

非線形最適化プログラム“HINLPS/HAP, HINLPS/M”

Nonlinear Programming Systems

解析すべき対象の大規模化による自動設計技術の確立に対する要求などにより、高度な非線形最適化技術への期待が高まっている。そこで、非線形最適化プログラムHINLPS/HAP(Hitachi Nonlinear Programming System/HAP)とHINLPS/M(HINLPS/M series)を開発した。HINLPS/HAPはHITAC S-820用であり、HINLPS/MはHITAC Mシリーズ用である。

HINLPSでは、非線形最適化手法として、最も評価の高い逐次二次計画法を採用するとともに、部分問題の解法である二次計画法、直線探索手法などに独自の新技术を採用した。さらに、使い勝手向上のために、自動的に微分などを行う数式処理機能を用意した。

田村正義* Masayoshi Tamura
 西方政春* Masaharu Nishikata
 野中久典** Hisanori Nonaka
 小林康弘*** Yasuhiro Kobayashi

1 緒 言

「最適化」は、すべての分野での課題である。構造解析、生産計画、投資など多くの分野で最適化すべき問題がある。さまざまな問題を表す手段として、非線形最適化問題は線形最適化問題とともに、古くから重要な位置を占めてきた。

さらに、近年、計算機利用技術の向上や計算機適用分野の拡大、解くべき問題の大規模化などにより、非線形最適化に対する要求が高まっている。

例えば、構造解析の対象が大規模化している。その対象物を最適に自動設計するために、非線形最適化技術への期待が高まっている。すでに航空機や自動車、橋梁(りょう)などの構造を最適化するために、非線形最適化技術を応用している例が知られている^{1)~3)}。

線形最適化問題を解くための手法として単体法⁴⁾が確立し、内点法^{4), 5)}が注目されている。同様に、非線形最適化問題を解くための手法として、逐次二次計画法⁶⁾や信頼領域法⁷⁾が高い評価を得ている。

また、数式処理の技術の進歩も目覚ましい。高速自動微分法⁸⁾はその名の示すとおり、高速に微分を得るための数式処理手法として注目を集めている。

さらに、HITAC S-820に代表されるスーパーコンピュータによる処理能力の大きな向上も大規模問題に有効である。

このような背景のもと、非線形最適化プログラムHINLPS/HAP(Hitachi Nonlinear Programming System/HAP)とHINLPS/M(HINLPS/M series)を製品化した。HINLPS/HAPはHITAC S-820用であり、HINLPS/MはHITAC M

シリーズ用である。本稿では、両者をまとめてHINLPSと呼ぶ。

HINLPSでは、非線形最適化手法として逐次二次計画法を採用するとともに、部分問題の解法である二次計画法、直線探索手法などに独自の新技术を採用した。さらに、使い勝手向上のための数式処理機能には、高速自動微分法⁸⁾を採用した。

本稿では、HINLPSの概要に続いて、HINLPSで採用している手法と改良点について述べる。さらに、HINLPSの性能評価に続いて結論を述べる。

2 HINLPSの概要

2.1 非線形最適化問題

HINLPSが解く非線形最適化問題の式による表現を図1に示す。ここで、 a_i , b_i , c_i , d_i は定数、目的関数 $f(x)$ 、不等号制約関数 $g_i(x)$ 、等号制約関数 $h_i(x)$ は二階微分可能であるものとし、非線形であることを許す。

