

I.F.I.P. 論文紹介*

90. 学校の時間表の作成

C.C. Gotlieb: The Construction of Class-Teacher Time-Tables [II-3, pp. 22~25]

30 あるいはそれ以上のクラスを持つ高等学校で、毎年時間表を作成する仕事はかなりの大仕事である。この論文は電子計算機をこれに用いる方法について述べたものである。

まず次の量を定義する、

- 教師の集合 $T = \{t_i\}$
- クラスの集合 $C = \{c_j\}$
- 学科時間の集合 $H = \{h_k\}$
- 計画マトリックス $\tilde{S}_k = \{\tilde{s}_{ij}^k\}$

また次を定義する、

$$\|A\| = \sum_k a_k$$

次に Availability ベクトル A^{ij} を定義する。これは教師 t_i がクラス C_j を計画によって担当する場合に 1、しない場合は 0 になるようなベクトルである。

以下の理論は一応教師中心で考える。 t_i のクラスの集合を C^i と書くと $D \subseteq C^i$ なる D について

$$\| \bigcup_{C_j \in D} A^{ij} \| \geq \sum_{C_j \in D} r_{ij} \quad (1)$$

が計画可能なための条件である。左辺の \cup は exclusive OR, 右辺の r_{ij} は t_i が C_j を担当する予定でまだ担当していない場合 1, 他は 0 となる量である。

(1) で等式がなり立ち、しかも D の部分集合ではなり立たないとき tight の状態と呼ぶ。この場合その教師とクラスは同じ時間 h_k では別の組合わせに使えない。この事情を考慮するため、各教師に対する計画相互の interaction Y^{ij} を考える (0 または 1 をとる)。すなわち

$$\begin{aligned} {}_{n+1}A^{ij} &= {}_nA^{ij} \cap {}_nY^{ij} \\ {}_0A^{ij} &= A^{ij} \end{aligned} \quad (2)$$

で Availability ベクトルの組 ${}_0A^{ij}, \dots, {}_nA^{ij}$ を作る。 \cap は AND である。もしこのうち一つでも Feasibility 条件(1)を満たさないものがあつたら、最初の問題は解をもたないことになるが、そうでなければこの組はある A^{ij} に収束する。

* 本号以後数回に分けて第 2 回国際情報処理学会の提出論文 (本誌 Vol. 3 No. 4 参照) の全部を Preprint of the Proceedings of the IFIP CONGRESS 62 (Aug. 1962) から紹介します。

次は学科時間について考える。与えられた t_i に対し、一つの tight 結合 D^i および P を考える。 $P^i \subseteq P$, P^i の部分集合 h_{k1}, \dots, h_{kM} に関し

$$\| \bigcup_{h_k \in P^i} A^{ik} \| \geq M \quad (3)$$

でなければならない。等号がなり立つとき、やはり tight と呼ぶ。すべての t_i に関して考えるために、さきほどと同様の操作を行なう。そうして得られるすべてのベクトルが Feasible, すなわち (3) を満たせば、その収束ベクトルが Final Availability ベクトルで、これから容易に解が求まる。

以上述べた方法は小さなモデルでテスト済みだが、現在 IBM 7090 用のプログラムを開発中である。

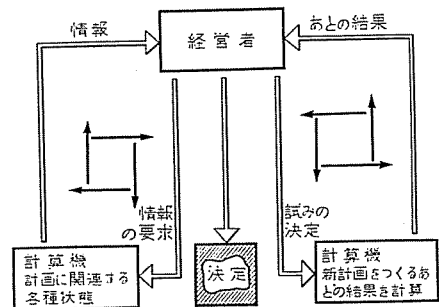
(岩城三郎)

91. 電子計算機による経営能力の拡大

A. Vazsonyi: Extending Management Capability by Electronic Computers [II-4, pp. 26~28]

今まで企業経営は経営者の経験と判断にたよってきたが、それが複雑になるにつれ系統的な方法が必要になってくる。ここ数年二つの面からの進歩が目立っている。一つは大形データ処理組織の登場、他は OR や経営科学における決定理論の発達である。

経営における政策決定の典型的な例を考えよう。上層部の会議でスタッフが問題点について現状と解決策を説明し、重役がそれについていろいろ質問する。あらゆる質問に対し答えられるということはまずないこ



経営における政策決定

とである。そこで図のように計算機を使う。

今度は経営者と計算機だけでよい。経営者の各種質

間に対し計算機は左側のループで情報の検索を行なう。これを何回かくりかえしたら、次は右のループで試みの決定を行なってみて予想結果をしらべる。右側のループを多数回くりかえし、時には再度左側のループも行なうようにすれば最良の政策を知ることができる。

このように計算機を、問題を解くための道具として on-line で計算尺のように使うためには、人間が計算機と話をする手段が大きな問題となる。

Thompson Ramo Wooldridge Inc. では、この目的のため特別の I/O 装置を開発した。一つは Display Analysis Console と呼ぶもので 17" 二つ、10" 一つ、計三つの CRT 出力装置、もう一つは Computer Communication Console と称する一つの CRT と多数のキーからなる入力装置である。30 個のキーに対しプラスチック製の字の入ったカバーをつけ、これによって必要な問を各キーに対応させて問の入力に使用する。カバーを変えれば異なる問題が扱える。これらコンソールは RW-300, RW-400, IBM 7090 で使用済みで、研究所で 1 年以上動作している。

最近 RW で公開された PERT (Program Evaluation and Review Technique) への応用を次に簡単にのべる。まず PERT 用のカバーをキーにかぶせる。各キー操作により 1. Analysis の Start 2. 比較する各計画を表示 (表示はすべて CRT で流れ図を含む) 3. そのうちの一つを選択 4. さらにその中の一部分に関する詳細な情報 (部分の選択は eight gun を CRT の像の特定のところ焦点を合わせて行なう) 5. Critical Path (太線で表示) と所要日数を表示 6. 上記 4・5 のくりかえしを行なう。さらに残業を考慮してシミュレーションをやりなおしたり、重要な activity を表示したり、労働力、費用などの表示要求がキーでできるようになっている。(岩城三郎)

