

切欠きのある場に対するペナルティ法適用の注意

村田健郎

神奈川大学理学部 情報科学科

拡散方程式: $(cu)_t + \text{div}(-k\nabla u) = f$ をCV法によって空間離散化して、得られた常微分方程式を台形法によって解く問題を考える。場が長方形でなく、切欠きがある(例えばいわゆるT形の場の)場合、まともに場の中のアクティブな節点だけに注目して離散化を行う方法と比べて、いわゆるペナルティ法の一つすなわち場を長方形でかこって、切欠き部に大きな拡散係数を与えて長方形上で解く手法がある。この方法は便利な反面、ソルバーにMICCG法を使う場合、その反復回数が元数の増大による当然増をはるかに超えて、元数増でははかれない大きなCPUタイムの増大をもたらすことがある。その諸例についてのべ、ペナルティ法の、安心して適用できる応用例(場の微調整)についてのべる。

Some notes for adaptation of the Penalty method to chipped fields

Kenro Murata

Faculty of Science, KANAGAWA Univ.

2946 Tsuchiya, Hiratsuka, Kanagawa 259-12, Japan

We want to discretize the time-dependent equation, $(cu)_t + \text{div}(-k\nabla u) = f$, using the control volume method, and solve the obtained ordinary equation with the trapezoidal method. In the case where we deal with a chipped field such as a T-shaped field, the Penalty method with a large diffusion coefficient on the chip is much more convenient than the method which discretizes the field using active nodes, but needs much CPU time according not only to the increased dimension size by the method, but also to the increased iteration count, especially with the MICCG technique for the Solver.

This paper shows some examples and discusses about the effective usage of the method, for example, applied to the "trimming" of the field.

1 はじめに

いわゆる、場の問題（保存則をあらわす偏微分方程式の初期値・境界値問題）を数値的に解くための、新しい計算技法を開発するにあたり、その適用性をテストするために、とりあえず用意するのは2次元の場である。その際、長方形の場だけでは不十分で、すくなくとも、いわゆるL形場、T形場を用意すべきとされている（図1a, b）。特にT形の場合は、離散化のデバッグに便利のためしばしば引用される。

筆者は近頃、時間依存の拡散方程式


$$\frac{\partial}{\partial t}(cu) + \text{div}(-k(x,y,u)\nabla u) = f(x,y,t) \quad (1)$$

の数値解法をいろいろ研究しているが、その際、T形の場を手軽に作るために、いわゆるペナルティ技法を使ってみた。そのとき遭遇したことから得られた、その技法の適用にあたっての注意事項について報告する。

— 以下(1)は、熱方程式だと思ふことにする。

2 問題の説明

(1)において、 c は定数とする。このとき、(1)の非線形性は拡散係数 k に u 依存性があるだけによる。(1)は移流項がない、いわゆる純拡散の系だから、非線形とはいえず、まだ御しやすい問題と言える。場は、離散化のデバッグの便宜上、左右対称のT形の場（図1b）とし、左右、および下辺の境界条件は脊次固定、上辺は $\nabla u = 0$ のいわゆるノイマン境界条件とした。

この問題を、図2のような長方形の場でかこって、切欠き部（をした部）に大きな拡散係数値 $DF+$ を与えて、実質的に図1bと同じ問題となるように仕向ける。— ペナルティ法と総称される手法の一種である。

$u(x,y,t)$ の所望精度として、 10^{-3} 程度を考える。よって $DF+$ としては 10^6 なら十分、 10^4 でもぎりぎり間に合う程度である（標準の拡散係数値を1.0とした）。

切欠きの大きさが、ペナルティ法によるときのCPUタイムの増大の程度に影響を与えるから、実際には、図2の長方形 $A_0B_0C_0D_0$ を固定し、 AA_0 、 B_0B およびE点、F点を調節した。長方形 $A_0B_0C_0D_0$ は、文献(1)で提示したと同じ場で、そのときと同じソース（ $f(x,y)$ ）を与える。^(図3) “温めたり、冷したり”する理由は、“モードをリッチに”するためである。ただ冷すだけとか温