非線形最適化問題の等高線図による表現を図2に示す。ここで、許容領域とはすべての制約条件を満足する領域である。

2.2 HINLPSの機能概要

HINLPSの機能を次に列挙する。

(1) 関数計算ルーチン定義

目的関数、制約関数とそれらに対応する導関数は、利用者がFORTRAN副プログラムとして定義することが可能である。

(2) 変数、制約関数の上下限值設定

図1に示した非線形最適化問題の a_i , b_i , c_i , d_i を値として設定することが可能である。

* 日立製作所 ソフトウェア工場 ** 日立製作所 エネルギー研究所 *** 日立製作所 エネルギー研究所 工学博士

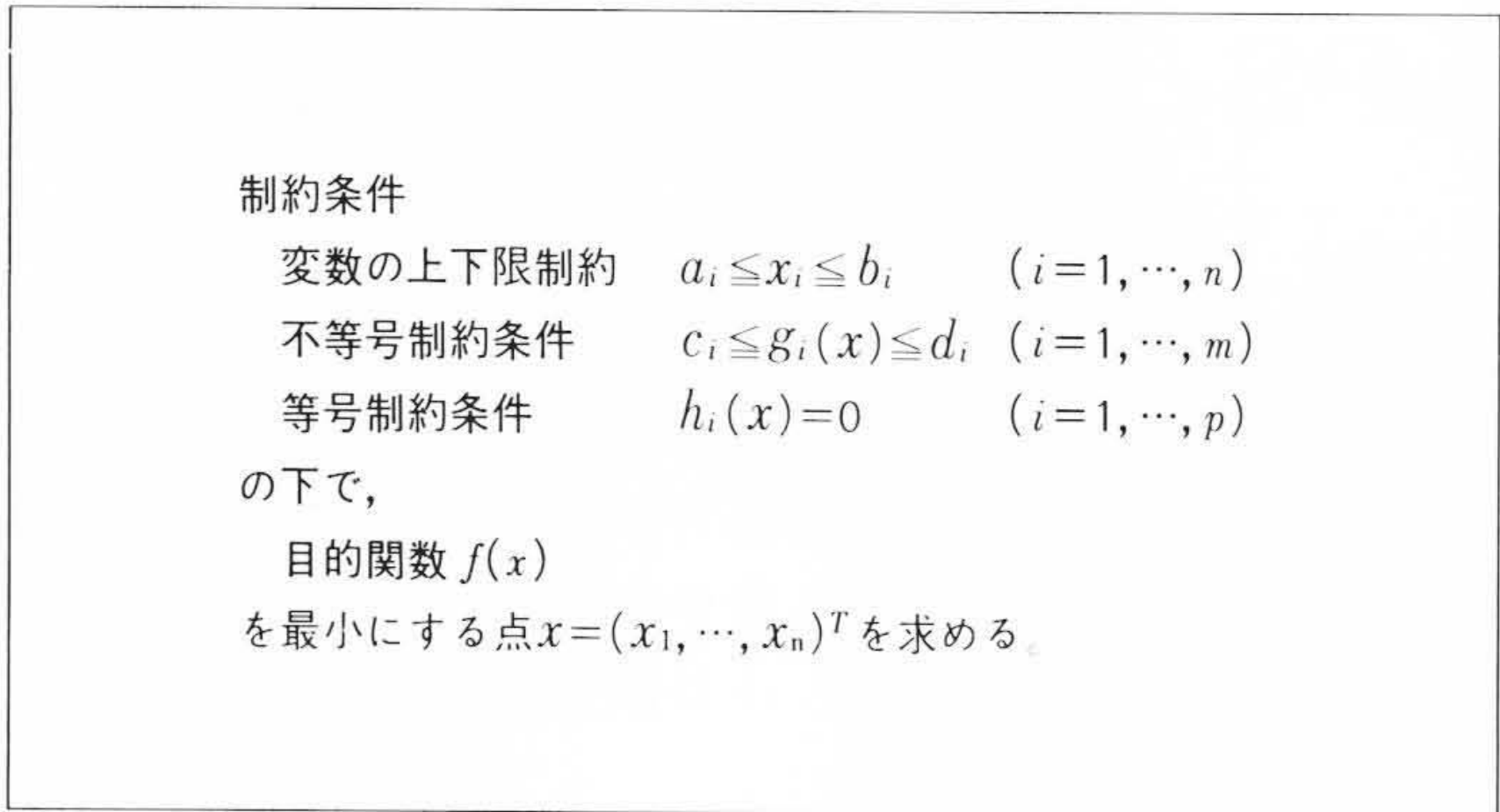


図1 非線形最適化問題(式による表現) HINLPSが解く非線形最適化問題を式で表した。

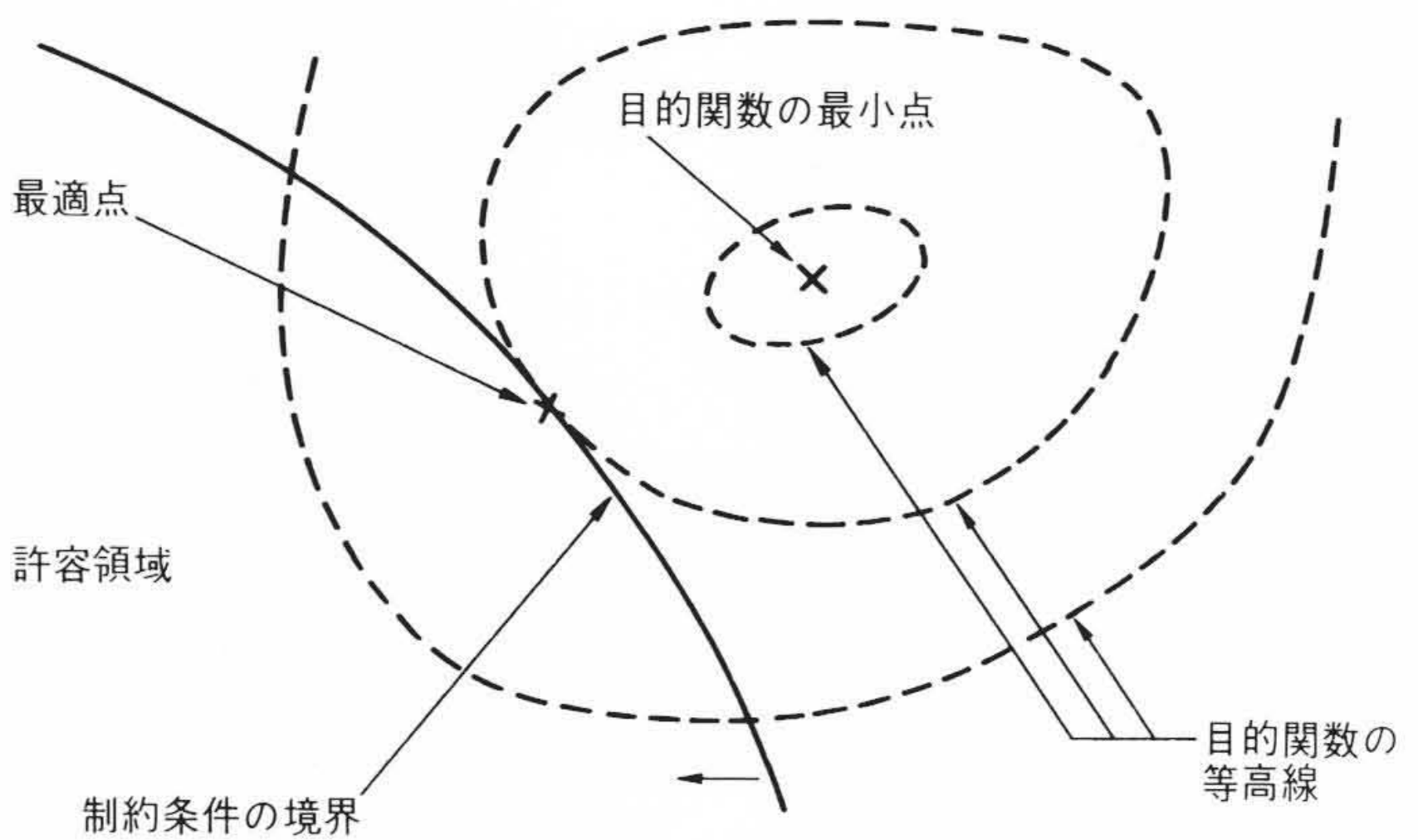


図2 非線形最適化問題(等高線図による表現) HINLPSが解く非線形最適化問題を等高線図で表した。

(3) 非線形最適化計算

数値計算により、非線形最適化を行う。手法、収束判定許容値などが可変である。

(4) 数式処理

数式として記述した非線形最適化問題から、(1)で述べた副プログラム群を自動的に生成する。

(5) 高精度数値微分

(1)で述べた副プログラム群のうち、導関数の計算ルーチンを定義せずにHINLPSの持つ高精度数値微分機能を用いることができる。

2.3 HINLPSの用法

HINLPSには、大きく分けて二つの用法がある。FORTRAN副プログラムとしての使用と数式入力による使用である。

FORTRAN副プログラムとして使用する場合の実行の流れを図3に示す。利用者は、目的関数と制約関数およびそれらの導関数の計算ルーチンを、FORTRAN副プログラムとして定義する必要がある。この用法では、利用者のシステムの一部として利用することが容易である。

