

線形計画法自習教材

森 重 雄*

A teaching material of Linear Programming for students to study by themselves

Shigeo MORI

要 旨

線形計画法の自習教材を作成した。この教材は、二段階シンプレックス法のアルゴリズムの理解を支援するもので、Excelのもとで動作する。線形計画の処理ステップごとに説明を表示して、解の推移を学生自身が学習できるようにした。また、グラフも表示できるので、図式と合わせた理解も得られる。

本報告では、この教材の機能、実現方法、効果などについて述べる。

1. はじめに

情報工学の教科の一つであるシステム工学では、最適化手法の一つとして線形計画法を教授している。線形計画法には各種の手法があるが、代表的かつ実用的な二段階シンプレックス法のアルゴリズムを中心に教えている。しかし、この手法は学生が理解するには難度が高く、消化不良で終わる者も多い。授業終了後、情報工学実験で二段階シンプレックス法のプログラミングを学生に課しているが、予定の時間で終了する学生は少ない。その第一の原因は、二段階シンプレックス法のアルゴリズムを十分理解していないためである。また、アルゴリズムを理解したとしても、その応用力をつけるには、各種の問題を解いてみなければならない。しかし、筆算では時間がかかり過ぎるので、限られた講義時間内での取り組みは難しい。

そこで、二段階シンプレックス法のアルゴリズムについて学生が自習し理解できるようになること、異なる問題を与えてその結果の確認と意義を理解できるようになること、および線形計画法の応用力をつけることを目的として、自習用の教材を作成した。

この教材は、解の展開過程を1ステップずつ把握でき、最大化・最小化問題、条件式の設定、目的関数の設定などが自由に行えるほか、グラフを表示させて、図式イメージでも理解できるようになっている。また、座学における補助教材としての利用が期待できる。

2. 二段階シンプレックス法

2. 1 概 要

二段階シンプレックス法は、第一段階と第二段階からなる。第一段階では、初期基底状態をつくり出して実行可能基底解を求め、第二段階では第一段階で得られた実行可能基底解を最大化して最適解を求める。

初期基底状態は人為変数（人工変数）を導入してつくるが、これに伴い、新たな目的関数を導入しなければならない。第一段階では、この新しい目的関数を最大化するように解を求めてゆく。もし、この目的関数の値がゼロにならないければ、元の線形計画問題の解は存在しない。目的関数の値がゼロであれば、第二段階の解へ進める。

第二段階では第一段階の目的関数と人為変数を捨てて、元の目的関数を最大化するように解を求めてゆく。

初期基底状態ができた後の第一段階と、その後の第二段階の最適解を求めるアルゴリズムは、対象の目的関数が異なる他は、基本的に同じである。

2. 2 最適解の求め方

最適解の求め方は、以下ようになる。

- (1) 目的関数式の右辺係数で、負で絶対値最大のものを検索し、その変数列（ピボット列）を特定する。
- (2) 各制約条件式の右辺定数を、(1)で求めた変数列の係数で割り、 θ を求める。
- (3) 最小の θ を求め、その所属行（ピボット行）

を特定する。

- (4) ピボット列とピボット行の交わるころの係数を軸として、すべての式の掃出し計算を行う。

- (5) 上記(1)から(4)までの処理を目的関数のすべての左辺係数が正になるまで繰り返す。

目的関数のすべての左辺係数が正になった時点で、目的関数の右辺定数は最大となり最適解が得られる。最小化の場合は、目的関数の右辺定数が負になっているので、正に戻す必要がある。

3. 教材の機能

今回開発した教材の機能等を以下に示す。

3.1 実行画面

実行画面は図1のようになっている。図のように、Excelのワークシートに、プログラムへ渡すパラメータ用コンボボックスとプログラム実行ボタン、および問題の式を設けている。

	A	B	C	D	E	F
1	二段階シンプレックス法による線形計画法					
2		最大化問題	中間表示学習	問題表の右に最適解表示		
3		小数2桁表示	条件記号	実行		
4		目的関数 $F = 25X_1 + 48X_2$				
5		F	X_1	X_2	条件	B_i
6		1	-25	-48	=	0
7			15	30	≤	45000
8			24	6	≤	24000
9			21	14	≤	28000
10						
11						
12						

図1 実行画面

利用者が、問題の式（図1の白枠内）、変数およびその係数を追加したり削除する場合は、Excelの編集機能を使用すればよい。コンボボックスやコマンドボタンは、行や列の追加削除には影響されない。

以下についてはコンボボックスで指定する。

- (1) 問題の区分（最大化、最小化）
- (2) 表示モード（解の中間結果表示、学習表示）
- (3) 最適解の表示位置
- (4) 小数または分数表示
- (5) 条件記号

実行画面はExcelのワークシートであるから、利用者は必要に応じてこれを複製できる。複製されたワークシートは、画面に表示されていれば、プログラムの処理対象として扱われる。これによ

