Soul Mπth

Jurnal Imiah Edukasi Matematika FKIP Universitas Dr. Soetomo

Vol. 2 ■ No. 3 ■ Agustus 2014 ■ Hal 107-169

ISSN 2337-9421

ISSN 2334-9421

JURNAL ILMIAH
"SOULMATH"
(Jurnal Edukasi Matematika)

Terbit dua kali setahun pada bulan Januari dan Agustus. Berisi tulisan yang berasal dari hasil penelitian, kajian, atau karya ilmiah di bidang Pendidikan Matematika

Penerbit

Fakultas Keguruan & Ilmu Pendidikan
Universitas Dr. Soetomo Surabaya

Alamat Penerbit:

Gedung C. 102 Universitas Dr. Soetomo Surabaya
Jalan Semolowaru 84 Surabaya 60118
Telp (031) 5944748

ISSN 2334-9421

JURNAL ILMIAH
"SOULMATH"
(Jurnal Edukasi Matematika)

Volume 2 Nomor 3, Agustus 2014
Halaman 107-159

IMPLEMENTASI ALGORITMA MODIFIKASI *BROYDEN-FLETCHER-GOLDFARB-SHANNO* (MBFGS)

Rahmawati Erma Standsyah
FKIP, Universitas Dr. Soetomo Surabaya

***Abstract:** BFGS algorithm is focused for unconstrained optimization with a convex objective function. BFGS can not be applied to non convex optimization, so that it is modified into MBFGS algorithm. It has been proven by theory of convergence but it has not been proven yet numerical method. Based on this problem, this paper is implemented and analyzed the numerical aspect of BFGS and MBFGS algorithm. This computation result showed that the effectiveness and the accuracy of MBFGS algorithm is more than BFGS for Banana function and one example of convex function.*

Key Words : *BFGS Algorithm , Unconstrained Optimization , Nonconvex Function*

Pendahuluan

Dalam menyelesaikan permasalahan optimasi di dunia nyata, pemodelan matematika yang diperoleh secara umum terdiri dari fungsi tujuan atau fungsi obyektif dan fungsi-fungsi kendala yang linear atau tidak linear. Fungsi obyektif sendiri terdiri dari dua tipe fungsi yaitu fungsi obyektif yang *convex* atau fungsi obyektif yang *non convex.* Banyak algoritma yang dapat digunakan untuk menyelesaikan masalah – masalah optimasi tersebut, misalnya algoritma *New Two-Point Stepsize Gradient* yang digunakan untuk menyelesaikan masalah optimasi tanpa kendala, algoritma BFGS untuk fungsi obyektif yang *convex* pada masalah optimasi tanpa kendala, algoritma *New Line Search* pada optimasi tanpa kendala dan banyak yang lainnya.

Pada optimasi tanpa kendala, algoritma Quasi Newton mengalami revolusi pada tahun 1960an untuk menghindari perhitungan yang rumit dari matriks Hessian. Sejak tahu 1970an, algoritma BFGS diterima sebag algoritma terbaik dari revolusi algoritma Qua Newton sehingga algoritma BFGS digunaka untuk menyelesaikan masalah optimasi tanp kendala. Hal ini disebabkan karena algoritm ini merupakan algoritma yang menguntungka dan hasil numeriknya secara teori lebih cep konvergen. Akibat hal tersebut algoritma i menjadi algoritma pilihan untuk insinyur d matematikawan yang tertarik dala menyelesaikan masalah optimasi.