数式処理機能を使用する場合の実行の流れを図4に示す。

この用法では、利用者は数式として非線形最適化問題を記述するだけで容易に非線形最適化を行うことができる。

これらの組み合わせとして、数式処理機能を使って生成した目的関数と制約関数およびそれらの導関数の計算ルーチンを使って、FORTRAN副プログラムとして使うこともできる。

3 非線形最適化

HINLPSは、非線形最適化手法として逐次二次計画法を用いている。逐次二次計画法では、部分問題の解法などとして、二次計画法、メリット関数、直線探索手法、近似ヘッセ行列の更新が必要である。次に、そのおののについて、また高精度数値微分機能についても述べる。

3.1 逐次二次計画法⁶⁾

HINLPSが採用した非線形最適化手法は、逐次二次計画法である。逐次二次計画法は、高い評価^{6),11),13)}を得ている。

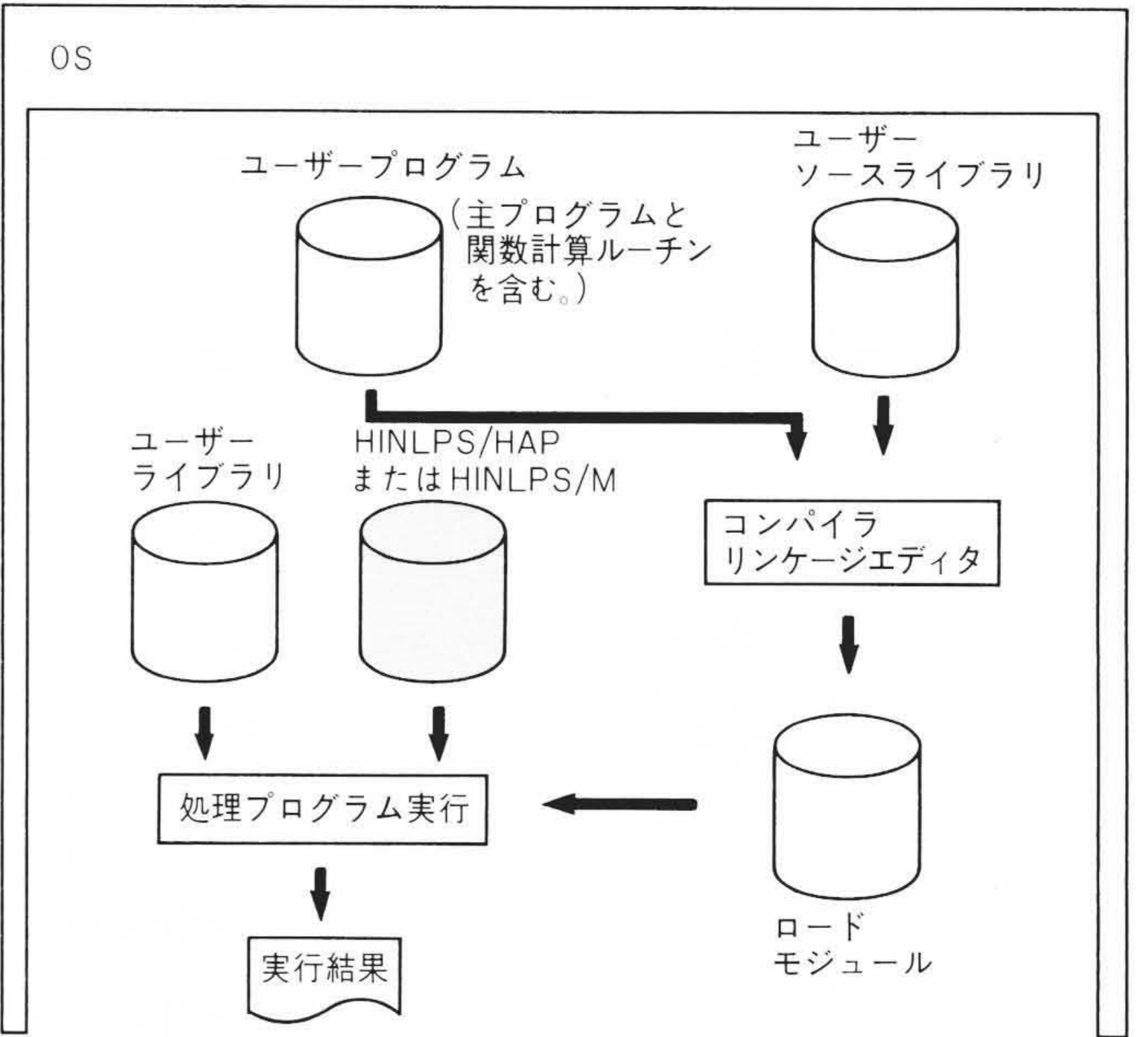
逐次二次計画法のアルゴリズムを図5に従って述べる。FORTRANで記述した数値計算プログラムによって、次の処理を実現している。

逐次二次計画法は反復法である。

図5中の①で、探索方向を決定するために二次計画問題を解く。二次計画問題は、解くべき非線形最適化問題の近似問題である。二次計画問題の目的関数は二次式であり、制約関数は一次式である。