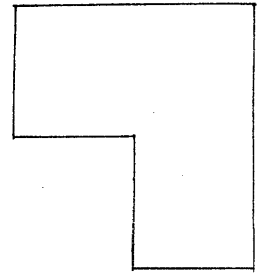


図1a L形場

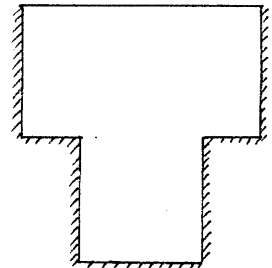


図1b T形場

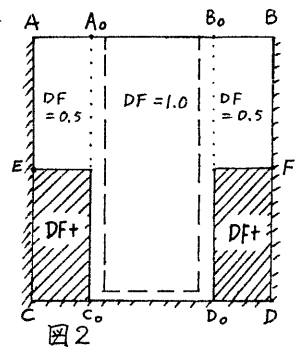


図2

めるだけだと高次モードが欠け、ペナルティ法の欠点が表面化しないおそれがある。基準となる長方形 $A_0B_0C_0D_0$ の寸法は、 $A_0B_0=16$ 、 $C_0D_0=25$ とし、基準のあみ目は1.0きざみとする。細分パラメータMJ, MJ1, MK, MK1により実際のあみ目を与える(図4)。たとえばMJ=2, MJ1=4のとき、 $C \sim LY$ 点間、 $LY1 \sim A$ 点間の(粗)メッシュ用の分点間隔は $1/2$ 、 $LY \sim LY1$ 間の(精)メッシュ用の分点間隔は $1/4$ である。場の内部の拡散係数の初期値に段差をもたせておく；図2の破線の内部のそれを1.0とし、その外側を0.5とした。やはり"モードをリッチに"するためである。