り、同じ目的関数について制約条件を変えて実行し、その違いを比較してみることができる。

3.2 処理ステップごとのガイダンス表示

以下の各処理において、処理ステップごとにガイダンスを表示して、理解を助けている。ガイダンスは、ワークシート上に表示され、[Enter]キーを押すか、ダイアログボックスの[OK]ボタンをクリックするまで表示される。

ガイダンス表示例					
◆ スラック変数案内表示 ◆					
表示継続=OK 表示中断=SKIP (次のステップまで全案内表示を中断します。) 表示停止=キャンセル (この案内表示のみを以降停止します。) 処理終了=END (処理を終了します。)					
目的関数 $F = 25X_1 + 48X_2$					
	X_1	X_2	条件	B_i	
1	-25	-48	=	0	
	15	30	≤	45000	
	24	6	≤	24000	
	21	14	≤	28000	
【 S_i 】のスラック変数を作成しました。					
第1段階	X_1	X_2	S_i		
-W					
F	-25	-48			
	15	30	1		
	24	6			
	21	14			

図2 ガイダンス表示例

(1) 目的関数の編集

最大化問題または最小化問題によって、目的関数の両辺の係数の符号を設定するが、このとき各係数の符号を設定するたびにガイダンスを表示する。

(2) 第一段階の式の作成

スラック変数の作成、人為変数の作成、第一段階の目的関数式の作成のそれぞれについて、ガイダンスを表示する。

(3) 各段階の最適化処理

以下の処理でガイダンスを表示する。

- ・目的関数の左辺係数について、負で絶対値最大の係数を検索するとき
- ・次に基底に入れる変数を特定したとき
- ・ θ を計算するとき
- ・最小の θ を特定したとき
- ・ピボット行の設定や他の行の掃出しのとき

負で絶対値最大の係数を検索中です。

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	-60	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	
	A_2	24	6		1			1		24000	
	A_3	21	14			1			1	28000	

負で絶対値最大の係数を検出しました。次に基底に入れる変数は $[X_1]$ です。

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	-60	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	
	A_2	24	6		1			1		24000	
	A_3	21	14			1			1	28000	

ピボット行を検出中です $[A_1]$ の式の θ 計算 $\theta = 45000.00 / 15 = 3000.00$

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	-60	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	3000.00
	A_2	24	6		1			1		24000	
	A_3	21	14			1			1	28000	

ピボット行(θ が最小の行)を検出しました。基底から掃き出す変数は $[A_2]$ です。

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	-60	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	3000.00
	A_2	24	6		1			1		24000	1000.00
	A_3	21	14			1			1	28000	1333.33

$[-W]$ の行の掃き出しをします。

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	-60	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	3000.00
	X_1	1.00	0.25	0.00	0.04	0.00	0.00	0.04	0.00	1000.00	1000.00
	A_3	21	14			1			1	28000	1333.33

変数 $[X_1]$ の係数 $[-60.00]$ の掃き出し計算をします。計算式は $[-60.00 - (1.00 \times -60.00) = 0.00]$ です。

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	-60	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	3000.00
	X_1	1.00	0.25	0.00	0.04	0.00	0.00	0.04	0.00	1000.00	1000.00
	A_3	21	14			1			1	28000	1333.33

変数 $[X_1]$ の係数 $[-60.00]$ の掃き出しをしました。

STEP 1	基底変数	X_1	X_2	S_1	S_2	S_3	A_1	A_2	A_3	B_1	θ
	-W	0.00	-50	-1	-1	-1				-97000	
	F	-25	-48							0	
	A_1	15	30	1			1			45000	3000.00
	X_1	1.00	0.25	0.00	0.04	0.00	0.00	0.04	0.00	1000.00	1000.00
	A_3	21	14			1			1	28000	1333.33

図3 各種ガイダンス表示例

図3 各種ガイダンス表示例

3. 3 ガイダンス表示抑止

すでに理解できた処理について、何度もガイダンス表示されることは利用者にとっては非常に煩わしい。そのため、現在表示ガイダンスを以降の処理で表示させない、次の掃出しまで表示させない、あるいは処理を終了させるということを選択できるようにしている。利用者は、これをダイアログボックスで指示すればよい。

ガイダンスを最初から表示したくない場合は、表示コンボボックスで「中間表示学習」以外のメニューを選択すればよい。

3. 4 グラフ作成

最適化問題の変数の数が2の場合のみ、グラフを作成できる。最適解が求まるとグラフを作成するか否かを利用者に問い合わせる。「はい」を選択すると、その問題のグラフが表示される。このグラフは、Excelのグラフ機能を利用して作成される。このグラフには、制約条件の式、各解の目的関数式および最終の目的関数式（最適解）が色違いで描画される。