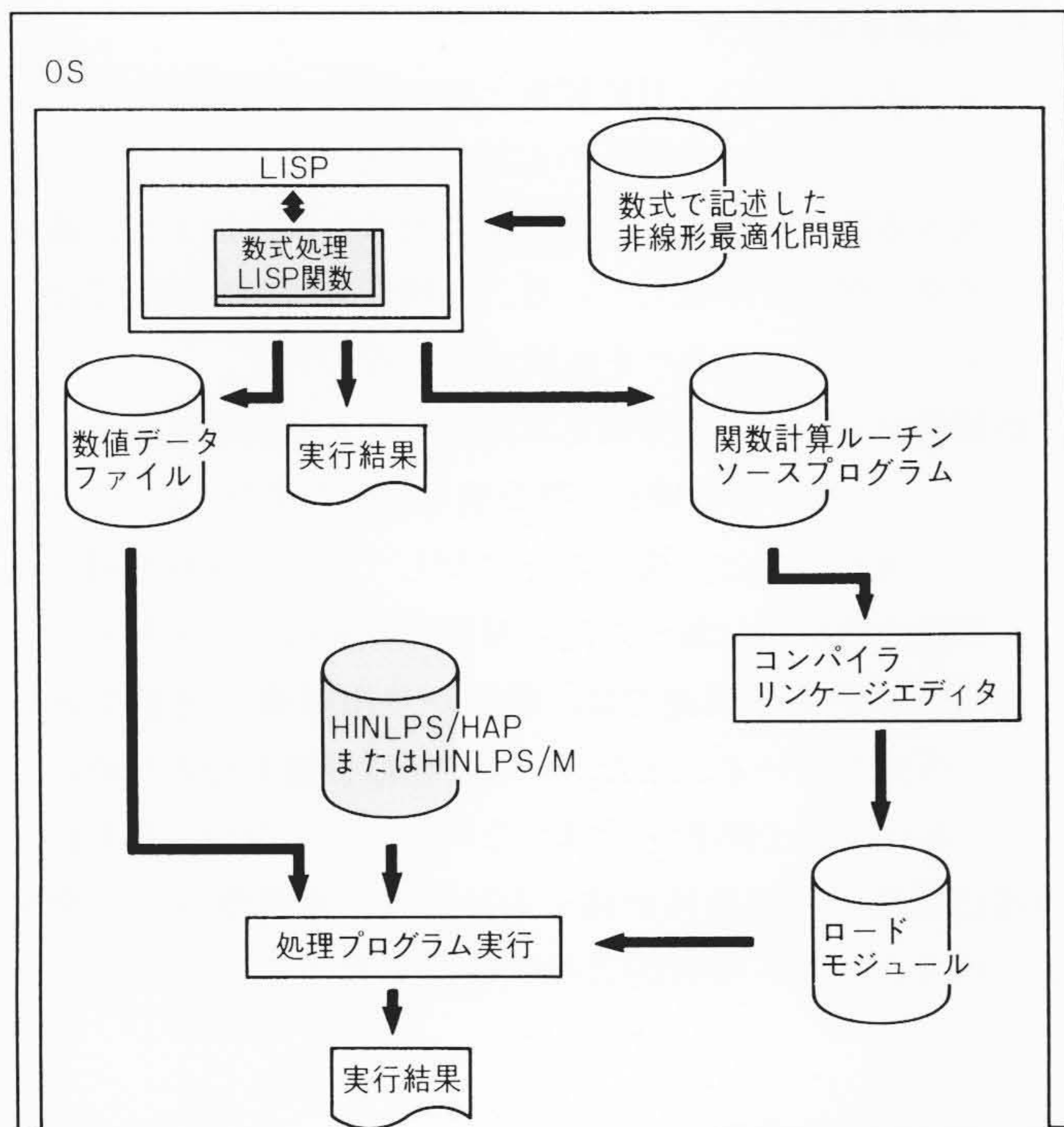
図5中の②で、同図①で求めた探索方向にどれだけ進むかを決める。これを直線探索と呼ぶ。

図5中の③で、近似ヘッセ行列を更新している。近似ヘッセ行列は、ラグランジュ関数のヘッセ行列(二階微分値の行列)



注：略語説明 HINLPS/HAP (Hitachi Nonlinear Programming System/HAP) HINLPS/M (HINLPS/M series)

図3 実行の流れ(I) HINLPSをFORTRAN副プログラムとして使用する場合の実行の流れである。



注：略語説明 LISP (List Processor)

