

スーパーコン 25 予選・認定問題: 地震活動推定問題

1 地震活動推定問題

地震活動の多くはゆっくりとしたプレートの運動によって岩盤にひずみが蓄積され、それが限界に達して短時間に一気に放出されることによって起こる。その簡単なモデルを考えよう^{*1}。まず概要を説明する。図 1 のような大きさ $L \times L$ の格子があり、この上での地震活動を考える。ある時刻 t において格子点 (x, y) に蓄積された「ひずみ」の大きさを $S(x, y, t)$ とする。ここで $x = 1, 2, \dots, L$ 、 $y = 1, 2, \dots, L$ である。時間 t とともにひずみの大きさは一定の割合で増加してゆき、ある格子点でのひずみの大きさが臨界値 S_c を超えると、その格子点を震源とした地震が始まる。これによってその格子点でのひずみが解消されるだけでなく、その周囲の格子点のひずみが増大する。これによって地震が連鎖的に起こる可能性がある。

詳しくモデルを定義しよう。簡単のためひずみは整数値 $S = 0, 1, 2, \dots$ を取り、時間も $t = 0, 1, 2, \dots$ のように整数値をとるとする。これらは本来連続的な値をとり、固有の次元を持った量であるが、それぞれある適当な単位で測った値を整数値で近似して表していると考えて欲しい。地震が起きていないとき、ひずみは時間 t が 1 進むごとに 1 増大する。ある時刻 t_0 におけるひずみの最大値、すなわち全ての格子点の中でのひずみの最大値が $S_{\max}$ であるとする。これが臨界値 S_c 未満であれば、時間 t が 1 進むごとに全ての格子点のひずみが 1 増大する。したがって、時刻 $t_0 + (S_c - S_{\max})$ においてひずみの最大値が S_c になり地震が始まる。地震の過程は非常に速く（プレートによってひずみが増大する速さに比べて）で起こり、その時間は無視できるとする。

では地震の過程を定義しよう。ある瞬間においても複数の場所（格子点）でひずみが丁度 S_c あるいはそれを上回っていれば、それらを震源とした地震が同時に起こる。その個々の地震は直前に震源にあったひずみ S を解消して 0 にするが、その震源に隣接する 4 つの格子点でのひずみを増大させる。この地震によって全ての格子点 (x, y) のひずみが次のように一斉に更新される。

$$S(x, y, t) \rightarrow \begin{cases} \Delta S & (x, y) \text{ が震源である場合} \\ S(x, y, t) + \Delta S & (x, y, t) \text{ が震源でない場合} \end{cases} \quad (1)$$

^{*1} きわめてゆっくりと起こるスロー地震が最近注目されているがここでは考えないことにする。

ここで ΔS は同時に周囲で起こっている地震からの寄与の和で

$$\Delta S = \sum_{(i),(ii),(iii) \text{ を満たす } (x',y')} \text{floor}(\alpha \times S(x', y', t)) \quad (2)$$

ただし $\sum_{(i),(ii),(iii) \text{ を満たす } (x',y')}$ は、次の 3 つの条件を満たす (x', y') だけに関する和である。

- (i) $(x+1, y), (x, y+1), (x-1, y), (x, y-1)$ のいずれか
- (ii) 格子 $(x = 1, 2, \dots, L)(y = 1, 2, \dots, L)$ の内部に属し、外部にはみ出していない
- (iii) この瞬間同時に震源になっている

ここで (1)(2) 式の右辺にある $S(\dots)$ は地震が起こる直前のひずみの値、 α は問題で指定される正の定数で、 $\text{floor}(x)$ は実数値に対して小数点以下を切り捨てて整数にする関数である。これによって更新後もひずみの値は整数値になる。(1)(2) による一斉更新の結果、ひずみが丁度 S_c あるいはそれを上回った格子点の一つまたは複数生ずる場合があれば、それらから次の地震が同時に始まる。以上のプロセスを繰り返し、全ての格子点でのひずみが S_c 未満になるまで連鎖反応が続く。一連の地震が終われば、その後、次の地震が始まるまで前述のように時間 t が 1 進むごとに全ての格子点でひずみが 1 ずつ増加する。

注意：一連の地震の連鎖反応は非常に短時間で起こり、その経過時間は無視するが、上で説明した各地震の「順序」(因果関係) は大事である。

ある時刻 t_0 での全ての格子点でのひずみ $S(x, y, t_0)$ がわかればその後の任意の時間 t のひずみ $S(x, y, t)$ を求めることができる。(3,2 級認定問題) 逆に、次のような問題も考えられる。ある時刻 t におけるひずみ $S(x, y, t)$ が与えられる。その情報からそれよりも前のある時間 t_0 における $S(x, y, t_0)$ を推定してみよう。(1 級認定問題兼予選問題)