92 微分方程式の近似解の新方法

J. Kuntzmann: Nouvelles Methodes Pour L'Integration Approchee des Equations Differentielles [V-1, pp. 70~73]

第 1 節では系 $x''=X(x, t)$ に対する Runge-Kutta type の公式の変形の原理をのべる。一階導函数を含まない系 $x''=X(x, t)$ は Runge Kutta type の公式

$$x_{i,\alpha} = x_i + h\theta_\alpha x_i' + \frac{1}{2}h^2 \sum_{\beta=0}^{\alpha-1} B_{\alpha,\beta} X_{i,\beta}$$

$$X_{i,\beta} = X(x_{i,\beta}, t_{i,\beta})$$

$$t_i + h\theta_\beta = t_{i,\beta}$$

$$x_{i+1} = x_{i0}$$

$$x'_{i+1} = x_i' + h \sum_{\beta=0}^{\alpha-1} A_{\alpha,\beta} X_{i,\beta}$$

によってとかれる。F. Ceschino は x_{i+1} が x'_{i+1} に関係なく求まる限り、 x'_{i+1} を表わす式の中の和は $\beta = q$ まで括げることが可能であるとの注意をしている。しかし、これは、この方法の価値をまさないとあてている。

第 2 節は公式の研究をする。P.J. Laurant はこの考え方によって、基本公式

$$x_{i,1} = x_i + hx_i' + \frac{1}{2}h^2 X_i$$

$$x'_{i+1} = x_i' + \frac{1}{2}h(X_i + X_{i+1})$$

を決定した。この公式の order は (2, 2) で示す。

$$\theta_1 = \frac{1}{2} \text{ とするときは}$$

$$x_{i,1} = x_i + \frac{1}{2}hx_i' + \frac{1}{8}h^2 X_i$$

$$x_{i+1} = x_i + hx_i' + \frac{1}{6}h^2(X_i + 2X_{i,1})$$

$$x'_{i+1} = x_i' + \frac{1}{6}h(X_i + 4X_{i,1} + X_{i+1})$$

で order は (3, 4) である。

さらに θ_1, θ_2 を用いた公式について、 θ_1, θ_2 にくつかの値を与える場合の結果をあげてある。

$$\theta_2 = \frac{3}{5} \text{ とするとき}$$

$$x_{i,1} = x_{i,0} + \frac{3}{5}hx'_{i,0} + \frac{9}{50}h^2X_{i,0} + \frac{9}{250}h^2DX_{i,0}$$

$$x_{i+1} = x_{i,0} + hx'_{i,0} + \frac{1}{2}h^2\left(\frac{29}{54}X_{i,0} + \frac{25}{54}X_{i,1}\right) + \frac{1}{36}h^3DX_{i,0}$$

$$x'_{i+1} = x'_{i,0} + \frac{1}{216}h[64X_{i,0} + 125X_{i,1} + 27X_{i+1}]$$

$$+ \frac{1}{36}h^2DX_{i,0}$$

ここで $x'''=DX$ とする。order (5, 6) の公式は

$$x_{i,1} = x_{i,0} + \frac{2}{3}hx'_{i,0} + \frac{2}{9}h^2x''_{i,0} + \frac{4}{81}h^3x'''_{i,0}$$

$$+ \frac{2}{243}h^4x''''_{i,0}$$

$$x_{i+1} = x_{i,0} + hx'_{i,0} + \frac{1}{2}h^2\left(\frac{53}{80}X_{i,0} + \frac{27}{80}X_{i,1}\right)$$

$$+ \frac{1}{240}h^3X'''_{i,0} + \frac{1}{240}h^4X''''_{i,0}$$

$$x'_{i+1} = x'_{i,0} + h\left(\frac{109}{380}X_{i,0} + \frac{243}{380}X_{i,1} + \frac{7}{95}X_{i+1}\right)$$

$$-\frac{7}{570}h^2X'''_{i,0} - \frac{1}{456}h^3X''''_{i,0}$$

となる。

第3節では、結論として次のように述べている。一般に、 q に対して、次の三つの式からなり立つ order $(q-1, q)$ の公式がある。すなわち

1. $\theta=(q-2)/q$ で、 h^{q-2} の項までの Taylor の公式
2. x_{i+1} を x_i とその $q-2$ 次までの導函数および $x_{i,1}$ の函数として与える式
3. x'_{i+1} を上のべた量および X_{i+1} によって与える式

第4節は $x''=x, x(0)=1, x'(0)=1$ に対する応用例をのべている。

第5節は次の問題として、 $x'=T(x, t)$ の系についての (multi step の Runge-Kutta type の公式、すなわち

$$x_{i,\alpha} = \sum_0^1 Q_{\alpha,m} X_{i-m} + h \sum_{\beta=0}^{\alpha-1} A_{\alpha,\beta} X_{i,\beta}$$

$$x_{i,q} = x_{i+1}$$

を考える。この公式は x_{i-m} を含んでおり、

$$\theta_1 = -1, \theta_2 = -2, \dots, \theta_l = -l$$

とおくことにより、 X_{i-m} もまた含まれる。すでにこのような方法として Nugeyre, Kuntzmann の論文があるが、ここには Siret の公式が使用される。これは特別の場合として Runge Kutta の公式 ($l=0$) multi step の classique な公式を含む。これを出すためには、

$$\theta_1 = -1, \theta_2 = -2, \dots, \theta_l = -l$$

$$q = l + 1 \text{ で}$$

$$x_{i,1} = x_{i-1}; x_{i,2} = x_{i-2}; \dots; x_{i,l} = x_{i-l}$$

が成り立つように、公式の係数を選ぶことで十分である。

また、さらに q と l の値の関係を表にまとめてある。

第6節では、解の安定性は $x_{i+1} = \sum_0^1 Q_{q,m} x_{i-m}$ の特性根に関係があること、また方程式

$$r^{l+1} = \sum_0^1 Q_{q,m} r^{l-m}$$

の 1 以外の根はすべてその絶対値が 1 より小さいことが必要であるとのべている。

第7節には、 q と l の値の五つの組合わせについて、公式の係数 $Q_{\alpha,m}, A_{\alpha,\beta}$ の表および数値計算例がある。