図4 実行の流れ(2) HINLPSの数式処理機能を使用する場合の実行の流れである。

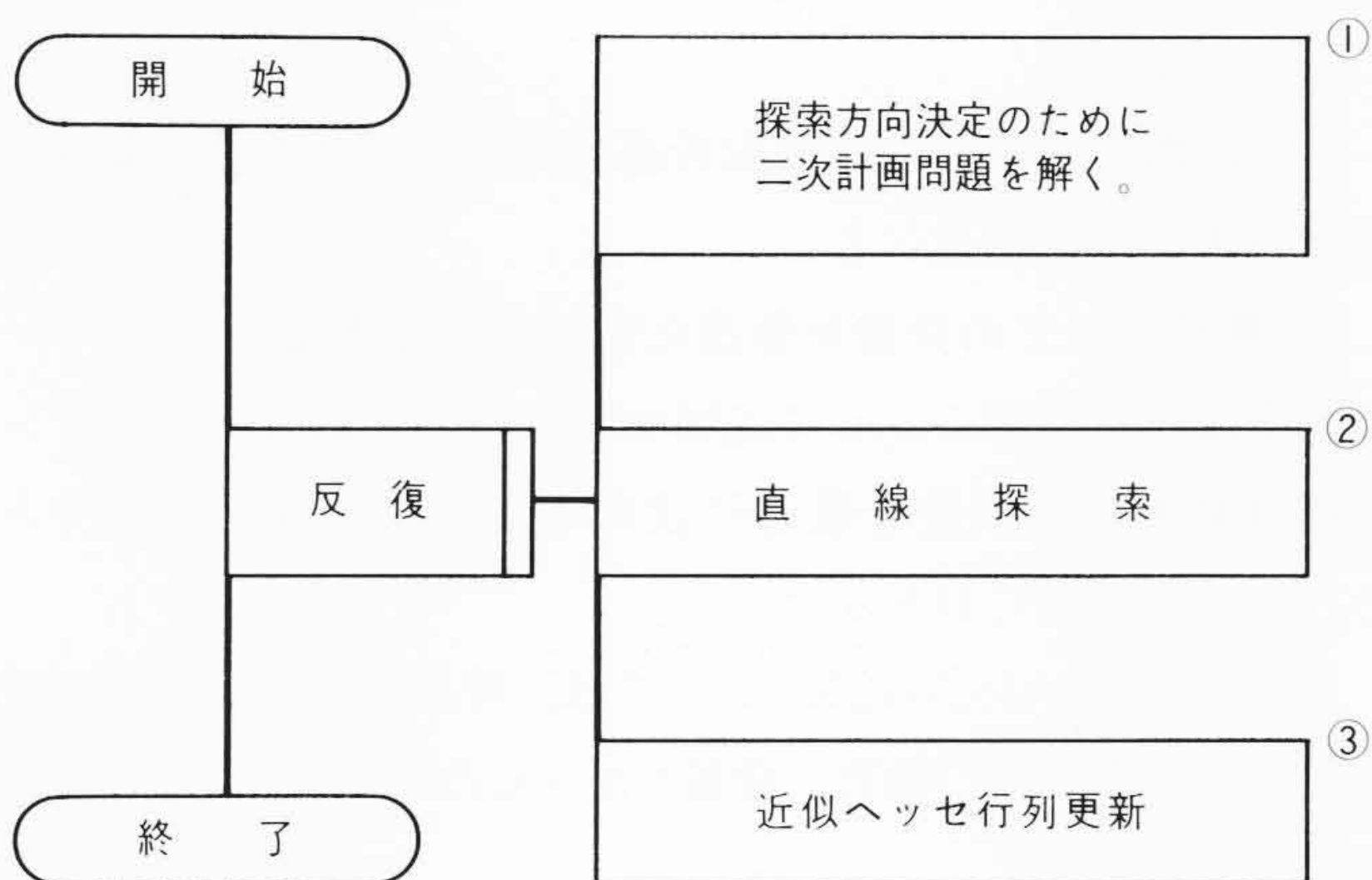


図5 逐次二次計画法 逐次二次計画法のアルゴリズムを示したPAD (Problem Analysis Diagram)図である。

の近似である。

つまり逐次二次計画法では、探索と同時に、二階微分に関する情報を更新する。

3.2 二次計画法

逐次二次計画法の各反復で探索方向を決めるために、二次計画問題を解く。二次計画問題を解くための二次計画法として、HINLPSは次の三つの手法を用意している。

- (1) GI (Goldfarb-Idnani) 法(標準手法)⁹⁾
- (2) 最小二乗法¹⁰⁾

(3) 上記(1)と(2)の自動選択

GI法は一般に最小二乗法に比べて速い。一方、二次計画問題の目的関数の二次係数行列である近似ヘッセ行列の性質が悪い場合には、最小二乗法でなければ解けない場合がある。そこで、上記(3)として両手法の特長を生かした手法を用意している。(3)ではまずGI法を適用し、二次計画問題が解けなかったときだけ最小二乗法を適用する。

通常はGI法だけで十分である。シトコウスキの問題集¹¹⁾の全問題をGI法だけを使って解くことができた。シトコウスキの問題集は、標準的な問題集として、非線形最適化手法の評価に多く用いられている。

3.3 メリット関数

直線探索時、点のよしあしを判断する基準がメリット関数の値である。制約条件を満足すると同時に目的関数を小さくする点がよりよい点であり、よりよい点に対してはメリット関数の値が小さくなる。

HINLPSはメリット関数として、次の二つの関数を用意している。

- (1) スカラーペナルティ形拡張ラグランジュ関数⁶⁾(標準関数)
- (2) ベクトルペナルティ形拡張ラグランジュ関数

上記はどちらも目的関数と制約関数を結合している。その際、制約関数に重みを付けている。この重みがペナルティパラメータである。

スカラーペナルティ形拡張ラグランジュ関数は、一般に拡張ラグランジュ関数と呼ばれている。拡張ラグランジュ関数はマラトス影響¹²⁾を防ぐことができる。マラトス影響とは、直線探索でほとんど探索点の改良を行えないことが何度も続き、最適化の効率が悪くなることである。メリット関数として、より初歩的なものを使うと起こる好ましくない現象である。

ベクトルペナルティ形拡張ラグランジュ関数の特長は、ペナルティパラメータの値が各制約条件ごとに変化することである。これにより、もともとの問題にスケーリングを施したのと同様の効果が得られる。スケーリングとは、互いにこう配などの大きく異なる目的関数や制約条件を、それぞれ定数倍することによって調整する操作のことである。

3.4 直線探索

直線探索では、探索方向に向かってどれだけ進めばよいかを決定する。現在の探索点を探索幅 $\alpha = 0$ の点、現在の探索点から探索方向ベクトル分進んだ点を探索幅 $\alpha = 1$ の点と言う。メリット関数 $\phi(\alpha)$ の値を十分小さくする探索幅 α^* を探す。この様子を図6に示す。

