7.　勾配法のプログラミング

田中雅博

最適化プログラミング

多変数関数の極値を数値的に求める代表的な手法その１。

1　１変数の関数

$f = f(x)$ の最大値を求める問題を考える。
解析的に求めるには、$f'(x) = 0$ になるような x を求める必要がある。

1.1　例 1

$$f(x) = -x^2 - 3x + 6$$

これを解くために、

$$f'(x) = -2x - 3 = 0$$

とすると、$x = -\frac{3}{2}$ を得る。このとき、極値（$f''(x) = -2$ より、極大値）は、

$$f(-\frac{3}{2}) = \frac{33}{4}$$

である。

1.2　例 2

では、

$$f(x) = \sin 4x - x^2 + 2x$$

ではどうだろうか。

$$f'(x) = 4\cos 4x - 2x + 2 = 0$$

を解かなければならないが、きれいな形では出てこない。こういう問題での解を数値的に求める（式を求めるのではなく、解となる x および $f(x)$ の値を求める）のがここのテーマである。

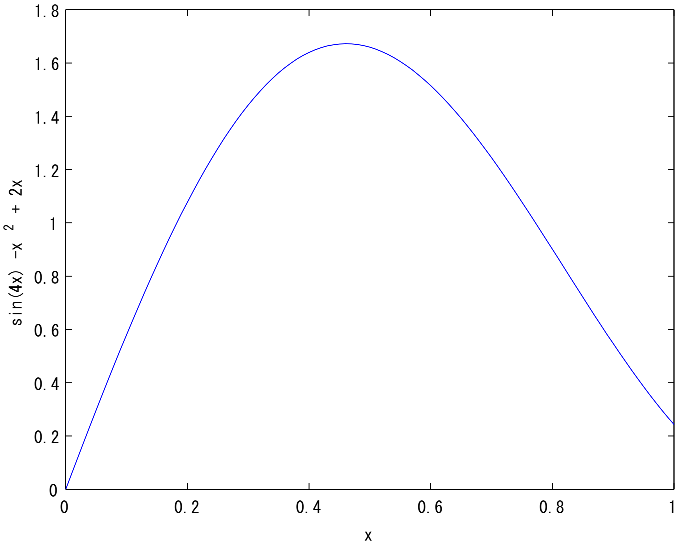

図 1: 関数 $f(x)$ のイメージ

1.3　考え方

関数 $f(x)$ をプロットすると 1 のようになる。1 変数の場合はこれを見て、集中的に良さそうなところを計算してみることもできるが、多変数の場合はそういうことができないので、以下はそのことを想定したアルゴリズムである。

勾配法とは、「関数 $f(x)$ は考えている領域で $f'(x) = 0$ となる x が 1 つしかない」ということがわかっているなら、次のようにして数値的に計算できる。（勾配法の詳しい説明は、専門書（例えば、シラバスに挙げている、**金谷「これならわかる最適化数学　基礎原理から計算手法まで」、共立出版**を参照のこと）

いま、$0 \leq x \leq 1$ で極値は 1 つしかないから、その範囲では勾配法が適用できる。

<u>極大点探索アルゴリズム</u>

1. x の初期値を与え、$h \leftarrow h_0$ とする。
2. 次のように置く。
$$h \leftarrow \mathrm{sgn}(f'(x))|h|, \quad X \leftarrow x, \quad X' \leftarrow x + h$$
$\mathrm{sgn}(\cdot)$ は、括弧内の値の符号が正なら 1、負なら −1。つまり、ここでは、$f'(x) > 0$ なら h の値を正、$f'(x) < 0$ なら h の値を負にする。大きさは個々では変えない。
X は現在の値、X' は h ほど変化させた値。
3. もし $f(x) < f(X')$ であれば（X' の方が良い場合）、次の計算を行う。
    (a) $f(x) \geq f(X')$ となるまで次の計算を繰り返す。
$$h \leftarrow 2h, \quad X \leftarrow X', \quad X' \leftarrow X + h$$