(藤野喜一)

93. コンボリューションの方程式の解の研究

J. Arsac: Recherche des Solutions d'une Equation de Convolution [V-2, pp. 74~77]

光学または電波天文学における星の見掛け上の直径の干渉の測定において現れる問題:

$$F(u) = \int_{-L}^L f(x) \exp(2\pi i u x) dx, -L \leq x \leq L$$

において $|F(u)|$ が与えられたとき、上式を満足する実、有界、非負の函数 $f(x)$ を決定することについての近似計算について述べられている。

解法の原理は筆者によって既に証明されている近似式:

$$f(x) \approx \sum_p f(p h) g\left(\frac{x-p}{h}\right)$$

(h は平行移動の幅)

を用いることにある。函数 $g(x)$ としては

$$g(x) = \frac{\sin \pi x}{\pi x}, \quad h = \frac{1}{2L}$$

とすることができて、上の近似式は

$$f(x) \approx 2L \sum_0^n a_p \frac{\sin \pi(2Lx-p)}{\pi(2Lx-p)}$$

の形に変形される。ここで a_p は未知数であり、 n は誤差の程度によって決定されるものである。さらに、コンボリューション $b(x) = f(x) * f(-x)$ に対して近似式

$$b(x) \approx 2L \sum_{-n}^n b_p \frac{\sin \pi(2Lx-p)}{\pi(2Lx-p)}$$

を用いれば b_p は既知であり、最初にのべた Fourier 変換の式から $F(u)$ の近似式を a_p を用いて表わすならば、結局 a_p と b_p との関係式が得られるが、その関係式は $z = \exp(-i\pi u/L)$ とおくことによって z および $1/z$ の多項式の間の関係に帰着される。そのことに基いて、一般的な解法は $-L \leq u \leq L$ で $|F(u)|^2$ に等しい週期 $2L$ の函数の Fourier 係数 b_p を計算すること、上の関係によって得られた一つの多項式の根を計算すること、その根が実根であるか虚根であるかを調べること、およびそれらを用いた一つの多項式をつくることによってその係数が求める a_p であるという 4 段階になっていることが示されている。

つぎに Hilbert の変換を用いるために必要な函数とその Fourier 展開が示され、それを予備手段として用いるならばプログラマーにとって非常に簡単な 3 段階による初等的な方法が述べられている。Hilbert の変換に現れる函数から直接計算するには 2 段階でよいことが示されている。

結語としては光学的意味における若干の注意が述べられている。(杉山昌平)

94. 擬線形一階偏微分方程式に対する初期値問題の近似解に対する差分法

R. Albrecht and W. Ulrich: A Difference Method for the Approximate Solution of The Initial Value Problem for Systems of Quasi-linear Partial Differential Equations of the First Order [V-3, pp. 78~79]

連立偏微分方程式

(1) a_k^{vi}(t, x, u) \frac{\partial u^k}{\partial x^v} = b^i(t, x, u), i=1, 2, \dots

..., m において, t=x^0, x=(x^1, x^2, \dots, x^n), u=(u^1, u^2, \dots, u^m) とし, k についての和は 1 から m まで, v についての和は 0 から n までとする。独立変数についての初期条件は

\lim_{t \rightarrow t_0+0} u^k(t, x) = \bar{u}^k(x)

とする。差分法を用いるには方程式 (1) が

(2) a_k^{vi} \left(\frac{du^k}{dt} \right)_{(ik)} = b_i, i=1, 2, \dots, m,

ここで

(3) \left(\frac{du^k}{dt} \right)_{(ik)} = \frac{a_{ik}^{v(i)}}{a_{ik}^{v(i)}} \frac{\partial u^k}{\partial x^v}, k=1, 2, \dots, m

と書かれることに基いている。(3) は擬特性曲線と呼ばれる常微分方程式

(4) \frac{dx}{dt} = \frac{a_{ik}^{v(i)}}{a_{ik}^{v(i)}}, v=1, 2, \dots, n

によって定められる積分曲線の方向導函数であることを意味している。

\tau > t_0 とし, 与えられた初期条件をもつ方程式 (1) の解 u^k(\tau, \xi) の近似はつぎの 3 段階の方法によって得られる:

a) 点 (\tau, \xi) を通る m^2 個の擬特性曲線を近似する直線を見つける。これらの直線と超平面 t=t_0 との交点を x_{(ik)} で表わせば, (4) を差分を用いて近似すると h=\tau-t_0 とし

(4') x_{(ik)}^v = \xi^v - h \frac{a_{ik}^{v(i)}(\tau, \xi, u(t_0, \xi))}{a_{ik}^{v(i)}(\tau, \xi, u(t_0, \xi))}

v=1, 2, \dots, n が得られる。ここで未知量 u^k(\tau, \xi) は既知量 u^k(t_0, \xi) によっておきかえられている。

b) u^k(\tau, \xi) の第 1 近似として v^k を得るために (2) を差分を用いて近似した連立一次方程式

a_k^{vi}(\tau, \xi, u(t_0, \xi))(v^k - u^k(t_0, x_{(ik)})) = hb^i(\tau, \xi, u(t_0, \xi))

