОГО ПРОГРАМУВАННЯ

Розглядається паралельний алгоритм розв'язування двоетапної задачі стохастичного програмування з фіксованою рекурсією, коли випадкові параметри мають скінченний дискретний розподіл ймовірностей. Алгоритм використовує методи недиференційовної оптимізації та орієнтований для реалізації на обчислювальному кластері у програмному середовищі MPI.

Вступ. Двоетапні моделі стохастичного програмування використовуються у багатьох прикладних задачах. Дослідженню цих моделей присвячено багато публікацій (див., наприклад, [1 – 5]). Якщо випадкові параметри моделі мають скінченний дискретний розподіл ймовірностей, задачу можна подати як детерміновану, і застосовувати весь арсенал методів детермінованої оптимізації. Але детерміновані моделі, як правило мають велику розмірність, і тому розробка ефективних алгоритмів для цих задач продовжує бути актуальною. Щоб скоротити час пошуку оптимальних розв'язків та досягти розумних значень часу обчислень можливо використання комп'ютерів паралельної архітектури.

В роботі [6] розглядався алгоритм розв'язування двоетапної задачі стохастичного програмування з фіксованою рекурсією, коли випадкові параметри мають скінченний дискретний розподіл ймовірностей. В даній роботі запропоновано паралельну версію такого алгоритму. Вона орієнтована для реалізації на обчислювальному кластері у програмному середовищі MPI [7]. Слід відмітити, що хоча в даній роботі розглядається лінійна модель, підхід до її розв'язання дозволяє розглянути й випадок, коли функціонал задачі першого етапу є нелінійним, наприклад, квадратичним чи кусочно-лінійним.

Математична модель задачі. Будемо розглядати лінійну двоетапну задачу стохастичного програмування з фіксованою рекурсією і з дискретно розподіленими випадковими технологічною матрицею $T(\omega)$ і вектором правих частин $h(\omega)$ у випадку, коли невизначеність представлена за допомогою випадкових змінних ω, визначених в деякому дискретному ймовірнісному просторі, множиною сценаріїв $\Omega = \{1, \ldots L\}$ з відповідними ймовірностями $\{p_1, \ldots, p_{L\}}\}$.

Цю модель можна подати у вигляді наступної еквівалентної детермінованої моделі:

$$\left.\begin{aligned}
&\min[(c,x) + \sum_{l=1}^{L} p_l q_l y_l] \\
&x \in X, \\
&T_l x + W y_l = h_l,\ l = 1, \ldots, L, \\
&y_l \ge 0.
\end{aligned}\right\}. \tag{1}$$

Тут

$x \in R^{n_1}$, $y_l \in R^{n_2}$ – змінні першого та другого етапу відповідно;

$c \in R^{n_1}$, $q_l \in R^{n_2}$ – коефіцієнти цільових функцій першого та другого етапу відповідно;

X – опукла множина обмежень першого етапу;

W – детермінована (фіксована) матриця рекурсії.

Очевидно, що задача (1) має блочно-діагональну структуру:

$$\left.\begin{array}{lllll}
\min[(c,x) + p_1 q_1 y_1 + \cdots + p_L q_L y_L] & & & & \\
x \in X, & & & & \\
 & W y_1 & & & = h_1 - T_1 x \\
 & & \ddots & & \vdots \\
 & & & W y_L & = h_L - T_L x \\
y_l \ge 0,\ l = 1, \ldots, L & & & &
\end{array}\right\},$$

яка може бути використана при розробці ефективних алгоритмів.

Для розв'язання задачі (1) в [6] розглядався алгоритм, що використовує схему декомпозиції за змінними [8] з поділом останніх на змінні «першого етапу» x та «змінні другого етапу» y. Для врахування обмежень першого етапу можна використовувати метод негладких штрафних функцій [8].

