グラフに対しての操作”により実現される．この操作は，どの関数をどの入力変数について微分するかを指定し実行される．また，1つの命令で複数の関数及び，複数の入力変数の指定を行うことができる．また，高階の微分は，1階の微分を繰返し行うことにより求められる．

(3)　計算グラフの部分グラフについて，その計算過程を必要最少限の中間変数を用いて出力することができる．

試作システムの評価として，試作システムとREDUCEを用いてトランジスタの動作モデルの1つであるEbers-MollモデルのI_B, I_Cを7個の入力変数について偏導関数を求め，その出力をFORTRANのサブルーチンにして計算を実行して，演算時間を測定した．その結果，試作システムが出力したプログラムの演算時間は，REDUCEが出力したプログラムの演算時間に対して，最適化コンパイラの最適化レベルが0の時0.13，最適化レベルが3の時0.69となり高速化が図れた．

今後の課題としては，操作性の高い入力インターフェイスとしての言語の設計，等価な節点の発見方法の検討などがある．

参考文献

1) Chikayama, T. (1981). UTILISP MANUAL, *Technical Report METR 81-6*, University of Tokyo.
2) Iri, M. (1984). Simultaneous computation of functions, partial derivatives and estimates of roundings errors-complexity and practicality. *Jpn. J. Appl. Math.*, **1**, 223-252.
3) 岩田憲和 (1984). 偏導関数計算の自動化，東京大学大学院工学系研究科情報工学専門課程修士論文．
4) Rall, L.B. (1981). *Automatic Differentiation: Techniques and Applications*, Lecture Notes in Computer Science, **120**, Springer-Verlag, Berlin.
5) 山下　稔 (1986). 関数及びその偏導関数の計算過程を表す計算グラフの自動作成，千葉大学工学部電気工学科卒業論文．

高速自動微分法の定式化の試みと利用のためのシステム

東京大学工学部　久保田　光　一・伊　理　正　夫

効率の良い偏導関数計算法である高速自動微分法は“計算グラフ”というデータ構造のもとでは最短路問題の解法と同等の算法であり，多変数関数の全ての変数に関する偏導関数を，変数の数に依存せずに，関数計算自身の手間の高々定数倍の手間で求めるものである．本稿では新たに“部分計算グラフ”を定義し，偏導関数の計算のための独立変数と従属変数の関係を明確にして，一般的な観点からの算法の定式化を試みた．

まず，四則演算や初等関数などの単項または二項演算を“基本演算”と呼び，各基本演算に演算の項数に応じて順序の付いた仮引数を対応付ける．また，基本演算の組み合わせにより定義されるような関数を“分解可能関数”と呼ぶ．分解可能関数を計算する過程で，基本演算の実行毎に結果を異なる変数（中間変数）に記憶し，各中間変数と（その値を計算した）基本演算とを対応付ける．関数fの計算グラフとは，fの中間変数，入力変数および定数を頂点とし，中間変数に対応する演算の仮引数をそれぞれ一本の有向枝（始点は実引数に対応する頂点，終点はその中間変数に対応する頂点）として作られるグラフ$G(V, E)$である．さらに，各枝eには，その終点の基本演算をその枝に対応する仮引数で偏微分したときの値$d(e)$(要素的偏導関

数値）を対応付ける．計算グラフ $G(V,\ E)$ の部分計算グラフ $G(U,\ F)$ とは次のようなグラフである：(a) 頂点集合 U は V の部分集合で，定数に対応する頂点を全て含んでいるものであり，(b) 枝集合 F は以下の W の要素を終点とする枝からなる E の部分集合である；W は，U の要素のうちそれを終点とする E の枝の始点もまた U に含まれるような頂点の集合の，任意の部分集合とする．(枝集合が空集合であるようなグラフも部分計算グラフとみなす．) 各枝はもとの計算グラフ $G(V,E)$ の枝に付随していた値 $d(e)$ を継承するとする．

計算グラフおよび部分計算グラフを用いると，二種類の部分計算グラフの拡大操作で偏導関数計算の算法を自然に記述することができる．第一の拡大操作は，初期部分計算グラフを $G(V_X\cup V_K,\ \phi)$ (V_X は入力変数の集合，V_K は定数の集合）として，これに頂点と枝を付加しながら部分計算グラフを拡大し，ある一つの変数に関する全ての頂点の偏導関数を求めて行く Bottom-Up 拡大（BU 拡大）である．第二の拡大操作は，初期部分グラフを $G(\{f\},\phi)$ (f は最終的な関数値に対応する頂点）として，これを拡大して全ての頂点に関する f の偏導関数を求めて行く Top-Down 拡大（TD 拡大）である．両者とも，v の u に関する偏導関数は，u から v へ到る道を形成する枝 e に関して $d(e)$ の積を作り，u から v へ到る全ての道に関してその様な積の和をとったものに等しい．高速自動微分法は，TD 拡大による算法として特徴づけられる．

高速自動微分法利用のための FORTRAN プリプロセッサ方式のシステムは，関数が FORTRAN 副プログラムとして記述されているとき，そのプログラムを高速自動微分法により関数値と偏微分係数の両方を計算する FORTRAN 副プログラムに変換するものである．プリプロセッサは副プログラム中に IF 文，DO 文，GOTO 文等の使用を許すので広範囲な関数に適用することができる．具体的な使用法は，(1) スカラー値関数についてはそれを関数副プログラムの形で記述し，勾配の値が計算され返される配列等のいくつかの仮引数を，その FUNCTION 文に付加してプリプロセッサの入力とする，(2) これをプリプロセッサで処理して，付加された仮引数に勾配の値が計算され代入されるような関数副プログラムを得る，(3) メインルーチンでは単にその関数副プログラムを呼び出すだけで，実引数に勾配の値が計算され戻ってくる，というものである．ベクトル値関数については関数副プログラムの代わりにサブルーチン副プログラムを用いて同様に記述すれば良い．

参考文献

1) Iri, M. (1984). Simultaneous computation of functions, partial derivatives and estimates of rounding errors — complexity and practicality, *Jpn. J. Appl. Math.*, **1**, 223-252.
2) 伊理正夫，土谷 隆，星 守 (1985). 偏導関数計算と丸め誤差推定の自動化の大規模非線形方程式系への応用，情報処理，**26**, 1411-1420.
3) 岩田憲和，伊理正夫 (1983). 多変数関数の勾配の計算方法，情報処理学会数値解析研究資料，**83-7**.
4) Kim, K.V. Nesterov, Yu.E. and Cherkasskiĭ, B.V. (1984). Otsenka trudoëmkosti vychisleniya gradienta, *Doklady Akademii Nauk SSSR*, Matematika, **275**, 1306-1309.
5) Kim, K.V. Nesterov, Yu.E. and Cherkasskiy, B.V. (1985). An algorithm for fast differentiations and its applications, *Abstracts of the 12th IFIP Conference on System Modelling and Optimization*, 181-182.
6) 久保田光一，伊理正夫 (1986). 高速微分法利用システム — FORTRAN プリプロセッサ，第15回数値解析シンポジウム論文集，84-87.
7) Rall, L.B. (1981). *Automatic Differentiation — Techniques and Applications*, Lecture Notes in Computer Science, Springer-Verlag, 120.