論文

角型構造の大規模多目的 ファジィ線形計画問題に対する 分解手法の有効性[†]

坂和 正敏* 加藤 浩介* 茂原 英樹*

本論文では、角型構造の大規模多目的ファジィ線形計画問題に焦点をあて、近年、坂和らによって提案されてきた対話型ファジィ満足化手法における Dantzig-Wolfe の分解手法の有効性について検討する。Dantzig-Wolfe の分解手法と対話型ファジィ満足化手法を概観した後、改訂シンプレックス法に対する Dantzig-Wolfe の分解手法の有効性について検討するため、3 目的 15 結合制約の線形計画問題を取り上げる。ワークステーション上で、50 変数と 200 変数の数値例に対する数多くの数値実験により、Dantzig-Wolfe の分解手法の有効性を処理時間や必要記憶容量に関して数値的に検証する。

キーワード：分解手法、角型構造、大規模線形計画法、ファジィ計画法、対話型手法

1. はじめに

現実の数多くの大規模な最適化問題は、角型構造の数理計画問題として定式化されることが多い。例えば、製造会社の各事業部ごとに、独自で利用可能な資源が与えられているとともに、会社全体として利用可能な最大量に制限のある共通の資源をわけあって利用しなければならないという制約のもとで、会社全体の利益を最大にするというような場合である。このような大規模な数理計画問題に対しては、なんらかの工夫により、いくつかの小規模な問題に分割して解くことが望ましい。このような観点からの試みは、1960 年代の初めに Dantzig と Wolfe[1]が、角型構造の大規模線形計画問題に対する分解原理を提案して以来、角型構造の大規模数理計画問題に対する数多くの研究が活発に行われてきた[2-6]。さらに一目的のみならず多目的をも考慮した角型構造の大規模数理計画問題に対する研究も行われてきた[7,8]。

筆者らは、近年、角型構造の大規模多目的線形計画問題の各目的関数に対する意思決定者の“だいたいある値以下にしたい”というファジィ目標を線形のメンバシップ関数で規定した後、凸ファジィ決定やファジィ決定に従って変換した問題に、Dantzig-Wolfe の分解原理を適用することにより意思決定者のファジィ目標を考慮した満足解を効率よく導出するという手法を提案するとともに[9]-[12]、対話型手法への拡張も試

みてきた[13]。

また、角型構造の大規模多目的線形計画問題に含まれる数多くのパラメータに対して、問題の定式化に携わった専門家の人間としての判断をより適切に表現するためには、従来のように何らかの経験的あるいは主観的な方法である値に設定するよりはむしろ、含まれるパラメータそのものをファジィ数としてとらえた方が、現実の多目的意思決定状況をより適切に表すことが可能になるものと期待できる[14,15]。このような観点から、筆者らは、角型構造の大規模多目的線形計画問題の各目的関数や制約式に含まれるパラメータのあいまい性がファジィ数として特性づけられているという、ファジィ数を含む角型構造の大規模多目的線形計画問題の定式化を試みるとともに、含まれるすべてのファジィ数のメンバシップ関数の帰属度が α 以上となる係数ベクトルや行列の中で、特に意思決定者が最も望ましいように、その値を設定するという、非ファジィな α -多目的線形計画問題を定義して、通常のパレート最適解の素直な拡張と見なされる α -パレート最適解の概念を導入した。さらに、意思決定者が主観的に設定した基準値にミニマックスの意味で近い (M-) α -パレート最適解は Dantzig-Wolfe の分解原理の適用により求められることを示し、得られた解に意思決定者が満足しなければ基準値を対話的に更新することにより、(M-) α -パレート最適解の集合の中から意思決定者の満足解を導出するという対話型意思決定を提案した[16,17]。

しかし、残念ながら、これまでの筆者らの論文においては、簡単な数値例により提案した手法の妥当性を示したのみにとどまっており、大規模な問題に対する処理時間や必要記憶容量に関する優位性についての比

[†] Efficiency of a Decomposition Method for Large-Scale Multiobjective Fuzzy Linear Programming Problems with Block Angular Structure
Masatoshi SAKAWA, Kousuke KATO and Hideki MOHARA

* 広島大学工学部第二類(電気系)
Faculty of Engineering, Hiroshima University

較・検討は今後の課題として残されていた。このような大規模な問題に対する処理時間や必要記憶容量に関する優位性についての比較・検討は、いうまでもなく使用するコンピュータに依存するが、現実の様々な意思決定状況を考慮すれば、近年特に性能の向上が著しいパソコンやワークステーション上で、提案した手法に基づいて作成されたプログラムによる意思決定者との対話が現実には有効に実行できるのかということは、極めて重要な課題になるといわざるを得ない。このような状況を鑑み、本論文では、角型構造の大規模多目的ファジィ線形計画問題に対して、ワークステーション上で、3目的15結合制約の数値例を取り上げて、50変数と200変数のそれぞれの場合に対する数多くの数値実験を試みることににより、Dantzig-Wolfeの分解手法を適用した場合の処理時間や必要記憶容量に関する有効性を数値的に検証する。

