カーマーカー法の観客席から

前田　英次郎

本当はプレーヤーでありたいのだが，このような題しかつけられないのが何とも残念である．

1988年8月29日から9月2日まで行なわれた国際数理計画シンポジウムでは，内点法に関する多数の発表が行なわれたが，とりわけ2日目火曜日はKarmarkar dayの感があった．10時10分～12時10分にはKarmarkarの研究グループによる4つの発表があり，16時～18時にはBell研究所で開発した内点法の商用コードKORBXに関する4つの話があった．私の英語能力では半分も聞き取れていないので，以下の話は誤解も記憶違いも少なくないことを覚悟の上で書かせていただいた．

午前のセッションの2番目に内点法の計算実験の話があった．連立方程式を共役傾斜法で解くやり方で，相当大型の問題を扱っており，MINOS (Stanford 大学が提供している単体法のコードでごく安く入手できる) と比較して数十倍から数百倍のスピードで解けたという．KORBXの方はアライアント社製の8個のプロセサをもつベクトル機構つきミニスーパー計算機の上で動くもので，手法は主としてaffine scaling法を使っていて，連立方程式はCholesky分解で解く．やはりMINOSの30倍とか50倍といった計算結果を発表していた．([1]によれば，変数30万個，制約20万個の問題を1分～10時間で解くと言っている．) ここまでの話で，内点法と単体法の優劣は決まったと考えた人は少なくなかったであろう．もっとも価格はハードウェア込みで890万ドル，120円で計算して11億6,800万円とあっては，KORBXを買う気になるかどうかはまた別である．

翌水曜日，IBMワトソン研究所のForrest氏の発表があった．彼はイギリスの故Martin Bealeの会社SCICONでLP, Integer Programmingのコードを扱っていた人で，基底逆行列の計算法に関する論文もあり，単体法の研究では年期が入っている．3年前IBMに移ったそうである．彼の話はIBMのLPコードMPSX(当然単体法である) にベクトルプロセサ向きの手直しを行なった成果の発表で，これまたMINOSに比べて30倍とか50倍速いという結果を出していた．相手がベクトルプロセサを使うならこちらも同じ条件でというわけである．単体法では，次に基底に入れる変数の選び方に任意性があり，これによって反復回数が大きく変る．反復が多い方法では，同種の問題を解いたときの所要時間のバラツキが大きくなる．基底に入れる変数の選択を良くするための情報を反復ごとに計算して，反復回数を少なくする手法にDEVEXというのがある．反復回数は減るが，反復当りの計算が増えて全体の計算時間は必ずしも小さくならない，というのがこれまでの評価であった．ところが，Forrestの結果ではDEVEXのための余分な計算はベクトルプロセサでは負担が軽く，反復回数減少の効果だけがはっきり出たということである．

結局，内点法と単体法の決着はまだついていないようである．はっきりしたのはMINOSが物差しとしては遅すぎることであろう．MINOSの遅さは反復回数が多いことによるらしく，そうだとすれば所要時間はかなりバラツキが大きいと見られ，MINOSの何倍という数値も信頼性が低くなる．MPSX, FMPS, MPSⅢ, SCICONICといった商用コードとのまともな比較，つまり同じ問題を同じ計算機で解いてみた比較が欲しいところである．もっともKORBXが特殊な機械の上でしか動かないので実現は難しいが，機械の性能は適当に換算するとして，同じ大型の問題を解いた結果はぜひ知りたいものである．

単体法が成功した原因は計算機の進歩につれて新しい手法を産み出し続けたことになるが，その最大のものは逆行列計算の省力化手法である．

$$Ax=b,\quad A=(a_1,a_2,\cdots,a_n)$$

というLPの制約に対して，Aの行数をmとすると，単体法ではAの列$a_1,a_2,\cdots,a_n$からm個を選び出して基底行列Bを作り，

$$Bx=b$$

を解くことですませる．そのためにB^{-1}を計算するので

まえだ　えいじろう　日本ユニシス㈱
〒107 港区赤坂2—17—51

あるが，A（したがってB）の要素のほとんどは0であって，普通のＬＰ問題ではB^{-1}は積行列形式でBの非零要素の数と同じ程度の非零要素で表現できている．またB^{-1}を計算する手間もほぼ同じである．a_jの非零要素数をp_j, p_jの平均をpとすると，B^{-1}の表現と計算量はmp程度ですむことになる．