を解く。

c) a), b) を繰り返すことによって改良された近似 v_{(k)}^k (\lambda=1, 2, \dots) を求める。v_{(k+1)}^k は a_k^{vi}, b^i(v_{(k)}^i = v^k) の変数において u^k(t_0, \xi) を v_{(k)}^k で置きかえることによって計算される。

数値計算のためには, 最初の超平面 t=t_0 の中正方形の網を考え, この網目の有限集合 G_0' 上で初期値が既知であるとする。G_0' のすべての点を含む一つの領域を G_0 とし, さらに近似された初期値が線形化されて計算されるとする。超平面 t=t_0+h, h>0 においても対応する網目を考え, この網目においては, 線形化された擬特性曲線が G_0 の点で最初の平面と交わるようなすべての網目よりなる集合 G_1' において決定される。超平面 t=t_0+h を新しい最初の平面として用いることにより, 超平面 t=t_0+2h における網目の集合 G_2' 上で近似解を得る。以下これを繰り返す。

この過程の収束に対する必要十分な条件は筆者等によって得られている (Numer. Math. 3(1961) 参照)。いままでの方法は ALGOL によってプログラムされ, コンパイルされ, ミュンヘン工科大学の計算機 PERM にかけている。また, 若干の数値例が満足に計算されている。

さらに, 函数 u^k(t, x) に関しての内挿法について公式が述べられ, ある場合には近似解を解析的に得ることができることが示され, その例として 2 変数に関しての偏微分方程式が例示されている。(杉山昌平)

95. 線形計画に対する近似解法の有用性

J. Habr: The Use of Approximation Methods in Linear Programming [VI-1, pp. 79~82]

この論文の主張は, LP に対する Simplex 解法は複雑で厳密すぎて Business 界の現実には不適当であろう。もって簡単な方法をとるべきであるとして, 一般の LP に対しては Bottleneck 法, 輸送型に対しては Frequency 法を提案している。これらの方法はいずれも計算原理が簡単で, 最適解を与える保証はないが, 多くの場合よい Feasible 解を与える。これらは近似解ではあるが, 一般に LP では Dual 問題をとくことによって最適解の目的関数と近似解のそれとの差の上限がわかるから, どれくらい近いかの保証をうることができる。

Bottleneck 法

目的関数の係数の大きさの順に x_i をきめていく。

(1) x_1 = \min(8/2, 6/2, 20/5) = 3

x_1 (2) (3) x_5 Max Sub 解 さ 関数 差は F1 つ こ すべ 列は の表

F zur dte: sch sitä 1 Asc pp. -

x_1 の分を資源量からひいた残りを表の I にかく。

(2) $x_2 = \min(0/3) = 0$

(3) $x_3 = \min(2/2, 5/1) = 1$

x_5 の分を I からひいて III にかく。以下同様

| Max | $10x_1 + 6x_2 + 4x_3 + 3x_4 + 5x_5$ | | I | II | III | | |
|-----|-------------------------------------|---------------|---------------|-----------|-----|---|--------|
| Sub | $2x_1$ | $+4x_3$ | $+2x_5$ | ≤ 8 | 2 | 0 | 0 |
| | $2x_1$ | $+3x_2 + x_3$ | | ≤ 6 | 0 | 0 | 0 |
| | $5x_1$ | | $+2x_4 + x_5$ | ≤ 20 | 5 | 4 | 0 |
| | | | $3x_4$ | ≤ 8 | 8 | 8 | 2 |
| 解 | 3 | 0 | 0 | 2 | 1 | | 目的値=41 |

さらに上の Dual 問題を同様な方法でとくと、目的関数の値が 43 となる。このことから真の最大値との差は $43 - 41 = 2$ をこえないことがわかる。

Frequency 法

つぎのような輸送形を考えよう。

これに対して、Frequencies 表 (各ルートの他のすべてのルートに対する評価) を作る (第 4 行, 第 6 列はチェック欄)。この小さい順に値をうめる。最後の表が近似 Feasible 解である。

| | | | | | |
|-----|-----|-----|-----|-----|----|
| 3 | 2 | 1 | 2 | 3 | 1 |
| 5 | 4 | 3 | -1 | 1 | 5 |
| 0 | 2 | 3 | 4 | 5 | 7 |
| 3 | 3 | 3 | 2 | 2 | 13 |
| 2 | -6 | -16 | 9 | 4 | 0 |
| 36 | 21 | 11 | -39 | -29 | 0 |
| -45 | -15 | 5 | 30 | 25 | 0 |
| 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 2 | 2 | 5 |
| 3 | 3 | 1 | 0 | 0 | 7 |
| 3 | 3 | 3 | 2 | 2 | 13 |

Frequency 法については、Die Frequenz-methode zur Lösung des Transportproblems und Verwandter Linear Programmierungsprobleme. Wissenschaftliche Zeitschrift der Technischen Universität Dresden 10 (1961) 5. 参照。(高橋啓郎)

96. 勾配法 (Gradient 法) のコンベックスプログラミングへの適用

T. Pietrzykowski: Application of the Steepest Ascent Method to Convex Programming [VI-2, pp. 83~86]

一般に線形計画, 二次計画などのようにある目的関

数 $f(p)$ ($p: n$ 次元ベクトル) を $\varphi_i(p) \geq 0$ ($i=1 \sim m$) の下に Max にするという問題のうち, f, φ_i が Convex なもの, つまり一般の Convex Programming の一つの逐次解法をこの論文は与えている。

考え方の骨子は, 条件式自身をまとめて目的関数の中に繰り入れたものを考え, 条件なしの Max 問題にやきなおそうということである。つまりあるパラメータ $\mu > 0$ に対して,

$$(1) G_\mu(p) = \mu f(p) + \frac{1}{2} \sum_{i=1}^m S(\varphi_i(p)) \varphi_i^2(p)$$

$$(S(x) = 0 \text{ if } x \geq 0, = -1 \text{ if } x < 0)$$

なる関数をつくり, $G_\mu(p)$ の Max 点を求める問題になおす。これについて,

[定理 1] 任意に $\varepsilon > 0$ を与えられたとき, $\mu > 0$ を十分小さくとれば, (1) の $G_\mu(p)$ の Max 点と真の解との距離を ε より小さくできる。(ただし $\varphi_i(p) \geq 0, i=1 \sim m$, できまる領域は有界, f, φ_i は連続で Convex)。