2. 角型構造の問題と等価な問題への変換

本節では、次のような角型構造の大規模線形計画(LSMOLP)問題を考え、Dantzig-Wolfeの分解手法について概観する[1-6]。

$$\text{minimize } z = \sum_{i=1}^n c_i x_i \quad (1)$$

$$\text{subject to } \sum_{i=1}^n A_i x_i = b_0 \quad (2)$$

$$B_i x_i = b_i, \quad i=1, \dots, n \quad (3)$$

$$x_i \geq 0, \quad i=1, \dots, n \quad (4)$$

ここで、 $m_0 \times n_i$ 行列 A_i ($i=1, \dots, n$)、 $m_i \times n_i$ 行列 B_i ($i=1, \dots, n$)、 n_i 次元行ベクトル c_i ($i=1, \dots, n$)、 m_i 次元列ベクトル b_i ($i=0, 1, \dots, n$)はすべて与えられた定数であり、変数 x_i ($i=1, \dots, n$)は n_i 次元列ベクトルである。この問題の変数の数は $\sum_{i=1}^n n_i$ 個であり、制約式(2)および(3)の個数は総計 $m_0 + \sum_{i=1}^n m_i$ 個であることに注意しよう。各変数 x_i ごとに独立な制約式(3)はブロック制約式といわれ、各変数 x_i を結合している制約式(2)は結合制約式といわれる。

図1 角型構造の問題の構造

この問題の係数行列、定数項、目的関数などをひとまとめにして図式的に表せば図1のようになる。もちろん、このような角型構造の線形計画問題に直接、改訂シンプレックス法を適用して $\sum_{i=1}^n m_i \times \sum_{i=1}^n n_i$ という大規模な基底行列を取り扱って解くことも理論的には可能である。しかし、制約式の形の特異性を考慮して、例えば行列 B_1, B_2 などを係数とするより小規模の個々の独立した部分問題

$$\left. \begin{array}{ll} \text{minimize} & c_i x_i \\ \text{subject to} & B_i x_i = b_i \\ & x_i \geq 0 \end{array} \right\} \quad (5)$$

に分解して解けば、計算機における占有記憶量や計算時間が減少することが期待される。ただし、 $A_1, \dots, A_n$ で結合されているため、各部分問題を独立に解いた解がそのままとの問題の最適解とはならない。しかし、以下に述べるDantzig-Wolfeの分解原理を用いれば、もとの問題を、 $(m_0 + n)$ 個の制約式をもつ一つの主問題と n 個の独立な部分問題に分解して、各部分問題の目的関数を主問題の解により更新して全体として最適解を繰り返しによって得ることができる。

さて、各ブロックの制約式を満たす凸体

$$K_i = \{x_i | B_i x_i = b_i, x_i \geq 0\} \quad (6)$$

が空でない有界な凸体、すなわち凸多面体である場合を考えてみよう。 K_i の端点の個数を l_i とし、 l_i 個の端点を $\{v_i^1, \dots, v_i^{l_i}\}$ とおけば、凸多面体の分解定理により凸多面体 K_i の任意の点 $x_i \in K_i$ すなわち、式(3)および(4)を満たす任意の x_i は次のような v_i^j の凸結合で表すことができる。

$$x_i = \sum_{j=1}^{l_i} \lambda_i^j v_i^j \quad (7)$$

$$\sum_{j=1}^{l_i} \lambda_i^j = 1 \quad (8)$$

$$\lambda_i^j \geq 0, \quad i=1, \dots, n, \quad j=1, \dots, l_i \quad (9)$$

いま、すべての v_i^j がわかっているとしよう。そこで

$$p_i^j = A_i v_i^j, \quad i=1, \dots, n; \quad j=1, \dots, l_i \quad (10)$$

$$f_i^j = c_i v_i^j, \quad i=1, \dots, n; \quad j=1, \dots, l_i \quad (11)$$

とおけば、もとの問題は形式的につぎのような λ_i^j を変数とする線形計画問題に書きかえられる。

$$\begin{aligned} & \text{minimize } z = \sum_{i=1}^n \sum_{j=1}^{l_i} f_i^j \lambda_i^j \\ & \text{subject to } \sum_{i=1}^n \sum_{j=1}^{l_i} p_i^j \lambda_i^j = b_0 \\ & \sum_{j=1}^{l_i} \lambda_i^j = 1, \quad i=1, \dots, n \\ & \lambda_i^j \geq 0, \quad i=1, \dots, n; \quad j=1, \dots, l_i \end{aligned} \quad (12)$$