Banyak algoritma BFGS yang dimodifika telah diusulkan selama bertahun- tahu misalnya tahun 1970 Powel menulis tenta algoritma baru untuk optimasi tanpa kenda tahun 1991 Davidon menulis algoritr *Variabel Metric* untuk meminimalk permasalahan optimasi, tahun 1999 Zhar Dhen dan Chen menulis tentang persama baru Quasi-Newton dan aplikasinya pa optimasi tanpa kendala, tahun 2001 Li dan Fukushima memodifikasi algoritma BFGS dan menganalisis global konvergensinya pada fungsi optimasi minimal yang *non convex* dan kembali menulis tentang global konvergensi algoritma BFGS untuk permasalahan optimasi yang *non convex* dan selanjutnya tahun 2004 Wei, Yu, Yuan dan Lian menulis tentang *Superlinear Convergence* pada algoritma modifikasi BFGS untuk optimasi tanpa kendala. Algoritma BFGS difokuskan untuk menyelesaikan optimasi tanpa kendala dengan fungsi obyektif yang *convex*. BFGS tidak dapat diterapkan pada optimasi *non convex*. Berdasarkan pemikiran tersebut maka Li dan Fukushima [Li, D dan Fukushima, M. 2001] membuat modifikasi pada algoritma BFGS yang dikenal dengan MBFGS. Secara teori konvergensi, MBFGS ini telah dibuktikan oleh Li dan Fukushima [Li, D dan Fukushima, M. 2001]. Yuan dan Lu [Yuan, G dan Lu, X. 2011] memanfaatkan MBFGS untuk menciptakan bentuk modifikasi baru, namun MBFGS belum diimplementasikan dan dibuktikan secara numerik oleh Yuan dan LU. Oleh karena itu maka dalam penulisan ini diimplementasikan dan dianalisis secara numerik algoritma BFGS dan MBFGS tersebut. Hasil yang diperoleh, menunjukkan keefektifan dan keakuratan algoritma MBFGS ini dibandingkan BFGS pada fungsi Banana dan salah satu contoh fungsi *convex* sehingga dapat dilanjutkan untuk menciptakan modifikasi yang baru [Yuan, G dan Lu,X. 2011].

Tinjauan Pustaka

Fungsi *Convex*

Fungsi *convex* didefinisikan sebagai berikut [Nocedal, J., Wright, S. 1999]:

$f: R^n \to R$ adalah *convex* jika domain f adalah himpunan *convex* dan $\forall\, x, y \in domain\, f$ dan $\theta\ dengan\ 0 \leq \theta \leq 1$ maka

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y)$$

(2.1)

Fungsi *convex* ini memiliki sifat sebagai berikut:

a. Penjumlahan dari fungsi *convex* adalah fungsi *convex*

b. Perkalian fungsi *convex* oleh skalar positif adalah fungsi *convex*

c. Fungsi linear $\sum_{i=1}^{k} a_i x_i$ adalah *convex* , dengan tanda positif untuk a_i

d. Fungsi linear $\sum_{i=1}^{k} a_i x_i$ adalah *concave* , dengan tanda negative untuk a_i

e. Jika f adalah *convex*, $-f$ adalah *concave*

f. Jika f adalah *concave*, $-f$ adalah *convex*

Optimasi Tanpa Kendala *(Unconstrained Optimazation)*

Fungsi optimasi tak linear tanpa kendala, secara umum menurut [Arora, J. 2004] memiliki bentuk:

$$f: R^n \to R$$

(2.2)

fungsi Obyektif atau fungsi tujuan:

$$\text{maks atau min } f(\bar{x}), \quad \bar{x} \in R^n$$

(2.3)

nilai awal:

$\bar{x}^{(0)} = (x_1, x_2, \cdots x_n)$

(2.4)

Algoritma Broyden-Fletcher-Goldfarb-Shanno (BFGS)

Dalam numerik optimasi, algoritma *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) adalah salah satu algoritma untuk menyelesaikan optimasi *nonlinear* tanpa kendala. Algoritma ini dikembangkan oleh *Broyden-Fletcher-Goldfarb-Shanno* dari algoritma Quasi-Newton. Persamaan umum quasi-Newton [Yuan, G dan Lu,X. 2011]:

$$B^{k+1}s^k = y^k$$

(2.5)

dengan

$$s^k = x^{k+1} - x^k$$

$$y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$$

sehingga matriks BFGS pembaruannya adalah

$$B^{k+1} = B^k - \frac{B^k s^k (s^k)^T B^k}{(s^k)^T B^k s^k} + \frac{y^k (y^k)^T}{(s^k)^T y^k}$$