Використовуючи схему декомпозиції за змінними, зафіксуємо значення змінних першого етапу $x = \bar{x}$. Задача (1) розпадається на L лінійних підзадач зі змінними другого етапу $y(\bar{x})$:

знайти

$$\left.\begin{array}{l} \min p_l q_l y_l \\ W y_l = h_l - T_l \bar{x}, \\ y_l \ge 0. \end{array}\right\}. \quad (2)$$

Задачу (2) замінюємо на наступну:

$$\left.\begin{array}{l} \min[p_l q_l y_l + R^+ y_l^+ + R^- y_l^-] \\ W y_l + y_l^+ - y_l^- = h_l - T_l \bar{x}, \\ y_l \ge 0, \\ y_l^+ \ge 0, \\ y_l^- \ge 0. \end{array}\right\}. \quad (3)$$

Додаткові змінні y_l^+, y_l^- додані в модель (3) для забезпечення сумісності обмежень. Символи R^+, R^- $(R^+, R^- > 0)$ позначають коефіцієнти штрафу. Припустимо, що обмеження першого етапу сумісні. Очевидно, якщо в оптимальному розв'язку змінні y_l^+, y_l^- модифікованої задачі (6) дорівнюють нулеві, то він є також розв’язком початкової задачі.

Нехай $\{u_l^*(\bar{x})\}$ – оптимальні значення двоїстих змінних для l-ої лінійної підзадачі. Тоді, виходячи з схеми декомпозиції за змінними, субградієнт у точці $\bar{x}$ можна обчислити, використовуючи оптимальні двоїсті змінні $\{u_l^*(\bar{x})\}$, за формулою:

$$g_\Phi(\bar{x}) = g_I(\bar{x}) - \sum_{l=1}^{L} T_l^T u_l^*(\bar{x}), \quad (4)$$

де $g_I(\bar{x})$ – субградієнт у точці $\bar{x}$ для задачі першого етапу.

Лінійні підзадачі в схемі декомпозиції можна розв’язувати спеціалізованими симплексними алгоритмами, а для розв’язання координуючої задачі відносно змінних першого етапу x можна використовувати субградієнтні алгоритми, наприклад, субградієнтний алгоритм з розтягом простору в напрямку двох послідовних субградієнтів – r-алгоритм [8].

З формули (4) для обчислення субградієнту видно, що даний алгоритм розв'язання двохетапної моделі з рекурсією можна досить легко розпаралелити, розв'язуючи лінійні підзадачі незалежно, і реалізувати обчислення, наприклад, на кластері.

Паралельний алгоритм розв'язання двохетапної стохастичної задачі з фіксованою рекурсією. Паралельний алгоритм розв'язання двохетапної стохастичної задачі з фіксованою рекурсією можна представити у вигляді наступної процедури.

Нехай процес розв'язання задачі проводиться на $p+1$ процесорах. Будемо використовувати модель "Master-Slave (Coordinator-Worker)". Один з процесорів умовно вибирається «провідним» (Master), а решта – «підлеглими» (Slave). У Master процесорі відбувається розв'язання координуючої задачі, а на p Slave-процесорах – розв'язання лінійних підзадач.

Master-процесор.

Крок 1. Обчислює поточну точку $\bar{x}$ згідно процедури субградієнтного алгоритму.

Крок 2. Обчислення субградієнту в точці $\bar{x}$.

Крок 2.1. Генерує параметри для l -ої лінійної підзадачі ($l = 1, \ldots, L$): p_l, q_l, $h_l - T_l \bar{x}$.

Крок 2.2. Згенеровані параметри для l -ої лінійної підзадачі пересилаються у вільний Slave-процесор.

Крок 2.3. Очікування відповіді з будь-якого Slave-процесора.

Крок 2.4. Отримання розв'язку $\{u_l^*(\bar{x})\}$ l -ої лінійної підзадачі і модифікація субградієнта згідно формули (4).

Крок 2.5. Якщо $l \le L$ – перехід до кроку 2.1.

Крок 2.6. Якщо ще є активні Slave-процесори – перехід до кроку 2.3.

Крок 3. Якщо виконується деякий критерій зупинки субградієнтного алгоритму – посилає сигнал СТОП у всі Slave-процесори і завершує роботу; інакше – перехід до кроку 1.