この問題(12)はもとの問題(1)～(4)の主問題といわれ、主問題を解くことは、もとの問題を解くこととまったく等価である。従って主問題の最適解 $\lambda_i^* = \lambda_i^{j*}$ が求まれば、それを式(7)に代入することによってもとの問題の最適解 $x_i = x_i^*$ が求められる。このとき、主問題の係数行列、定数項、目的関数などを図式的に表せば、図2のようになる。もとの問題を形式的に主問題の形に変換することによって制約式の数 $m_0 + \sum_{i=1}^n m_i$ から $m_0 + n$ に減少する一方、変数の数は $\sum_{i=1}^n n_i$ から $\sum_{i=1}^n l_i$ になっているので、一般的にはいちじるしく増加している。主問題を構成するためには凸多面体 K_i のすべての端点を求めなければならない。

しかし、すべての v_i, f_i, p_i を求めるわけではないので、主問題が完全な形で再現されることはなく、行数に等しい数の列を出発基底とすれば以後、基底に入ってくる列は計算によって生成される。

3. 角型構造のファジィ多目的線形計画問題

本節では、筆者らによって提案された角型構造の大規模多目的ファジィ線形計画問題に対する対話型ファジィ満足化手法における M - α -パレート最適解が Dantzig-Wolfe の分解原理により求められることを概観する[16,17]。

3.1 問題の定式化と解の概念

一般に、ファジィパラメータを含む角型特殊構造の大規模多目的線形計画(LSMOLP-FP)問題は次のように定式化される。

$$\left. \begin{array}{l} \text{minimize } \tilde{c}_{11}x_1 + \cdots + \tilde{c}_{1p}x_p \\ \text{minimize } \tilde{c}_{21}x_1 + \cdots + \tilde{c}_{2p}x_p \\ \vdots \\ \text{minimize } \tilde{c}_{k1}x_1 + \cdots + \tilde{c}_{kp}x_p \\ \text{subject to } \tilde{A}_1x_1 + \cdots + \tilde{A}_px_p \leq \tilde{b}_0 \\ \qquad \qquad \qquad \geq \tilde{b}_1 \\ \qquad \qquad \qquad \tilde{B}_px_p \leq \tilde{b}_p \\ \qquad \qquad \qquad x_j \geq 0 \quad j=1, \dots, p \end{array} \right\} \quad (13)$$

ここで、 $\tilde{c}_{ij} (i=1, \dots, k; j=1, \dots, p)$ は要素がファジィ数の n_j 次元係数行ベクトル、 $x_j (j=1, \dots, p)$ は n_j 次元列ベクトルの決定変数である。さらに、 $\tilde{A}_j$ は要素がファジィ数の $m_0 \times n_j$ の係数行列で、 $\tilde{B}_j$ は要素がファジィ数の $m_i \times n_j$ の係数行列である。

LSMOLP-FP は、目的関数や制約式に含まれる係数がファジィ数として特性づけられているため、これまでの LSMOLP に対して定義されてきたパレート最適解の概念をそのまま適用することはできない。そこで、

図2 主問題の構造

LSMOLP-FP に含まれているすべてのファジィ数のベクトルや行列のメンバシップ関数の値が α 以上であるような α -レベル集合を導入して非ファジィな α -LSMOLP を定式化するとともに、 α -LSMOLP の各目的関数に対する意思決定者 (Decision maker : DM) の“だいたいある値以下にしたい”というファジィ目標 (fuzzy min) や“だいたいある値以上にしたい”というファジィ目標 (fuzzy max) のみならず“だいたいある値ぐらいにしたい”というファジィ目標 (fuzzy equal) をもつ場合をも考慮すれば、一般化された α -LSMOLP ($G\alpha$ -LSMOLP) は次のように表される。

$$\left. \begin{array}{l} \text{fuzzy min } c_i x \quad i \in I_1 \\ \text{fuzzy max } c_i x \quad i \in I_2 \\ \text{fuzzy equal } c_i x \quad i \in I_3 \\ \text{subject to } x \in X(A, B, b) \\ \qquad \qquad (A, B, b, c) \in (\tilde{A}, \tilde{B}, \tilde{b}, \tilde{c})_\alpha \end{array} \right\} \quad (14)$$

ただし、 $c_i x = c_{i1}x_1 + \cdots + c_{ip}x_p, I_1 \cup I_2 \cup I_3 = \{1, 2, \dots, k\}, I_i \cap I_j = \emptyset, i=1, 2, 3, i \neq j$ である。

このような fuzzy equal を含む $G\alpha$ -LSMOLP においては、目的関数の大小関係に基づいて定義した α -パレート最適解の概念を適用することはできない。しかし、目的関数の大小関係の代わりにメンバシップ関数の大小関係を用いれば、次のような M - α -パレート最適解の定義を行うことが可能である。