このグラフにより、線形計画の意義が視覚的によく分かるようになる。

3. 5 最適解の表示

最適解が求まると、その結果を表示する。表示する内容は以下のとおりである。

- (1) 変数の値（変数の解）と基底状態
- (2) 目的関数の値（目的関数の最適解）
- (3) 最終の目的関数式
- (4) 縮小コスト（reduced cost）
- (5) 潜在価格（shadow price）
- (6) 最適解への影響

このうち、縮小コストは最終の目的関数式に現れる、非スラック変数の係数である。また、潜在価格は、最終の目的関数式に現れるスラック変数の係数である。

最適解への影響とは、縮小コストまたは潜在価格が最適解に与える影響を示す。例えば、最大化の問題で、変数 X_i が基底にあるが制約条件の下限または上限に接しているとき、潜在価格が最適解への影響として表示される。このとき、その値が負であれば、変数 X_i を1増加させると目的関数は潜在価格の分だけ減少し、逆に変数 X_i を1減少させると潜在価格の分だけ増加することが読

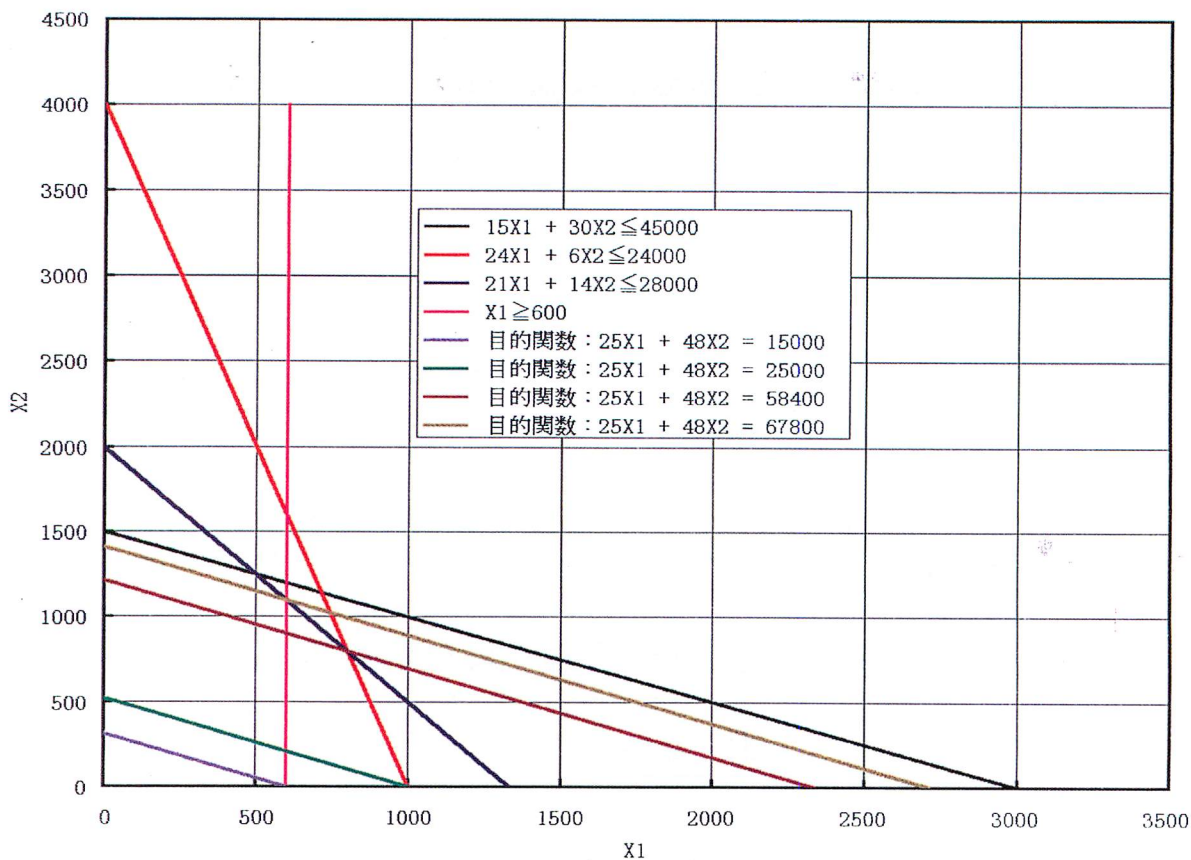


図4 グラフ表示例

み取れる。なお、変数 X_i を増減させるには、制約条件式の右辺定数を変更すればよい。

縮小コストは、基底に入っていない非スラック変数 X_i の、最終目的関数式における係数である。この X_i は基底に入っていないのでゼロである。しかし、もし制約条件式の右辺定数を変えて強制的にこの X_i を基底に入れると、目的関数は最適解への影響として表示された縮小コスト分だけ悪化する。そして今度は潜在価格が出現し、 X_i を増加させると目的関数がどの程度悪化するかが最適解への影響として表示されるようになる。

