線形計画問題の強多項式解法について

藤重　悟

1. はじめに

線形計画モデルとそれに対する最適化技法である単体法が G. B. Dantzig によって1947年に示されて以来，一般的な線形計画問題に対する実用的解法として単体法に勝るものがない状況が続いてきた．一方，1970年代に入って，いわゆる計算複雑度（computational complexity）の理論研究が活発に行なわれ，計算複雑度の観点から，多項式時間で解ける“やさしい”問題に対して多項式時間で解けないであろうと思われる“むずかしい”問題（NP一完全な問題）の存在が示され，ORなどで定式化されている膨大な数の問題に関してそれらが“むずかしい”問題群に属するか否かの分類が行なわれた（現在でもその活動が続けられている）．

そんな中にあって，実際的に効率がよい単体法が理論的には多項式時間の解法ではないことが明らかにされ[6]，また，線形計画問題が多項式時間で解ける“やさしい”問題であることが，1979年に Khachiyan [5]によって多項式時間の解法（楕円体法）が示されることにより判明した．この楕円体法は実用的にはとても単体法に太刀打ちできるものではなくて「多項式時間の解法＝効率がよい解法」という考えに対する反省を促すことになった．そして1984年に Karmarkar [4] によって実用的にも効率がよい（単体法を凌ぐと主張される）多項式時間の解法が提案され，その解法は理論と実用の両面から注目されて，その改良などを含む周辺の研究が現在活発に行なわれている．

ところが線形計画問題に対する Khachiyanや Karmarkar の解法の手間は入力データのサイズ（＝問題を記述するデータを２進表現したときの総ビット数）の多項式の時間でおさえられるというものであって，最適解を得るまでに実行する四則演算の回数が入力データのサイズに依存している．問題を記述している数値をごく微小量変化させると入力データのサイズは大きく増加する．しかし，最適解を得るまでに実行する四則演算の回数が入力データをごく微小量変化させることによって大きく増加するとは（きわめて直観的には）考えにくいであろう．したがって，線形計画問題が係数行列の大きさ（行数m，列数n）の多項式回の四則演算で解くことができるのではないかと予想することもできよう（このような解法を強多項式解法という；正確には次節で定義する）．この方向の成果として，最近得られた Tardos の解法[8]が注目される．本小文で，その解法の概略を紹介しよう．

2. 強多項式解法とは

ある問題Pを表現するデータ中の数値の個数をそのデータの**次元**という．（たとえば，$n \times n$ 係数

ふじしげ　さとる　筑波大学　社会工学系
〒305 茨城県新治郡桜村天王台 1―1―1

行列をもつ連立1次方程式を解くという問題の場合，この問題の入力データの次元は $n(n+1)$ である.）問題Pの解法が次の (1)，(2)を満たすとき，その解法は**強多項式時間の解法**あるいは**強多項式解法**であるといわれる.

(1)問題Pを解くまでに実行される四則演算の回数が入力データの次元の多項式でおさえられる

(2)入力データが有理数であるとき，問題Pを解く過程で現われる任意の数値のサイズが入力データの次元とサイズの多項式でおさえられる.

（ここで，有理数 p/q (p, q：整数) のサイズは $\lceil \log_2(p+1)\rceil+\lceil \log_2(q+1)\rceil$ であり，入力データのサイズは入力データ中の数値のサイズの総和である．ただし，$\lceil x\rceil$ はxより小さくない最小整数を表わす.）

この定義により強多項式時間の解法は多項式時間の解法であるがこの逆は必ずしも成立たない.

$n\times n$に係数行列をもつ連立1次方程式がGaussの消去法によって$O(n^3)$ 回の四則演算によって解かれることはよく知られているが，このとき上記の条件 (2) が満足されることを証明することはそんなに簡単なことではない．乗除算のくりかえしによって得られる数値のサイズが入力データのサイズの多項式でおさえられなくなることが容易に起こり得るからである．たとえば，$x=2$ として $x:=x^2$ を k 回くりかえすと $x=2^{2^k}$ になることに注意されたい．Gauss の消去法が強多項式解法であることは Edmonds[1] によって示されている．線形計算に関係する算法で，四則演算の回数が入力データの次元の多項式でおさえられることが古くから知られていても，条件 (2) の確認が最近になってできたものが少なくない.

