

モデリングシステムを用いた生産計画問題の解法(2)

——引っ張り型生産指示方式の多目標モデル——

渡 辺 展 男

1. はじめに

トヨタ自動車が2006年5月に発表した2006年3月期連結決算(米国会計基準)は、売上高が21兆369億円と、日本の製造業で初めて20兆円を上回った。原油高の中で低燃費車の展開が遅れた欧米勢は低迷が予想され、2007年3期にはトヨタが自動車の売上高世界一の座に就く公算が大きくなっている[26]。トヨタの強さの根底を支えているといわれるトヨタ生産方式においては、在庫量の削減、ムダの徹底的排除による原価低減を図ることを基本目標として、「必要な物を、必要なときに、必要な量だけ作ること」として定義されるジャストインタイム(Just-in-Time; JIT)を実現するために、引っ張り型生産指示方式が採用されている[13]。引っ張り型生産指示方式においては、各工程における生産・引き取り指示量はその直後工程で実際に消費された量に基づいて決定されるが、カンバン方式は引っ張り型生産指示方式を実現する一つの情報システムであり、カンバンは生産情報の伝達手段となっている¹⁾。本稿は、この引っ張り型生産指示方式を対象として、数理計画法によるモデリングについて論じたものである。

生産指示方式に関する数理計画法によるモデリングにおいて、その実際問題への適用可能性を検討する場合、次の2点、つまり(1)定式化された問題を解くための計算量の削減、および(2)数式モデルの計算機への入力、モデルとデータの分離等の計算環境の改善が大きな課題となる。対象となる問題は、多くの整数変数を持つ整数計画問題に定式化されるため、前者の解決は重要な課

題となる。この整数計画問題を解くためのアルゴリズムは、近年、飛躍的に向上している。特に、分枝限定法 (branch-and-bound method) と切除平面法 (cutting plane method) の組み合わせと考えられる分枝カット法 (branch-and-cut method) と呼ばれる解法が注目されており、現在は商用ソフトウェアにおいても採用されている²⁾。一方、後者の解決方法として1970年代以降開発が進められてきたモデル記述言語といわれるソフトウェアが1990年代以降日本においても入手が可能となったが、近年このモデル記述言語が大きな進展を見せている。その先進性を表すためにモデル記述言語に代わり、新たな進展をみせたソフトウェアに対しては、モデリングシステムという言葉が用いられている³⁾。前稿[23]においては、近年の数理計画ソフトウェアの求解性能の向上を例示するとともに、前者の観点から著者らが提案している近似計算手続きの有効性を検証し、また後者の観点からモデリングシステムの先進性を活用した解法アプローチを提案した。


本稿のねらいは、前稿において課題として示した内容、つまりモデリングシステムによる解法アプローチを引っ張り型生産指示方式の多目標モデルに展開し、数値計算を通して提案アプローチの有効性を検証するとともに今後の課題を明らかにすることにある。

本稿の構成は以下の通りである。まず2節で多目標モデルの意義と従来の解法システムの枠組みについて述べた後、3節において本稿が対象としている引っ張り型生産指示方式の多目標モデルを示す。次に4節において多目標モデルに対する解法を示した後、5節においてモデリングシステムを用いた解法アプローチについて述べる。そして6節においてある自動車部品製造工程を用いて行った数値検証の結果を示した後、最後に7節において本稿のまとめと今後の課題について述べる。

2. 多目標モデルと従来の解法システム

本稿では生産指示方式に対するアプローチとして数理計画法を用いている。数理計画モデルでは、ある与えられたデータのもとで実際に問題を解いた場

合、実行可能解が存在しないという結果になる可能性がある。しかし、本稿で提示するモデルが対象としている小日程計画レベルでは生産計画の実行が不可能となつてはならず、実際には生産の実行前に生産能力等の手当が行われている。従つて、解くべき問題の実行可能性の保証は数理計画法によるモデリングの重要な要点の一つと考えられ、基準生産能力に対する負荷の超過がどの工程でどの程度発生する可能性があるかについて、事前に判明するモデルの構築が必要となつてくる。そこで本稿では、生産必要量に合わせ生産能力を用意するという生産能力調整に必要なデータを与え得る多目標モデルを提示する⁴⁾。具体的には、(1)計画期間中の各工程、各品目の補充目標在庫水準の総和を最小にすること、および(2)生産必要量に対し用意された基準生産能力の他に、さらに追加手当すべき生産能力の総和を最小にすることを目標とした多目標モデル⁵⁾である。

多目標モデルに対する解法アプローチの試みとして、これまで著者らは反復的目標計画法 (sequential goal programming) ⁶⁾ の枠組みを適用し、まずワークステーション上において、モデル記述言語を用いた解法システムを構築し、数値計算を通してその有効性を検証した。次にパソコン上のシェル環境を用いた解法システムを提案してきた[20, 46, 49]。しかしながら、 1 に示すようにその解法システムの枠組みを実現するためには、スクリプト環境など何らかの形で処理全体を制御する仕組みが別途必要となっており、さらには多目標モデルを対象としている関係で、スクリプト言語による処理の記述は極めて複雑なものとなっていた。

しかしながら、解法システムにおいて主要な役割を果たすモデル記述言語が、近年さらに大きな進展を見せている。既に述べたように新たな進展をみせたソフトウェアに対しては、モデリングシステムという言葉が用いられている。その特徴を簡潔に表現するとすれば、モデルの記述 (model describing) とモデルの解法 (model solving) を一つの環境で実現しているということができ

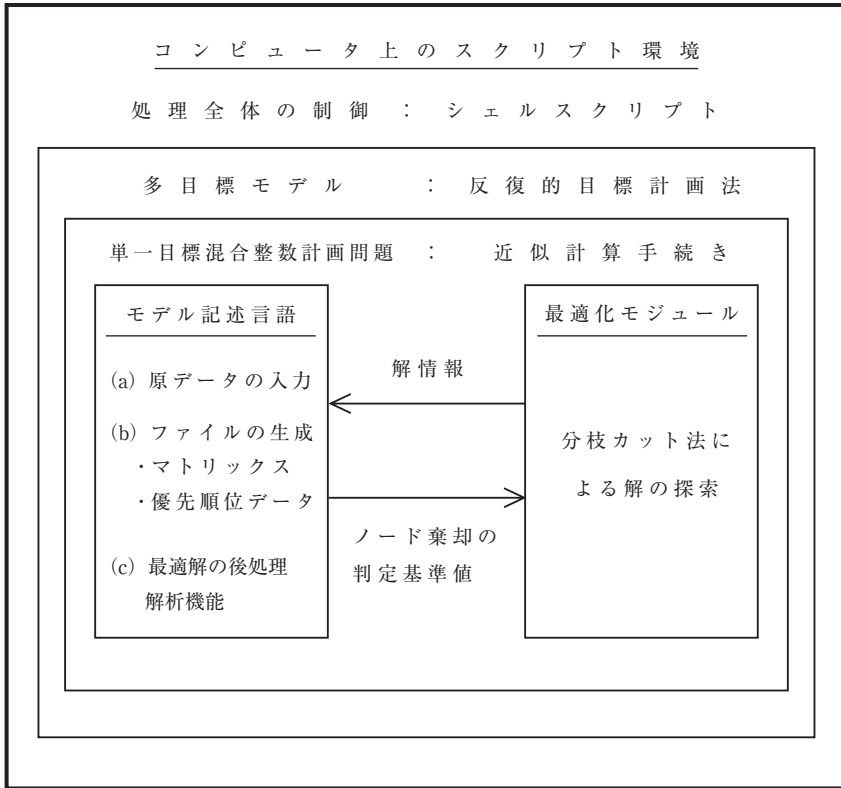


図1 従来の解法システムの枠組み

本稿では、前稿[23]で提案したモデリングシステムを用いた解法アプローチを基礎として多目標モデルへの拡張を試みる。以下、著者らが提案した引っぱり型生産指示方式の多目標モデルを対象として、代表的なモデリングシステムの一つである Xpress-MP の Mosel [32]を用いた解法アプローチを例示する。

3. 引っぱり型生産指示方式の多目標モデル

3.1 モデルの条件

本稿で対象とするモデルは、著者らが提案した引っぱり型生産指示方式の多目標モデルである[20, 46, 49]。このモデルが対象とするシステムは次のような

多段階、多品目生産・在庫・運搬システムである。

- (1) 一つの組立工程に収束していく多段工程で N 工程から構成されており、 $n \in \{1, 2, \dots, N\}$ で工程を表す。なお最終工程は $n=1$ とする。
- (2) 各工程は生産工程，加工済み在庫点および後続工程加工待ち在庫点 ($n=1$ では納入待ち製品在庫点) から成る。
- (3) 期間を $t \in \{0, 1, \dots, T\}$ で表す。計画期間は1期より始まり T 期で終了する。