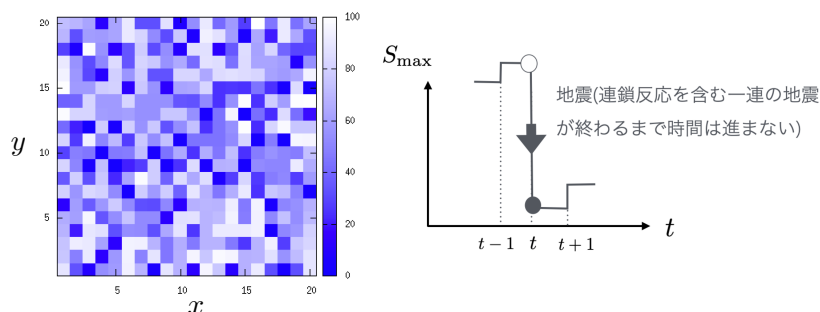


図 1 ある時刻 t におけるひずみ $S(x, y, t)$ の例 (左). ひずみの最大値の時間変化 (右).

2 問題

2.1 SuperCon 3 級認定問題

ひずみの初期値 $S_0[I][J]$ ($I = 1, 2, \dots, L, J = 1, 2, \dots, L$) が問題として与えられる。その最大値はすでに S_c になっている。この状態から地震が始まり、全ての格子点でのひずみが S_c 未満になると地震が終了する。地震がその連鎖反応を含めて終了した時点で残されるひずみ $S_f[I][J]$ ($I = 1, 2, \dots, L, J = 1, 2, \dots, L$) を求めるプログラムを作れ。

入力

標準入力から以下の形式の入力ファイルを読み込む。このための関数はあとで説明するように提供するヘッダーファイルに含まれる。

```
S0[1][1]  
S0[2][1]  
⋮  
S0[L][L]
```

出力

標準出力に以下の形式のファイルを出力する。このための関数もあとで説明するように提供されるヘッダーファイルに含まれる。

```
Sf[1][1]  
Sf[2][1]  
⋮  
Sf[L][L]  
入力読み込み終了からの経過時間 (秒)
```

2.1.1 3 級認定問題詳細

ヘッダーファイル `sc3.h` に必要な関数と配列が定義されているので,

```
#include "sc3.h"
```

として使うこと。また格子の大きさなどのパラメータの値も定義されているものを使うこと。定義されているものは以下である。

```
void SC_input() : 問題を標準入力から読み込む関数
void SC_output() : 解答を標準出力に出力する関数
# define SC_L 10 : 格子の x および y 軸方向の長さ L
# define SC_SC 100: ひずみの臨界値 Sc
# define SC_alpha 0.20 :  $\alpha$  の値 ((2) 式参照)
int SC_S0[SC_L+1][SC_L+1]: 問題のひずみの初期値を格納する 2 次元整数配列
int SC_Sf[SC_L+1][SC_L+1] : 解答を格納する 2 次元整数配列
```

2.1.2 3 級認定基準

1 問を解くプログラムを作成して応募する。審査では 10 問の問題 (すなわち上で説明した入力ファイルが 10 個) が用意され, 問題を変えてプログラムを 10 回実行する。制限時間内にすべてに正解した場合に 3 級を認定する。制限時間は入出力を除く計算時間で 10 問合計 1 分とする。

2.2 SuperCon 2 級認定問題

時刻 0 でのひずみの値 $S_0[I][J]$ ($I = 1, 2, \dots, L$, $J = 1, 2, \dots, L$) が問題として与えられる。ただし時刻 $t = 0$ で地震が起こる場合はそれが連鎖反応も含めて全て終了した状態でのひずみが与えられる。問題として与えられる時刻 $t(> 0)$ におけるひずみ $S_f[I][J]$ ($I = 1, 2, \dots, L$, $J = 1, 2, \dots, L$) を求めるプログラムを作れ。ただし時刻 t で地震が起こる場合はそれが連鎖反応も含めて全て終了した状態でのひずみを求めよ。

入力

標準入力から以下の形式の入力ファイルを読み込む。このための関数はあとで説明するように提供するヘッダーファイルに含まれる。

```
t  
S0[1][1]  
S0[2][1]  
⋮  
S0[L][L]
```

出力

標準出力に以下の形式のファイルを出力する。このための関数もあとで説明するように提供されるヘッダーファイルに含まれる。

```
Sf[1][1]  
Sf[2][1]  
⋮  
Sf[L][L]  
入力読み込み終了からの経過時間 (秒)
```