なる保証を与えている。

$G_\mu(p)$ の Max 点を求めるには Gradient 法を用いる。つまり第 0 次近似 p_0 を任意にとり, つぎの手順で近似を高める:

$$(2) q_k = VG_\mu(p_k) \quad (VG_\mu \text{ は } G_\mu \text{ の Gradient ベクトル})$$

$$(3) \frac{d}{dt} G_\mu(p_k + tq_k) = 0 \quad (G_\mu(p) \text{ は } p \text{ について連続でしかも一次微分可能) の } t \text{ についての根を } t_k \text{ として}$$

$$(4) p_{k+1} = p_k + t_k q_k$$

[定理 2] 上の手順できまる p_k は $k \rightarrow \infty$ で $G_\mu(p)$ の Max 点に達する(ただし f, φ_i は定理 1 の条件の他に 2 次微分可能の条件をみたとする)。

ことが保証される。

LP, 二次計画 (φ_i 一次), 二次計画 (φ_i も二次) について, (2), (3) の手順を具体化している。1 回の iteration (p_k から p_{k+1} を求める操作) の計算量の order はつぎのごとし; LP は mn , 二次 (φ_i 一次) は $mn + n^2$ 二次 (φ_i 二次) は $n^2 m$ のほか平方根演算, 3 次方程式の解が必要。

[訳者注] 一般の Lagrange 関数は条件式の数だけ Lagrange 乗数を導入するのに対して, この方法は, Lagrange 乗数なしで強引に $G_\mu(p)$ という 1 本の式にまとめていところに特徴がある。しかし実際に数値解をえたいとき, [定理 1] の μ をどうとるかがかなり問題であろう。その具体的手順にはふれていないし, 数値例もない。

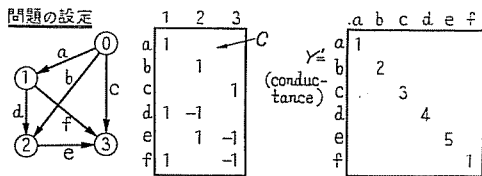
(高橋啓郎)

97. 回路問題に対する折線法

J.M. Bennett: A Breakpoint Technique for Network Problems [VI-3, pp. 87~90]

電気回路網は多くの物理現象のアナログに用いられるだけでなく、Dennis* が指摘しているように、直線形回路網の問題は線形不等式条件下の二次形式の最大化問題たる二次計画の問題と同等である。ここではある特種な直線形回路の問題をあつかうが、この手法は上記の広い分野に応用をもつだろう。

ここで考える問題は、線形直線回路で各 Branch は Conductance をもち電流の Capacity 制限があるとす。基準 Node 0 (電位 0) からのすべての Node Pairs (01, 02, 03) に一定の比で電流 q を与え、これを 0 から次第に増加させていくとき、回路の状態



(u, e, q') が次第にどう変化していくかを計算する手順を与えようとするものである。

$$Y=C^T Y' C = \begin{pmatrix} 6 & -4 & -1 \\ -4 & 11 & -5 \\ 1 & -5 & 9 \end{pmatrix}$$

(Admittance)

(1) $q = h \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix}$ (Node 0 をふくむ Branch の電流。 $h > 0$ をいろいろにうごかし q を与える)

(2) $Y u = q$ (u は Node 電位)

(3) $e = C u$ (e は Branch の電位差)

(4) $q' = Y' e$ (q' は Node 0 をふくまない Branch の電流)

(5) a, q' の各エレメントの絶対値は、たとえば、1 をこえない (電流 Capacity 制限)

もし Capacity 制限がなければ与えられた q に対して (2) (3) (4) の解 u, e, q' を求めればよい。つまり主として (2) なる連立一次方程式の解 (あるいは行列の逆転) が計算の主要部である。しかし Capacity 制限があるため単なる行列の逆転ではいかない。これに対して Dennis* の Break Point Tracing Procedure を適用する。

* Dennis, Mathematical Programming and Electrical Network, (Wiley) 1959.

つまり、 h を 0 から次第に増加させていくとき、どの Branch かではじめて Capacity に Saturate するところがあるのである。そしたらその Branch をとりのぞき、そのときの各 Branch の電流を Capacity からひいて Capacity を修正する。この操作をつづけていくとき、すでに Saturate した Branch の電圧が降下して、その Branch がふたたび影響を与えるような場合もでてくる。このときはその Branch をふたたび回路の中にとり入れなければならない。以上の操作をつづけてやがて h の Max 値に達することができる。

以上の操作を具体化する計算手順は結局、行列の逆行列を求めることと、その行列のある列 (または行) をとりのぞいたり入れかえたりした場合にその逆行列を前にえられたものから補正して求めるということである。この補正を n^2 (n は Node 数) の order で可能にした。(高橋啓郎)

98. 刻みの問題への整数についての線形計画法の応用

F. Genuys: Application de la Phogrammation Lineaire en Nombres Entiers a un Probleme de Decoupe [VI-4, pp. 91~93]

$Ax = a, x \geq 0$ の下に fx を最大ならめるという線形計画法を応用した Dantzig のシンプレックス法において、量 $f^s - uA^s$ が正であるような列 A^s の計算には繰り返えしが必要であることから、さらに Dantzig および Wolfe によって用いられた分解法に関する注意を用いることから出発して、種々なる変形のうち、結局つぎのような問題:

$L, l^i (1 \leq i \leq n)$ が与えられた自然数であるとき、

$$\sum_{i=1}^n l^i x_i + y = L$$

を満足し、かつ

$$\sum_{i=1}^n u^i - y > 0$$

ならしめる正または 0 の整数 $x_i (1 \leq i \leq n)$ および y を求めることについて説明されている。

この問題は整数解をもつときの線形計画法であって、Gomory による方法が用いられている。また、プログラミングのコードは FORTRAN で書かれたと述べられている。

実際のプログラミングには主プログラミングと補助プログラミングの 2 種類があり、主プログラミングでは初期値のとり方および繰り返えしのための基本式が

与え
ラミ
める
てい
た形
計
=8,
れて
えし
返え
る。
た結
つい
ミン
り返
算を
ロク
がね
しい
てい
9
R
Tec
ror
pp.
3
正角
きて
ジョ
状
デ
にラ
な
tch
Me
の
こ
1

与えられている。補助プログラミングでは、主プログラミングを実行しかつ問題を解くために、最大ならしめる補助の函数およびそのときの制約条件が定義されている。計算を容易ならしめるためにそれらを変形した形について述べられている。

計算例については、まず小規模のものについて、 $n=8$, $L=800$, および l_1, \dots, l_n などの値が与えられている。しかしながら、主プログラミングの繰り返えしが増加するにつれて、補助プログラミングの繰り返えしが極めて多くなるという実験報告がなされている。また、他の見地からの補助プログラミングを考えた結果に対しての主および補助プログラミングの数についても報告されている。しかしながら、主プログラミングの繰り返えしに必要な補助プログラミングの繰り返えしの数が急速に増加するために、やむを得ず計算を中断したが、その結果を分析してみると、補助プログラミングの中に用いた函数の係数に起因することがわかったので、そのような不便を取り去るために新しいプログラミングを作ったが、それはまだ実験されていないと報告されている。(杉山昌平)

99. デジタルコンピュータによる気象データ処理技術

R.B. Stauffer and T.H. Lewis: Met Watch: A Technique for Processing and Scanning Meteorological Data with a Digital Computer [VII-3, pp. 105~107]

ジェット航空機の発達に伴ない、その安全な操縦が正確な気象データに依存する度合いは大いに高まってきている。特に着陸時にはその天候に左右される故、ジェットパイロットに着陸する前にその天候の正しい状況を早く知らせ、種々の警告を発するためには気象データを連続的に監視しなければならない。このように天候を監視したり、天気予報を Check したり適切な助言を与えたりするこれら一連の機能は、Met-Watch として知られている。この中で述べられている Met-Watch の技術は、軍事用として調査された最新のものである。

この問題は次の三つに大別できる。

- 莫大な量のデータを扱えるシステムである。
- これらのデータの連続的なモニタリングができる。
- Offut の AFB にある Global Weather Center とこのシステムとを連動できること。

Met-Watch を人力で行なうとすれば、危険なばかりでなく大きな無駄である。ここで多くの電信回線からのデータを高速に処理し、かつ簡単な論理判断と適確な助言を発することのできる装置の必要に迫られたのである。以上のことより大容量、高速の Computer が全体のシステムの重要な部分をしめることは明らかであるが、これだけでは直接、通信路からデータを入力とすることができない。そこでまず第一に磁気テープにデータをストアしてから Met-Watch をやらせる方法を考えたが、結局入力がパッチ式になる、入力のあるたびに予報ルーチンを犠牲にし遅れが生じる、Computer の 24 時間使用で一寸の休止も許されない、などの理由で大形 Computer だけでは難点があることがわかった。

一方もっと小形の、安価な Computer で紙テープで入力を与えようという考えも扱うデータが 1 日 2 千万文字からのデータになると紙テープだけでも 10 字/in のもので 1 日 200 万インチにも達し、やはり時間遅れは避けられない。

このような研究の結果この二つの考えを結合させるという案にまとまった。その方式は電信回線を Computer に直結して一つの Computer を Met-Watch 用として使用する。また、この連続監視ができるよう 2 台の Computer により分割された大容量メモリユニットをもつ中央システムをもうける。このような形により完全なオートマチックなシステムとして、小形 Computer が Met-Watch の警告を発している時だけしか人間の介入は起らない。

さて入力データはコード化されたメッセージに分割され、これは集められて見出しのついた気象報告となる。この見出しは観測時の場所、時間、どの回線か、などを表わし、また、そのメッセージが地表のものか上空のものか、どのようなグループに属するかなどの分類にも用いられる。メッセージは、5 文字 1 語の多くの語を含む一つないしそれ以上のラインと呼ばれる 1 区切りで構成され、そのラインは直列のパルス信号として扱われる。

このような電信回線からのパルス列をコード化文字として認識しフトモードを考慮しながら CRLF などの不必要な文字を捨て適当な BCD に変換するのがこのシステムの初めの仕事である。20 個の回線は同時に働くので、総ての情報を Computer に送るためには高速の scanning が必要である。一番早い所では少なくとも 1 ms に 1 回のチェックが必要となる。各文

字は到着すると文字の順序が一つの論理的なブロックになるまでバッファーされる。このブロックはテレタイプ情報の一つのラインを含み、このラインより短い情報は認識しない。すなわちブロックでチェックを行っている。

小形 Computer は通信路のチェックと共にデータを有効にストアしなおす。こうして似た性質のメッセージはグループ別にストアされ、見出しによりメッセージの情報のタイプを区別する。通信路のエラーはこのシステムに重大な結果を及ぼすので、この方面に多くの努力をせねばならない。飛行場近辺の Met-Watch は我々の興味の対照となる所で 13,000 の観測ステーションのうち、1%がこの近辺にある。これらから少なくとも1時間ごとに信号が入り、データのトラフィックは天気の数となる。