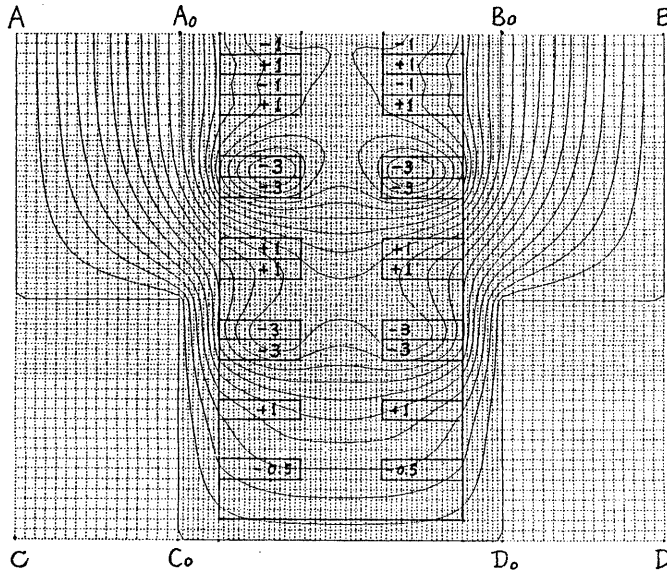


図3: ソースのパターン

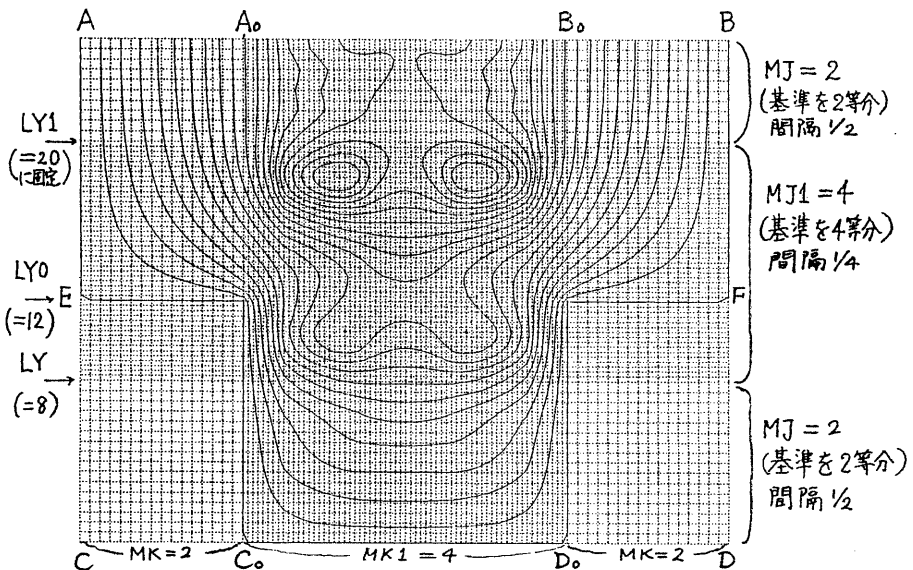


図4 MJ, MJ1, MK, MK1 = 2, 4, 2, 4, LY = 8, 切欠きパラメータ LY0 = 12, AA0 = 8 の例。

3 解法の大筋

CV法(コントロールボリューム法)によって空間離散化を行い、あみ目上の u 値(温度)の離散変数 $w(t) = (w_1(t), \dots, w_n(t))^T$ に関する常微分方程式

$$C w'(t) + A^{(w)} w(t) = f(t) \quad (2)$$

が得られると考え、それから台形法による $w(t)$ の時間的離散値 $w^{(\nu)}$ に関する非線形方程式

$$\left[C + \frac{\tau_\nu}{2} A^{(\nu)} \right] w^{(\nu)} = \left[C - \frac{\tau_\nu}{2} A^{(\nu-1)} \right] w^{(\nu-1)} + \tau_\nu f^{(\nu-\frac{1}{2})} \quad (3)$$

を考え、 $w^{(\nu-1)} \rightarrow w^{(\nu)}$ のために次の反復スキームを入れる:

```

b = [ C - \frac{\tau_\nu}{2} A^{(\nu-1)} ] w^{(\nu-1)} + \tau_\nu f^{(\nu-\frac{1}{2})} とおき、
w_0 = w^{(\nu-1)}; A = A^{(\nu-1)}; \Delta = b - [ C + \frac{\tau_\nu}{2} A ] w_0 から始めて、
do i = 1, niter (niter は 16 が標準)
  if (i \le 2) then
    J (= C + \frac{\tau_\nu}{2} A) の不完全LU
    solve J \Delta w = \Delta (MICCG(1,2) による)
    w_i = w_{i-1} + \Delta w
    make (new) A = A^{(w_i)}
    \Delta = b - [ C + \frac{\tau_\nu}{2} A ] w ; cnr = \| \Delta \|_2
    if (cnr < epsn) then
      if (\| \Delta w \|_\infty < epsm) go to exit
exit
    
```

— 精しくは文献(1)を見よ。プログラムがのっている。

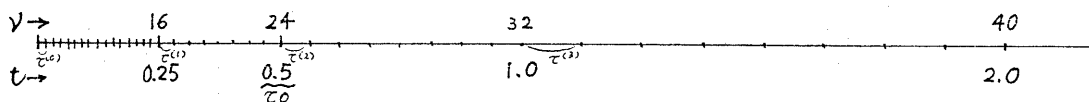
($epsn = 0.4D-4$, $epsm = 0.4D-3$, MICCG(1,2)の収束判定は

$$\| \text{残差} \|_2 < (epsn/16.0) * \| b \|_2 \text{ によった })$$

τ_ν のパターンについて: $\tau^{(0)} = \tau_0 / (16 \times KF)$, $n_0 = 8 \times KF$ から始めて、

$$n_1 = 4 \times KF \text{ おきに } \tau^{(\nu)} = 2 \times \tau^{(\nu-1)}, \nu = 2, 3, \dots$$

の形で漸増させる: たとえば $KF = 2$, $\tau_0 = 0.5$ ($n_0 = 8 \times 2$, $n_1 = 4 \times 2$, $\tau^{(0)} = 0.5 / (16 \times 2)$) の場合、下図のようになる: (KF と τ_0 が $\{\tau_\nu\}$ を決める)