この縮小コストと潜在価格が目的関数に与える影響を、座学を通して具体的に学生に理解させることは非常に難しい。しかし、本教材を利用して教えると結果を対比して見せられるようになる。

なお最適解は、最初の式の右側か最終タブローの下のいずれかに表示できる。この選択は、コンボボックスで行う。

最適解を表示するときの書式もコンボボックスから指定する。

3. 6 入力支援

制約条件式の条件記号として " $=$ ", " $\leq$ ", " $\geq$ " を指定できるようにしている。これらは全角の記号文字であるため、かな漢字変換機能では表示できない。この入力をプログラムで支援している。入力するときは、対象のセルを選択し、条件記号のコンボボックスから希望する記号を選択すればよい。

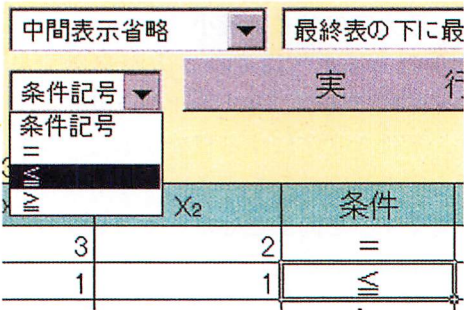


図 5 条件記号の入力支援

H	I	J	K	L	M	N	O
最適解 最適解への影響縮小コスト 潜在価格 $X_1 = 600.00$ (基底) -47 該当なし $S_3 = 3.43$ $X_2 = 1100.00$ (基底) $S_4 = 47.00$ $F = 67800.00$ 最終の目的関数式 $F + 3.43 S_3 + 47.00 S_4 = 67800.00$							

図 6 最適解の表示 1

最適解	最適解への影響	縮小コスト	潜在価格
$X_1 = 0.0000$ (非基底)	-6.9821	$X_1 = 6.9821$	$S_1 = 0.1429$
$X_2 = 10.2679$ (基底)			$S_2 = 0.4554$
$X_3 = 15.6250$ (基底)			$S_3 = 2.0000$
$X_4 = 27.9464$ (基底)			
$F = 375.2679$			
最終の目的関数式			
$F + 6.9821 X_1 + 0.1429 S_1 + 0.4554 S_2 + 2.0000 S_3 = 375.2679$			

図 7 最適解の表示 2

3. 7 数値の書式

各係数の書式として分数表示ができるようにした。システム工学や線形計画法の書籍には、分数による表示が多く、そのほうが理解しやすい場合もある。Excel には書式として分数を表示できるので、それを利用している。なお、最終の目的関数式については、分数による表示は実現できていない。これは、最終の目的関数式を文字列としてセルに格納しているためである。

3. 8 添え字のサイズ縮小

変数などの添え字は小さい方が自然で見やすい。Excel には文字ごとにフォントサイズを設定したり、下付き文字にする機能がある。本教材では、添え字の数字のみを小さいフォントサイズにして、見やすさを向上させた。下付き文字にすると、添え字の下端が変数文字の下端より下がり、実行中に行のサイズが変化して見苦しいので、フォントサイズによる方式にした。

利用者は添え字のある変数名を入力するときには、添え字のサイズ変更をしなくてよい。本教材のプログラムが、自動的にサイズを設定する。

なお、添え字のある変数名がグラフの凡例などに表示される場合、これらの個々の文字列に対してはフォントサイズ設定ができないので、添え字の大きさは変数名の文字の大きさと変わらない。

3. 9 人為変数掃出し省略

第一段階で作成した人為変数は最適化の過程で掃出し処理されるが、この変数は本来ゼロであるから、掃出しを省略しても最適解は得られる。その実験が可能となるよう、掃出しを省略した最適化が行えるようにした。この選択は、コンボボックスで指定する。

3. 10 セルの背景色設定

Excel はセル背景色を設定できる。この機能を利用して、ピボット行、ピボット列、第一段階と第二段階の最適解（目的関数式の右辺定数）の各セルに背景色を設定し、より見やすくした。

また、最適化の過程を示す表のセルとその外側のセルの背景色を変え、表全体を見やすくした。

4. 実現方法

4. 1 開発環境

本ツールは、次の環境で開発した。

- (1) Microsoft Windows95
- (2) Microsoft Excel VBA
(Visual Basic for Applications)
- (3) Microsoft Excel97

4. 2 動作環境

本ツールは、以下の環境で動作する。

- (1) Microsoft Windows95
- (2) Microsoft Windows98
- (3) Microsoft WindowsNT/WS
- (4) Microsoft Excel97
- (5) ExcelVBA 日本語ライブラリ

4. 3 実行シート

実行シートは通常の Excel のシートである。利用者は、問題の目的関数や制約条件をこのシートに設定する。あらかじめ用意してあるテンプレートシートを複写して独自のシートに変更できるので、Excel の知識があれば操作は簡単である。