各メッセージは雲の高さ、雲の量、天気、視界、気圧、温度、露点、風速、霧、判別用文字などの情報グループをもっており、Met-Watch は気圧、温度、露点については行なわない。各情報は分類されて、実際の滑走路での安全操縦ができるはんいの天気状況リストに参照され、適当なパラメータが選ばれる。

いかなるパラメータもない時は情報はプリントされ警報を發す。また天気予報もチェックが行なわれている。このシステムの次の段階では予報自体が Computer で行なわれるようになるだろう。この装置は20回線のテレタイプラインから情報を受ける Interface-unitがあり、1回線に1チャンネルが用意され、直列ビットはソフトレジスタにより並列ビットになり、マルチプレクサで規則的に走査される。データは変換マトリックスで変換される。このデータを処理する Processor で用いられるコードは種々あり、気象報告の見出しも2,000に分類されて用途に応じて分類格納されるので、その処理は非常に複雑となる。このため大容量のランダムアクセスメモリは是非必要なものである。この Computer は数学的演算よりもむしろ文字などの論理判断を考慮して設計されたものが望ましい。

以上の研究の結果 Met-Watch 手順のオートメ化の可能性は、現在の設備にそれほど莫大な金をかけなくても、ほとんど完全な天候データの入力システムが可能となるということである。(庄司 喬)

100. デジタル計算機による脳波解析

M.G. Saunders: Digital Computer Usage in Analysis of Electroencephalograph and Similar

Quasi-rhythmic Patterns [XIII-1, pp. 195~198]

脳の神経細胞の活動に伴う電位差(100 μ V 程度)は脳波と呼ばれ、その発現機構が一切わかっていないにもかかわらず、研究上、臨床上重要な道具になっている。熟練者による脳波記録の直観的な判定は比較的正確だし費用も安い、客観性の要請、モデルの検討などの面から大きな計算能力を前提とした数量化が必要となってきた。その一例をここに報告する。

普通に使われているペン書きチャートを透過して8個のホト・セルで走査し、線を横切るときのパルスを読みとり紙テープに穿孔(人手?)した。16.7 ms(チャート上では0.5 mm)ごとにとり1,000個の数値を得た。複雑な脳波パターンをそのまま扱うことはできないので少数の統計学的パラメータに捨象することにし、周波数が重要なこと、フィルタを通したノイズがある種の脳波に酷似していること、分析結果からフィルタの性質を推定できることなどから周波数分析をやってみた。正常人のアルファ波の例では10サイクル前後が卓越しており、それより0.5~1サイクル低いところにも二次的な活動がみとめられた。

なお生体では長い時間をとると異質な原因からくる変動があるので面倒になる点があり、数秒程度の小さいデータしか計算機にかけられないようである。

この研究はカナダ保健局精神衛生部でLGP 30を用いてなされたが、ごく素朴な試みにすぎず、脳波の統計的性質を記述したにとどまる。(西村 悠彦)

101. 両眼立体視の実験

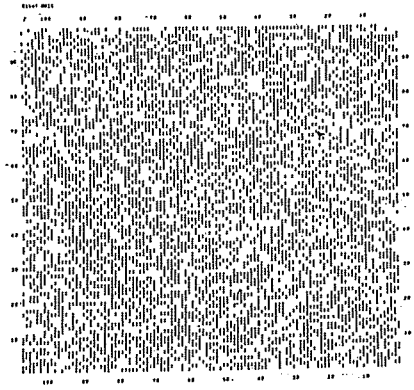
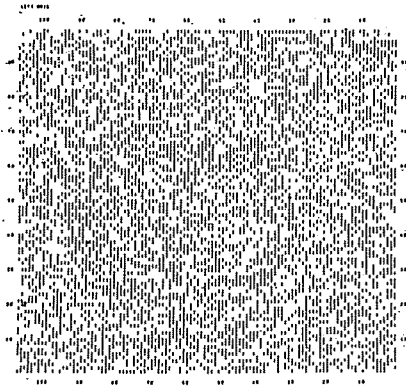
B. Julesz: Towards the Automation of Binocular Depth Perception [XIII-2, pp. 199~203]

三次元の環境を二次元の網膜投影によって把握するのに多くの手掛りが用いられるが、ここでは両眼視の手掛りだけをとりだして実験した。ある種の航空写真の作図機では明確に対応のつくようになっているが、ここに報告する方法は特定の図形について左右を対応づけるものではない。

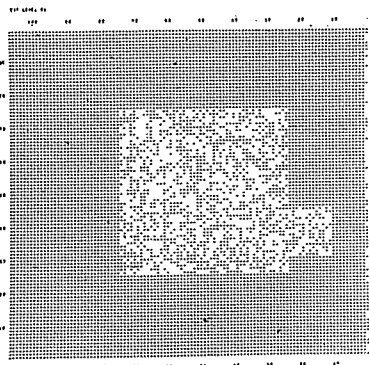
100点 \times 100点のランダムな黒白図形を計算機につくらせた。右眼視野と左眼視野は領域により異なった視差をもって対応がつくようになっている。

まず左右の視野を適当な視差 i だけ水平にずらしてかさね、一致点 minimum points をプロットしたものを視差 i の対応野 difference field D_i と呼ぶ(第2図)。これは高さ i における切断面(等高線)をあらわすはずだが、正しく定位された点の他に偶然かさな

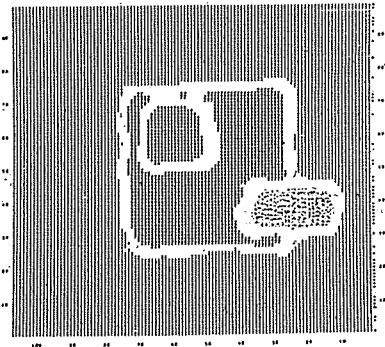
度) について 的 討 必 8 を (チ 直 を き に が イ ヤ ル いる さ 用 統 的) 00- する の 真 応 った ても 第 ち な