またこのモデルは以下の条件で示される生産状況を対象としている。

- (1) 受注先から最終製品の各期の納入量についての内示があり，受注残は認められない。
- (2) 各工程で M 種類の品目が生産される。 $i \in \{1, 2, \dots, M\}$ で品目を表す。
- (3) 各期の各品目について，計画期間全体の生産および引き取り割当量が定まっている。
- (4) 各期の各品目に対する生産および引き取り指示量は前期の期末に計算される。
- (5) 資材在庫は十分にあるが，各工程での生産および引き取りは加工待ちおよび加工済み在庫量の制約を受ける。
- (6) 第 n 工程での生産リードタイムは LP^n である。即ち， t 期中に生産された品目は $t+LP^n$ 期中に加工済み在庫点に納入される。また引き取りリードタイムは LH^n である。即ち， t 期中に引き取られた品目は， $t+LH^n$ 期中に納入待ち製品在庫点あるいは加工待ち在庫点に納入される。
- (7) 各品目の段取り替え時間および単位量当たり加工時間は既知で計画期間中は一定である。
- (8) 各工程の各品目について期末目標在庫量が設定されている。
- (9) 段取り替えが必要な工程においては，サブロットの大きさが定まっており，生産はこのサブロット単位で行われる。

$N=5$ の場合のモデルの概念図を図2に示す。

3.2 記号

ここで示す多目標モデルでは、前項 3.1 で述べた記号のほかに次の記号を用いる。

- J : 工程全体の集合 $J = \{1, 2, \dots, N\}$
- $J1$: 最終工程を除いた工程の集合 $J1 = \{2, 3, \dots, N\}$
- K : 段取り替えが必要な工程の集合
- sn : 第 n 工程の直後工程 ($n \in J1$)
- $D_t^{(i)}$: i 製品の t 期の納入内示量
- W_t^n : 第 n 工程の t 期の基準生産能力 (時間)
- $a^{n(i)}$: 第 n 工程での i 品目の単位量当たり加工時間
- $S^{n(i)}$: 第 n 工程での i 品目の段取り替え時間 ($n \in K$)
- $L^{n(i)}$: 段取り替えが必要な工程で加工される i 品目のサブロットの大きさ ($n \in K$)
- $I_0^{n(i)}$: 第 n 工程での i 品目の初期加工済み在庫量
- $B_0^{n(i)}$: i 製品の初期納入待ち在庫量 ($n=1$ の場合) 及び第 n 工程の後工程 sn への i 部品の初期加工待ち在庫量 ($n \in J1$ の場合)
- $P_{j-LP^n}^{n(i)}$: 第 n 工程の i 品目についての生産仕掛量 ($j = 1, 2, \dots, LP^n$)
- $d_{j-LP^n}^{n(i)}$: 第 n 工程の i 品目についての引き取り仕掛量 ($j = 1, 2, \dots, LH^n$)
- $SI_t^{n(i)}$: 第 n 工程の加工済み在庫点における i 品目の t 期末目標在庫量
- $SB_t^{n(i)}$: 納入待ち製品在庫点及び加工待ち在庫点における i 品目の t 期末目標在庫量
- $Q^{n(i)}$: 第 n 工程の i 品目についての計画期間全体の生産割当量
- $R^{n(i)}$: 第 n 工程の i 品目についての計画期間全体の引き取り割当量
- $e^{sn(i)}$: 直後工程 sn の i 品目を 1 個作るのに必要な第 n 工程の i 品目の個数
 $e^{sn(i)} \in \{1, 2, \dots\}$

なお、納入内示量、在庫量、仕掛量およびサブロットの大きさに関する上記の記号は全て非負の整数である。

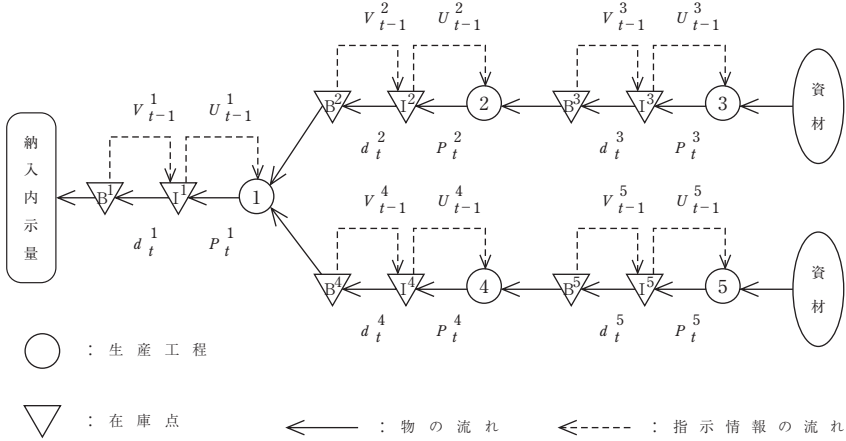


図2 モデルの概念図の例 (N=5)

- $I_t^{n(i)}$: 第 n 工程での i 品目の t 期末における加工済み在庫量
- $B_t^{n(i)}$: i 製品の t 期末における納入待ち在庫量 ($n=1$ の場合) 及び第 n 工程の後工程 sn への i 部品の t 期末の加工待ち在庫量 ($n \in J1$ の場合)
- $U_t^{n(i)}$: 第 n 工程の i 品目について t 期末に計算される $t+1$ 期の生産指示量
- $V_t^{n(i)}$: 第 n 工程の i 品目について t 期末に計算される $t+1$ 期の加工済み在庫からの引き取り指示量
- $P_t^{n(i)}$: 第 n 工程での i 品目の t 期中の実際の生産量
- $d_t^{n(i)}$: 第 n 工程での i 品目の t 期中の実際の引き取り量
- $X_t^{n(i)}$: 第 n 工程で加工される i 品目についての t 期における段取り替えの回数を表す変数 ($n \in K$)
- $U_0^{n(i)}$: 第 n 工程の i 品目についての初期生産指示量 (決定変数)
- $V_0^{n(i)}$: 第 n 工程の i 品目についての加工済み在庫からの初期引き取り指示量 (決定変数)
- z_r^-, z_r^+ : 補充目標在庫水準の総和の最小化に関する目標制約式に対する差異

変数

z_t^{n-} , z_t^{n+} : 第 n 工程の t 期での生産能力調整に関する目標制約式に対する差
異変数 (決定変数)

3.3 定式化

本稿で対象とする引っ張り型生産指示方式の多目標モデルは、次のような目標計画法の問題に定式化される。

lexicographically minimize

$$u = \{(z_r^+ - z_r^-), \sum_{n=1}^N \sum_{t=1}^T z_t^{n+}\} \quad (1)$$

subject to

$$\sum_{n=1}^N \sum_{i=1}^M (I_0^{n(i)} + \sum_{j=1}^{LP^n} P_{j-LP^n}^{n(i)} + U_0^{n(i)} + B_0^{n(i)} + \sum_{j=1}^{LH^n} d_{j-LH^n}^{n(i)} + V_0^{n(i)}) + (z_r^- - z_r^+) = 0 \quad (2)$$

$$\sum_{i=1}^M a^{n(i)} P_t^{n(i)} + \sum_{i=1}^M S^{n(i)} X_t^{n(i)} + (z_t^{n-} - z_t^{n+}) = W_t^n \quad (n \in K; t = 1, 2, \dots, T) \quad (3)$$

$$\sum_{i=1}^M a^{n(i)} P_t^{n(i)} + (z_t^{n-} - z_t^{n+}) = W_t^n \quad (n \in J-K; t = 1, 2, \dots, T) \quad (4)$$

$$I_t^{n(i)} = I_{t-1}^{n(i)} + P_{t-LP^n}^{n(i)} - d_t^{n(i)} \quad (i = 1, 2, \dots, M; n \in J; t = 1, 2, \dots, T) \quad (5)$$

$$B_t^{1(i)} = B_{t-1}^{1(i)} + d_{t-LH^1}^{1(i)} - D_t^{1(i)} \quad (i = 1, 2, \dots, M; t = 1, 2, \dots, T) \quad (6)$$

$$B_t^{n(i)} = B_{t-1}^{n(i)} + d_{t-LH^n}^{n(i)} - e^{sn(i)} P_t^{sn(i)} \quad (i = 1, 2, \dots, M; n \in J1; t = 1, 2, \dots, T) \quad (7)$$

$$U_t^{n(i)} = U_{t-1}^{n(i)} - P_t^{n(i)} + d_t^{n(i)} \quad (i = 1, 2, \dots, M; n \in J; t = 1, 2, \dots, T) \quad (8)$$

$$V_t^{1(i)} = V_{t-1}^{1(i)} - d_t^{1(i)} + D_t^{1(i)} \quad (i = 1, 2, \dots, M; t = 1, 2, \dots, T) \quad (9)$$

$$V_t^{n(i)} = V_{t-1}^{n(i)} - d_t^{n(i)} + e^{sn(i)} P_t^{sn(i)} \quad (i = 1, 2, \dots, M; n \in J1; t = 1, 2, \dots, T) \quad (10)$$

$$P_t^{n(i)} \leq U_{t-1}^{n(i)} \quad (i = 1, 2, \dots, M; n \in J; t = 1, 2, \dots, T) \quad (11)$$

$$d_t^{n(i)} \leq V_{t-1}^{n(i)} \quad (i = 1, 2, \dots, M; n \in J; t = 1, 2, \dots, T) \quad (12)$$

$$P_t^{n(i)} = L^{n(i)} X_t^{n(i)} \quad (i = 1, 2, \dots, M; n \in K; t = 1, 2, \dots, T) \quad (13)$$

$$\sum_{t=1}^T P_t^{n(i)} \geq Q^{n(i)} \quad (i = 1, 2, \dots, M; n \in J) \quad (14)$$

$$\sum_{t=1}^T d_t^{n(i)} \geq R^{n(i)} \quad (i = 1, 2, \dots, M; n \in J) \quad (15)$$