Slave-процесор.

Крок 1. Очікує на сигнал з Master-процесора.

Крок 2. Якщо отримано сигнал СТОП від Master-процесора – завершує роботу.

Крок 3. Отримує значення параметрів лінійної підзадачи p_l, q_l, $h_l - T_l \bar{x}$ і знаходить оптимальні розв'язки $\{u_l^*(\bar{x})\}$.

Крок 4. Посилає значення $\{u_l^*(\bar{x})\}$ у Master-процесор.

Крок 5. Перехід до кроку 1.

Висновки. В роботі розглянуто паралельний алгоритм розв'язання двохетапної стохастичної задачі з фіксованою рекурсією та з стохастичними правими частинами та технологічною матрицею обмежень задачі другого етапу. Алгоритм орієнтований для реалізації на обчислювальному кластері у програмному середовищі MPI. Описаний підхід дозволяє розглянути й випадок, коли функціонал задачі першого етапу є нелінійним, наприклад, квадратичним чи кусочно-лінійним, а також його може бути використано і для інших застосувань.

Вищеописаний паралельний алгоритм реалізовано на мові програмування C++ у програмному середовищі MPI. Для розв'язання координуючої задачі використовувався r-алгоритм. Вхідні дані задаються в SMPS форматі [9]. В теперішній час проводяться обчислювальні експерименти з перевірки ефективності паралельного алгоритму на кластерному комплексі СКІТ-3 Інституту кібернетики НАН України [10]. В майбутньому планується публікація результатів обчислювальних експериментів.

А.П. Лиховид

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ РЕШЕНИЯ ДВУХЭТАПНОЙ ЗАДАЧИ СТОХАСТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Рассматривается параллельный алгоритм решения двухэтапной задачи стохастического программирования с фиксированной рекурсией, когда случайные параметры имеют конечное дискретное распределение вероятностей. Алгоритм использует методы недифференцированой оптимизации и ориентирован для реализации на вычислительном кластере в програм-мной среде MPI.

O.P. Lykhovyd

A PARALLEL ALGORITHM FOR SOLVING TWO-STAGE STOCHASTIC PROGRAMMING PROBLEM

A parallel algorithm for solving two-stage stochastic programming problem with fixed recourse, when random parameters have a finite discrete probability distribution is considered. The algorithm uses methods of non-differential optimization and is oriented for implementation on a computing cluster in the MPI software environment.

Список літератури

1. Ермольев Ю.М. Методы стохастического программирования. М.: Наука, 1976.
2. Шор Н.З., Бардадым Т.А., Журбенко Н.Г., Лиховид А.П., Стецюк П.И. Использование методов негладкой оптимизации в задачах стохастического программирования. *Кибернетика и системный анализ*. Киев. Институт кибернетики имени В.М. Глушкова. 1999. № 5. С. 33 – 47.
3. Kall P., Wallace S.W. Stochastic programming. Chichester: Wiley, 1994.
4. Mayer J. Stochastic Linear Programming Algorithms: A Comparison Based on a Model Manegement System. Amsterdam: ODP. 1998. 153 p.
5. Ruszczynski A. Decomposition methods in stochastic programming. *Math. Prog.* 1997. Vol. 79. P. 333 – 353.
6. Лиховид А.П. Использование методов негладкой оптимизации для решения некоторых классов задач стохастического программирования. Праці міжнародного симпозіуму "*Питання оптимізації обчислень*" (ПОО-XXXIII). Київ, 2007. С. 174.
7. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002. 608 с.
8. Шор Н.З. Методы минимизации недифференцируемых функций и их приложения. К.: Наук. думка, 1979. 199 с.
9. Birge J.R., Dempster M.A.H., Gassmann H.I., Gunn E., King A.J., Wallace S.W. A standard input format for multiperiod stochastic linear programs. Working Paper WP-87-118, IIASA, Laxenburg, Austria, 1987.
10. Кластерний комплекс Інституту кібернетики. Кластерний комплекс СКІТ. https://icybcluster.org.ua/.

Одержано 18.03.2019