(2.6)

dan algoritmanya sebagai berikut :

1. Tentukan nilai $x^0, \varepsilon \in (0,1)\ dan\ B^0$, jika B^0 tidak diketahui dapat menggunakan $B^0 = I$
2. Hitung $c^k = \nabla f(x^k)$ dengan $k = 0$ (menandakan iterasi)
3. Hitung norm dari $c^k\ yaitu\ \|c^k\|$
4. Uji jika $\|c^k\| < \varepsilon$ maka berhenti dan jika tidak maka lanjut ke tahap berikutnya
5. Mendapatkan nilai d^k dari persamaan *linear* $B^k d^k = -c^k$
6. Hitung nilai $\lambda^k = \lambda$ dari fungsi $f(x^k + \lambda^k d^k) = \min_{\lambda > 0} f(x^k + \lambda d^k)$
7. Hitung $x^{k+1} = x^k + \lambda d^k$
8. Hitung $s^k = x^{k+1} - x^k = \lambda d^k$, c^{k+1} $y^k = c^{k+1} - c^k$, $untuk\ menghitung$ matriks pembaruan BFGS:
$$B^{k+1} = B^k - \frac{B^k s^k (s^k)^T B^k}{(s^k)^T B^k s^k} + \frac{y^k (y^k)^T}{(s^k)^T y^k}$$
9. Kembali ke langkah 3 dengan k = k+1

Metode Penelitian

Studi Literatur

Pada tahap ini penulis mempelajari segala hal yang berkaitan dengan materi penelitian, antara lain mengenai fungsi *convex*, algoritma BFGS dan algoritma MBFGS

Implementasi Algoritma

Setelah mempelajari materi yang berkaitan, selanjutnya mengimplementasikan algoritma tersebut kedalam bahasa Matlab yang dapat diproses pada langkah selanjutnya.

Simulasi Hasil Implementasi

Hasil implementasi dari algoritma tersebut disimulasikan terlebih dahulu dengan menggunakan program simulink - Matlab 7.8.

Analisis Hasil Simulasi

Pada tahap ini penulis melakukan analisis terhadap hasil yang telah diperoleh dari simulasi yang selanjutnya ditarik kesimpulan dari hasil numerik yang didapat.

Analisis Dan Pembahasan

Algoritma Modifikasi *Broyden-Fletcher-Goldfarb-Shanno* (MBFGS)

Metode Quasi-Newton untuk permasalahan tak linear sering membutuhkan matriks pembaruan dari matriks B^k pada tiap iterasinya. Algoritma BFGS adalah salah satu algoritma terbaik dari Quasi Newton dalam pembaruan matriks B^k. Sifat konvergensi BFGS telah dikenal baik jika digunakan untuk meminimalkan masalah optimasi *convex*, namun tidak demikian jika digunakan untuk meminimalkan masalah optimasi *non convex.* Oleh karena itu Li dan Fukushima [Li, D dan Fukushima, M. 2001] memodifikasi persamaan Quasi Newton. Persamaan (2.5) $B^{k+1}s^k = y^k$ dimodifikasi menjadi

$$B^{k+1}s^k = (y^k)^{1*} \quad (4.1)$$

dengan

$$(y^k)^{1*} = y^k + t^k\|c^k\|s^k$$

$$c^k = \nabla f(x^k)$$

$$t^k = 1 + \max\left\{-\frac{(s^k)^T y^k}{\|s^k\|^2}, 0\right\} \text{ dengan } t^k > 0$$

sehingga matriks pembaruan BFGS termodifikasi menjadi

$$B^{k+1} = B^k - \frac{B^k s^k (s^k)^T B^k}{(s^k)^T B^k s^k} + \frac{(y^k)^{1*}((y^k)^{1*})^T}{(s^k)^T (y^k)^{1*}}$$