ここで,

$$R^{1(i)} = \max \{ 0, \sum_{t=1}^T D_t^{(i)} - B_0^{1(i)} + SB_T^{1(i)} \} \quad (i = 1, 2, \dots, M) \quad (16)$$

$$Q^{n(i)} = \max \{ 0, R^{n(i)} - I_0^{n(i)} + SI_T^{n(i)} \} \quad (i = 1, 2, \dots, M; n \in J) \quad (17)$$

$$R^{n(i)} = \max \{ 0, e^{sn(i)} Q^{sn(i)} - B_0^{n(i)} + SB_T^{n(i)} \} \quad (i = 1, 2, \dots, M; n \in JI) \quad (18)$$

$$B_t^{1(i)} \geq SB_t^{1(i)} \quad (i = 1, 2, \dots, M; t = 1, 2, \dots, T) \quad (19)$$

$$B_t^{n(i)} \geq SB_t^{n(i)} \quad (i = 1, 2, \dots, M; n \in JI; t = 1, 2, \dots, T) \quad (20)$$

$$I_t^{n(i)} \geq SI_t^{n(i)} \quad (i = 1, 2, \dots, M; n \in J; t = 1, 2, \dots, T) \quad (21)$$

$$X_t^{n(i)} : \text{非負の整数} \quad (i = 1, 2, \dots, M; n \in K; t = 1, 2, \dots, T) \quad (22)$$

$$P_t^{n(i)}, d_t^{n(i)} : \text{非負の整数} \quad (i = 1, 2, \dots, M; n \in J; t = 1, 2, \dots, T) \quad (23)$$

$$U_0^{n(i)}, V_0^{n(i)} : \text{非負の整数} \quad (i = 1, 2, \dots, M; n \in J) \quad (24)$$

$$z_t^{n-}, z_t^{n+} \geq 0 \quad (n \in J; t = 1, 2, \dots, T) \quad (25)$$

$$z_r^-, z_r^+ \geq 0 \quad (26)$$

本稿が対象とするモデルは、前節で述べたように、(1)計画期間中の各工程、各品目の補充目標在庫水準の総和を最小にすること、および(2)生産必要量に対し用意された基準生産能力の他に、さらに追加手当すべき生産能力の総和を最小にすることを目標とした多目標モデルである。また本稿で提示する解法は、具体的には、多目標モデルを目標計画法 (goal programming) ⁷⁾の問題に定式化し、反復的目標計画法 (sequential goal programming) の枠組みを採用するというものである。

式(1)は辞書式最小化、つまり補充目標在庫水準の総和の最小化を第1優先目標とし、追加手当すべき生産能力の総和の最小化を第2優先目標とすることを示す達成関数である。式(2)は、補充目標在庫水準の総和の最小化に関する目標制約式である⁸⁾。式(3)、(4)は各工程の各期における生産必要量と基準生産能力との差異を表現しており、追加手当すべき生産能力の総和の最小化に関する目標制約式である。目標制約式(3)、(4)に対する差異変数 z_t^{n-} 、 z_t^{n+} の値が、生産能力調整に必要なデータとなる。つまり、この値によって事前に用意すべき生産能力の設定を行うことができる。

式(5)~(7)は各在庫点の各期末における在庫量のバランス式である。式(8)

～(10)は各工程，各品目の生産指示量，引き取り指示量のバランス式である。またこれらのバランス式(8)～(10)は，各工程における生産・引き取り指示量はその直後工程で実際に消費された量に基づいて決定されるという引っ張り型生産指示方式の概念[43]を表現している。式(11)，(12)は指示量による生産量制約および引き取り量制約を示している。式(13)は3.1項で述べた条件(9)に対応するもので，段取り替えが必要な工程での生産量とサブロットとの関係を表している。式(14)，(15)は3.1項で述べた条件(3)に対応するもので，計画期間全体の割当量による生産・引き取り量制約を表現している。式(16)～(18)はその割当量を定めたものである。式(19)～(21)は各在庫点における期末在庫量に対する制約を表しているが，同時に式(19)は製品納入の保証を表している。また同様に，式(20)，(21)は在庫による実際の生産量と引き取り量に対する制約を意味している。式(22)～(24)は段取り回数，生産量，引き取り量および初期指示量に対する非負整数制約である。式(25)，(26)は差異変数の非負制約である。

なお $X_t^{n(i)}$ ， $P_t^{n(i)}$ ， $d_t^{n(i)}$ ， $U_0^{n(i)}$ ， $V_0^{n(i)}$ および納入内示量，初期在庫量，仕掛量およびサブロットの大きさの非負整数性と式(11)，(12)および式(19)～(21)により，各期の指示量 $U_t^{n(i)}$ ， $V_t^{n(i)}$ および期末在庫量 $I_t^{n(i)}$ ， $B_t^{n(i)}$ の非負整数性は保証されている。

4. 多目標モデルに対する解法

4.1 解法の枠組み

本稿では，引っ張り型生産指示方式の多目標モデルに対して，前稿で提示したモデリングシステムを用いた解法アプローチを基礎として，反復的目標計画法の枠組みを用いた解法を示す。その概要は次の通り。

- (1) 整数変数の数を減らすために数式モデルを再定式化する⁸⁾。
- (2) 反復的目標計画法の枠組みを採用した結果，単一目標問題に帰着される問題に対して，前稿で提案した近似計算手続きを繰り返し適用し近似最適解を求める。

前節では多目標モデルを多くの整数変数を持つ目標計画法の問題に定式化した。この目標計画法の問題は、一般的には以下のような整数目標計画問題として表される。

lexicographically minimize

$$\mathbf{u} = \{f_1(\mathbf{z}^-, \mathbf{z}^+), f_2(\mathbf{z}^-, \mathbf{z}^+), \dots, f_G(\mathbf{z}^-, \mathbf{z}^+)\} \quad (27)$$

subject to

$$\sum_{y=1}^Y d_{hy} x_y + z_h^- - z_h^+ = l_h \quad (h = 1, 2, \dots, H) \quad (28)$$

$$\mathbf{Ax} \leq \mathbf{b} \quad (29)$$

$$\mathbf{x} = (x_1, x_2, \dots, x_Y) : \text{非負の整数} \quad (30)$$

$$\mathbf{z}^-, \mathbf{z}^+ \geq \mathbf{0} \quad (31)$$

ここで、 $f_g(\mathbf{z}^-, \mathbf{z}^+)$ は優先度 g の目標制約式に対する差異変数の線形関数 ($g = 1, 2, \dots, G$),

$$\mathbf{z}^- = (z_1^-, z_2^-, \dots, z_H^-), \quad \mathbf{z}^+ = (z_1^+, z_2^+, \dots, z_H^+)$$

$$z_h^-, z_h^+ \text{ は } h \text{ 番目の目標制約式に対する差異変数 } (h = 1, 2, \dots, H),$$

$$x_y \text{ は非負の整数変数 } (y = 1, 2, \dots, Y),$$

$$d_{hy} \text{ は } h \text{ 番目の目標制約式における変数 } x_y \text{ に対する係数 } (h = 1, 2, \dots, H),$$

$$(y = 1, 2, \dots, Y),$$

$$l_h \text{ は } h \text{ 番目の目標制約式における希求水準 } (h = 1, 2, \dots, H),$$

\mathbf{A} および \mathbf{b} は厳密に守るべき制約条件式に関わる係数行列および右辺ベクトル。

つまり、達成関数(27)は式(1)に、式(28)は目標制約式(2)～(4)に、式(29)は厳密に守るべき制約条件式(5)～(21)に、条件(30)は変数に対する非負整数条件(22)～(24)に、そして条件(31)は差異変数に対する非負条件(25)、(26)に各々対応している。

本稿では整数変数を減らすためモデルを再定式化した後、式(27)～(31)で表される整数目標計画問題に対して、反復的目標計画法の枠組みを用いて、次のような計算手続きを提案する。

手順1: $g=1$ とおく。

手順2：以下に示すような優先度 g の差異変数の線形関数に対する最小化問題を設定し、これを解く。ここで得られた近似最適解における $f_g(z^-, z^+)$ の値 u_g^* とする。

$$\begin{aligned} & \text{minimize} && f_g(z^-, z^+) \\ & \text{subject to} && (28) \sim (31) \end{aligned}$$