その他に，効率よく解かれている組合せ最適化問題のほとんどすべてに対して強多項式解法が存在する．そのような問題の中で，最近強多項式解法が見つかったものとして，最小費用流問題がある．1984年に Tardos [7] によって最小費用流問題の強多項式解法が示され，その後 Fujishige [2]，Galil-Tardos [3] の改良を経て，現在最小費用流問題は $O(n^2(m+n\log n)\log n)$ 回の四則演算で解けるようになっている（ただし，m：枝数，n：点数).

3. 線形計画問題に対するTardosの解法

Tardos [8] は最小費用流問題に対する強多項式解法[7]のアイデアを線形計画問題に拡張して，計算複雑度の観点から興味ある結果を導いている．その結果の概略を示そう.

次のような線形計画問題Pについて考える.

(3)　目的関数：$\boldsymbol{cx}\to$最大

(4)　制約条件：$A\boldsymbol{x}=\boldsymbol{b}$，$\boldsymbol{x}\geqq \boldsymbol{o}$.

ここで，$A=(a_{ij})$は$m\times n$ 行列，$\boldsymbol{b}=(b_i)$は m 次元(実)列ベクトル，$\boldsymbol{c}=(c_j)$は n 次元(実)行ベクトル，$\boldsymbol{x}=(x_j)$ は変数 x_j $(j=1,\cdots,n)$ を要素にもつn次元列ベクトルであって，Aの各要素 a_{ij} は整数値であるとする．そして，

(5)　$\triangle(A)\geqq \max\{|\det B|\,|B{:}A$ の正方部分行列$\}$ ($\det B$: Bの行列式) を満たす $\triangle(A)$ が与えられているとする．どんな A に対しても $\triangle(A)=m^m\cdot\alpha^m$ (α: A の要素の絶対値の最大値) は不等式(5)を満たすので，行列Aの構造に関する特別な情報がなければ $\triangle(A)$ としてこの値を採用する．$\triangle(A)$のサイズがAの次元とサイズの多項式でおさえられることに注意する.

補題1：　任意のn次元(実)行ベクトル $\boldsymbol{y}$に対して，$\bar{\boldsymbol{c}}=\boldsymbol{c}+\boldsymbol{y}A$ とする．このとき，目的関数の係数ベクトル $\boldsymbol{c}$ を $\bar{\boldsymbol{c}}$ に変えても線形計画問題 P の最適解の全体は不変である.

(証明) 問題 P の実行可能解$\boldsymbol{x}$に対して，$\bar{\boldsymbol{c}}\boldsymbol{x}=\boldsymbol{cx}+\boldsymbol{y}A\boldsymbol{x}=\boldsymbol{cx}+\boldsymbol{yb}$ であるから. ▮

Tardos の解法を構成するうえで次の補題は重要である．以下では，$N=\{1,\cdots,n\}$ とする.

補題2：$\boldsymbol{x}^*$ を問題 P の任意の実行可能解，ε を任意の正数，$\boldsymbol{y}$ を次の条件を満足するn次元行ベクトルとする.

(6)　$\boldsymbol{y}A\geqq \boldsymbol{c}-\varepsilon\cdot\boldsymbol{1}$, (7)　$\forall j \in N: (\boldsymbol{ya}_{\cdot j} > c_j \Rightarrow x_j^* = 0)$.

ここで，$\mathbf{1}=(1,1,\cdots,1)$ (n 次元行ベクトル)，$\boldsymbol{a}_{\cdot j}$ は A の第 j 列ベクトルである．このとき，問題 P の任意の最適解 $\hat{\boldsymbol{x}}$ に対して，

(8)　$\forall j \in N: (\boldsymbol{ya}_{\cdot j} \geqq c_j + \varepsilon n \triangle(A) \Rightarrow \hat{x}_j = 0)$

が成り立つ．

(証明) いま，ある $j_0 \in N$ に対して，

(9)　$\boldsymbol{ya}_{\cdot j_0} \geqq c_{j_0} + \varepsilon n \triangle(A)$，$\hat{x}_{j_0} > 0$

が成り立つと仮定して矛盾を導こう．$\hat{\boldsymbol{x}} - \boldsymbol{x}^*$ は線形等式・不等式系

(10)　$A\boldsymbol{z} = \mathbf{0}$，　$z_{j0} > 0$，

　　$z_j \geqq 0$ ($x_j^* = 0$で ある各 $j \in N$に対して)，

　　$\boldsymbol{cz} \geqq 0$

の解である．したがって，(10)の解 $\boldsymbol{z} = \bar{\boldsymbol{z}}$ で，次の(11)～(13)を満たすものが存在する．

(11)　$\bar{z}_{j0} = 1$.