α^* を探すための直線探索手法として次の五つの手法を用意している。

- (1) 適応形アルミーホ法(標準手法)¹³⁾
- (2) 簡易形アルミーホ法
- (3) 番犬付きアルミーホ法

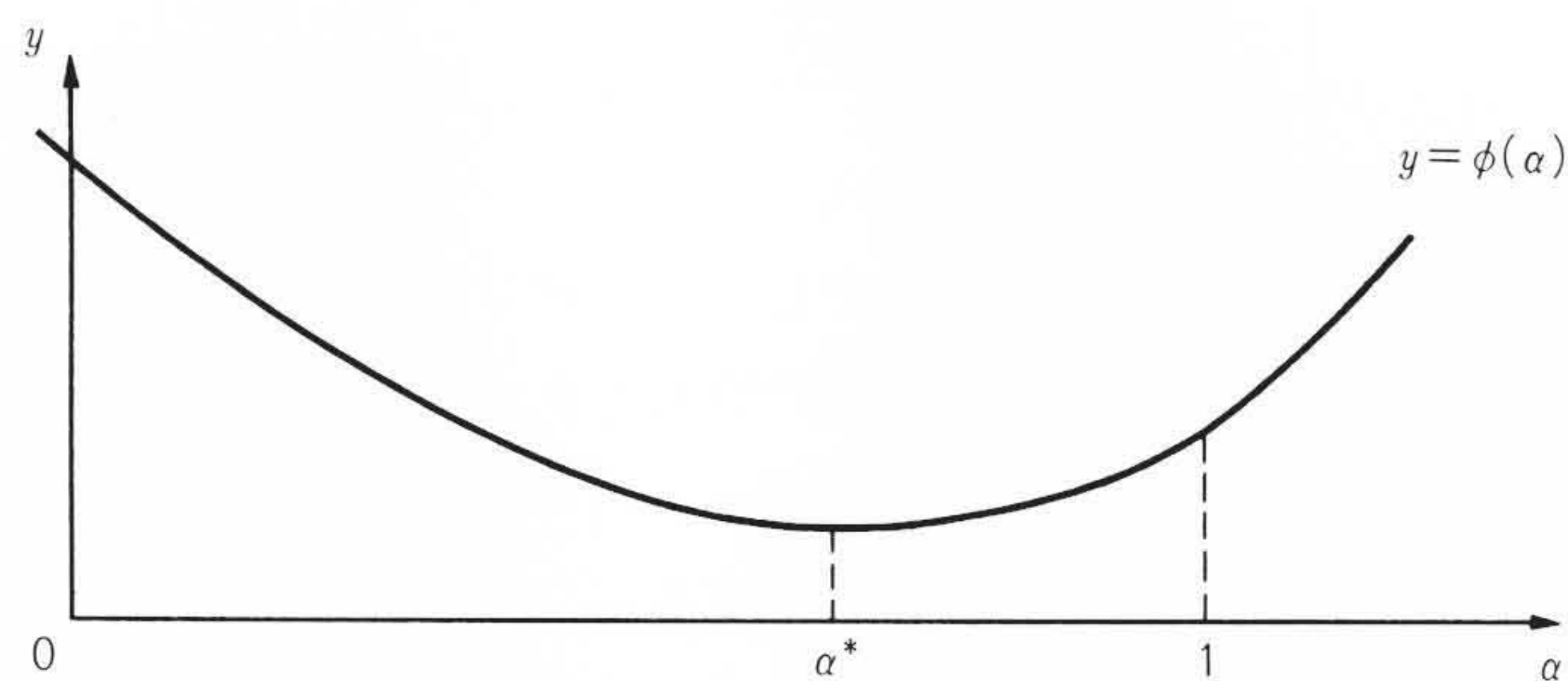


図6 メリット関数と探索幅 α の関係 HINLPSでの直線探索の目的関数であるメリット関数の例を示す。

(4) 黄金分割法¹⁴⁾

(5) 信頼領域法⁷⁾

上記(2)の簡易形アルミーホ法は、一般にアルミーホ法^{6), 14)}と呼ばれている手法である。(1)の適応形アルミーホ法は、アルミーホ法の日立製作所独自の改良である。アルミーホ法には結果に影響を与える定数がある。この定数を適応的に決めることによって、一般に安定させることに成功している。

ただし、適応形アルミーホ法では、目的関数や制約関数の計算回数が、簡易形アルミーホ法に比べて若干多い。そのため、スーパーコンピュータHITAC S-820でのベクトル処理に向かない逐次処理の部分の比重が高くなる。したがって、HITAC S-820では簡易形アルミーホ法のほうが効率のよいこともある。

上記(3)の番犬付きアルミーホ法は、番犬法¹⁵⁾と拡張ラグランジュ関数の組み合わせである。これもアルミーホ法の日立製作所独自の改良である。逐次二次計画法の何回かの反復で、探索点が悪くなることを許している。そのため、制約条件の性質が厳しくなっているときなどに有効なことがある。

上記(4)の黄金分割法は、各反復での直線探索をより精密に行うので、簡単な問題に対しては有効なことがある。しかし、逐次処理の比重が高いのでベクトル処理には向かない。

上記(5)の信頼領域法は、五つの手法の中で最も逐次処理の比重を小さくしているため、ベクトル処理による加速が最も高い。しかし、他の手法に比べて安定性が低い。

3.5 近似ヘッセ行列の更新

近似ヘッセ行列の更新には、Powellの修正BFGS(Broyden, Fletcher, Goldfarb and Shanno)公式⁶⁾を用いている。

この更新公式によれば、理論的には近似ヘッセ行列の正定値性が保証されている。近似ヘッセ行列の正定値性は、逐次二次計画法の収束に必要な条件である。しかし、計算誤差によって数値的に正定値性が崩れることがある。そこで、HINLPSでは計算誤差を自動的に検出し、正定値性が崩れたときには近似ヘッセ行列を単位行列にリセットするという処理を施している。これをリスタートと呼んでいる。

3.6 高精度数値微分

逐次二次計画法は、目的関数と制約関数の導関数の値を必要とする。しかし、導関数の定義は面倒である。そこで、4章で述べる数式処理機能による導関数の生成に加えて、高精度数値微分機能を用意している。高精度数値微分機能では、リチャードソン補外法¹⁶⁾を採用している。

数値微分として、微分値を求めたい点 x と微小正数 ε に対して、 $x \pm \varepsilon$ での関数値から微分値を得る方法がある。リチャードソン補外法では、複数の ε に対して、この操作を繰り返し、複数の数値微分値から高い精度の数値微分値を得る。

高精度数値微分機能では、関数が無限階微分可能であることを仮定している。また、一点での微分値を得るために、通常、約10の点で関数を計算しなければならない。さらに、極端に関数の非線形性が強くなければ、単精度以上の微分値を得られるが、解析的に導関数を定義するよりは精度が劣る。

4 数式処理機能

HINLPSは数式処理機能を用意した。数式として記述した非線形最適化問題を入力すれば、目的関数、制約関数とその導関数の計算ルーチンを生成する。その際、関数の微分には高速自動微分法⁸⁾を用いている。また、生成するFORTRAN副プログラムは極力HITAC S-820のベクトル処理に向くようにしている。

次に、数式としての非線形最適化問題の記述方法と高速自動微分法について述べる。

4.1 数式としての非線形最適化問題の記述方法

図7に示した例に従って説明する。この問題は、あるピン連結骨組モデル構造を最適に設計するために解く問題である。；が文の切れ目を示す。

タイトル語VARIABLE；以下では、非線形最適化問題の変数を宣言する。例の場合、変数が $x_1 \sim x_4$ の四つであることを宣言している。

タイトル語DATA；以下では、中間変数を定義する。例の場合、中間変数R3に $\sqrt{3}$ を設定している。このように、何度も現れる式などを設定しておくことと便利である。

タイトル語OBJECTIVE；以下では、目的関数を定義する。

タイトル語CONSTRAINTS；以下では、不等号制約関数を定義する。

タイトル語BOUND；以下では、変数の上下限制約を定義している。

このほか、配列、 Σ (総和関数)、 Π (総積関数)なども使用可能である。また、非線形最適化での出発点、収束判定許容値、手法選択などのための制御パラメータ群を設定することもできる。