手順3：もし、 $g = G$ ならば手順5へ。そうでない場合は手順4へ。

手順4：手順2で得られた値 u_g^* を用いて、以下に示す制約式(32)を厳密に守るべき制約式(29)に追加し、 $g = g+1$ とおき、手順2へ。

$$f_g(z^-, z^+) \leq u_g^* \quad (32)$$

手順5：手続きを終了する。

最後の優先度 G に対する最小化問題を解いて得られた解が、元の整数目標計画問題の近似最適解となる。手順2で設定される単一目標の問題は、多くの整数変数を含む混合整数計画問題となるが、本稿では、その解法として前稿で提案した近似計算手続きを適用し近似最適解を求める。

以下、本稿では代表的なモデリングシステム Mosel を用いた新たな解法アプローチを例示するが、その前に単一目標の混合整数計画問題を解くために本研究で使用している数理計画ソフトウェア Xpress-MP の最適化モジュールである Optimizer [32] (これ以降、Optimizer は最適化モジュールを意味しているものとする) において、初期設定で定められている計算手続きと本研究で採用している近似計算手続きについて述べておく。

4.2 標準手続き

本研究で使用している数理計画ソフトウェアの最適化モジュール Optimizer では、整数計画問題を解くために分枝カット法が採用されている。分枝カット法とは、分枝限定法による探索の過程で切除平面 (cut) を加えながら、緩和問題である線形計画問題を解いていくことで整数解探索の効率化を図ろうとする解法である。いわば分枝限定法と切除平面法の組み合わせと考えられるが、整数計画法の研究においては現在最も注目されているアプローチの一つであ

る²⁾。

このように分枝カット法では基本的には分枝限定法の手続きを進行させるため、その計算戦略が計算時間に大きな影響を及ぼす。Optimizerでは、パラメータを操作することにより分枝限定法に関わる計算戦略の設定が可能である。

本研究では、次の項目に関するパラメータを操作し計算手続きを作成している。

- (1) ノードの選択
- (2) 分枝変数の選択
- (3) ノード棄却の判定基準値

これらの項目に関するパラメータに対して、Optimizerの初期設定は次の通りである。

- (1) ノードの選択
 - ・下界値優先則と奥行き優先則との折衷則。
 - ・つまり、最後に解いたノードの二つの子問題のうち、良いノードを選ぶ。
 - ・両方の子問題とも捨てられた場合は、待ちノード全体から良いノードを選ぶ。
 - ・良いノードとは、選択の対象となっている子問題の中で最良の下限值（最小化問題の場合）を持つノードとする。
- (2) 分枝変数の選択
 - ・擬コスト（pseudo-cost）を用いて評価関数の劣化が最も大きいと予想される変数を選ぶ⁹⁾。
- (3) ノード棄却の判定基準値
 - ・基本的には、最良整数解における評価関数値であるが、式(33)で示される設定であるため評価関数値が同じ整数解は探索されない。

$$\text{CUTOFF} = \text{IPOBJ} + \text{ADDCUT} \quad (33)$$

ここで、 CUTOFF：ノード棄却の判定基準値

IPOBJ：その時点での最良整数解における評価関数値

$$\text{ADDCUT} = \min (-1.0 \text{ E} - 5, -1.0 \text{ E} - 6 \times \text{LPOBJ})$$

LPOBJ : 変数に対する整数条件が緩和された線形計画問題
の最適解における評価関数値

以下、単一目標の混合整数計画問題のレベルにおいて、これらの設定を用いて構成した多目標モデルに対する計算手続きを標準手続きと呼ぶ。

4.3 近似計算手続き

標準手続きに対して、近似計算手続きの計算戦略は次の通りである。

(1) ノードの選択

・標準手続きと同様、下界値優先則と奥行き優先則との折衷則。

(2) 分枝変数の選択

・モデリングシステム Mosel によるモデル記述の中で優先順位を指定する。

・その優先順位は順に、(1)段取り替えの回数を表す変数 $X_i^{n(i)}$ 、(2)生産および引き取りの初期指示量 $U_0^{n(i)}$ 、 $V_0^{n(i)}$ 、(3)実際の生産量および引き取り量 $P_i^{n(i)}$ 、 $d_i^{n(i)}$ とする。

なおトヨタ生産方式においては段取り替え作業の取扱いが一つの要点¹⁰⁾となるが、この段取り替えに関わる変数を優先的に分枝させるという優先順位の導入が、本研究で採用している近似計算手続きの特徴¹¹⁾となっていることが今回行った数値計算においても明らかになっている (後述 6 節を参照)。

(3) ノード棄却の判定基準値

・次の式(34)で示される値をノード棄却の判定基準値として設定する。

$$\text{CUTOFF} = \text{IPOBJ} / (1 + \alpha) \quad (34)$$

ここで、CUTOFF : ノード棄却の判定基準値

IPOBJ : その時点での最良整数解における評価関数値

α : 下限値 (最小化問題の場合) からの相対誤差

ここで示した近似計算手続きは、下限値を基準とした相対誤差に基づいた計算手続きである。つまり少ない計算量で相対誤差がある値 α 以内であること

を保証する近似最適解が得られる。以下、単一目標の混合整数計画問題のレベルにおいて、これらの設定を用いて構成した多目標モデルに対する近似計算手続きを単に**近似手続き**と呼ぶ。

4.4 数値計算で用いた計算手続き

ここでは、本研究で行った数値計算で用いた計算手続きを示す。各計算手続きの計算結果については6節において詳述する。

(1) 標準手続き

4.2項で述べた標準手続き、即ち Optimizer の初期設定による多目標モデルの計算手続き。

(2) 優先順位

分枝変数の選択以外は、標準手続きと同じ計算手続き。またノード棄却の判定基準値の設定以外は、近似手続きと同じ計算手続きでもある。つまりノード棄却の判定基準値については初期設定(4.2項を参照)を用いる計算手続きであり、この計算手続きと標準手続きおよび近似手続きの計算結果を比較することで、近似計算手続きにおける優先順位データの導入の効果そしてノード棄却の判定基準値の設定が計算時間に及ぼす影響が明らかになる。

(3) 近似手続き

4.3項で述べた近似手続きの内、相対誤差 $\alpha=0.01$ とした多目標モデルの計算手続き。

5. モデリングシステムを用いた解法アプローチ

多目標モデルに対する解法アプローチの試みとして、これまで著者らは図1に示すような枠組みを構築してきた。しかし、その解法システムを実現するためにはスクリプト環境など何らかの形で処理全体を制御する仕組みが別途必要となっており、さらには多目標モデルを対象としている関係で、スクリプト言語による処理の記述は極めて複雑なものとなっていた。

これに対して、モデリングシステムでは対象とする数理計画問題のモデルと

その解き方を同時に記述し問題解決を図ることができる。つまり、モデルの記述 (model describing) とモデルの解法 (model solving) を一つの環境で実現している。以下では、3節で示した引っ張り型生産指示方式の多目標モデルを対象として、モデリングシステム Mosel を用いて実現した新たな解法アプローチを例示する。

5.1 モデルを記述する

まず目標制約式 (2)～(4)は、次のように記述する。なお式 (35)は計画期間中の各工程、各品目の補充目標在庫水準の総和の最小化に関する目標制約式を意味する記述であるが、ここでは定数項部分 (初期在庫量および仕掛量) を消去している。式 (36)、(37)が各工程の各期における生産必要量と基準生産能力との差異を表現しており、追加手当すべき生産能力の総和の最小化に関する目標制約式の記述である。ここで、各々の2行目に示されているのは、工程 n が段取り変え作業の考慮が必要である場合 ($SK(n)=1$)、あるいは考慮が必要でない場合 ($SK(n)=0$) にのみ、この制約を生成しなさいという条件付きの制約生成を表現している。

!!! Goal Constraint (2)

GC2 :=

$$\begin{aligned} & \text{sum}(n \text{ in } 1..N, i \text{ in } 1..M) V0(i, n) + \\ & \text{sum}(n \text{ in } 1..N, i \text{ in } 1..M) U0(i, n) + z\text{rminus} - z\text{rplus} = 0 \end{aligned} \quad (35)$$

!!! Goal Constraints (3) - (4)

forall(n in 1..N, t in 1..T| SK(n) = 1)

GC3(n, t) :=

$$\begin{aligned} & \text{sum}(i \text{ in } 1..M) (a(n, i) * L(n, i) + S(n, i)) * X(i, t, n) + \\ & z\text{minus}(t, n) - z\text{plus}(t, n) = W(n, t) \end{aligned} \quad (36)$$

forall(n in 1..N, t in 1..T| SK(n) = 0)

GC4(n, t) :=

$$\text{sum}(i \text{ in } 1..M) a(n, i) * P(i, t, n) +$$

$$zminus(t, n) - zplus(t, n) = W(n, t) \quad (37)$$

次に達成関数を表わす式(1)の記述であるが、上記の目標制約式の記述を踏まえ、式(38)が補充目標在庫水準の総和の最小化を第1優先目標とすることに対応し、式(39)が追加手当すべき生産能力の総和の最小化を第2優先目標とすることに対応している。

!!! Lexicographically minimize

$$goal(1) := zrplus - zrminus \quad (38)$$

$$goal(2) := \text{sum}(n \text{ in } 1..N, t \text{ in } 1..T) zplus(t, n) \quad (39)$$