(4.2)

dan algoritma MBFGS dapat dijelaskan sebagai berikut:

1. Tentukan nilai $x^0, \varepsilon \in (0,1), \sigma \in (0,1), \rho \in (0,1) dan\ B^0$, jika B^0 tidak diketahui dapat menggunakan $B^0 = I$
2. Hitung $c^k = \nabla f(x^k)$ dengan $k = 0$ (menandakaan iterasi)
3. Hitung norm dari $c^k\ yaitu\ \|c^k\|$
4. Uji jika $\|c^k\| < \varepsilon$ maka berhenti dan jika tidak maka lanjut ke tahap berikutnya
5. Mendapatkan nilai d^k dari persamaan *linear* $B^k d^k = -c^k$
6. Mencari nilai integer j yang terendah yang memenuhi persamaan $f(x^k + \rho^j d^k) \leq f(x^k) + \sigma\rho^j (c^k)^T d^k$ dengan $\lambda^k = \rho^{j^k}$
7. Hitung $x^{k+1} = x^k + \lambda^k d^k$
8. Hitung $s^k = x^{k+1} - x^k = \lambda^k d^k$

   $$c^{k+1}$$

   $$y^k = c^{k+1} - c^k$$

   $$\|s^k\|$$

   $$t^k = 1 + max\left\{-\frac{(s^k)^T y^k}{\|s^k\|^2}, 0\right\}\ dengan\ t^k > 0$$

   $$(y^k)^{1*} = y^k + t^k\|c^k\|s^k$$

   untuk menghitung matriks pembaruan BFGS:

   $$B^{k+1} = B^k - \frac{B^k s^k (s^k)^T B^k}{(s^k)^T B^k s^k} + \frac{(y^k)^{1*}((y^k)^{1*})^T}{(s^k)^T (y^k)^{1*}}$$
9. Kembali ke langkah 3 dengan k = k+1

Fungsi Optimasi

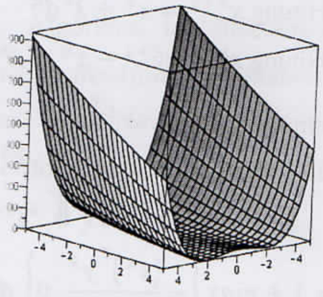
Fungsi optimasi yang digunakan dalam analisis hasil numerik pada permasalahan ini terdiri dari 2 fungsi yaitu fungsi optimasi *convex* dan *non convex*. Fungsi optimasi *convex* adalah fungsi optimal yang memiliki fungsi obyektif *convex,* sedangkan fungsi optimasi *non convex* adalah fungsi optimal yang memiliki fungsi obyektif *non convex.* Sebelum mengimplementasikan algoritma BFGS dan MBFGS ditentukan salah satu contoh fungsi optimasi *convex* dan *non convex* untuk menunjukkan keefektifan algoritma MBFGS dibandingkan BFGS. Fungsi *non convex* yang diambil sebagai contoh adalah fungsi Banana yang didefinisikan sebagai berikut:

Min $f(x_1, x_2) = (1 - x_1)^2 + (x_2 - (x_1)^2)^2$

(4.3)

dengan nilai awal $x_1 = -3 \; dan \; x_2 = 5$.