定義1 (M - α -パレート最適解)

$G\alpha$ -LSMOLP に対して、 $\mu_i(c_i x) \geq \mu_i(c_i^* x^*), i=1, \dots, k$ で、しかも、ある l について $\mu_l(c_l x) > \mu_l(c_l^* x^*)$ となるような $x \in X(A, B, b), (A, B, b, c) \in (\tilde{A}, \tilde{B}, \tilde{b}, \tilde{c})_\alpha$ が存在しないとき、 $x^* \in X(A^*, B^*, b^*)$ を M - α -パレート最適解、対応するパラメータ値 (A^*, B^*, b^*, c^*) を α -レベル最適係数とよぶ。

3.2 ミニマックス問題

DM の設定した α に対して、各メンバシップ関数に対する DM の志望水準を反映する基準メンバシップ値 $\bar{\mu}_i$ ($0 \leq \bar{\mu}_i \leq 1$), $i=1, \dots, k$ を DM が主観的に設定すれば、DM の満足解の候補となりうる M- α -パレート最適解は、次のミニマックス問題を解くことにより求められる。

$$\left. \begin{array}{l} \text{minimize} \quad \max_{i=1, \dots, k} (\bar{\mu}_i - \mu_i(\mathbf{c}_i \mathbf{x})) \\ \text{subject to} \quad \mathbf{x} \in X(A, B, \mathbf{b}) \\ (A, B, \mathbf{b}, \mathbf{c}) \in (\tilde{A}, \tilde{B}, \tilde{\mathbf{b}}, \tilde{\mathbf{c}}) F_\alpha \end{array} \right\} \quad (15)$$

この問題を解くことにより、DM の設定した α に対する α レベル集合 $(A, B, \mathbf{b}, \mathbf{c}) \in (\tilde{A}, \tilde{B}, \tilde{\mathbf{b}}, \tilde{\mathbf{c}})_\alpha$ 内で、DM が主観的に設定した基準メンバシップ値 $\bar{\mu}_i$ に、ミニマックスの意味で近い解が得られる。

ここで、係数ベクトルや行列 $(A, B, \mathbf{b}, \mathbf{c})$ は、もはや単に係数としてではなく、変数として取り扱われているので、制約条件式はもはや線形ではなく、非線形になってしまっている。そこで、便宜上、fuzzy min と fuzzy equal の右側の強意単調減少のメンバシップ関数を $d_{iR}(\mathbf{c}_i \mathbf{x})$ ($i \in I_1 \cup I_{3R}$)、fuzzy max と fuzzy equal の左側の強意単調増加のメンバシップ関数を $d_{iL}(\mathbf{c}_i \mathbf{x})$ ($i \in I_2 \cup I_{3L}$) と表して、ミニマックス問題を次のように変形してみよう。

$$\left. \begin{array}{l} \text{minimize} \quad v \\ \text{subject to} \quad \mathbf{c}_i \mathbf{x} \leq d_{iR}^{-1}(\bar{\mu}_i - v), \\ \qquad \qquad \qquad i \in I_1 \cup I_{3R} \\ \qquad \qquad \qquad \mathbf{c}_i \mathbf{x} \geq d_{iL}^{-1}(\bar{\mu}_i - v), \\ \qquad \qquad \qquad i \in I_2 \cup I_{3L} \\ \mathbf{x} \in X(A, B, \mathbf{b}) \\ (A, B, \mathbf{b}, \mathbf{c}) \in (\tilde{A}, \tilde{B}, \tilde{\mathbf{b}}, \tilde{\mathbf{c}})_\alpha \end{array} \right\} \quad (16)$$

ここで、ファジィ数の行列 A_i, B_i 、ファジィ数のベクトル $\mathbf{b}_i, \mathbf{c}_j$ に対する α -レベル集合の性質より、 $A_i, B_i, \mathbf{b}_i, \mathbf{c}_j$ の実行可能領域は、それぞれ対応する α -レベル集合の左右の端点により、閉区間 $[A_{i\alpha}^L, A_{i\alpha}^R], [B_{i\alpha}^L, B_{i\alpha}^R], [\mathbf{b}_{i\alpha}^L, \mathbf{b}_{i\alpha}^R], [\mathbf{c}_{i\alpha}^L, \mathbf{c}_{i\alpha}^R]$ のように表されることを考慮し、問題の制約の実行可能領域を最大にするように係数をとると、ミニマックス問題の最適解は次の問題を解くことにより得られることがわかる。