その他の制約条件式の記述も含め、上記に示したように、Moselでは数式モデルに近い表現でモデルの記述を行うことができ、これによってOptimizerの入力データであるマトリックスファイルの生成を指示している。

5.2 解き方を記述する：近似計算手続きの記述

4.3項で示した分枝限定法に関わる近似計算手続きのうち、分枝変数の選択に関わる優先順位は次のように記述する。式(40)は段取り替えの回数を表す変数 $X_t^{n(i)}$ に第1優先があることを、式(41)、(42)は生産および引き取りの初期指示量 $U_0^{n(i)}$ 、 $V_0^{n(i)}$ に第2優先があることを示している。分枝変数の選択に関わる優先順位を表すパラメータ XPRS_PR の標準値は500であり、この値よりも小さな値を設定することで優先順位を表現できる。なお式(40)の1行目に示されているのは、工程 n が段取り替え作業の考慮が必要な工程の場合、この優先順位の指定が有効であることを示しており、制約の生成のみならずパラメータの設定についても条件付の設定が可能であることを示している。

最後の式(43)はここで設定された分枝変数の選択に関わる優先順位を、あるファイルに書き出すことを示している。この例ではファイル名として ex.dir という名前が設定されている。

$$\begin{aligned} & \text{forall}(i \text{ in } 1..M, t \text{ in } 1..T, n \text{ in } 1..N \mid SK(n) = 1) \\ & \quad \text{setmipdir}(X(i, t, n), XPRS_PR, 100) \\ & \text{forall}(i \text{ in } 1..M, n \text{ in } 1..N) \end{aligned} \quad (40)$$

```
setmipdir (V 0 (i, n), XPRS_PR, 200) (41)
```

```
forall (i in 1..M, n in 1..N)
```

```
setmipdir (U 0 (i, n), XPRS_PR, 200) (42)
```

```
writedirs ('ex.dir') (43)
```

次に、ノード棄却の判定基準値の設定については以下のように記述する。これは4.3項で述べたノード棄却の判定基準値の式(34)で示される値を設定するためのもので、手続き名 `setcutoff` としてその手続き (procedure) が記述されている。式(34)に対応させるとパラメータ `XPRS_mipobjval` は整数解が得られた時点での評価関数值 `IPOBJ` を、`XPRS_mipabscutoff` は Optimizer に指示するノード棄却の判定基準値 `CUTOFF` を表している。なお `ALPHA` は相対誤差 α に対応しており、その値はモデル記述の最初の段階で設定しておけばよい。6節で示す数値計算例である近似手続きでは、この値を0.01に設定している。

また `getparam` および `setparam` は、Optimizer からその時点でのあるパラメータの値を受け取る (`getparam`) あるいはパラメータを設定し Optimizer へ与える (`setparam`) 役割を果たすものである。Mosel は、このようにパラメータの受け渡しを行うことで Optimizer に対して細かな求解指示を与えることができる。

```
procedure setcutoff
  declarations
    objval : real
    cutoff : real
    cutoffnew : real
  end-declarations (44)
  objval := getparam ('XPRS_mipobjval ')
  cutoff := getparam ('XPRS_mipabscutoff ')
  cutoffnew := objval / (1 + ALPHA)
  setparam ('XPRS_mipabscutoff ', cutoffnew)
end-procedure
```

これらの記述の後、モデリングシステム Mosel は以下のようなコマンドを発行し最適化モジュールである Optimizer に求解の指示を与える。式(45)は callback 機能と言われるもので、整数解が得られた時点で求解を一時停止し、式(44)で示した procedure setcutoff の手続きを実行した後、求解を再開せよという指示を表している。また式(46)の3行目は計算時間 (CPU time) が 3600 秒を経過した時点で計算を打ち切れという指示に相当する。最後の式(47)は式(40)～(42)で設定し、式(43)の指示でファイル名 ex.dir に保存されている分枝変数の選択に関わる優先順位情報を読み取り、評価関数を式(38)で示している補充目標在庫水準の総和、つまり第1優先目標 goal(1)とする混合整数計画問題の最小化 (minimize) を実行せよという Optimizer への指示を表している。

```
setcallback(XPRS_CB_INTSOL, 'setcutoff ') (45)
```

```
setparam('XPRS_loadnames ', true)
setparam('XPRS_verbose ', true) (46)
```

```
setparam('XPRS_maxtime ', - 3600)
setparam('XPRS_solutionfile ', 1)
```

```
loadprob(goal(1))
readdir('ex.dir ') (47)
minimize(goal(1))
```

5.3 解き方を記述する：反復的目標計画法の記述

4.1項で示した反復的目標計画法の計算手続きにおいては、上述の式(47)は手順2に対応する。この式(47)に続けて、次のように記述することで反復的目標計算法の計算手続きを実現することができる。まず式(48)は、第1優先目標の最小化の結果判定を行う記述である。ここでは第1優先目標の最小化において近似最適解が得られている場合、その解の出力を指示している。

```
if getparam('XPRS_mipsols ') = 0 then
  writeln('Goal 1 : NO MIP Solutions ')
  exit(1) (48)
```

else

```
command('writeprtsol')
writeln('Goal 1 : MIP Solution is obtained')
writeln('Go to the stage of Goal 2')
```

式(48)において、第1優先目標の最小化において近似最適解が得られていると判定された場合には、以下の手続きに進む。式(49)は、第1優先目標の最小化から第2優先目標の最小化へ移行する前準備として、第2優先目標の最小化におけるノード棄却の判定基準値を設定している記述である。ここでは、第1優先目標の最小化が完了した時点の第2優先目標の値 `getact(goal(2))` をもとに、ノード棄却の判定基準値を新たに設定することで、第2優先目標の最小化における計算量の削減を図っている。なお式(49)の第1行目は、第1優先目標の最小化が完了した時点で、既に第2優先目標の最小化も完了している場合、手続きを終了させるための記述である。

```
if getact(goal(2)) <> 0 then
  writeln('goal(2) value = ', getact(goal(2)))
  cutoff 2 := getact(goal(2)) / (1 + ALPHA)
  setparam('XPRS_mipabscutoff', cutoff 2)
  writeln('cutoff value of Goal 2 = ', cutoff 2)
```

(49)

次の式(50)の記述は、第1優先目標の近似最適解における目標値 `getobjval` を用いて、第1優先目標の値を制約条件式に追加しているものである。これは4.1項で示した反復的目標計画法の計算手続きの手順4における式(32)に対応する記述である。式(51)は、手順2に戻り第2優先目標の最小化を行うことを指示しているものである。なお、この第2優先目標の最小化の段階においても、5.2項で示した分枝限定法に関わる近似計算手続きの記述が有効であり、少ない計算量で近似最適解を求めようとしている。

```
goal(1) := goal(1) <= getobjval
```

(50)

```
loadprob(goal(2))
readdirs('ex.dir')
minimize(goal(2))
```

(51)

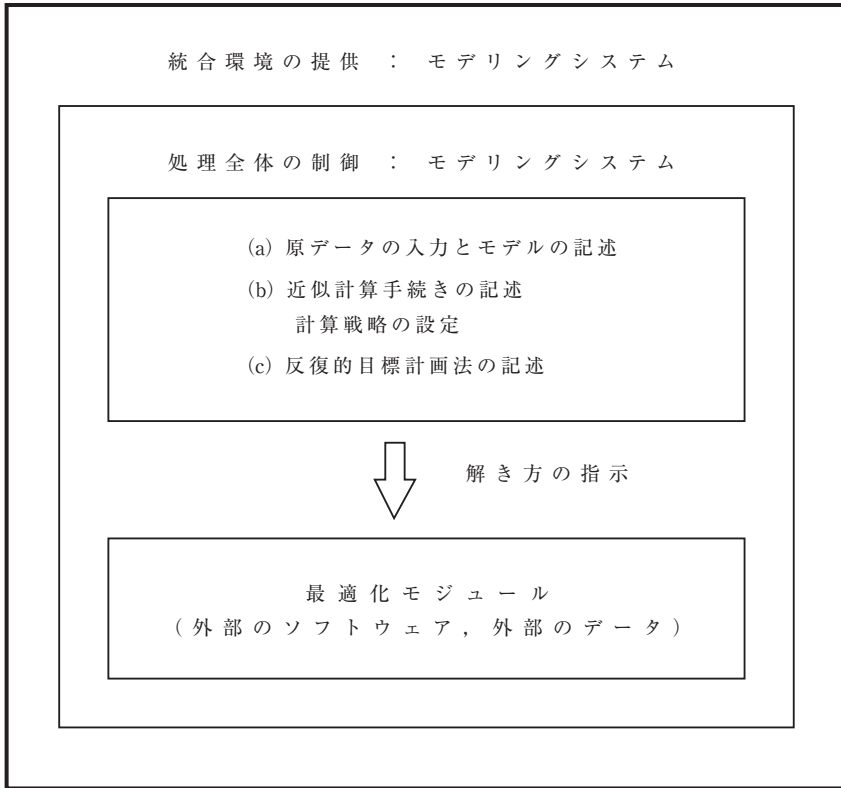


図3 モデリングシステムを用いた解法アプローチ

図3に、本稿で提案するモデリングシステム Mosel を用いた解法アプローチを示す。多目標モデルに対するこれまでの枠組み、つまり2節の図1で示したモデル記述言語を用いた場合と異なり、モデリングシステムが処理全体の制御を行っており、別途必要であったスクリプト言語などによる煩雑な処理手順の記述はない。モデル記述によるマトリックスファイルの生成のみならず、分枝限定法に関わる近似計算手続きおよび反復的目標計画法の記述を行うことで、モデリングシステムが最適化モジュールに対してモデルの解き方を指示する形式となっている。具体的には、最適化モジュールとのインターフェースの