(12)　$J = \{j \mid j \in N, j \neq j_0,\ \bar{z}_j \neq 0\}$

とおいて，

(13)　$|\bar{z}_j| \leqq \triangle(A)$　$(j \in J)$

が成り立つ (A が整数行列であることと Cramer の公式による)．

(10)より，$\bar{z}_j < 0$ ならば$x_j^* > 0$ であり，さらに，(7)より $\boldsymbol{ya}_{\cdot j} \leqq c_j$ であることに注意して，(6)，(9)，(10)，(13)より，

(14)
$$
\begin{aligned}
0 &\leqq \boldsymbol{c}\bar{\boldsymbol{z}} = (\boldsymbol{c} - \boldsymbol{y}A)\bar{\boldsymbol{z}} \\
&\leqq (c_{j0} - \boldsymbol{ya}_{\cdot j0})\bar{z}_{j0} + \sum\{\varepsilon \bar{z}_j \mid j \in J, \bar{z}_j > 0\} \\
&\leqq -\varepsilon n \triangle(A) + \varepsilon(n-1)\triangle(A) \\
&< 0
\end{aligned}
$$

となって，矛盾である．▮

補題2にもとづいて，次のような解法を得る．ここで，ベクトル $\boldsymbol{c}$ に対して $\|\boldsymbol{c}\|_\infty = \max\{|c_j| \mid j \in N\}$ であり，実数xに対して $\lfloor x \rfloor$ は x を超えない最大整数を表わす．

基本算法

入力：(3)，(4)の線形計画問題Pを表現するデータ $A, \boldsymbol{b}, \boldsymbol{c}$ (ただし，問題Pは実行可能解をもつとする；算法中の K は N の部分集合)．

出力：問題 P の最適解 $\boldsymbol{x}^*$ または目的関数が上に有界でないことの判定．

ステップ0：$K = \phi$ とする．

ステップ1：連立1次方程式　$A\boldsymbol{x} = \boldsymbol{b}, x_j = 0$ $(j \in K)$をまとめて$\bar{A}\boldsymbol{x} = \bar{\boldsymbol{b}}$ と表わす．$\bar{A}$の行ベクトルの張る空間へのベクトル$\boldsymbol{c}$の正射影を$\boldsymbol{c}_0$として，$\boldsymbol{c}' = \boldsymbol{c} - \boldsymbol{c}_0$ とおく．

もし $\boldsymbol{c}' = \mathbf{0}$ ならば，$F_K = \{\boldsymbol{x} \mid \bar{A}\boldsymbol{x} = \bar{\boldsymbol{b}}, \boldsymbol{x} \geqq \mathbf{0}\}$ 中の元 $\boldsymbol{x}^*$ を見いだし停止する．

ステップ2：各$j \in N$ に対して，

$\bar{c}_j = \lfloor c'_j n^2 \triangle(A) / \|\boldsymbol{c}'\|_\infty \rfloor$ とおく．

ステップ3：問題 $\bar{P}: \max\{\bar{\boldsymbol{c}}\boldsymbol{x} \mid \bar{A}\boldsymbol{x} = \bar{\boldsymbol{b}}, \boldsymbol{x} \geqq \mathbf{0}\}$ の双対問題 $\bar{P}^*: \min\{\boldsymbol{y}\bar{\boldsymbol{b}} \mid \boldsymbol{y}\bar{A} \geqq \bar{\boldsymbol{c}}\}$ の最適解 $\bar{\boldsymbol{y}}$ を求める．