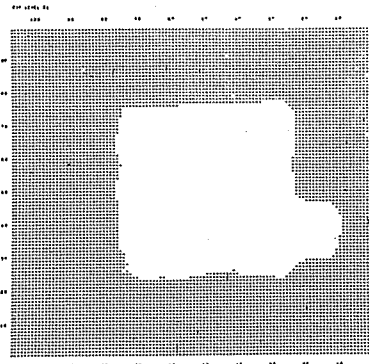
第1図 左眼視野と右眼視野



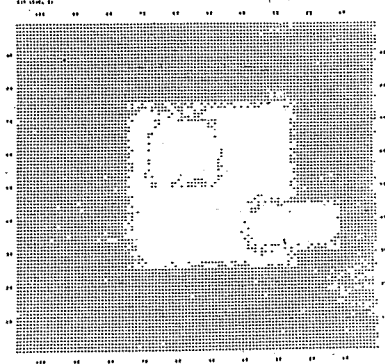
第2図 対応野 (D_1)



第4図 等高線図



第3図 クリーナにかけた対応野 (D_1')



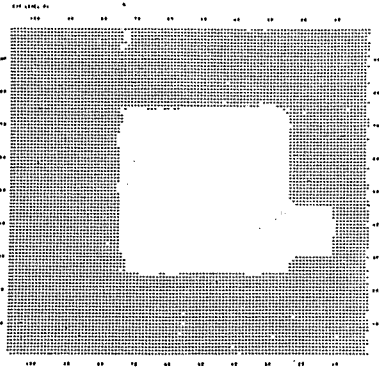
第5図 D_1^c と D_1' の結合

った点も含まれている。

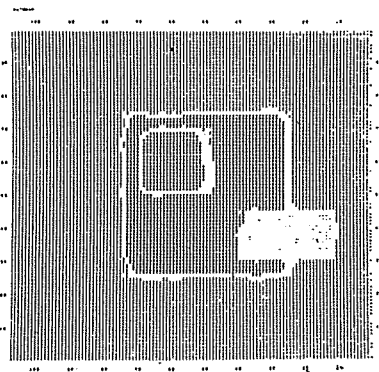
そこで強いノイズ・クリーナをかけ、つながりが40点以下の孤立した点を削ってしまう(第3図)。残った点はたしかにこの面*i*に定位されたわけだから、別の面の処理においては除外できる。この手続を*i*につ

いて順次くりかえし等高線図(第4図)ができる。

ここで3種の点が区別できる。1は確定点でたしかに一つの面に定位されたもの、2はあいまいな点で二つ以上の面に定位されたもの、ただし強いクリーナのおかげでこれはほとんどない。3はクリーナで削りと



第6図 第5図に弱いクリーナをかけたもの



第7図 第1次の回復による等高線図

られた不確定点で第4図にはblankで示されている。

これを回復するために不確定点だけについて対応野 D_i^c をつくり、 i の確定点 D_i' と結合し(第5図)、弱いクリーナをかけ6点以下の孤立した点を削る(第6図)。こうしてできたのが第1次の回復による等高線図(第7図)である。中央に正方形の台($i=2$)があり、その左に小さな正方形($i=4$)がのり、右に長方形の凹み($i=0$)がある。blankは結局片眼でしかみえない部分であり、隣接する領域のうち、遠い方に属するものとみなす。

この実験は、IBM 7090 のシミュレーション・ルーチン AUTOMAP-1 と General Dynamics SC-4020 マイクロフィルム・プリンタを用いて行なわれた。表面が一樣なテクスチャで、おとなしい高低をもつ面ではうまく動作するが、別な場面では人間ほどうまくはやれない可能性もある。

左右視野で形や反射の著しく異なる図形を対応づけた、単眼立体視の手掛り(たとえばテクスチャの勾

配)を評価することも考慮中である。(西村 恕彦)

102. 2台の計算機のあいだのプログラムの自動翻訳

A. Opler et al: Automatic Translation of Programs from one Computer to Another [XV-3, pp. 245~248]

コンピュータ間のプログラムの自動翻訳についてのべよう、研究したところによると、この方法を拡張すれば、IBM 705 のプログラムのかなりの部分が、翻訳できそうである。しかし全部できるということはたしかではない。この方法でできたプログラムは、シミュレーションよりは早いけれど、つくりなおされたプログラムよりは、おそいだろう。この仕事は、翻訳プログラムを完成したのではない、必要な知識をうのが目的であった。

翻訳のまえにおこなうことは、「もとの計算機」のプログラム(Source Computer program)の情報の復活(recovery)と、もとと計算機のメモリーから、「さきの計算機」(target computer)への写像(remapping)である。プログラムの内容は、ワードが、命令か、定数か、データか、どれかになっている。これをきめるのがむずかしいというのが翻訳を困難にしているのだ。プログラムをだんだんとアブソリュートなマシンコードへかえてくると、情報の損失がでてくる。これを反対にたどれるかということが問題である。そして復活は、プログラムをつくるよりずっと大変な仕事である。こうしてえられた情報は、もとの計算機の、シンボリックな言語のかたちで再現される。翻訳のレベルをどの程度(アッセンブル、コンパイルなど)までもどして行なうかも問題で、うんともどるなら復活の労力は大変なものになり、すこしもどるなら間接的なシミュレーションと大して違わないことになる。

もとの計算機がさきの計算機のなかでどうあらわされるかもきめておかなければならないが、とくに両方の計算機の論理設計がかなりちがっているとき、もとの計算機の性質が、写像によくあらわれている必要がある。復活された情報が十分でないときは、シミュレーションをおこなわなければならない。そこで、できたプログラムは、シミュレートされる部分と、翻訳でいく部分の二つからできることになる。もとのメモリーをうつすには、シミュレーションと部分だけでよい。

実施した仕事は、もと-さきとして、IBM 705-IBM 7074 の計算機のものをとって行なった。(和田英一)