働きをする `mmxprs` といわれるモジュールを介して最適化モジュールを制御している。上述の `setcallback`, `readdir` および `minimize` などのコマンドも全てそのモジュールを介して `Optimizer` へ伝達されている [32]。

なお Mosel ではインターフェースの働きをする別のモジュールを介在させることで、`Xpress-MP` の最適化モジュールだけではなく外部のソフトウェアあるいは外部のデータを制御することも可能となっている。このインターフェースの働きをする一群の Mosel Modules の存在、つまり `modularity` の概念がモデリングシステムの機能をさらに拡張させるものとして Mosel の特徴の一つだといわれている [38]。また図 3 に示しているように、本研究で使用している数理計画ソフトウェア `Xpress-MP` では、モデルの記述 (`model describing`) とモデルの解法 (`model solving`) を一つの画面上で行う `Xpress-IVE` といわれる統合環境が提供されている。この統合環境の提供も広い意味でのモデリングシステムの機能といえる。

6. 数値計算例

6.1 数値計算の条件

モデリングシステムの先進性を活用した解法アプローチの有効性を検証するため、ある自動車部品製造メーカーにおける製造工程を対象に数値計算を行った。対象とした製造工程はガソリントankに使用する小物自動車部品を製造しており次の 5 工程より構成される。その流れ図は図 2 と同じものである。

- (1) 組立工程 (ロー付け) : $n = 1$
- (2) プレス工程 1 (タンデム工程) : $n = 2$
- (3) プレス工程 2 (フープライン) : $n = 3$
- (4) ベンディング工程 (ベンダー) : $n = 4$
- (5) パイプ加工工程 (自動切断機) : $n = 5$

また、用いた入力データ等の具体的な数値計算の条件は次の通りである。

- (1) 計画期間は 2 週間であり、 t は 1 日単位で $T = 10$ 日である。
- (2) 各工程で生産する品目として、代表的な 3 車種の部品を考える。

- (3) 段取り替えが必要な工程はプレス工程 1($n=2$) およびプレス工程 2($n=3$) である。
- (4) 組立工程($n=1$) およびプレス工程 1($n=2$) における生産リードタイムは 1 日であり, その他の工程における生産リードタイムおよび全ての工程における引き取りリードタイムは十分に短い。

(5) 入力データ

(a) 納入内示量

$$D_t^{(1)} = 20, \quad D_t^{(2)} = 15, \quad D_t^{(3)} = 5 \quad (t = 1, 10),$$

$$D_t^{(1)} = 30, \quad D_t^{(2)} = 25, \quad D_t^{(3)} = 5 \quad (t = 2, 3, \dots, 9)$$

(b) 基準生産能力

$$W_t^n = 420 \text{ 分} \quad (n = 1, 2, \dots, 5; t = 1, 2, \dots, 10)$$

(c) 単位量当たり加工時間

$$a^{n(i)} = 6 \text{ 分} \quad (i = 1, 2, 3; n = 1, 2),$$

$$a^{n(i)} = 3 \text{ 分} \quad (i = 1, 2, 3; n = 3, 4, 5)$$

(d) 段取り替え時間

$$S^{2(i)} = 15 \text{ 分}, \quad S^{3(i)} = 10 \text{ 分} \quad (i = 1, 2, 3)$$

(e) サブロットの大きさ

$$L^{2(i)} = 10, \quad L^{3(i)} = 10 \quad (i = 1, 2, 3)$$

(f) 初期在庫量

$$B_0^{n(1)} = 14, \quad B_0^{n(2)} = 12, \quad B_0^{n(3)} = 5$$

$$I_0^{n(1)} = 14, \quad I_0^{n(2)} = 12, \quad I_0^{n(3)} = 5 \quad (n = 1, 2, \dots, 5)$$

(g) 期末目標在庫量

$$SB_t^{n(1)} = 10, \quad SB_t^{n(2)} = 8, \quad SB_t^{n(3)} = 3$$

$$SI_t^{n(1)} = 10, \quad SI_t^{n(2)} = 8, \quad SI_t^{n(3)} = 3 \quad (n = 1, 2, \dots, 5; t = 1, 2, \dots, 10)$$

(h) 生産仕掛量

$$P_0^{1(1)} = 25, \quad P_0^{1(2)} = 20, \quad P_0^{1(3)} = 5$$

$$P_0^{2(1)} = 30, \quad P_0^{2(2)} = 20, \quad P_0^{2(3)} = 0$$

なお, 部品構成を表す $e^{sm(i)}$ は全て 1 である。

以上の生産条件の下での実際に解くべき計画問題の規模は、目標制約式が 51 制約、厳密に守るべき制約式が 606 制約、整数変数は 330 変数そして差異変数が 102 変数である。

なお、数値計算で用いた計算環境は次の通りである。

- | | |
|-----------------------|--|
| (1) CPU | Pentium M 1.40 GHz |
| (2) RAM | 1 GB |
| (3) オペレーティングシステム (OS) | Windows XP Professional SP 1 |
| (4) 数値計画ソフトウェア | Xpress-MP Release 2006 A |
| | モデリングシステム Xpress Mosel Version 1.6.2 |
| | 最適化モジュール Xpress Optimizer Version 17.01.02 |
| | 統合環境 Xpress-IVE Version 1.17.04 |

6.2 計算結果と考察

前項 6.1 で示した条件のもとで実施した数値計算の結果を以下に述べる。まず第 1 優先目標である補充目標在庫水準の総和の最小化における三つの計算手続き（標準手続き、優先順位および近似手続き）の計算結果を表 1 に示す。表の見方は次の通り。

- (1) CPU Time : 計算手続きの実行に要した計算時間。
- (2) 生成ノード : 分枝限定法の過程で生成されたノードの数。但し、最終結果が計算打ち切りの場合においては、実際の生成ノード数は、その表示数から +100 の範囲にある。
- (3) 評価関数値 : 補充目標在庫水準の総和の内、定数項（初期在庫量および仕掛量）を除いた値。
- (4) goal(2) : 計算打ち切りも含め、第 1 優先目標の最小化が完了した時点での第 2 優先目標である追加手当てすべき生産能力の総和の値。

表 1 で示されているように、第 1 優先目標の最小化における標準手続きと優先順位との大きな違いは、探索を完了できているか否かである。既に述べたよ

表1 計算結果 (第1 優先目標)

第1 優先目標	標準手続き	優先順位	近似手続き $\alpha = 0.01$
最初の整数解			
CPU Time	2 秒	1 秒	1 秒
生成ノード	63	64	64
評価関数値	611	600	600
最良の整数解	23 回目	10 回目	6 回目
CPU Time	486 秒	358 秒	91 秒
生成ノード	133806	102292	25432
評価関数値	561	561	561
最終結果			
CPU Time	3600 秒	921 秒	120 秒
生成ノード	1121800	276085	39509
評価関数値	561	561	561
判定	計算打ち切り	探索完了	探索完了
goal(2)	150	180	195

うに整数計画問題を解くためのアルゴリズムは、近年、飛躍的に向上しており、最適化モジュールの初期設定による計算である標準手続きにおいても最良解は、486 秒で得られている。実は、結果としてこの最良解は最適解であるのだが、標準手続きにおいては、分枝限定法の手続きによる最適判定を 3600 秒以内に行うことができていない。一方、優先順位 of 計算手続きは、分枝変数の選択において 4.3 項で示した段取り替えに関わる変数を優先的に分枝させるという優先順位を導入したものである。これによって最良解への到達時間も早く (358 秒と 486 秒)、さらには 921 秒の時点で最適解の判定を行い、計算を完了することができる。

近似手続きは、分枝変数の選択における優先順位 of 導入に加え、さらにノード棄却 of 判定基準値を導入し少ない計算量で相対誤差がある値 α 以内であることを保証する近似最適解を得ようとする計算手続きである。今回の数値計算では、相対誤差 $\alpha = 0.01$ の場合であるが、優先順位 of 計算手続きに対して約 $1/8$ (120 秒と 921 秒) の計算時間で計算を完了することができる。しかも

近似最適解ではなく、優先順位の計算手続きと同じく最適解が得られている。

次に、第2優先目標である追加手当すべき生産能力の総和の最小化に移行した後の各計算手続きの計算結果を表2に示す。表の見方は次の通り(説明のないものは、表1と同じである)。