2.2.1 2 級認定問題詳細

ヘッダーファイル `sc2.h` に必要な関数と配列が定義されているので,

```
#include "sc2.h"
```

として使うこと. 定義されているものは以下である。

```
void SC_input() : 問題を標準入力から読み込む関数
void SC_output() : 解答を標準出力に出力する関数
# define SC_L 10 : 格子の x および y 軸方向の長さ L
# define SC_SC 100: ひずみの臨界値  $Sc$ 
# define SC_alpha 0.20 :  $\alpha$  の値 ((2) 式参照)
int SC_t: 問題の時刻  $t$  を格納する整数
int SC_S0[SC_L+1][SC_L+1]: 問題のひずみの初期値を格納する 2 次元整数配列
int SC_Sf[SC_L+1][SC_L+1] : 解答を格納する 2 次元整数配列
```

2.2.2 2 級認定基準

1 問を解くプログラムを作成して応募する。審査では 10 問の問題 (すなわち上で説明した入力ファイルが 10 個) が用意され, 問題を変えてプログラムを 10 回実行する。制限時間内にすべてに正解した場合に 2 級を認定する。制限時間は入出力を除く計算時間で 10 問合計 1 分とする。

2.3 SuperCon 1 級認定問題兼予選問題

問題としてある時刻 t でのひずみの値 $S_f[I][J](I = 1, 2, \dots, L, J = 1, 2, \dots, L)$ が与えられる。ただし時刻 t で地震が起こる場合はそれが連鎖反応も含めて全て終了した状態でのひずみが与えられる。また時刻 $t = 0$ から時刻 t までに起こった地震の回数 N_e も情報として与えられる。ただし、 N_e の定義においては同時刻に起こっている地震は離れた場所にあっても同じ地震とみなして数え、一つの地震をその連鎖反応も含めて一回と数える。これらの情報に基づいて時刻 $t = 0$ でのひずみの値 $S_0[I][J](I = 1, 2, \dots, L, J = 1, 2, \dots, L)$ を推定するプログラムを作れ。ただし時刻 $t = 0$ で地震が起こる場合はそれが連鎖反応も含めて全て終了した状態のものを求めよ。

推定の良し悪しは次のように評価する。実際に $S_0[I][J](I = 1, 2, \dots, L, J = 1, 2, \dots, L)$ を初期値として時間 t だけ時間発展させる。時刻 t における全ての地震が終了した後のひずみ $S[I][J](I = 1, 2, \dots, L, J = 1, 2, \dots, L)$ とする。また時刻 $t = 0$ から時刻 t までに起こる地震の回数を「地震の回数」とする。これらを用いた評価関数

$$E = \frac{1}{L^2} \sum_{I=1}^L \sum_{J=1}^L |S[I][J] - S_f[I][J] + A| \text{地震の回数} - N_e|$$

の大きさに推定の良し悪しを評価する。ここで $|\dots|$ は絶対値を表す。第 2 項で A はパラメータで正の定数である。

入力

標準入力から以下の形式の入力ファイルを読み込む。このための関数はあとで説明するように提供するヘッダーファイルに含まれる。

```
 $N_e$   
 $S_f[1][1]$   
 $S_f[2][1]$   
 $\vdots$   
 $S_f[L][L]$ 
```

出力

標準出力に以下の形式のファイルを出力する。このための関数もあとで説明するように提供されるヘッダーファイルに含まれる。

```
 $S_0[1][1]$   
 $S_0[2][1]$   
 $\vdots$   
 $S_0[L][L]$   
入力読み込み終了からの経過時間 (秒)
```

出力するひずみに対する条件

どの I, J においても $S_0[I][J]$ は 0 以上 S_c 以下の正の整数とする。これを満たさない出力は不適である。

2.3.1 1 級認定問題詳細

ヘッダーファイル `sc1.h` に必要な関数と配列が定義されているので、

```
#include "sc1.h"
```

として使うこと。定義されているものは以下である。

```
void SC_input() : 問題を標準入力から読み込む関数  
void SC_output() : 解答を標準出力に出力する関数  
# define SC_L 20 : 格子の x および y 軸方向の長さ L  
# define SC_SC 100: ひずみの臨界値  $S_c$ 
```

```
# define SC_alpha 0.20 :  $\alpha$  の値 ((2) 式参照)
# define SC_t 10 : 下にあたえられるひずみの時刻 t
# define SC_A 10.0 : 評価関数 E のパラメータ A
int SC_Ne: 問題の地震の回数 Ne を格納する整数
int SC_Sf[SC_L+1][SC_L+1]: 問題の時刻 t でのひずみを格納する 2 次元整数配列
int SC_S0[SC_L+1][SC_L+1] : 解答を格納する 2 次元整数配列
```