テンプレートシートには、パラメータとなるコンボボックスが配置されている。このコンボボックスは、シートを複写すると同時に新しいシートにも複写される。

4. 4 問題の式数と変数の数の自動把握

問題の式の数と変数の数は、表の大きさをプログラムで求めることにより把握している。このため、利用者は式や変数の数を指定しなくてよい。

なお問題式の先頭セルのアドレスは、処理の簡略化のため固定にしている。従って、問題式の先頭セルの上または左に、行または列を挿入するとプログラムは正しく動作しない。

4. 5 学習用ガイダンス表示

プログラムはガイダンスをセルに格納表示し、その後の応答操作についてはダイアログボックスに表示して応答を待つようにしている。これらの処理を統括的に行う関数を作成し、ガイダンス表示元からこの関数を呼び出すようにした。呼び出す際に、セルに格納するメッセージ、セルのアドレス、ダイアログボックスに表示するメッセージを引き渡す。掃出し処理では、個々の係数が掃出した後にどう変わるかをガイダンスで表示する。また、同時に書式制御なども行うため、この処理プロシジャはやや複雑になった。（リスト1）

リスト1 掃出し処理のプロシジャ

```

'----- 掃き出し
'ピボット行以外の行を掃き出してピボット変数を基底状態にする。
Sub 掃き出し(ピボット行 As Integer, ピボット列 As Integer, 開始行 As Integer, 終了行 As Integer, 変数開始列 As Integer, _
    変数終了列 As Integer, 右辺定数列 As Integer, ゼロ仮定値 As Double, 数値表示 As Integer)
    Dim 行 As Integer
    Dim 列 As Integer
    Dim セル値 As Double
    Dim ピボット列セル値 As Double
    Dim ピボット行セル値 As Double
    Dim 表示続行 As String

    表示続行 = "続行"
    For 行 = 開始行 To 終了行 Step 1
        If 不一致(行, ピボット行) Then 'ピボット行はスキップする
            セル選択 行, 基底変数列, 行, 0 列
            学習モード対応 " [ " & セル値取得(行, 基底変数列) & " ] の行の掃き出しをします。", "行掃き出しの案内表示", 行掃き出し案内
            ピボット列セル値 = セル値取得(行, ピボット列)
            For 列 = 変数開始列 To 変数終了列 Step 1
                ピボット行セル値 = セル値取得(ピボット行, 列)
                セル値 = セル値取得(行, 列)
                学習モード対応 "変数 [ " & セル値取得(見出し行, 列) & " ] の係数 [ " & Format(セル値, 数値編集形式) & _
                    " ] の掃き出し計算をします。" & _
                    "計算式は [ " & Format(セル値, 数値編集形式) & _
                    " - ( " & Format(ピボット行セル値, 数値編集形式) & " × " & Format(ピボット列セル値, 数値編集形式) & _
                    " ) = " & Format(みなしゼロ取得((セル値 - (ピボット行セル値 * ピボット列セル値)), _
                        ゼロ仮定値), 数値編集形式) & " ] です。", _
                    "係数掃き出しの案内表示", セル掃き出し案内, 行, 列
                セル値設定 みなしゼロ取得((セル値 - (ピボット行セル値 * ピボット列セル値)), ゼロ仮定値), 行, 列
                書式設定 行, 列, ゼロ仮定値, 数値表示
                学習モード対応 "変数 [ " & セル値取得(見出し行, 列) & " ] の係数 [ " & Format(セル値, 数値編集形式) & _
                    " ] の掃き出しをしました。", _
                    "係数掃き出しの案内表示", セル掃き出し案内

            Next
            ピボット行セル値 = セル値取得(ピボット行, 右辺定数列)
            セル値 = セル値取得(行, 右辺定数列)
            学習モード対応 "変数 [ " & セル値取得(見出し行, 列) & " ] の係数 [ " & Format(セル値, 数値編集形式) & _
                " ] の掃き出し計算をします。" & _
                "計算式は [ " & Format(セル値, 数値編集形式) & " - ( " & Format(ピボット行セル値, 数値編集形式) & " × " & _
                Format(ピボット列セル値, 数値編集形式) & _
                " ) = " & Format(みなしゼロ取得((セル値 - (ピボット行セル値 * ピボット列セル値)), ゼロ仮定値), _
                    数値編集形式) & " ] です。", _
                "係数掃き出しの案内表示", セル掃き出し案内, 行, 列
            セル値設定 みなしゼロ取得((セル値 - (ピボット行セル値 * ピボット列セル値)), ゼロ仮定値), 行, 右辺定数列
            書式設定 行, 右辺定数列, ゼロ仮定値, 数値表示
            学習モード対応 "変数 [ " & セル値取得(見出し行, 列) & " ] の係数 [ " & Format(セル値, 数値編集形式) & _
                " ] の掃き出しをしました。", _
                "係数掃き出しの案内表示", セル掃き出し案内

            セル選択 行, 基底変数列, 行, 0 列
            学習モード対応 " [ " & セル値取得(行, 基底変数列) & " ] の行の掃き出しをしました。", "行掃き出し完了の案内表示", _
                行掃き出し案内
            セル値消去 見出し行 - 1, 基底変数列
        End If
    Next
End Sub
```