(b) $x \leftarrow X, h \leftarrow h/2$ と置く。

4. そうでなければ（X' の方が悪い場合)、次の計算を行う。

   (a) $f(X) \leq f(X')$ となるまで、次の計算を繰り返す。

$$h \leftarrow \frac{h}{2}, \quad X' \leftarrow X' - h$$

   (b) $x \leftarrow X', h \leftarrow 2h$ と置く。

5. ステップ 2 に戻り、これを $|f'(x)| \leq \epsilon$ となるまで繰り返す。

6. 得られた x を返す。

これの MATLAB プログラムを示そう。関数 $f(x)$ を `rei7(x)`、その導関数 $f'(x)$ を `rei7d(x)` とする。

```
function f= rei7( x )
f = sin(x * 4) - x.^2 + 2 * x;
end

function df = rei7d( x )
df = 4 * cos(x * 4) - 2 * x + 2;
end

function x = search(x0)
epsilon = 0.0001;
h0 = 0.1;
h = h0;
x = x0;
%Step 2
while abs(rei7d(x)) > epsilon
    h = sign(rei7d(x)) * abs(h); %MATLAB の符号関数は sign
    X = x;
    Xprime = x + h;
    if rei7(X) < rei7(Xprime)
        while rei7(X) < rei7(Xprime)
            h = 2 * h;
            X = Xprime; %(X' のこと)
            Xprime = X + h;
        end
        x = X;
        h = h / 2;
    else
        while rei7(X) > rei7(Xprime)
            h = h / 2;
            Xprime = Xprime - h;
```

```
        end
        x = Xprime;
        h = 2*h;
    end
end
end
```

計算する。

```
>> search(0.3)

ans =

    0.4609
```

最大値（極大値）は、

```
>> rei7(search(0.3))

ans =

    1.6724
```

と求まる。
この関数をもっと広い範囲で見てみよう。すると、

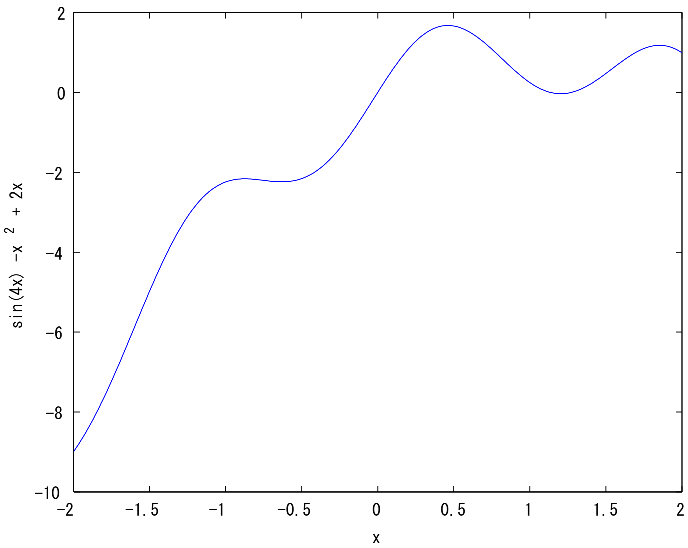

図 2: 関数 $f(x)$ のイメージ

これを見るとわかるように、極値はいくつかあり、それに従って、初期値によっては違うところに収束することが考えられる。アルゴリズムの特性上、必ずしもその初期点に対応する極大点に収束するとは限らず、別の極大点にいくこともある。