2.3.2 1 級認定基準

1 問を解くプログラムを作成して応募する。審査では 10 問の問題 (すなわち上で説明した入力ファイルが 10 個) が用意され, 問題を変えてプログラムを 10 回実行する。制限時間内にすべての問題について結果が出力され, 評価関数 E の合計が 200 以下である場合に 1 級を認定する。制限時間は入出力を除いて 10 問合計 150 秒とする。

2.3.3 本選出場チーム選考基準

問題は 10 問与えられる。評価関数 E の合計によって小さい順にランキングする。上位 20 チームを本選出場チームとする。制限時間は入出力を除いて 10 問合計 150 秒とする。以上の基準でもし 20 チーム決まらなければ, 問題数を増やして (制限時間もそれに比例して増やす) 順位を決定する。成績がほとんど変わらない時には, レポートを参考に, アルゴリズム上の工夫が見られたチームを優先する。なお, 本選出場と 1 級認定の基準は違うので, 本選出場チームでも 1 級認定が与えられない場合がある。

3 言語と実行環境その他の注意

- 言語: C++ を使用すること。
- OS: 審査は macOS Sonoma 14.6.1, CPU: プロセッサ Apple M1, メモリ 16GB を搭載したコンピュータ上でこなう。
- コンパイラは Homebrew GCC 14.2.0.1 を使用し, 以下のように `-O2` オプションでコンパイルする。

```
g++-14 -O2 sc3.cpp
```
- 出力に関する注意: 実行時に問題は標準入力から読まれ, 解答は標準出力に出力される。ヘッダーファイルに定義された入力, 出力用の関数を用いること。すべての級で, `SC_input` 関数は必ずプログラムの最初の実行文であること。つまり,

SC_input の前に実行文を書いてはならない。入出力は SC_input と SC_output のみで行ない、他の出力をプログラムに含めてはならない

- 計算時間は time.h に含まれる clock() 関数で測定する。入力終了後に測定を開始し、出力直前までの所要時間を測る。また、開始時刻と終了時刻を表す変数 SC_starttime と SC_endtime の値はプログラム中で参照してよいが、変えてはならない。なお、ヘッダーファイルは改変してはならない。これらの注意に反するプログラムは失格とする。

4 提出方法

ひとつの級にだけ応募してもよいし、複数の級に応募してもよい。作成したプログラムはファイル名を

1 級は sc1.cpp

2 級は sc2.cpp

3 級は sc3.cpp

として、以下のファイルと一緒に zip ファイルにまとめて Google フォーム「SuperCon2025 予選・級認定申込」<https://forms.gle/5ZSQvzsECLmzL8za6> から提出する。zip ファイルの名前は”〈チーム名〉.zip”にすること。整理の都合上、予選に応募しない場合も (ひとりでも) チーム名はつけること。なお、チーム名は 8 文字以内の英数字とします。ただし、1 文字目は英字でなければなりません。

zip ファイルにまとめて提出するものは以下の 3 つである。

- 提出プログラム: sc1.cpp, sc2.cpp, sc3.cpp のうち応募する級に対応するもの。予選応募には sc1.cpp が必須である。
- プログラム実行環境の説明 (下で説明するサンプル問題の一つに対する実行結果と計算時間、使用コンパイラなど) ”〈チーム名〉-env.txt”
- アルゴリズムのポイントや工夫などをまとめたレポート: ファイル名はテキストファイルの場合は”〈チーム名〉-algo.txt”, word ファイルの場合は”〈チーム名〉-algo.docx”, pdf ファイルの場合は”〈チーム名〉-algo.pdf”

回答期限までは回答の再編集およびファイルの再提出は可能である。ただし過度な頻度でのファイル再提出は控えて欲しい。

級認定のみの応募はひとりでも構わないが、SuperCon 予選に応募する場合には 2 ない

し 3 名のチームを作ること（予選通過後のチーム構成変更は認められないので注意）。1 級応募者は何も書かれていなければ予選参加とみなすので、級認定のみ希望の場合はその旨上記 Google フォームで明記すること。

5 注意事項

問題解法を他のチーム（学校が同じか違うかにかかわらず）と相談してはならない。実施委員会としてはこのことを監督する手段はないが、厳守をお願いする。また、募集締め切りまでは、問題解法に関して SNS やブログ等、チームメンバー以外の目に触れる場所に投稿することを禁止する。

6 問題のサンプル

問題のサンプルをウェブサイトにて公開する。