4. 6 グラフの作成

変数が二つの場合、最適解が求まった後でグラフを表示できる。このグラフは Excel の散布図として描画される。さらに、散布図のうちの折れ線グラフを採用している。通常、グラフの系列データをセルのアドレスで指定するが、本教材のプログラムでは系列データに数値を直接設定してグラフを描画させる。実際には、筆者が開発した ExcelVBA 日本語ライブラリにある、[直線グラフ作成関数]と[直線グラフ系列追加関数]を呼び出して描画させる。(リスト2)

4. 7 添え字のフォントサイズ設定

添え字のフォントサイズは以下の方法で設定する。

- (1) 変数名の添え字を除いた文字列を、記号配列として作成しておく。
- (2) 対象のセルに格納する文字列について、記号配列の変数名を検索し、それらに続く添え字の位置と桁数を把握する。添え字の終了は空白か最後の文字かで判定する。
- (3) 対象のセルに文字列を格納する。
- (4) 対象のセルにつき、各添え字の文字列上の開始位置と桁数を指定して、フォントサイズを設定する。(リスト3)

リスト2 直線グラフ作成・直線グラフ系列追加関数

```
Function 直線グラフ作成(Optional タイトル As Variant, Optional X軸名 As String, Optional Y軸名 As String, _
    Optional X軸最小値 As Variant, Optional X軸最大値 As Variant, _
    Optional Y軸最小値 As Variant, Optional Y軸最大値 As Variant, _
    Optional X軸交点 As Variant, Optional Y軸交点 As Variant, Optional X軸反転 As Variant, _
    Optional Y軸反転 As Variant, Optional Y軸対数目盛 As Variant, _
    Optional X軸目盛間隔 As Variant, Optional Y軸目盛間隔 As Variant, _
    Optional X値1 As Variant, Optional Y値1 As Variant, Optional 直線名1 As Variant, _
    Optional 線色1 As Variant, Optional 線種1 As Variant, Optional X値2 As Variant, _
    Optional Y値2 As Variant, Optional 直線名2 As Variant, Optional 線色2 As Variant, _
    Optional 線種2 As Variant, Optional X値3 As Variant, Optional Y値3 As Variant, _
    Optional 直線名3 As Variant, Optional 線色3 As Variant, Optional 線種3 As Variant) As String
    Dim シート名 As String
    シート名 = ActiveSheet.Name
    Charts.Add
    With ActiveChart
        .ChartType = xlXYScatterLinesNoMarkers '// 散布図の折れ線グラフ
        If IsMissing(直線名1) Then
            ActiveChart.HasLegend = False
        Else
            End If
            直線グラフ系列追加 X値1, Y値1, 直線名1, 線色1, 線種1, , 1
            If Not IsMissing(X値2) Then
                直線グラフ系列追加 X値2, Y値2, 直線名2, 線色2, 線種2, , 2
            End If
            If Not IsMissing(X値3) Then
                直線グラフ系列追加 X値3, Y値3, 直線名3, 線色3, 線種3, , 3
            End If
            グラフタイトル設定 タイトル
            プロット領域設定
            グラフX軸設定 X軸名, "目盛線", "補助目盛線なし", X軸最小値, X軸最大値, X軸目盛間隔, X軸交点, X軸反転
            グラフY軸設定 Y軸名, "目盛線", "補助目盛線なし", Y軸最小値, Y軸最大値, Y軸目盛間隔, Y軸交点, Y軸反転, , _
                Y軸対数目盛
            グラフの場所選択 "シート埋込", シート名 '// グラフをアクティブなシート上に確保
            直線グラフ作成 = ActiveChart.Name
        End With
    End Function

Function 直線グラフ系列追加(X値 As Variant, Y値 As Variant, Optional 直線名 As Variant, _
    Optional 線色 As Variant, Optional 線種 As Variant, Optional 線幅 As Variant, _
    Optional 指定系列 As Variant) As Variant
    Dim 系列 As Long
    Dim 色数値 As Variant
    色数値 = 色番号数値取得(Variant 引数による値設定(線色, "黒"))
    With ActiveChart
        .SeriesCollection.NewSeries
        If IsMissing(指定系列) Then
            系列 = .SeriesCollection.Count
        Else
            系列 = 指定系列
            If .SeriesCollection.Count > 系列 Then
                .SeriesCollection(.SeriesCollection.Count).Delete
            End If
        End If
        直線グラフ系列追加 = 系列
        With .SeriesCollection(系列)
            If IsArray(X値) Then
                .XValues = "=" & 数値配列文字列化(X値) & "'" '// 変数値
            Else
                .XValues = X値 '// セルデータ
            End If
            If IsArray(Y値) Then
                .Values = "=" & 数値配列文字列化(Y値) & "'" '// 変数値
            Else
                .Values = Y値 '// セルデータ
            End If
            If Not IsMissing(直線名) Then
                .Name = 直線名 '// 凡例
            End If
            With .Border
                If TypeName(色数値) = 短整数型 Then
                    .ColorIndex = 色数値
                Else
                    .Color = 色数値 '// 直線の色
                End If
                .Weight = Variant 引数による値設定(線幅, 中太線) '// 直線の幅
                .LineStyle = Variant 引数による値設定(線種, 実線) '// 直線の種類
            End With
        End With
    End With
End Function
```