- (1) CUTOFF : 第1優先目標の最小化が完了した時点での第2優先目標である追加手当すべき生産能力の総和の値をもとに設定するノード棄却の判定基準値。標準手続きと優先順位については、表1のgoal(2)の値から0.0001を引いた値を設定している。近似手続きについては、4.3項の式(34)に基づいた値が設定される。
- (2) 評価関数値 : 第2優先目標である追加手当すべき生産能力の総和の値。
- (3) 経過時間 : 計算手続き全体、つまり第1優先目標の最小化の計算開始から第2優先目標の計算終了までの経過時間。

なお、表2で示されている第2優先目標の最小化においては、いずれの計算手続きにおいても最初の整数解が分枝限定法の手続きにおいて最適と判断され

表2 計算結果 (第2優先目標)

第2優先目標	標準手続き	優先順位	近似手続き $\alpha = 0.01$
CUTOFF	149.9999	179.9999	193.069
最初の整数解			
CPU Time	905 秒	363 秒	612 秒
生成ノード	164001	57327	84231
評価関数値	120	120	120
最良の整数解			
最終結果			
CPU Time	905 秒	363 秒	612 秒
生成ノード	164001	57327	84231
評価関数値	120	120	120
判定	探索完了	探索完了	探索完了
経過時間	5161 秒	1354 秒	771 秒

計算を終了している（カットの生成後に得られている下限値と一致したためである）。

表2に示されている計算結果より、第1優先目標の最小化と同様、第2優先目標の最小化においても分枝変数の選択における優先順位の導入が有効であることがわかる。優先順位の計算手続きは、ノード棄却の判定基準値が大きな値（179.9999と149.9999）であるにもかかわらず、標準手続きよりも短い計算時間（363秒と905秒）で計算を終了している。

近似手続きについては、(1)ノード棄却の判定基準値が大きな値（193.069）で第2優先目標の最小化を開始していること、(2)本数値例の場合、最初の整数解が最適解であることもあるが、表1でみられた大きな優位性はなく、優先順位の計算手続きよりも多くの計算時間を要している。しかしながら計算全体を通して見た場合、優先順位の計算手続きに対して、約1/2（771秒と1354秒）の計算時間で計算を完了することができている。

表1および表2の計算結果は、解くべき問題の特徴を踏まえた上で現在考えられている技法を組み入れることで、さらに計算量の削減を図ることができることを示している。本研究が対象としている数理計画モデルはトヨタ生産方式におけるカンバン方式の概念に基づいたモデルであるが、トヨタ生産方式においては段取り替え作業の取扱いが一つの要点となっている。この段取り替えに関わる変数を優先的に分枝させるという優先順位の導入が、計算量を削減させる上で有効であるという結果は極めて興味深いものである¹¹⁾。

追加手当すべき生産能力

目標制約式(3)、(4)に対する差異変数 z_i^{n-} 、 z_i^{n+} の値が、生産能力調整に必要なデータとなる。つまり、この値によって事前に用意すべき生産能力の設定を行うことができる。具体的には、提示したモデルの決定変数の一つである差異変数 z_i^{n+} の値が、基準生産能力 W_i^n の他に、さらに追加手当すべき生産能力の値を示している。今回の計算では追加手当すべき生産能力の総和は、120分となっている。近似手続きの計算結果における追加手当すべき生産能力の配

分は以下の通りであった。つまり、いずれもプレス工程1（タンデム工程）において生産能力の追加手当てが必要であることを示すものであった。

$$z_3^{2+} = 30 \text{ 分}, \quad z_5^{2+} = 30 \text{ 分}, \quad z_6^{2+} = 30 \text{ 分}, \quad z_7^{2+} = 30 \text{ 分}$$

初期指示量および補充目標在庫水準

決定変数，つまり計画期間のはじめに提示する第 n 工程の i 品目についての初期生産指示量および初期引き取り指示量は次の通りである。なおこれらの値も，近似手続きにおいて得られている値である。

$$\begin{aligned} U_0^{1(1)} &= 31, U_0^{1(2)} = 26, U_0^{1(3)} = 3; & V_0^{1(1)} &= 26, V_0^{1(2)} = 27, V_0^{1(3)} = 3; \\ U_0^{2(1)} &= 27, U_0^{2(2)} = 26, U_0^{2(3)} = 16; & V_0^{2(1)} &= 26, V_0^{2(2)} = 21, V_0^{2(3)} = 3; \\ U_0^{3(1)} &= 34, U_0^{3(2)} = 26, U_0^{3(3)} = 10; & V_0^{3(1)} &= 26, V_0^{3(2)} = 26, V_0^{3(3)} = 8; \\ U_0^{4(1)} &= 26, U_0^{4(2)} = 20, U_0^{4(3)} = 3; & V_0^{4(1)} &= 26, V_0^{4(2)} = 21, V_0^{4(3)} = 3; \\ U_0^{5(1)} &= 26, U_0^{5(2)} = 19, U_0^{5(3)} = 3; & V_0^{5(1)} &= 26, V_0^{5(2)} = 20, V_0^{5(3)} = 3. \end{aligned}$$

この値，初期在庫量および仕掛量によって，提案したモデルの各工程における補充目標在庫水準が与えられる。引っ張り型生産指示方式は，この補充目標在庫水準のもとに運用される。

7. まとめ

本稿では，前稿において課題として示した内容，つまりモデリングシステムによる解法アプローチを引っ張り型生産指示方式の多目標モデルに展開し，数値計算を通して提案アプローチの有効性を検証した。モデリングシステムによって，モデル記述のみならず計算手続き全体を記述することが可能となり，これまで解法システムの構築にあたり別途必要であったスクリプト言語などの環境は，もはや不要のものとなっている。

多目標モデルに対しては反復的目標計画法の枠組みを用いているため，個々の最小化の計算は単一目標の混合整数計画問題に帰着されている。従って本稿においても，これまで著者らが提案してきた計算量削減の指針，つまり求解の困難な問題に対しては，解くべき問題の特徴を踏まえた上で，複数のアプロー

チを組み合わせることで対処すれば有効な成果が得られることを例示したものとなっている。本研究で示している近似計算手続きは、いわば(1)分枝カット法によって対象としている問題の解の下限 (lower bound) を切り上げる、(2) ノード棄却の判定基準値において α という相対誤差を導入することによって解の上限 (upper bound) を切り下げる、さらに(3)解くべき問題の特徴を捉え、段取り替えに関わる変数を優先的に分枝させる優先順位データを導入するという3点の視点から構成されている求解戦略に基づくものである。今回の数値計算においては、このうち、(3)の方策が極めて有効であることをあらためて認識する結果となった。

その一方で、第2優先目標の最小化における計算結果は、さらに計算量削減の方策を検討する余地があることを示すものであった。今後は、数理計画法によるアプローチと他の方法論との併用などによる改善策を検討するとともに、引き続きモデル記述言語の動向についてもサーベイを進めていきたいと考えている。

謝辞

本稿は平成18年度専修大学研究助成「モデリングシステムを用いた数理計画問題の解法に関する研究」による研究成果の一部であることを記して、関係各位に厚くお礼申し上げます次第である。

<注>

*本稿中のシステム名および製品名は一般に各社の登録商標または商標です。

- 1) トヨタ生産方式、カンバン方式および引っ張り型生産指示方式については、秋庭他[1]、平木[4]、黒田他[10]、門田[13]、村松[14]および日本生産管理学会編[15]などを参照するとよい。
- 2) 分枝限定法および分枝カット法については、藤江[2]、茨木[5]、茨木—福島[6]、今野—鈴木編[8]、久保[9]、Beasley(ed.)[28]、Carter-Price[30]、Martin[42]、Nemhauser-Wolsey[44]およびWolsey[51]などを参照するとよい。また線形計画法および整数計画法の発展の歴史については、今野[7]、Bixby[29]などを参照するとよい。あわせて具体的な数値検証も含め整数計画問題に対する求解戦略のサーベイおよびソフトウェアの近年の動向については、宮代—松井

- [12], Atamturk-Savelsbergh[27], Johnson-Nemhauser-Savelsbergh[37]および Linderoth-Savelsbergh[39]などを参照するとよい。
- 3) モデリングシステムを含めたモデル記述言語の詳細については前田[11], 渡辺[20, 23], Kallrath (ed.) [38], Sharda-Rampal[45]および Williams[50]などを参照するとよい。また Linear Programming FAQ のホームページ[40]上にもモデル記述言語についての解説がある。なおモデル記述言語の近年の進展については, Atamturk-Savelsbergh[27], Kallrath (ed.) [38]などが詳しい。
 - 4) MRP (Material Requirements Planning) とカンバン方式との関連性を含め, 生産能力に関わる計画・管理活動については, 大野監修・門田編著 [16] を参照するとよい。また本稿が対象としている引っ張り型生産指示方式の数理計画モデルにおける生産能力制約の表現に関わる検討の詳細については, 渡辺 [20] を参照。
 - 5) 多目標モデルについては, Ehrgott-Gandibleux (eds.) [33], Ignizio[34], Ignizio-Cavalier[36]などを参照するとよい。
 - 6) 反復的目標計画法の詳細については, 瀬見[18], Crowder-Sposito[31], Ignizio [34], Ignizio-Peris[35], Ignizio-Cavalier[36], Markland-Vickery[41]などを参照するとよい。
 - 7) 目標計画法については, 注5および注6で示した文献の他に伏見他[3], 坂和[17]などを参照するとよい。
 - 8) 補充目標在庫水準の表現およびモデルの再定式化の詳細については渡辺[20], Watanabe-Hiraki[48]を参照。
 - 9) 擬コストの詳細については茨木[5], 今野, 鈴木編[8]などを参照するとよい。
 - 10) 段取り替えの詳細については門田[13]を参照するとよい。
 - 11) 本研究で提案している近似計算手続きの特徴については渡辺[20]-[22], Watanabe-Hiraki[47], [48]において詳細な数値検証を加えている。