非線形性の与え方: 拡散係数 $k = k(x, y, u)$ の u 依存性を、各要素矩形毎に、

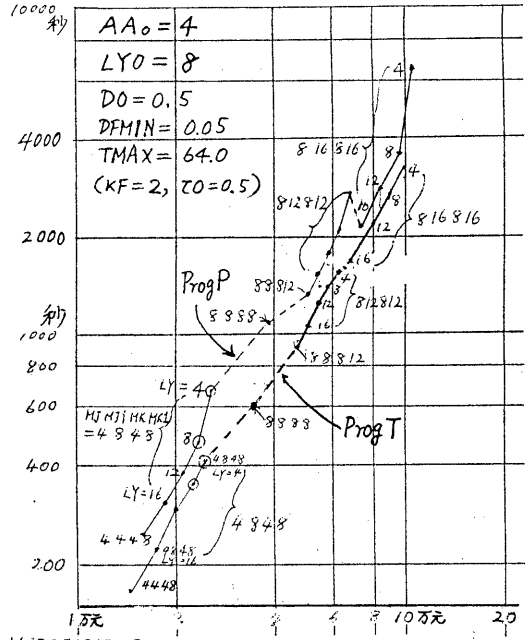
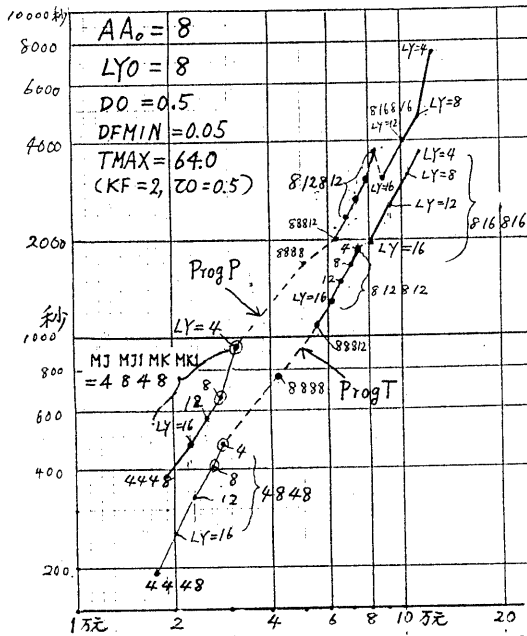
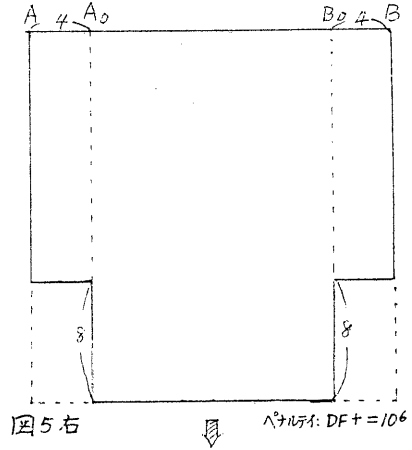
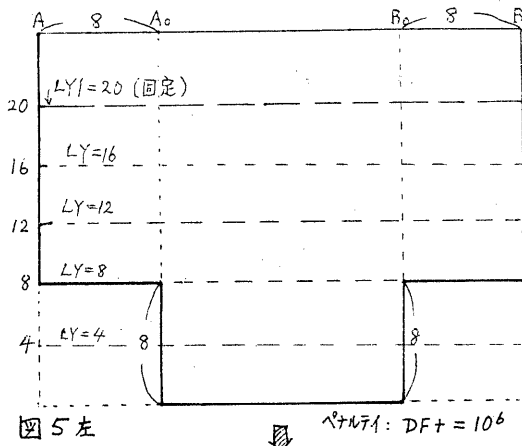
$$k(u) = k_0(u) \times DF; \quad k_0(u) = \max(1 + D_0 \times u, DFMIN)$$

によって与える。DFは要素毎に初めに与えた拡散係数、 D_0 とDFMINがパラメータ。

4 取得したデータとコメント:

アクティブな節点にだけ一連番号づけを行って離散化したプログラムを ProgT とし、ペナルティ法によって離散化したプログラムを ProgP と名付ける。
元数 N を横軸にとり、両者による CPU タイムを縦軸にプロットしてみる:

[ProgP の方が、同じ問題に対する N の値が ProgT のときより大きいのはやむを得ないが、それ以上に CPU タイムが大きい。原因は 1 回の solve 当りの MICCG 反復回数が増すことにある。下表参照]



CPU タイムは HITAC M680IAP による。

表 1 MJ, MJ1, MK, MK1 = 4, 8, 4, 8 ; IN は総準ニュートン反復, 平均 IC は平均 MICCG 反復回数。

LY	N (P/f)	Prog P		Prog T	
		sec (")	IN, 平均 IC	sec (")	IN, 平均 IC
8	N (P/f)	28268 (1.08)	26220	28220	26220
	sec (")	652" (1.61)	406"	406"	406"
	IN, 平均 IC	492回, 17.0回	492回, 10.9回	492回, 10.9回	492回, 10.9回
4	N (P/f)	31824 (1.11)	28252	28252	28252
	sec (")	935" (1.98)	472"	472"	472"
	IN, 平均 IC	493回, 22.4回	492回, 11.9回	492回, 11.9回	492回, 11.9回

LY	N (P/f)	Prog P		Prog T	
		sec (")	IN, 平均 IC	sec (")	IN, 平均 IC
8	N (P/f)	23532 (1.05)	22058	23532	22058
	sec (")	470" (1.33)	353"	470"	353"
	IN, 平均 IC	476回, 15.1回	476回, 11.6回	476回, 11.6回	476回, 11.6回
4	N (P/f)	26076 (1.06)	24540	26076	24540
	sec (")	691" (1.69)	409"	691"	409"
	IN, 平均 IC	477回, 20.5回	477回, 12.3回	477回, 12.3回	477回, 12.3回

Prog TとProg Pの、表1右上段の場合; すなわち $AA_0=4$, $LY=8$ の場合の初期行列 A の λ_{max} , λ_{min} , $Cond(A) = \lambda_{max}/\lambda_{min}$ を測ったものを表2に示す。

これは、常微分方程式の硬さ比にほかならないが、Prog Pの硬さ比が甚だ大きいけれども、別に障害はない。

MICCG法の反復回数を問題にするとき、行列の条件数がしばしば引合に出されるので、今の場合の行列 $C + \frac{T}{2}A$ の条件数を調べた。表3。

反復回数のProg P/Prog T比^(*)は、表1の場合、最大で約2である。表3での条件数の違いは甚だしいにもかかわらず。MIC(1.2)の前処理で、そのちがいがどうなるかに興味もたれるが、それでは説明できるとは思えない。

MICCGの反復回数のP/T比をもう少し詳しく調べたものを図6と表4に示す。図6の太線のものは、表2のものと同じく $DO=0.5$ (強非線形) のものである ($AA_0=4$, $LY=8$)。図6の見方: 例えば $\tau=8.0$ にて、1.62とあるが、それは $4.0 < \tau \leq 8.0$ におけるMICCG反復のP/T比が1.62ということを示している。

細線は $DO=0.25$ (弱非線形) としたときのもので、他の条件は同じである。MICCG反復の回数そのものは両者でかなりちがうが、P/T比は大体似た傾向を示している。

表4は、 $DO=0.5$ の場合の、各時刻 τ までのMICCG反復、準ニュートン反復の累積回数を示している。

ソースを簡単に、上段の-3,-3としたところ2ヶ所だけにし、拡散係数も $Df=1.0$ (all) にし (急冷を起させるため、-3の熱源は-4に直して) 測ったP/T比を、図6に点線で書きこんだ。

(*) 以後これをP/T比と略称する。

表2 Aの条件数

	N(元数)	λ_{max}	λ_{min}	Cond.
Prog T	22058	098D+1	0.28D-3	3.44D+4
Prog P	23532	0.77D+7	0.28D-3	2.73D+10

表3 $C + \frac{T}{2}A$ の条件数 (Df=all 1.0のもの)