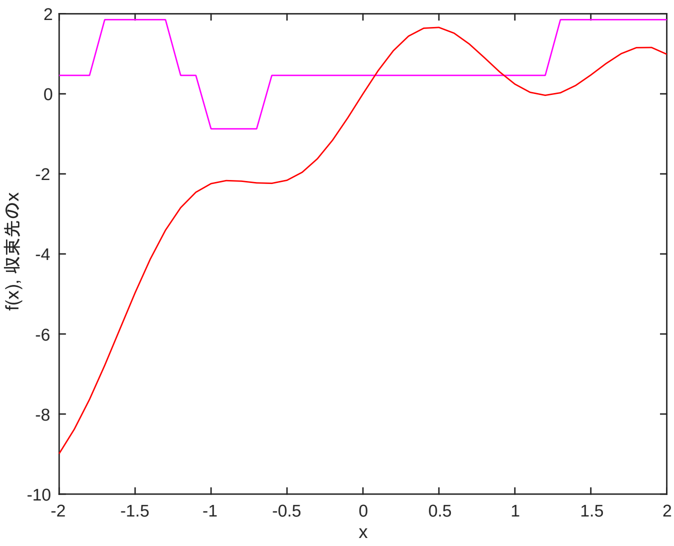

図 3: 描く図

課題

1. 区間 $[-5, 5]$ において関数 $f(x) = \sin 4x - x^2 + 2x$ の**最大化**を行う。
   (a) まず、f(x) の関数を赤でプロット (図 3 を参考にして）
   (b) 同じ図に出発点をごとに収束点を関数値とする線を描く（マゼンタ）
   (c) この区間における最大値を与える x と最大値 $f(x)$ を記載せよ。
2. 関数 $f(x) = (x - \cos x)^2$ の**最小化**を行う。最小化を行う際には、f(x) の符号を変えれば、最大化のアルゴリズムを用いることができる。最後に得られた結果 $f(x)$ の符号を変えればよい。

レポート作成について

問題１，２それぞれについて、プログラムリストと図、得られた結果を示せ。それらを、１つの Word ファイルにして、ファイル名 “学籍番号氏名 optprog7.doc” という名称のファイルに保存しておくこと（例　“11671140 甲南太郎 optprog7.doc”）。
また、後日ファイル提出の指示があったときに、すぐに送信できるようにしておくこと。本日提出する必要はない (採点時にタイムスタンプを見ることもある。解説後に書いたものとは区別して採点する。書き上げたのちに修正すると、その時刻が残るので修正しないこと（もし自分のために修正するのであれば、別のファイルを作ること)。

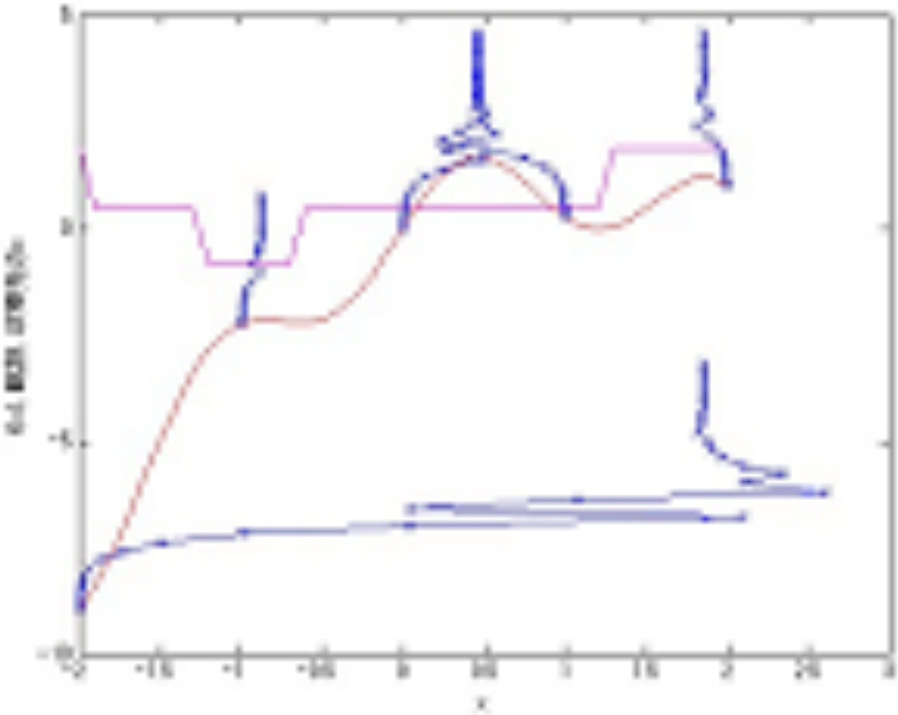

図 4: （参考）収束の様子