参考文献

- [1] 秋庭雅夫, 黒田充, 田部勉, 石井和克, 宮崎晴夫, 市村隆哉:「生産管理システムの設計—その研究と活用—」, 日本能率協会, 東京 (1986)。
- [2] 藤江哲也:“整数計画問題に対する分枝カット法とカットの理論”, オペレーションズ・リサーチ, Vol. 48, No. 12, pp. 935-940 (2003)。
- [3] 伏見多美雄, 福川忠昭, 山口俊和:「経営の多目標計画—目標計画法の考え方と応用例—」, 森北出版, 東京 (1987)。
- [4] 平木秀作:「自動車の現地生産と部品調達」, 溪水社, 広島 (1996)。

- [5] 茨木俊秀：「組合せ最適化一分枝限定法を中心として」，産業図書，東京（1983）。
- [6] 茨木俊秀，福島雅夫：「最適化の手法」，共立出版，東京（1993）。
- [7] 今野浩：「役にたつ一次式—整数計画法「気まぐれな王女」の50年—」，日本評論社，東京（2005）。
- [8] 今野浩，鈴木久敏編：「整数計画法と組合せ最適化」，日科技連，東京（1982）。
- [9] 久保幹雄：「組合せ最適化とアルゴリズム」，共立出版，東京（2000）。
- [10] 黒田充，田部勉，圓川隆夫，中根甚一郎：「生産管理」，朝倉書店，東京（1989）。
- [11] 前田英次郎：「数理計画支援システム」，第5回 RAMP シンポジウム論文集，pp. 57-72（1993）。
- [12] 宮代隆平，松井知己：「ここまで解ける整数計画」，システム／制御／情報，Vol. 50, No. 9, pp. 363-368（2006）。
- [13] 門田安弘：「トヨタプロダクションシステム—その理論と体系—」，ダイヤモンド社，東京（2006）。
- [14] 村松林太郎：「新版生産管理の基礎」，国元書房，東京（1979）。
- [15] 日本生産管理学会編：「トヨタ生産方式」，日刊工業新聞社，東京（1996）。
- [16] 大野耐一監修，門田安弘編著：「トヨタ生産方式の新展開」，日本能率協会，東京（1983）。
- [17] 坂和正敏：「経営数理システムの基礎<線形計画法に基づく意思決定>」，森北出版，東京（1991）。
- [18] 瀬見博：「目標計画法の研究」，泉文堂，東京（1989）。
- [19] 反町洋一編：「線形計画法の実際」，産業図書，東京（1992）。
- [20] 渡辺展男：「多段階生産・在庫・運搬システム—数理計画法によるモデリング—」，溪水社，広島（1999）。
- [21] 渡辺展男：「パソコン上のシェル環境を用いた生産計画問題の解法」，広島大学経済論叢，第24巻，第2号，pp. 53-70（2000）。
- [22] 渡辺展男：「生産計画問題における Cut-and-Branch 法の数値検証」，広島大学経済論叢，第25巻，第1・2号，pp. 13-29（2001）。
- [23] 渡辺展男：「モデリングシステムを用いた生産計画問題—モデルを記述しながら整数計画問題を早く解く—」，専修経営研究年報，No. 29, pp. 27-55（2005）。
- [24] 渡辺展男，安范俊，平木秀作：「引っ張り型生産指示方式の数理計画的アプローチ」，日本経営工学会誌，Vol. 44, No. 6, pp. 478-486（1994）。
- [25] 渡辺展男，錦織昭峰，平木秀作：「モデル記述言語を用いた生産計画問題の解法」，平成7年度第2回 OR セミナーテキスト，数理計画モデルの応用—構築

- と解法と分析一, pp. 14-28, 日本 OR 学会 (1995)。
- [26] 読売新聞, 2006年5月11日。
- [27] Atamturk, A. and Savelsbergh, M. W. P.: “Integer-Programming Software Systems”, *Annals of Operations Research*, Vol. 140, pp. 67-124 (2005).
- [28] Beasley, J. E. (ed.): *Advances in Linear and Integer Programming*, Oxford University Press, Oxford (1996).
- [29] Bixby, R. E.: “Solving Real-World Linear Programs – A Decade and More of Progress –”, *Operations Research*, Vol. 50, No. 1, pp. 3-15 (2002).
- [30] Carter, M. W. and Price, C. C.: *Operations Research – A Practical Introduction*, CRC Press, Florida (2000).
- [31] Crowder, L. J. and Sposito, V. A.: “Sequential Linear Goal Programming: Implementation via MPSX/370 E”, *Computers and Operations Research*, Vol. 18, No. 3, pp. 291-296 (1991).
- [32] Dash Optimization: Xpress-MP Getting Started Release 2006 (2006);
Xpress-Mosel Language Reference Manual Release 1.6 (2006);
Xpress-Optimizer Reference Manual Release 17 (2006);
<http://www.dashoptimization.com/>
- [33] Ehrgott, M. and Gandibleux, X. (eds.): *Multiple Criteria Optimization – State of the Art Annotated Bibliographic Surveys*, Kluwer Academic Publishers, Massachusetts (2002).
- [34] Ignizio, J. P.: *Linear Programming in Single- & Multiple-objective Systems*, Prentice Hall, New Jersey (1982).
- [35] Ignizio, J. P. and Perlis, J. H.: “Sequential Linear Goal Programming: Implementation via MPSX”, *Computers and Operations Research*, Vol. 6, pp. 141-145 (1979).
- [36] Ignizio, J. P. and Cavalier, T. M.: *Linear Programming*, Prentice Hall, New Jersey (1994).
- [37] Johnson, E. L., Nemhauser, G. L. and Savelsbergh, M. W. P.: “Progress in Linear Programming-Based Algorithms for Integer Programming: A Exposition”, *INFORMS Journal on Computing*, Vol. 12, No. 1, pp. 2-23 (2000).
- [38] Kallrath, J. (ed.): *Modeling Languages in Mathematical Optimization*, Kluwer Academic Publishers, Massachusetts (2004).
- [39] Linderoth, J. T. and Savelsbergh, M. W. P.: “A Computational Study of Search Strategies for Mixed Integer Programming”, *INFORMS Journal on Computing*,

- Vol. 11, No. 2, pp. 173–187 (1999).
- [40] Linear Programming FAQ :
<http://www-unix.mcs.anl.gov/otc/Guide/faq/linear-programming-faq.html>
- [41] Markland, R. E. and Vickery, S. K. : “The Efficient Computer Implementation of a Large-scale Integer Goal Programming Model”, *European Journal of Operational Research*, Vol. 26, pp. 341–354 (1986).
- [42] Martin, R. K. : *Large Scale Linear and Integer Optimization : A Unified Approach*, Kluwer Academic Publishers, Massachusetts (1999).
- [43] Muramatsu, R., Ishii, K. and Takahashi, K. : “Some Ways to Increase Flexibility in Manufacturing Systems”, *International Journal of Production Research*, Vol. 23, No. 4, pp. 691–703 (1985).
- [44] Nemhauser, G. L. and Wolsey, L. A. : *Integer and Combinatorial Optimization*, John Wiley & Sons, New York (1988).
- [45] Sharda, R. and Rampal, G. : “Software Survey : Algebraic Modeling Languages on PCs”, *OR/MS Today*, Vol. 22, No. 3, pp. 58–63 (1995).
- [46] Watanabe, N. : “A PC-based Solution to a Multi-stage Production Ordering System”, *Proc. of the Special International Conference on Production Research (Special ICPR 2000)* provided in a CD-ROM, 6 pages (2000).
- [47] Watanabe, N. and Hiraki, S. : “A Mathematical Programming Model for a Pull Type Ordering System including Lot Production Processes”, *International Journal of Operations & Production Management*, Vol. 15, No. 9, pp. 44–58 (1995).
- [48] Watanabe, N. and Hiraki, S. : “A Modeling Approach to a JIT-based Ordering System”, *Annals of Operations Research*, Vol. 69, pp. 379–403 (1997).
- [49] Watanabe, N. and Hiraki, S. : “An Integer Goal Programming Model of a Multi-stage Production Ordering System”, *Proc. of the 14th International Conference on Production Research*, Vol. 2, pp. 1670–1673 (1997).
- [50] Williams, H. P. : *Model Building in Mathematical Programming* 4th ed., John Wiley & Sons, Chichester, England (1999).
- [51] Wolsey, L. A. : *Integer Programming*, John Wiley & Sons, New York (1998).