Fungsi diatas jika digambar sebagai berikut:



Gambar 1 Fungsi Banana

Dari gambar diatas menunjukkan bahwa fungsi Banana adalah fungsi *non convex*. Selain itu juga dapat ditunjukkan dengan menggunakan definisi fungsi *convex* yaitu persamaan (2.1)

$\forall \, x, y \in domain \, f$ dan $\theta \; dengan \; 0 \leq \theta \leq 1$ maka

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$$

Jika fungsi banana adalah fungsi *convex* maka memenuhi persamaan (2.1) dengan asumsi $x_1 = x \; dan \; x_2 = y$. Oleh karena itu diselidiki apakah fungsi tersebut persamaan (4.3) memenuhi persamaan (2.1). Hasil penyelidikan kontradiksi dengan persamaan (2.1) jadi fungsi Banana adalah fungsi *non convex*.

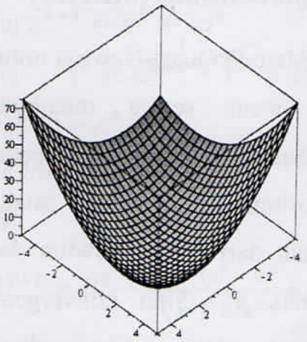
Selanjutnya contoh fungsi *convex* didefinisikan sebagai berikut:

$\min f(x_1, x_2) = {x_1}^2 + {x_2}^2 - x_1 x_2$

(4.4)

dengan nilai awal $x_1 = 1 \; dan \; x_2 = 2$.

Fungsi diatas jika digambar sebagai berikut:



Gambar 2 Fungsi Optimasi *Convex*

Dari gambar diatas menunjukkan bahwa fungsi tersebut adalah fungsi *convex*. Selain itu juga dapat ditunjukkan dengan menggunakan definisi fungsi *convex* yaitu persamaan (2.1)

$\forall \, x, y \in domain \, f$ dan $\theta \; dengan \; 0 \leq \theta \leq 1$ maka

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y),$$

dengan asumsi $x_1 = x \; dan \; x_2 = y$. Hasil penyelidikkan terbukti bahwa $f(x, y) = x^2 + y^2 - xy$ adalah fungsi *convex*.

Simulasi dan Analisis Hasil Rancangan Algoritma

Dari hasil rancangan pada subbab sebelumnya akan dilakukan dua macam simulasi, yaitu penyelesaian masalah optimasi *non convex* dan *convex* dengan algoritma BFGS dan MBFGS. Hal ini dilakukan untuk menganalisis hasil numerik dan membandingkan keefektifan dari dua algoritma tersebut.

Simulasi Fungsi *Convex*

Berikut hasil numerik yang didapat untuk masalah optimasi *convex* seperti pada persamaan (4.4) dengan menggunakan algoritma BFGS:

Tabel 1 Hasil Numerik Algoritma BFGS

```
Penyelesaian fungsi x1^2-x1*x2+x2^2 dengan nilai awal
(1,2)
menggunakan metode BROYDEN-FLETCHER-
GOLDFARB-SHANNO (BFGS)
-----------------------------------------------------------------
-----
masukkan nilai error yang diinginkan dengan
0<error<1:0.0001
masukkan nilai maksimum iterasi yang diinginkan:5
data awal yang diketahui adalah
 x0      b0     c0
  1   1   0   0
  2   0   1   3

 iterasi  norm   lambda  turunan_ke2   x1      x2
fungsi
_________________________________________________

1.0000   3.0000   0.5000  18.0000   1.0000   0.5000
0.7500

 2.0000  1.5000  0.6667  3.3750     0     0     0

norm_c =

  0

nilai x yang optimal =

x =

  0
  0

maka nilai fungsi yang optimal adalah

fun =

  0

bn =

  2  -1
 -1   2

maksimum iterasinya adalah :

i =

  3
```

Selanjutnya hasil numerik dari algoritma MBFGS dengan $\sigma = 0.0001$ dan $\rho = 0.8$ adalah sebagai berikut:

Tabel 2 Hasil Numerik Algoritma MBFGS

```
Penyelesaian fungsi x1^2-x1*x2+x2^2 dengan nilai awal
(1,2)
menggunakan metode MODIFIKASI BROYDEN-
FLETCHER-GOLDFARB-SHANNO (MBFGS)
-----------------------------------------------------------------
----
masukkan nilai error yang diinginkan dengan
0<error<1:0.0001
masukkan nilai konstanta tho yang diinginkan dengan
0<tho<1:0.0001
masukkan nilai konstanta Ro yang diinginkan dengan
0<ro<1:0.8
masukkan nilai maksimum iterasi yang diinginkan:20
data awal yang diketahui adalah
 x0      b0      c0
  1   1   0   0
  2   0   1   3

___________________________________________________

iterasi   norm    lambda    x1       x2     fungsi
___________________________________________________

  1.0000   3.0000   0.8000   1.0000  -0.4000   1.5600

  2.0000   3.0000   1.0000  -1.0400  -0.4480   0.8164

  3.0000   1.6383   1.0000  -0.7054  -0.4064   0.3760

  4.0000   1.0101   1.0000  -0.4004  -0.3079   0.1318

  5.0000   0.5379   1.0000  -0.1899  -0.1907   0.0362

  6.0000   0.2691   1.0000  -0.0623  -0.0867   0.0060

  7.0000   0.1174   1.0000  -0.0095  -0.0259   0.0005

  8.0000   0.0428   1.0000   0.0018  -0.0044   0.0000

  9.0000   0.0134   1.0000   0.0016  -0.0002   0.0000

 10.0000   0.0039   1.0000   0.0004   0.0002   0.0000

 11.0000   0.0006   1.0000   0.0000   0.0001   0.0000

norm_c =

 6.9931e-005

nilai x yang optimal =
```

```
x =

  1.0e-004 *

   0.3857
   0.5189

maka nilai fungsi yang optimal adalah

fun =

  2.1789e-009


bn =

   2.0425  -1.0912
  -1.0912   2.1994

maksimum iterasinya adalah :

i =

   12
```

Dari hasil diatas diketahui bahwa tidak ada perbedaan yang signifikan antara kedua algoritma tersebut, namun untuk menyelesaikan masalah optimasi *convex* lebih efektif jika menggunakan algoritma BFGS karena iterasinya lebih cepat.

Algoritma MBFGS dapat menghasilkan iterasi yang beragam tergantung nilai konstanta σ dan ρ yang diambil. Dari hasil tersebut jika $\sigma \in (0{,}0.5)$ dan $\rho \in (0{,}1)$ iterasi yang didapat lebih cepat sedangkan jika $\sigma \in [0.5{,}1)$ dan $\rho \in (0{,}1)$ iterasi yang didapat lebih lambat. Hal ini telah dibuktikan Li dan Fukushima [Li, D dan Fukushima, M. 2001] secara teori konvergensi bahwa jika $\sigma \in (0{,}0.5)$ maka lebih konvergen. Dari hasil numerik yang didapat telah membuktikan kebenaran dari pembuktian Li dan Fukushima [Li, D dan Fukushima, M. 2001].

Simulasi Fungsi *Non Convex* Berikut has numerik yang didapat untuk masalah optima *convex* seperti pada persamaan (4.3) denga menggunakan algoritma BFGS:

Tabel 3 Hasil Numerik Algoritma BFGS

```
Penyelesaian fungsi (1-x1)^2+(x2-x1^2)^2 dengan nil
awal (-3,5)
menggunakan    metode    BROYDEN-FLETCHEF
GOLDFARB-SHANNO (BFGS)
--------------------------------------------------------
------
masukkan   nilai   error   yang   diinginkan   denga
0<error<1:0.0001
masukkan nilai maksimum iterasi yang diinginkan:5
data awal yang diketahui adalah
 x0      b0     c0
 -3    1    0  -56
  5    0    1   -8

 iterasi              norm                    lambda
-
________________________________________________________
-
              x1                         x2       fungsi
________________________________________________________
-
  1.0000              56.5685              0.0953 - 0.0000
0.0161 - 0.0000i  0.0532 + 0.0000i  -2.1011 - 0.0000
5.1284 - 0.0000i  10.1263 + 0.0000i

  2.0000            1.4419 - 0.0000i  2.6924 + 0.0000i  -
6.7028 - 3.6014i  -6.7028 + 3.6014i  -3.7118 - 0.8654i
14.6612 + 5.1219i  22.4212 + 3.9020i

  3.0000            19.5999 -15.6366i   1.2569 - 0.4267i
0.4287 + 1.7852i  -0.8606 - 1.4101i  -2.1489 - 1.3397i
1.9146 + 5.2055i  8.6412 + 9.4410i

  4.0000          11.2783 +12.3312i  -0.0676 - 0.5085i  -
0.1924 + 0.2600i  -1.7921 + 0.4609i  -2.0879 - 0.8430i
2.0956 + 2.3182i  9.7914 + 8.9389i

  5.0000          15.3921 +17.1145i   1.7680 + 1.4106i
1.0285 - 0.3044i  -0.4358 - 0.7764i  -1.1698 - 0.2063i  -
0.7911 - 0.0433i  8.8699 + 3.1224i


bn =

 14.2830 - 1.6754i   1.9442 + 2.1818i
  1.9442 + 2.1818i   0.5334 + 1.1697i

maksimum iterasinya adalah :

i =

   5
```

Selanjutnya hasil numerik dari algoritma MBFGS dengan $\sigma = 0.0001$ dan $\rho = 0.001$ adalah sebagai berikut:

Tabel 4 Hasil Numerik Algoritma MBFGS

Penyelesaian fungsi (1-x1)^2+(x2-x1^2)^2 dengan nilai awal (-3,5)
menggunakan metode MODIFIKASI BROYDEN-FLETCHER-GOLDFARB-SHANNO (MBFGS)
----
masukkan nilai error yang diinginkan dengan 0<error<1:0.0001
masukkan nilai konstanta tho yang diinginkan dengan 0<tho<1:0.0001
masukkan nilai konstanta Ro yang diinginkan dengan 0<ro<1:0.001
masukkan nilai maksimum iterasi yang diinginkan:20
data awal yang diketahui adalah

x0	b0		c0
-3	1	0	-56
5	0	1	-8

iterasi	norm	lambda	x1	x2	fungsi
1.0000	56.5685	0.0010	-2.9440	5.0080	28.9444
2.0000	51.5006	1.0000	-2.6302	5.3109	15.7617
3.0000	24.3825	1.0000	-0.2978	0.6590	2.0096
4.0000	2.2300	1.0000	-0.2577	0.5985	1.8649
5.0000	2.2364	1.0000	0.1647	-0.0020	0.6986
6.0000	1.6524	1.0000	0.3182	-0.1923	0.5510
7.0000	1.1510	1.0000	0.3475	-0.2082	0.5339
8.0000	1.0730	1.0000	0.4296	-0.2078	0.4792
9.0000	0.9129	1.0000	0.5295	-0.1424	0.4001
10.0000	0.8467	1.0000	0.6468	0.0381	0.2693
11.0000	0.8094	1.0000	0.7390	0.3148	0.1216
12.0000	0.4899	1.0000	0.8001	0.5363	0.0507
13.0000	0.2184	1.0000	0.8704	0.7128	0.0188
14.0000	0.1365	1.0000	0.9447	0.8698	0.0036
15.0000	0.0518	1.0000	0.9828	0.9550	0.0004
16.0000	0.0233	1.0000	0.9971	0.9919	0.0000
17.0000	0.0052	1.0000	0.9997	0.9993	0.0000
18.0000	0.0003	1.0000	1.0000	1.0000	0.0000

norm_c =

7.5921e-006

nilai x yang optimal =

x =

1.0000
1.0000

maka nilai fungsi yang optimal adalah

fun =

3.7222e-011

bn =

10.4756 -4.1831
-4.1831 2.0710

maksimum iterasinya adalah :