	($KF=2$ $DO=0.5$)	λ_{max}	λ_{min}	Cond
Prog T	T=4にて	1.20D0	0.11D-2	1.12D+3
	T=16にて	4.83D0	0.12D-2	3.94D+3
Prog P	T=4にて	9.73D+5	0.11D-2	9.03D+8
	T=16にて	3.88D+6	0.12D-2	3.16D+9

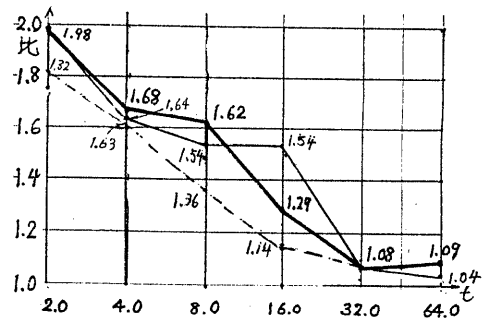


図6 ICCG反復のP/T比

表4 ICCG反復、準ニュートン反復の累積値

$DO=0.5$ τ ↓	Prog P		Prog T	
	ICCG 反復(回)	準ニュートン 反復(回)	ICCG 反復(回)	準ニュートン 反復(回)
2.0	1313	96	661	46
4.0	1833	125	970	125
8.0	2653	171	1475	171
16.0	3874	246	2420	246
32.0	5125	328	3583	328
64.0	7216	477	5502	476

以上、ペナルティ法するときMICCG反復回数が増す状況を見てきた。何故増すかについては、定性的には多少は分かるが、定量的理解からは甚だ遠い。

もうひとつ、(これも理由はわからないが)時間間隔 τ を小さくすると、 P/T 比が上昇の傾向を示す。その状況を図7に示しておく。(今回は、精度0.1%程度をめじとしたから、 $\tau_0 = 0.5$ 、 $KF = 2$ で十分であるが、もっと高精度を要求するときは、“ペナルティーが大きくなることを覚悟せねばならない”)

図7は、ソースは単純に冷すだけ、DF値はall 1.0 という簡単な場合のものである。

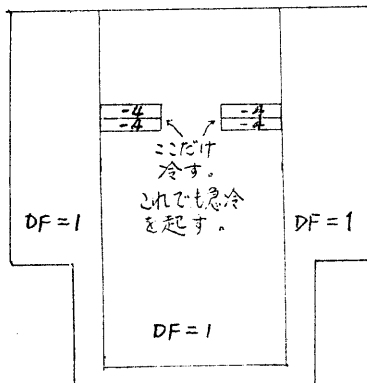


図7の場合。

5 ペナルティ法の適用法について

以上、ペナルティ法は、問題はあっても(特に原因がよく分からないので不安ではあるが)便利である。たとえば、ここで示したような切欠きのある場に対して、いきなりProgTを作ってそれが意図通り正しいかどうかをチェックしようとする

と大変である。予じめProgPを作って、それで動かして、のちProgTを作って、ProgPと照合する方が断然楽である。実際、筆者はそうしている。ProgTの、離散化プログラムは意外にめんどうである。また、MICCGのプログラムも、馴れていけば離散化ほどではないが、けっこう長い。

また、場の大筋はProgTで作ってのち、場の微調整にペナルティ技法を使うぶんには、CPUタイムの損失も上記とくらべてごく僅かなものに収まることが多からう。実際そのような応用例を以下に示そう。図8aに示すような、内側に、テーパのついた切欠きを入れた場を手軽に作成するために、図8bに示すように、ごく狭い細長の4本足の“ひも状の”部分に $DF = 10^6$ の拡散係数を入れて試みた。すなわち、場の微調整用にだけペナルティ技法を使った。これで図8a相当の場をうまく近似できる筈である。