リスト3 添え字の設定処理

```
For 件数 = 1 To 下付き文字列数 Step 1
セル.Characters(Start:=下付き文字列位置(件数), Length:=下付き文字列長(件数)).Font.Size = 文字列フォントサイズ
Next
```

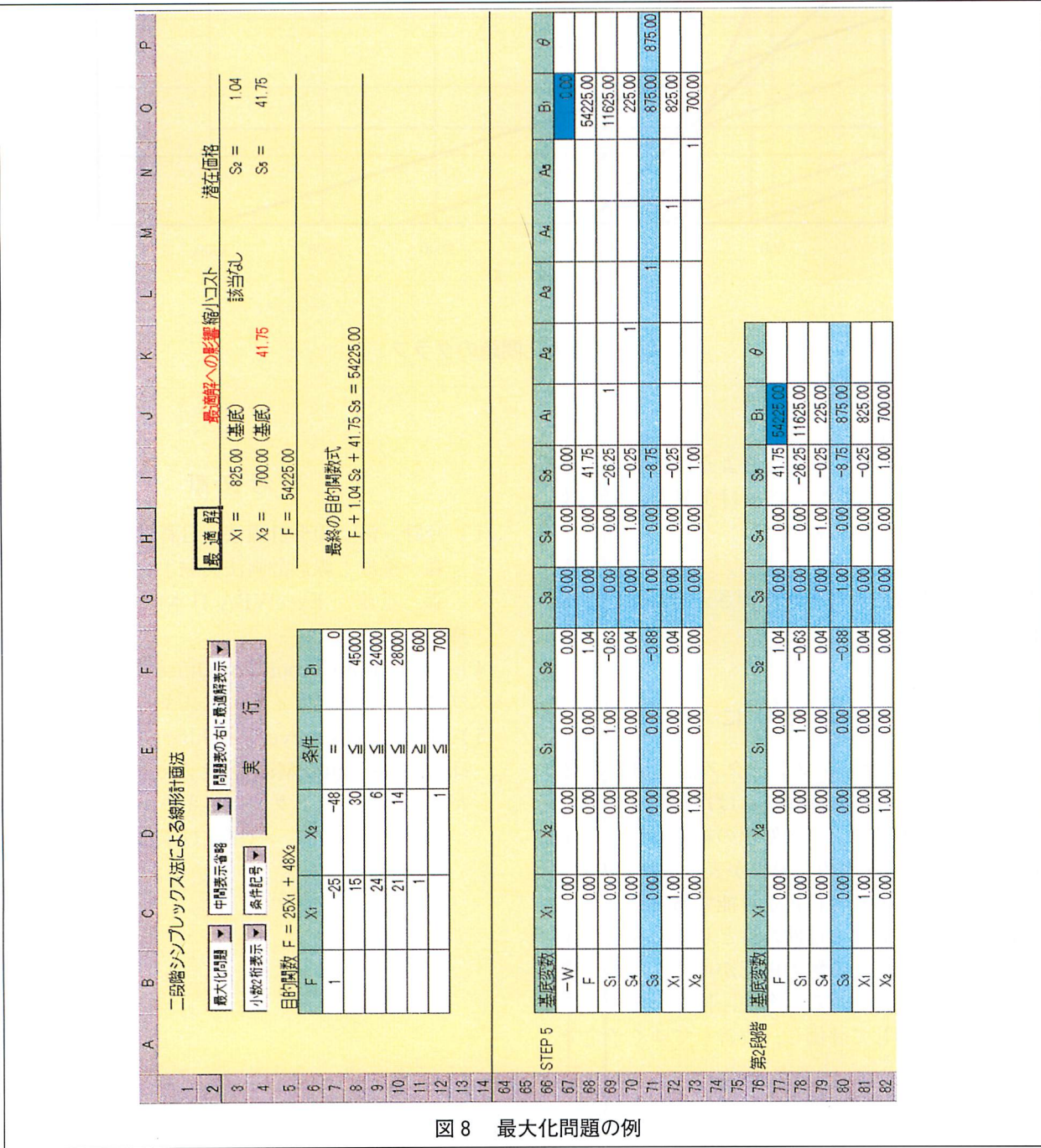
4. 8 分数表示

セルのデータを分数で表示する場合は、そのセルの書式を分数にすればよい。セルの書式設定はNumberFormatLocal プロパティで行う。

```
Cells (行, 列).NumberFormatLocal = " #
???/???"
```