i =

19

Dari hasil diatas diketahui bahwa ada perbedaan yang signifikan antara kedua algoritma tersebut. Pada algoritma BFGS ditampilkan semua nilai λ (lambda). Pada iterasi pertama nilai lambda yang didapat bernilai imaginer yaitu 0.0953 - 0.0000i, 0.0161 - 0.0000i, 0.0532 + 0.0000i. Nilai 0.0000i berarti bahwa ada nilai imaginer yang mendekati 0, namun jika pada iterasi pertama dipilih nilai λ=0.0161 - 0.0000i untuk melanjutkan iterasi. Pada iterasi ke-2 dihasilkan nilai lambdanya yaitu 2.6924 + 0.0000i,-6.7028 - 3.6014i, -6.7028 + 3.6014i. Ketiga nilai lambda tersebut tidak memenuhi syarat nilai lambda untuk algoritma BFGS. Hal ini berarti bahwa algoritma BFGS tidak dapat diterapkan untuk optimasi *non convex.* Berbeda dengan algoritma MBFGS, algoritma ini lebih efektif untuk menyelesaikan optimasi *non convex.*

Seperti yang telah dijelaskan sebelumnya bahwa dari beragam nilai konstanta σ dan ρ yang lebih efektif ialah pilih nilai $\sigma \in (0{,}0.5)$ dan $\rho \in (0{,}1)$.

Simpulan

Kesimpulan yang dapat diambil dari hasil numerik antara algoritma BFGS dan MBFGS adalah:

1. Algoritma BFGS dimodifikasi menjadi MBFGS agar algoritma tersebut dapat menyelesaikan masalah optimasi *non convex*. Hal ini telah dibuktikan sebelumnya secara teori konvergensi.
2. Algoritma BFGS lebih cepat konvergen untuk menyelesaikan masalah optimasi *convex* dibandingkan dengan algoritma MBFGS pada salah satu contoh fungsi *convex*.
3. Algoritma MBFGS lebih cepat konvergen menyelesaikan masalah optimasi *non convex* yaitu fungsi Banana dibandingkan dengan algoritma BFGS. Algoritma BFGS tidak dapat diterapkan pada optimasi *non convex* karena nilai lambda yang dihasilkan tidak sesuai dengan nilai lambda yang memenuhi syarat algoritma BFGS
4. Algoritma MBFGS lebih cepat konvergan jika memilih nilai konstanta $\sigma \in (0{,}0.5)$ dan $\rho \in (0{,}1)$ karena hal ini telah dibuktikan sebelumnya secara teori konvergensi.

Daftar Pustaka

Arora, J. 2004. *Introduction to optimum design (edisi ke 2).* California: Elsevier Academic Press.

Bazaraa, M., Sherall, H dan Shetty, C.1979. *Nonlinear Programming (edisi ke 2).* New York: John Wiley & Sons, Inc.

Nocedal, J. dan Wright, S. 1999. *Numerical Optimization.* New York: Springer-Verlag.

Li, D dan Fukushima, M. 2001. *A modified BFGS method and its global convergence in nonconvex minimization.* Jurnal of Computational and Applied Mathematic 129: 15–35.

____________________. 2001. *On the global convergence of the BFGS methods for nonconvex unconstrained optimization problems.* SIAM Jurnal Optimization 11: 1054–1064.

Yuan, G dan Lu,X. 2011. *An active set limited memory BFGS algorithm for bound constrained optimization. Applied Mathematical Modelling* 35: 3561–3573.

Fakultas Keguruan & Ilmu Pendidikan
Universitas Dr. Soetomo Surabaya

Gedung C. 102 Universitas Dr. Soetomo Surabaya
Jalan Semolowaru 84 Surabaya 60118

Telp (031) 5944748
http://unitomo.ac.id