$$\left. \begin{array}{l} \text{minimize} \quad v \\ \text{subject to} \\ \mathbf{c}_{i1\alpha}^L \mathbf{x}_1 + \dots + \mathbf{c}_{ip\alpha}^L \mathbf{x}_p \leq d_{iR}^{-1}(\bar{\mu}_i - v), \\ \qquad \qquad \qquad i \in I_1 \cup I_{3R} \\ \mathbf{c}_{i1\alpha}^R \mathbf{x}_1 + \dots + \mathbf{c}_{ip\alpha}^R \mathbf{x}_p \geq d_{iL}^{-1}(\bar{\mu}_i - v), \\ \qquad \qquad \qquad i \in I_2 \cup I_{3L} \\ A_{1\alpha}^L \mathbf{x}_1 + \dots + A_{p\alpha}^L \mathbf{x}_p \leq \mathbf{b}_{\alpha\alpha}^R \\ B_{1\alpha}^L \mathbf{x}_1 \leq \mathbf{b}_{1\alpha}^R \\ \vdots \\ B_{p\alpha}^L \mathbf{x}_p \leq \mathbf{b}_{p\alpha}^R \\ \mathbf{x}_j \geq 0, j=1, \dots, p \end{array} \right\} \quad (17)$$

この問題の最小の v を求めることは、この問題の制約式の許容領域の存在する最小の v の値を求めることと等価である。そこで、 v の値を 2 分法のアルゴリズムにより離散的に変化させて、この問題の許容領域の存在する最小の v の値を求めればよい。このようにして、式(16)の制約式すなわち、式(17)の許容領域の存在する最小の v の値 v^* が求まれば、式(17)に $v=v^*$ を代入して、最も重要と考えられる目的関数(ここでは $\mathbf{c}_{i1\alpha}^L \mathbf{x}$ で $i \in I_1$ とする)を用いて解を一意的に決定するため、次の線形計画問題を解く。

$$\left. \begin{array}{l} \text{minimize} \quad \mathbf{c}_{i1\alpha} \\ \text{subject to} \\ \mathbf{c}_{i1\alpha}^L \mathbf{x}_1 + \dots + \mathbf{c}_{ip\alpha}^L \mathbf{x}_p \leq d_{iR}^{-1}(\bar{\mu}_i - v), \\ \qquad \qquad \qquad i \in I_1 \cup I_{3R} \\ \mathbf{c}_{i1\alpha}^R \mathbf{x}_1 + \dots + \mathbf{c}_{ip\alpha}^R \mathbf{x}_p \geq d_{iL}^{-1}(\bar{\mu}_i - v), \\ \qquad \qquad \qquad i \in I_2 \cup I_{3L} \\ A_{1\alpha}^L \mathbf{x}_1 + \dots + A_{p\alpha}^L \mathbf{x}_p \leq \mathbf{b}_{\alpha\alpha}^R \\ B_{1\alpha}^L \mathbf{x}_1 \leq \mathbf{b}_{1\alpha}^R \\ \vdots \\ B_{p\alpha}^L \mathbf{x}_p \leq \mathbf{b}_{p\alpha}^R \\ \mathbf{x}_j \geq 0, j=1, \dots, p \end{array} \right\} \quad (18)$$

4. 数値実験

前節で定式化した問題に対して、Dantzig-Wolfe の分解手法を適用した場合と適用しなかった場合に対して計算機プログラムを作成して、処理時間と必要記憶容量に関する比較・検討を行う。

4.1 実験環境

使用した計算機は Sun の SPARCStation 10 (主記憶: 32 MB) で、計算機プログラムを C 言語を用いて作成した。このような実験環境でのシミュレーションにおいては、3 目的、15 結合制約式の問題を対象として、問題に含まれる係数は次のような規則に基づいて

生成することにした。

- (1) 問題の係数ベクトル c_i と係数行列 A_i, B_i は $(-5, 5)$ の区間でランダムに生成する。
- (2) (1) のようにして生成された c_i, A_i, B_i に対して、実行可能領域が必ず存在するように、制約式の右辺定数ベクトル b_0, b_i は $(5000, 10000)$ の区間でランダムに生成する。

4.2 比較手法と比較内容

本論文では次のような3種類の手法の比較を行う。

(1) Dantzig-Wolfe の分解手法

2節で述べたように Dantzig-Wolfe の分解原理を適用することにより、もとの問題を1つの主問題と n 個の独立な部分問題に分解して解く。

(2) 改訂シンプレックス法

生成された問題に対して、分解手法を適用しないで直接改訂シンプレックス法で解く場合には、図3に示されているように、各ブロック制約式に現れない変数の係数はすべて0に設定した等価な問題を取り扱うことにする。このようにもとの問題を改訂シンプレックス法で解く場合には、ブロック制約式に現れない変数の係数をすべて0に設定した問題を対象とすることになり、本来不要なデータをも取り扱うことになる。