内点法のアルゴリズムはいろいろあるが，結局，

$$Hd=(ADDA^t)d=f$$

の形の方程式を作って解く部分が計算時間の大半を占める．HやH^{-1}の表現をいかに疎に保つかが計算量を左右する．Dは$n\times n$の対角行列である．対角要素を$d_1, d_2, \cdots, d_n$とすると，

$$H=ADDA^t=\sum_{j=1}^{n} d_j{}^2 a_j a_j{}^t$$

となる．$a_j a_j{}^t$の非零要素は$p_j{}^2$個である．Hはこれをn個加えたもので，非零要素の重なる部分も多いからnp^2になるというわけではないが，pよりはp^2に比例する非零要素をもっていそうである．とすればこれを解く手間も少なくともp^2のオーダーになる．pが小さければ内点法，大きければ単体法が有利と言えるかも知れない．

Hは対称行列で，B^{-1}を求めるための手法はほとんど役に立たない．Aの行をうまく並べ換えておけば，Cholesky 分解の結果の三角行列を疎にできる．KORBXでも良い並べ換えを見つけるために計算時間の相当部分が費やされていた．しかし，どう並べ換えれば良いかについてのあまり有効な理論はないようである．B^{-1}の場合には，Bを最小のブロック三角行列に分解する効率の良い算法があって大いに役立っている．係数行列が対称の連立方程式を解くことは，数理計画ではあまり問題にしてこなかったが，構造解析などの物理的問題の世界では数十年来の問題であった．現在大した理論的助けがないのであれば，将来もあまり期待できないのかも知れない．

単体法ではやさしい方程式を多数回解き，内点法では解きにくい方程式を少ない回数解く．現状では，不思議なことに両者の計算時間があまり違わない．計算機の世界は並列処理の方に動いている．ベクトルプロセサでは従来の計算機とは基本演算の所要時間の考え方がずいぶん違ってくるので，アルゴリズムもそれに合せて考えなければならないだろうが，ベクトルプロセサの機能自体に注文をつける必要もありそうである．疎行列を素直に扱えるものはぜひ欲しいと思う．

汎用計算機ディーラーに身を置いていると，20年前とくらべてＬＰなど数理計画の比重は大変下がったと痛感する．計算機が安く使えるようになって，利用分野が広がり，数学モデル作りなどといった付加価値は高くても手間のかかる問題をやろうという人がいなくなったように見える．絶対数は増えているのかも知れないが，計算機関係者の増え方がものすごいので，霞んでしまって見えない．ＬＰコードの需要も計算機ユーザー数から見ると大変少ない．[2] はアメリカで売られているパソコンで使えるＬＰコードの調査であるが，コードは20種類以上あって，そのうち15種類はここ３年以内に売り出されたものだそうである．制約が数千の問題が解け，使い勝手のよさそうなものも多い．値段も50～2,000ドル程度で手頃である．何か土壌が違うのか，それとも日本でもこの方向に動くことを期待できるのだろうか．

内点法といえば特許問題が気になるところである．理論家が金銭的に報われるのは良いことだと思うが，特許料を取れる身になりそうもないことを棚に上げて言わせていただけば，アルゴリズムを特許で保護するのは有難くない．たとえば，内点法の改良案は数限りなくあり，その多くはすでに知られていることを使ってみたら効果があったといったものだろう．それらは新発見ではないかも知れないが，使うことを思いつかない場合よりはるかに良い結果をうむ．このようなものを特許と認めれば，特許の洪水になって，ちょっとした仕事をするにも膨大な特許情報を調べなければ始められなくなり，ソフトウェア開発費用が跳ね上がるし，特許権者との交渉による時間遅れも深刻になる．特許権者からすれば，権利の侵害を発見することはほとんど不可能であろう．あまり意味のない権利となりそうである．特許と認めないことにすれば，境界の設定は大変難しい．ソフトウェアの効率はごく細かい工夫の積み重ねで決まる部分が多いがこれらは盗まれたことがわかることを期待できないから公開しないことによってだけ保護が可能である．特許を侵害したと疑われるたびに中を覗き込まれるのではたまらないが，ソフトのコピーと違ってプログラムを読んでみなければ判定できない．よほどうまく運用しないと，特許による保護は混乱を生むばかりのように思われる．

参考文献

[1] OR/MS Today, Oct. 1988, vol. 15, no. 5, pp. 44.

[2] Sharda, R., "The State of the Art of Linear Programming on Personal Computers", INTERFACES, July-August 1988, vol. 18, no. 4, pp. 49-58.