図8の場の、微調整を入れる前の、 $MJ, MJ1, MK, MK1 = 4, 8, 4, 8$ のあみ目

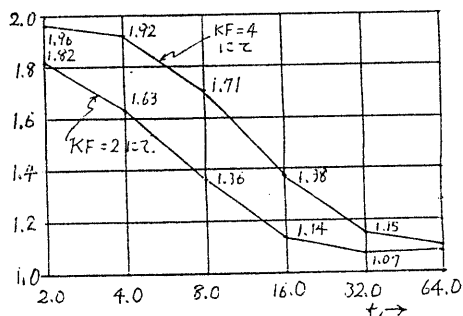


図7 KF=2のときとKF=4のときのP/T比; 図の見方は図6と同じ。

によって計算したときの総MICCG反復回数は、 $T=64$ にて 5502回であった。それが、微調整を入れると 5547回となった。準ニュートン反復回数はどちらも 478回で、HITAC-M680 IAPによるCPUタイムは前者が 363秒、後者が 366秒であった。この程度なら問題ない。(この図8aの場を、アクティブな点だけによる離散化を行って解いたとしても、元数が 22508元が僅かに減って 21996 になるだけゆえ 3%程度速くなるだけである)。

むしろ図8aのように斜線(近似的ではあるが)を入れた場を作るか作らないかのほうが実際のシミュレーションにおいては重要である。実際図8aの場で最低温度 -8.0094 を示したのに対し、斜線でなく直角に切欠いたもの(図8b点線)では -7.8965 となった。1.5%の相異である。LY0=12, LY=12 で同じ実験をすると -5.9660 対 -5.7664 , これは 3.5%の相異である。

6 おわりに

ペナルティ法は、図8に示した程度の切欠き全体に応用すると、MICCG反復が正直に離散化したときとくらべてかなり増し、CPUタイムにして 70%程度の増になる。

図8bに示したような、場の微調整用に使うぶんには(3%程度ゆえ)差しつかえない。場の微調整は、シミュレーションの上で大切に、それを楽に実行できる点で、ペナルティ技法の利便は大きいと言える。

文献 (1) 村田: 時間依存の非線形拡散方程式の数値解法, 応用数理 SEP. 1992.

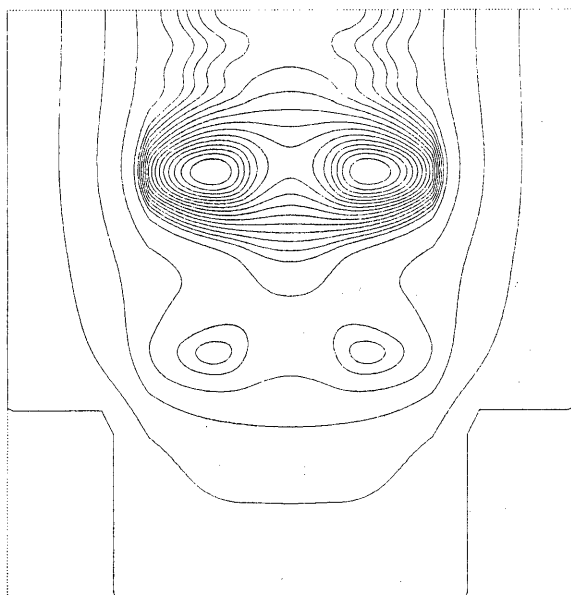


図8a 斜線(テーパー)入り切り欠きのある場。

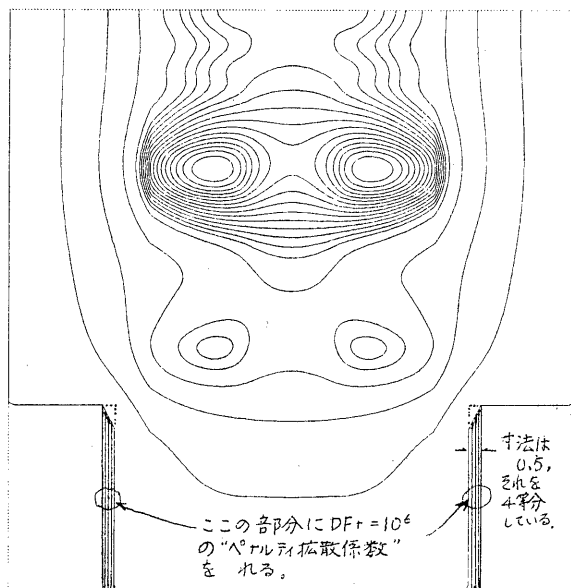


図8b 図8aの場をペナルティ法による微調整によって作る, 説明図