図3 問題の変換

(3) 行列の疎密性を考慮した改良改訂シンプレックス法

(2) のようにもとの問題に直接改訂シンプレックス法を適用すれば、本来不要なデータをも取り扱うことになるので、分解手法を適用するときよりも余計なデータを付加することになり得策とはいえない。このような無駄なデータの取り扱いを回避して余計な処理時間を削除するため、余計なデータを付加しないで、各ブロック制約式に現れる変数の係数のみを対象としてピボット項を決定するという工夫を付加した行列の疎密性を考慮して改良した改訂シンプレックス法を適用する。

より詳しく述べれば、シンプレックス法の手順

「手順1 相対費用係数 $\bar{c}_j$ を用いて $\min_{c_j < 0} \bar{c}_j = \bar{c}_s$ となる s を求める。 $\bar{c}_s \geq 0$ であれば、最適解を得て終了。

手順2 すべての $\bar{a}_{is} \leq 0$ ならば、最小値が有界でないという情報を得て終了。

手順3 $\bar{a}_{is}$ に正のものがあれば

$$\min_{\bar{a}_{is} > 0} \frac{\bar{b}_i}{\bar{a}_{is}} = \frac{\bar{b}_r}{\bar{a}_{rs}} = \theta$$

となる r を求める。

手順4 $\bar{a}_{rs}$ に関するピボット操作を行う。」

において、手順2と手順3を

「手順2 ここで結合制約式が m_0 行から n_0 行、 s 列のブロック制約式が m_s 行から n_s 行とする。このとき $m_0 < i < n_0$ もしくは $m_s < i < n_s$ で $\bar{a}_{is} \leq 0$ ならば、最小値が有界でないという情報を得て終了。

手順3 $\bar{a}_{is}$ に正のものがあれば $m_0 < i < n_0$ もしくは $m_s < i < n_s$ において

$$\min_{\bar{a}_{is} > 0} \frac{\bar{b}_i}{\bar{a}_{is}} = \frac{\bar{b}_r}{\bar{a}_{rs}} = \theta$$

となる r を求める。」

に変更する。

これらの3種類の手法を適用して比較する内容は、次に述べるような処理時間と最大必要記憶容量である。

(1) 処理時間

- ・比較的小規模の50変数の問題と大規模で処理時間が1時間程度の取り扱いやすい200変数の問題におけるブロックの個数と処理時間の関係。
- ・200変数、20ブロックの問題において各ブロックの中の変数の数が均一でなく突出して変数の個数が多いブロックが1つだけある場合、その変数の個数が突出したブロックの変数の個数と処理時間の関係。

(2) 必要記憶容量

- ・分解手法を適用した場合と分解手法を適用しない場合のブロックの個数と必要な記憶容量の関係を2000変数の問題について調べる。

5. 実験結果と考察

5.1 処理時間

まず初めに、問題の中に含まれる分解の個数、すなわちブロックの個数によって処理時間がどのように影響するかを調べる。そのために、ブロックの個数が変化した場合の処理時間の比較を行う。ただし、ここでは各ブロックの変数の個数は均一に振り分け、端数は初めのブロックから1つずつ加える。得られた結果を図4と図5に示す。ただし、50変数における処理時間を示す図4ではグラフを見やすくするため改良したシンプレックス法のデータは省略している。これより、問題に対するブロックの個数が多くなればなるほど分解手法を適用した方が処理時間がより短くなっていることが分かる。これに対して、分解の数がかなり少ないとき、分解手法を適用した場合がより多くの処理時

間を必要としている。これは、分解した後の部分問題がまだ大規模であるにもかかわらず、部分問題から主問題へのデータの受け渡しの分だけより多くの時間が要求されるためであると思われる。

図4 50変数における処理時間

図5 200変数における処理時間

図6 50変数における処理時間の比

図7 200変数における処理時間の比

次に、分解手法を適用しない場合に対する分解手法を適用した場合の処理時間の比率を求める。得られた結果を示すと図6と図7のようになる。50変数に比べて200変数の方が分解手法を適用する場合の処理時間の短縮の有効性が向上している。このことにより、問題が大規模になるにつれて処理時間の差が拡大することになり、分解手法を適用した方が有効になることがわかる。これまでの比較においては、問題の中に含まれる各ブロックの変数の数が均一の場合を対象としてきたが、次に変数の個数が突出して多いブロックが1つだけあると仮定した場合を考察する。さらに、そのブロックの変数の個数が変化した場合の処理時間の比較を行う。得られた結果を図8に示す。これにより変数の個数がある程度突出したブロックがあっても分解手法を適用する方が処理時間が短いことがわかる。しかし、問題の中に他のブロックより極端に変数の個数が多いブロックがあるとそのブロックの処理に多くの時間を必要とするためかなりの処理時間がかかってしまうことになる。