(ア) $\bar{P}^*$ が実行可能でなければ，停止する(このとき，問題Pの目的関数の値は上に有界ではない)．

(イ) $\bar{P}^*$ が実行可能であれば，

$K_0 = \{j \mid \bar{\boldsymbol{y}}\bar{\boldsymbol{a}}_{\cdot j} - (n^2\triangle(A)/\|\boldsymbol{c}'\|_\infty)c'_j \geqq n\triangle(A)\}$

とおく(ただし，$\boldsymbol{a}_{\cdot j}$ は A の第 j 列)．

$K := K \cup K_0$ として，ステップ1へもどる．

(算法終了)

上記の「基本算法」において次のことに注意する．

(15)ステップ1の正射影 $\boldsymbol{c}_0$ は強多項式時間で求められる．$\bar{A}$ の階数がその行数に等しければ，$\boldsymbol{c}_0 = \boldsymbol{c}\bar{A}^T(\bar{A}\bar{A}^T)^{-1}\bar{A}$ で与えられる．

(16)補題1，2($\varepsilon=1$)より，ステップ3の(イ)のようにKを変更しても最適解の全体は不変である．

(17)ステップ3の(ア)の妥当性は，任意の実行可能解 $\boldsymbol{x}$ に対して $\boldsymbol{cx} = \boldsymbol{c}'\boldsymbol{x} \geqq (\|\boldsymbol{c}'\|_\infty / n^2\triangle(A))\bar{\boldsymbol{c}}\boldsymbol{x}$ が成り立つことによる．

次に，ステップ3の(イ)でKを変更したときKの要素数$|K|$が必ず増加することを示そう．

補題3：「基本算法」のステップ3の(イ)でKを変更したとき必ず$|K|$が増加する．

(証明)ステップ3の(イ)の実行時 (Kの変更前)に得られている $\boldsymbol{c}', \bar{\boldsymbol{c}}, \bar{\boldsymbol{y}}$ を用いて

(18)
$$
c_j'' = \begin{cases} 0 & (j \in K \text{ のとき}) \\ (n^2\triangle(A)/\|\boldsymbol{c}'\|_\infty)c_j' - \bar{\boldsymbol{y}}\bar{\boldsymbol{a}}_{\cdot j} & \end{cases}
$$

$(j \in N-K$ のとき$)$

によって，$\boldsymbol{c}''$ を定義する，補題1より，このように目的関数の係数ベクトルを変えても最適解の全体は不変である．ステップ1の $\boldsymbol{c}'$ の決め方と(18)より，

(19) $\|\boldsymbol{c}''\|_2 \geqq (n^2\triangle(A)/\|\boldsymbol{c}'\|_\infty)\|\boldsymbol{c}'\|_2$

(ただし，$\|\boldsymbol{c}\|_2=(\sum_{j\in N} c_j{}^2)^{1/2}$)が成り立つので，

(20) $\|\boldsymbol{c}''\|_\infty \geqq \frac{1}{n}\|\boldsymbol{c}''\|_2 \geqq (n\triangle(A)/\|\boldsymbol{c}'\|_\infty)\|\boldsymbol{c}'\|_2 \geqq n \times\triangle(A)$ を得る．したがって，少なくとも1つの $j\in N-K$ に対して

(21) $|(n^2\triangle|(A)/\|\boldsymbol{c}'\|_\infty)c'_j-\bar{\boldsymbol{y}}\bar{\boldsymbol{a}}_{\cdot j}|\geqq n\triangle(A)$.

ここで，$\bar{\boldsymbol{y}}$ の決め方から $\lfloor(n^2\triangle(A)/\|\boldsymbol{c}'\|_\infty)c'_j\rfloor \leqq \bar{\boldsymbol{y}}\bar{\boldsymbol{a}}_{\cdot j}$ であるので，

(22) $\bar{\boldsymbol{y}}\bar{\boldsymbol{a}}_{\cdot j}-(n^2\triangle(A)/\|\boldsymbol{c}'\|_\infty)c_j' \geqq n\triangle(A)$