5. 実行例

最大化問題の実行例を図に示す。この問題は変数が2、制約条件式が5の問題である。X₁, X₂とも基底にあるが、X₂が制約条件値となっているため最適解への影響が赤字で表示されている。



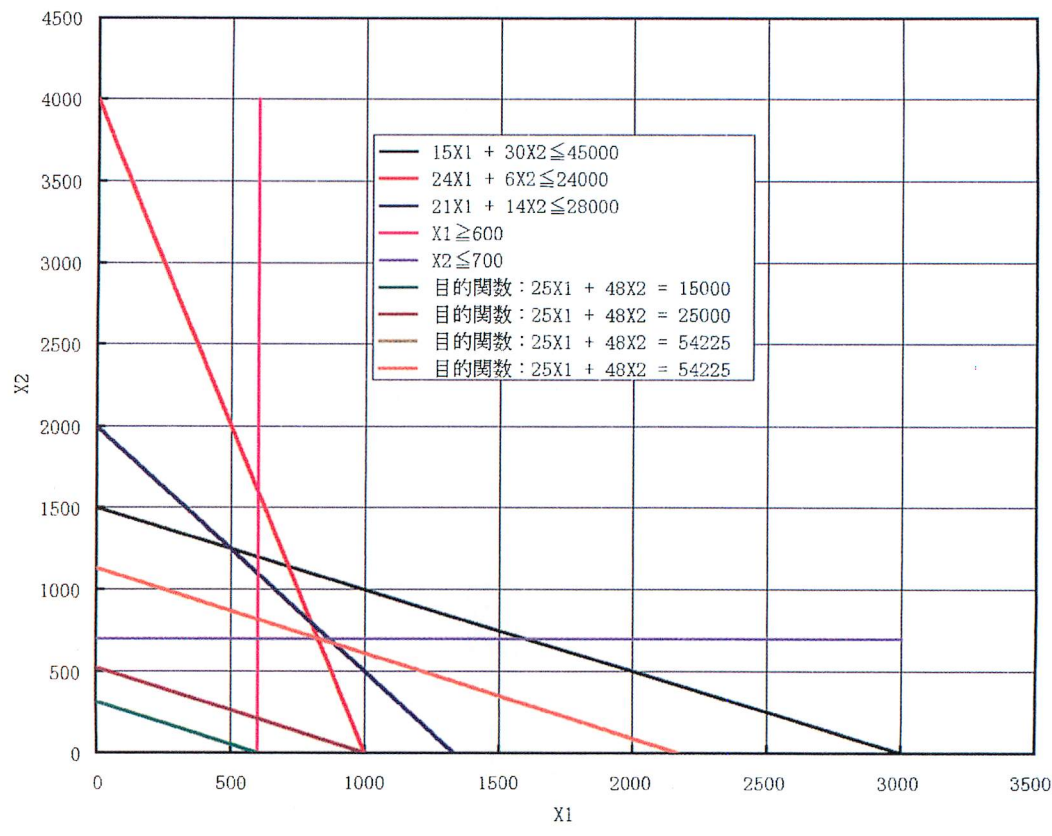


図9 最大化問題のグラフ

この場合、 X_2 が1大きくなるように制約条件を変えると、目的関数は41.75だけ上昇し、解が改善されることが読み取れる。

図はこの問題のグラフである。 $X_2 \leq 700$ という制約条件によって、最適解が押さえられている様子がよく分かる。

6. おわりに

作成した二段階シンプレックス法の自習教材は、若干のExcelの知識があればだれでも操作できる。詳細なガイダンスが処理の経過を追って表示され、さらに解を求めた過程が表示されるので、学生の理解力に応じた自習が可能である。

また、座学における補助教材としても利用できる。特にグラフによる解の確認が可能となり、視覚的に理解を深められる。今後、授業などでこの教材を学生に供して評価し、改善してゆく予定である。

参考文献

- 1) 今野 浩：線形計画法，日科技連，1991
- 2) 森 重雄：線形計画法実験手引書，1995
- 3) 森 重雄：ExcelVBA 日本語ライブラリ仕様書，1999
- 4) Microsoft Corporation: Office97 Visual Basic プログラマーズガイド，アスキー出版局，1997
- 5) Eric Wells・Steve Harshbarger・Micro Modeling Associates: Microsoft Excel97 デベロッパーズハンドブック，アスキー出版局，1997

(平成11年11月30日受理)