5.2 必要記憶容量

作成したプログラムにおいて、分解手法を適用する場合と適用しない場合の最大必要記憶容量配列となる基底行列は、大まかにはそれぞれ、次のようになる。

・分解手法を適用する場合

$$(m_0 + n + \text{目的関数から変換された制約の数})^2$$

・分解手法を適用しない場合

$$\left(m_0 + \sum_{i=1}^n m_i + \text{目的関数から変換された制約の数}\right)^2$$

である。例えば、2000変数、20ブロック、3目的、結合制約式50、各ブロック制約式50の問題を解くとき、分解手法を適用する場合と適用しない場合の最大必要記憶容量は、上の最大基底配列にdouble型で宣言するための8byteを掛けて、

・分解手法を適用する場合

$$(50 + 20 + 2)^2 \times 8 = 41472 \text{ byte}$$

図8 ブロックの変数の個数と処理時間

・ 分解手法を適用しない場合

$$(50+20 \times 50+2)^2 \times 8 = 8853632 \text{ byte}$$

となる。2000 変数においてブロックの個数を変化させてみると最大必要記憶容量は図 9 のようになる。

分解手法を適用する場合、配列の一边は結合制約式にブロックの個数を加えただけでよく、最大必要記憶容量はブロックの個数に依存する。よって、ブロックの個数が少ないときは分解手法を適用しない場合よりもはるかに少ない必要記憶容量ですむので有効である。

今回の環境下で実行したところ、目的関数 3, 結合制約式 15, 分解の数 50, ブロック制約の制約式 40, ブロック制約式の変数の数 40 (すなわち, 2000 変数) で, 分解しない場合は最大配列を確保することができなかった。これに対して, 分解手法を用いた場合は最大配列を確保することが可能で, 望ましい結果が得られた。

6. おわりに

本論文では, 角型構造の大規模多目的ファジィ線形計画問題に対して, Dantzig-Wolfe の分解手法を適用したときの処理時間の短縮や使用メモリの削減などの有効性について, ワークステーション上での数多くのシミュレーションにより数値的に検証した。数値実験により得られた結論を一般に述べれば, 分解手法を適用すれば適用しない場合よりもより望ましい結果が得られたといえる。特に, 問題が大規模になるにつれて分解手法を適用する場合の優位性が見られた。しかし, 問題の中のブロックがまだ大規模であったり, 他のブロックと比べて突出して変数の数の多いブロックがあったりすると分解手法の処理速度が遅くなるという欠点も見いだされた。なお, 本論文では, Dantzig-Wolfe の分解手法を適用した場合を, 直接改訂シンプレックス法を適用した場合と行列の疎密性を考慮した改良改訂シンプレックス法を適用した場合との比較・検討を行ったが, 今後, 行列の疎密性を考慮した他の手法や内点法などとの比較や Dantzig-Wolfe の分解手法にみられた欠点を補うような新たな分解手法の提案が望まれる。

参 考 文 献

- [1] G.B. Dantzig and P. Wolfe, "The decomposition algorithm for linear programming", *Econometrica*, Vol.29, pp.767-778 (1961).
- [2] W. Orchard-HAYS, "Advanced Linear Programming: Computing Techniques", McGraw-Hill, 1968 (小国坂本共訳), コンピュータによる線形計画法, 培風館, 1973.
- [3] L.S. Lasdon, "Optimization Theory for Large Systems", Macmillan, New York, 1970 (志水訳), 大規模システムの最適化理論, 日刊工業新聞社, 1973.

図 9 2000 変数における必要記憶容量

- [4] J.K. Ho and R.P. Sundarraj, "An advanced implementation of the Dantzig-Wolfe decomposition algorithm for linear programming", *Mathematical Programming*, Vol.20, pp.303-326 (1981).
- [5] J.K. Ho and R.P. Sundarraj, "Computational experience with advanced implementation of decomposition algorithms for linear programming", *Mathematical Programming*, Vol.27, pp.283-290 (1983).
- [6] J.K. Ho and R.P. Sundarraj, "DECOMP: an implementation of Dantzig-Wolfe Decomposition for Linear Programming", Springer-Verlag, New York (1989).
- [7] M. Sakawa and F. Seo, "Interactive multiobjective decisionmaking for large-scale systems and its application to environmental systems", *IEEE Trans. Systems, Man, and Cybernetics*, Vol.SMC-10, pp.796-806 (1980).
- [8] M. Sakawa, H. Yano and T. Yumine, "An interactive fuzzy satisficing method for multiobjective linear-programming problems and its application", *IEEE Trans. Systems, Man, and Cybernetics*, Vol. SMC-17, pp.654-661 (1987).
- [9] 坂和, 乾口, 澤田, "角型構造の大規模線形計画問題に対する凸ファジィ決定に基づくファジィ満足化手法", *日本ファジィ学会誌*, 5巻, 3号, pp.516-527 (1993).
- [10] 坂和, 乾口, 澤田, "角型構造の大規模線形計画問題に対するファジィ計画法", *電子情報通信学会論文誌*, J77-A 巻, 3号, pp.430-438 (1994).
- [11] M. Sakawa, K. Sawada and M. Inuiguchi, "A fuzzy satisficing method for large-scale linear programming problems with block angular structure", *European Journal of Operational Research*, Vol.81, No.2, pp.399-409 (1995).
- [12] M. Sakawa, M. Inuiguchi and K. Sawada, "A fuzzy satisficing method for large-scale multiobjective linear programming problems with block angular structure", *Fuzzy Sets and Systems*, Vol.78, No.3, pp.279-288 (1996).
- [13] M. Sakawa and K. Sawada, "An interactive fuzzy satisficing method for large-scale multiobjective linear programming problems with block angular structure", *Fuzzy Sets and Systems*, Vol.67, pp.5-17 (1994).
- [14] M. Sakawa, "Fuzzy Sets and Interactive Multiobjective Optimization", Plenum Press, New York (1993).