が成り立つ．▮

以上から，「基本算法」はステップ1～3を高々 n 回くりかえして(3)，(4)の線形計画問題 P を解く．右辺ベクトル $\boldsymbol{b}$ が整数ベクトルであるとき，ステップ1で $\boldsymbol{F}_K$ 中の元 $\boldsymbol{x}^*$ を見いだすためと，ステップ3で双対問題 $\bar{P}^*$ を解くために，多項式算法[4]，[5] を使うことにすると，「基本算法」は高々 $\lfloor\log_2\max|a_{ij}|\rfloor$，$\lfloor\log_2\max|b_i|\rfloor$，$m$，$n$ の多項式回の四則演算を実行して問題 P を解く．ここで，四則演算の回数が目的関数の係数ベクトルのサイズに依存しないことに注意する．

さらに右辺ベクトル $\boldsymbol{b}$ のサイズに依存しない解法を得るために，まず，制約式の係数行列のサイズだけの多項式でおさえられる回数の四則演算で実行可能解を見いだす算法を考えよう．

線形不等式系

(23) $A\boldsymbol{x}\leqq\boldsymbol{b}$

を満たす解 $\boldsymbol{x}$ を見いだすために，新たな目的関数の係数ベクトル $\tilde{\boldsymbol{c}}$ を

(24) $\tilde{\boldsymbol{c}}=\sum_{i=1}^{m}(\triangle(A)+1)^i\boldsymbol{a}_{i\cdot}$

($\boldsymbol{a}_{i\cdot}$ は A の第 i 行) のように定義する．そして，上述の「基本算法」を適用して，線形計画問題 $\tilde{P}$：$\max\{\tilde{\boldsymbol{c}}\boldsymbol{x}\|A\boldsymbol{x}\leqq\boldsymbol{b}\}$ の双対問題

(25) $\tilde{P}^*$：$\min\{\boldsymbol{y}\boldsymbol{b}|\boldsymbol{y}A=\tilde{\boldsymbol{c}},\boldsymbol{y}\geqq\boldsymbol{0}\}$

を解く．ここで，$\tilde{\boldsymbol{c}}$ の定義(24)より，双対問題 $\tilde{P}^*$ は実行可能解をもつ．また，$\tilde{P}^*$ の目的関数の値が下に有界でなければ，(23)は解をもたない．$\tilde{\boldsymbol{c}}$ の定義(24)より，問題 $\tilde{P}^*$ は「基本算法」を用いて $\boldsymbol{b}$ のサイズとは無関係に高々 $\lfloor\log_2\max|a_{ij}|\rfloor$，$m$，$n$ の多項式回の四則演算で解くことができる．双対問題 $\tilde{P}^*$ の解 $\tilde{\boldsymbol{y}}$ が得られたとき，$\tilde{I}=\{i|i\in\{1,\cdots,m\},\tilde{y}_i>0\}$ とおいて，

(26) $\tilde{\boldsymbol{F}}=\{\boldsymbol{x}|\forall i\in\tilde{I}:\boldsymbol{a}_{i\cdot}\boldsymbol{x}=b_i\}$

が(23)の解多面体 $\tilde{\boldsymbol{P}}=\{\boldsymbol{x}|A\boldsymbol{x}\leqq\boldsymbol{b}\}$ の (非空の) 極小面であることを示すことができる (証明は省略するが，このことが成り立つように $\tilde{\boldsymbol{c}}$ が(24)で定義されている)．したがって単に連立1次方程式

(27) $\forall i\in\tilde{I}:\boldsymbol{a}_{i\cdot}\boldsymbol{x}=b_i$

を解くことによって(23)の解 $\boldsymbol{x}\in\tilde{\boldsymbol{P}}$ を得ることができる．このようにして，線形不等式系(23)の解を見いだす問題を高々 $\lfloor\log_2\max|a_{ij}|\rfloor$，$m$，$n$ の多項式回の四則演算で解くことができる．また，この算法が条件(2)を満たすことも容易にわかる．この算法を「算法 F」と名づけておこう．

任意の線形等式・不等式系は線形不等式系(23)の形で表現することができ，このときのデータのサイズは高々定数倍になるだけであるから，「算法 F」は任意の線形等式・不等式系を高々その係数行列のサイズの多項式回の四則演算で解くことに注意する．