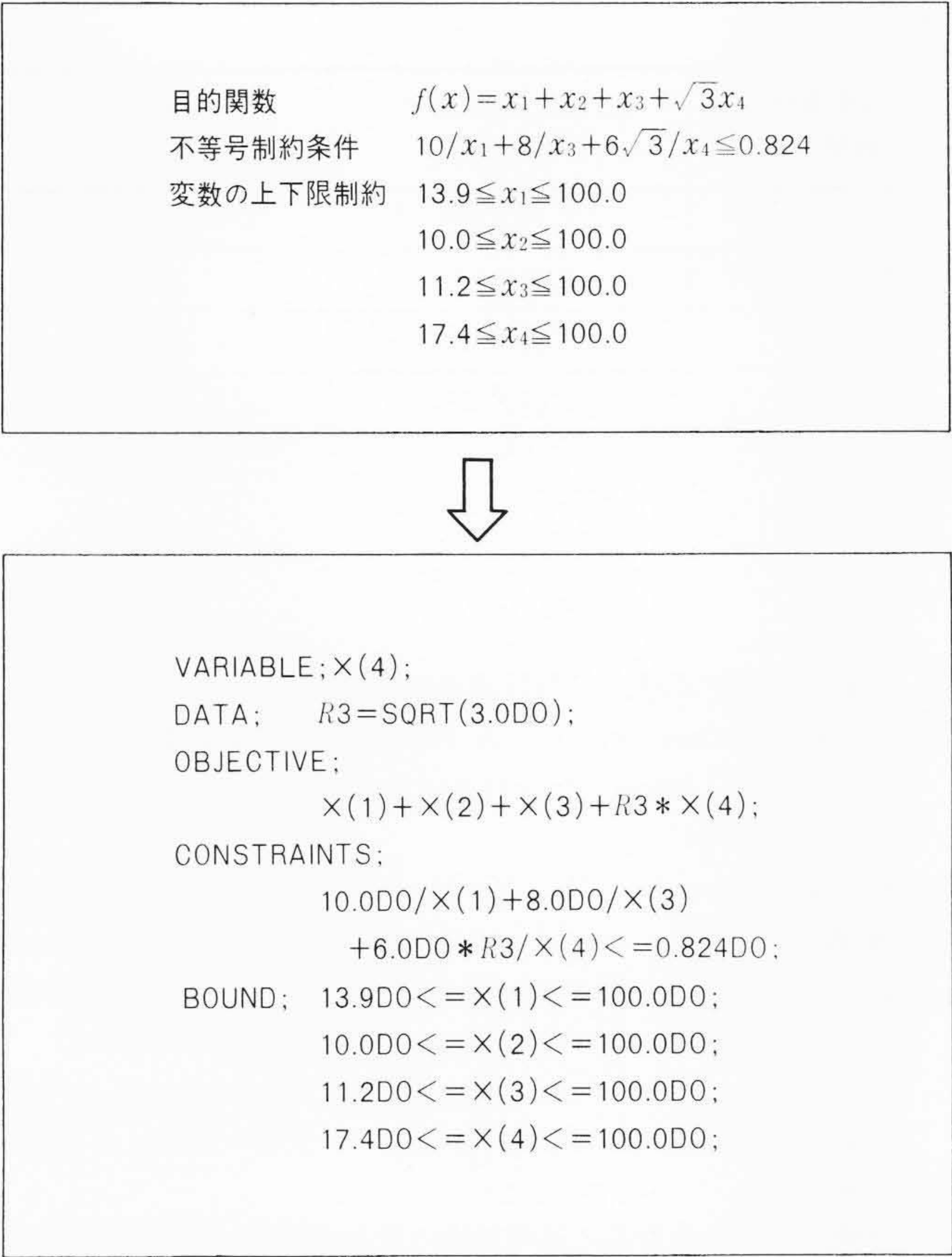


図7 数式として記述した非線形最適化問題 HINLPSの数式処理機能への入力例を示す。

4.2 高速自動微分法⁸⁾

高速自動微分法では、図8に示すようなグラフを作成する。例えば、 x_1 の導関数は、 x_1 から上にたどっていき、枝の式をすべて掛けることによって求めることができる。上にたどる経路が複数あるときには、経路ごとに掛けた結果を加えればよい。