- [15]坂和, 石井, 西崎, “ソフト最適化”, 朝倉書店 (1995).
- [16]坂和, 加藤, 溝内, “ファジィパラメータを含む大規模多目的線形計画問題に対する対話型意思決定”, 日本ファジィ学会誌, 7巻, 3号, pp.612-623 (1995).
- [17]坂和, 加藤, “ファジィパラメータを含む大規模多目的線形計画問題に対する対話型ファジィ満足化手法”, 日本ファジィ学会誌, 8巻, 3号, pp.547-557 (1996).

(1997年3月27日 受付)

[問い合わせ先]

〒739 東広島市鏡山1-4-1

広島大学工学部第二類(電気系)

計数管理工学講座

坂和 正敏

TEL : 0824(24)7694

FAX : 0824(24)7694/(22)7195

E-mail : sakawa@msl.sys.hiroshima-u.ac.jp

著者紹介

坂和 正敏 (さかわ まさとし)

広島大学工学部第二類(電気系)

昭和45年 京大・工・数理卒, 昭和50年同大学院博士課程了。同年 神大・工・システム助手, 昭和56年 同助教授, 昭和62年 岩手大・工・教授を経て, 平成2年 広島大・工・第二類(電気系)計数管理工学講座教授となり現在に至る。大規模システム, 多目的システム, ファジィシステムにおける意思決定手法とその応用に関する研究に従事, 工博。著書線形システムの最適化, 非線形システムの最適化, ファジィ理論の基礎と応用, 経営数理システムの基礎(森北出版), ソフト最適化, 遺伝的アルゴリズム(朝倉書店), Fuzzy Sets and Interactive Multiobjective Optimization (Plenum Press)等。

茂原 英樹 (もはら ひでき)

広島大学工学部第二類(電気系)

平成7年 広島大・工・第二類(電気系)卒。平成9年 同大学大学院博士課程前期了。同年(株)日本テレコム入社, 現在に至る。在学中は, 大規模多目的ファジィ線形計画問題に関する研究に従事。

加藤 浩介 (かとう こうすけ)

広島大学工学部第二類(電気系)

平成3年 阪大・基礎工・生物工卒。平成5年 同大学大学院博士課程前期了。平成6年 同後期退学。同年 広島大・工・第二類(電気系)計数管理工学講座助手。データ圧縮, システム最適化手法とその応用に関する研究に従事。

Efficiency of a Decomposition Method for Large-Scale Multiobjective Fuzzy Linear Programming Problems with Block Angular Structure

by

Masatoshi SAKAWA, Kousuke KATO and Hideki MOHARA

Abstract :

In this paper, we focus on large-scale multiobjective fuzzy linear programming problems with the block angular structure and examine the efficiency of the Dantzig-Wolfe decomposition method in the interactive fuzzy satisficing method recently proposed by Sakawa et al. After overviewing the Dantzig-Wolfe decomposition method and the interactive fuzzy satisficing method, three-objective linear programming problems with 15 coupling constraints are considered in order to demonstrate the efficiency of the Dantzig-Wolfe decomposition method over the revised simplex method. Through a lot of computational experiments on workstation for numerical examples with both 50 and 200 variables, the advantages of the Dantzig-Wolfe decomposition method are discussed with respect to processing time and required memory storage.

Keywords : Decomposition Methods, Block Angular Structure, Large-scale Linear Programs, Fuzzy Programming, Interactive Methods

Contact Address : Masatoshi SAKAWA, Kousuke KATO and Hideki MOHARA

Faculty of Engineering, Hiroshima University

Higashi-Hiroshima, 739 Japan

TEL : 0824(24)7694

FAX : 0824(24)7694/(22)7195

E-mail : sakawa@msl.sys.hiroshima-u.ac.jp