さて，「基本算法」のステップ1で $\boldsymbol{F}_K$ 中の元を見いだすためにこの「算法 F」を用い，さらにステップ3で双対問題 $\bar{P}^*$：$\min\{\boldsymbol{y}\bar{\boldsymbol{b}}|\boldsymbol{y}\bar{A}\geqq\bar{\boldsymbol{c}}\}$ の実行可能性を判定するためにも「算法 F」を用いることができる．双対問題 $\bar{P}^*$ が実行可能であるとき，$\bar{P}^*$ を解くために元の「基本算法」(をコピーしたもの) をサブルーチンとして使うと，$\bar{P}^*$ が高々 $\lfloor\log_2\max|a_{ij}|\rfloor$，$m$，$n$ の多項式回の四則演算で解ける．

以上のように「基本算法」を変更することによって，「基本算法」は条件(2)を満たして高々

$\lfloor \log_2 \max|a_{ij}| \rfloor$, m, n の多項式回の四則演算で線形計画問題Pを解く．

4. Tardos の解法の意味すること

Tardos の解法によって条件 (2) を満たし，目的関数の係数と右辺のデータのサイズによらなくて制約条件の係数行列のサイズだけの多項式回の四則演算で線形計画問題が解けることが明らかにされた．したがって制約条件の係数行列のサイズが入力データの次元の多項式でおさえられれば，その問題は強多項式時間で解けることになる．このことは線形計画問題の“むずかしさ”が係数行列の構造だけにあることを明らかにしている．

最小費用流問題に組合せ的制約が付加された問題や多種流問題などでは制約条件の係数行列Aの各要素 a_{ij} の絶対値が問題のサイズによらない一定値でおさえられるので，Tardos の結果から強多項式時間で解くことが可能であることがわかる．他にも組合せ最適化問題の中に Tardos の結果から強多項式解法の存在が判明するものが数多く見いだされるものと思われる．

計算複雑度の研究のある部分では，対象とする問題が NP-完全な問題であることが証明されて，その問題に対する効率のよい厳密解法を見いだすことを諦めるというある意味で“否定的”効用があったのに対して，Tardos の結果はある種の問題の強多項式解法の存在を示し，それに対して効率のよい強多項式解法の研究を促進するという“肯定的”効用があることは重要である．

しかし，あくまでもここで紹介した Tardos の解法は計算複雑度の理論発展上の一ステップであって，この解法をそのまま一般の線形計画問題に適用しても，とても実用的解法とはなり得ないであろう．Tardos の結果は，線形計画問題の計算複雑度や算法開発の研究における 1 つの道標として理解されるべきである．理論的には，一般の線形計画問題に対する強多項式解法が存在するか否かを決定することが今後に残された大きな未解決問題である．

参 考 文 献

[1] Edmonds, J.: Systems of distinct representatives and linear algebra. *Journal of Research of the National Bureau of Standards*, Vol. 71B(1967), pp. 241-245

[2] Fujishige, S: A capacity-rounding algorithm for the minimum-cost circulation problem—a dual framework of the Tardos algorithm. *Mathematical Programming*, Vol. 35 (1986), pp. 298-308

[3] Galil, Z., and Tardos, E.: An $0(n^2 \log_n (m + n\log_n))$ minimum cost flow algorithm. MSRI 04518-86, Mathematical Sciences Research Institute, Berkeley, California (March 1986) (to appear in Journal of ACM)

[4] Karmarkar, N.: A new polynomial-time Algorithm for linear programming. Combinatorica, Vol. 4 (1984), pp. 374-395

[5] Khachiyan, L. G.: A polynomial algorithm in linear programming. *Soviet Mathematics Doklady*, Vol. 20(1979), pp. 191-194. (また，Polynomial algorithms in linear programming. *U. S. S. R. Computational Mathematics and Mathematical Physics*, Vol. 20(1980), pp. 53-72, も見よ)

(6) Klee, V., and Minty, G. J.: How good is the simplex algorithm. In: Inequalities—III (O. Shisha, ed., Academic Press, 1972), pp. 159-175.

[7] Tardos, É.: A strongly polynomial minimum cost circulation algorithm. *Combinatorica*, Vol. 5 (1985) pp. 247-255.

[8] Tardos, É.: A strongly polynomial algorithm to solve combinatorial linear programs. Report No. 84360-OR, Institut für Ökonometrie und Operations Research, Universität Bonn (January 1985) (to appear in Operations Research).