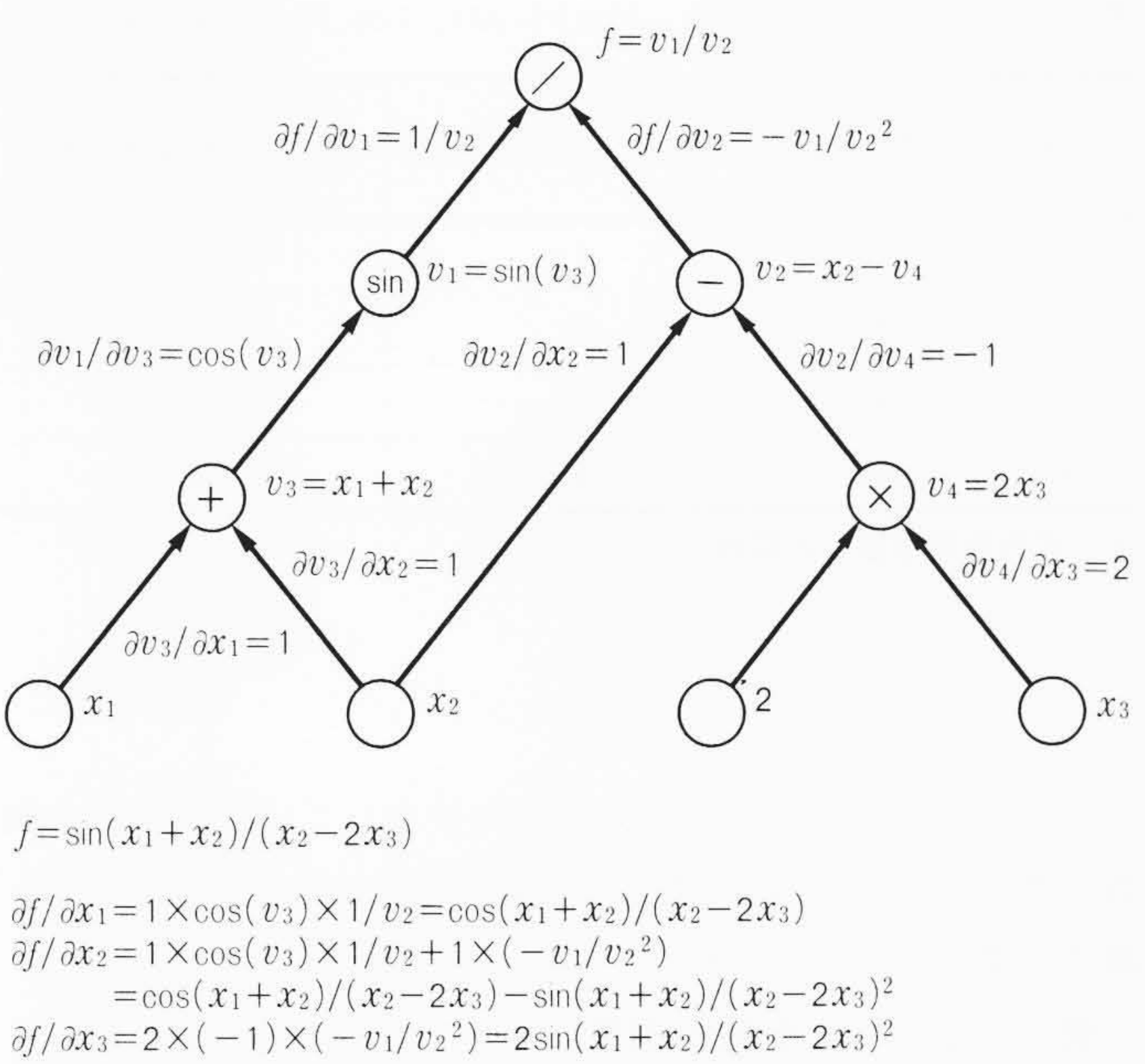


図8 高速自動微分法による導関数の生成 高速自動微分法で、式から生成する木構造図を示す。

5 性能評価

HINLPS/HAPおよびHINLPS/Mの性能評価を行う。特に、大規模問題に対するHINLPS/HAPとHINLPS/Mの性能比について述べる。

用いた問題は次のとおりである。

(1) 問題 1

0 以上200以下という変数の上下限制約と、各変数が定数以上という不等号制約条件の下で、変数の二乗和である目的関数を最小化する。

(2) 問題 2

ある湖の毎年の結氷日数から、地球が温暖化していると言えるかどうかを検証する問題である。

表1 問題1の計算結果 HINLPS/HAP, HINLPS/Mの実行結果と性能比である。

項番	変数の数	変数の上下限制約の数	不等号制約条件の数	(1) HINLPS/MのCPU時間(s)	(2) HINLPS/HAPのCPU時間(s)	CPU時間比 (1)/(2)
1	100	200	100	2.98×10^3	4.24×10^2	7.0
2	200	400	200	2.23×10^4	1.96×10^3	11.4
3	300	600	300	7.43×10^4	5.06×10^3	14.7
4	400	800	400	1.73×10^5	1.44×10^4	12.0
5	500	1,000	500	3.37×10^5	2.01×10^4	16.8
6	600	1,200	600	5.79×10^5	3.81×10^4	15.2

注：●HINLPS/MはHITAC M-682上で実行
●HINLPS/HAPはHITAC S-820モデル80上で実行
●手法は標準手法

表2 問題2の計算結果 HINLPS/HAP, HINLPS/Mの実行結果と性能比である。

項番	変数の数	変数の上下限制約の数	不等号制約条件の数	(1) HINLPS/Mの CPU時間(s)	(2) HINLPS/HAPの CPU時間(s)	CPU時間比 (1)/(2)
1	112	1	110	2.48×10^4	4.26×10^3	5.8
2	224	2	220	1.98×10^5	2.76×10^4	7.2
3	336	3	330	6.66×10^5	7.17×10^4	9.3
4	448	4	440	——	2.69×10^5	——

注：実行条件は表1と同様

この問題の変数の個数は、年数に依存している。変数の個数を変化させたときのHINLPS/HAPとHINLPS/Mの性能を表1, 2に示す。

表1, 2から、変数の個数が多い大規模問題に対して、HINLPS/HAPはHINLPS/Mに比べて10倍前後速く、HITAC S-820を有効に利用していることがわかる。

6 結 言

非線形最適化プログラムHINLPS/HAPおよびHINLPS/Mを開発した。HINLPS/HAPはHITAC S-820用であり、HINLPS/MはHITAC Mシリーズ用である。性能比較により、HINLPS/HAPはHINLPS/Mに比べて、大規模問題に対して有効であることがわかった。

非線形最適化手法として、HINLPSで採用した逐次二次計画法について述べた。逐次二次計画法は、現在最も評価が高い。

逐次二次計画法のための部分問題の解法などについて述べた。この中には、ベクトルペナルティ形拡張ラグランジュ関数、適応形アルミホ法など、多くの日立製作所独自の改良がある。また、数式処理機能について述べた。この機能は導関数計算ルーチンを自動的に生成するなど、使い勝手向上を実現している。微分のためには、高速自動微分法を採用した。

今後もHINLPSの使い勝手、効率などの向上のためにさまざまな機能拡張などを計画している。

参考文献

1) 鈴木, 外: 数理計画法の応用(実際編), 産業図書(1981)
2) G. N. Vanderplaats, et al.: An Efficient Approximation Technique for Frequency Constraints in Frame Optimization, International Journal for Numerical Methods, 26, 1057~1069(1988)
3) S. Kodiyalam, et al.: Shape Optimization of Three Dimensional Continuum Structures Via Force Approximation Technique, AIAA Journal, 27, 9, 1256~1263(1989)

4) 今野: 線形計画法, 日科技連出版社(1987)
5) N. Karmarker, et al.: A New Polynomial-Time Algorithm for Linear Programming, Combinatorica, 4, 373~395(1984)
6) K. Schittkowski: The Nonlinear Programming Method of Wilson, Han, and Powell with an Augmented Lagrangian Type Line Search Function-Part 1: Convergence Analysis, Numerische Mathematik, 38, 83~114(1981)
7) R. Fletcher: Practical Methods of Optimization—Volume 2 Constrained Optimization—, John Wiley & Sons(1981)
8) 伊理, 外: 数値計算の基礎技術の最近の話題から—ノルム, 丸め誤差, 偏微分と高速自動微分法を中心として—, 電子情報通信学会誌, 72, 10, 1044~1052(1989)
9) D. Goldfarb, et al.: A Numerically Stable Dual Method for Solving Strictly Convex Quadratic Programs, Mathematical Programming, 27, 1~33(1983)
10) K. Schittkowski: The Nonlinear Programming Method of Wilson, Han, and Powell with an Augmented Lagrangian Type Line Search Function-Part 2: An Efficient Implementation with Linear Least Squares Subproblems, Numerische Mathematik, 38, 115~127(1981)
11) W. Hock, et al.: Test Examples for Nonlinear Programming Codes, Springer-Verlag(1980)
12) M. J. D. Powell: Convergence Properties of Algorithms for Nonlinear Optimization, SIAM Review, 28, 4, 487~500(1986)
13) M. Tamura, et al.: Application of Sequential Quadratic Programming Software to an Actual Problem, Mathematical Programming series B (to be published)
14) 今野, 外: 非線形計画法, 日科技連出版社(1978)
15) R. M. Chamberlain, et al.: The Watchdog Technique for Facing Convergence in Algorithms for Constrained Optimization, Mathematical Programming Study, 16, 1~17(1982)
16) 伊理, 外: 数値計算の常識, 共立出版(1985)
17) R. E. Barlow, et al.: Statistical Interface under Order Restrictions, John Wiley & Sons(1972)