浙江大学

本科生课程作业

课程名称： 优化实用算法

姓名： 林皓泓

学院： 数学科学学院

专业： 自动化控制

学号： 3170105156

指导老师： 王何宇

2020 年 3 月 4 日

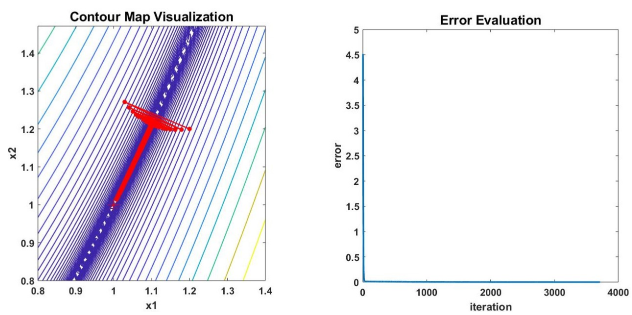
Algorithm 1 (P3.1) Backtracking Line Search for steepest descent

Require: α_0, c_1, ρ, x_0, maxiter, tolerance

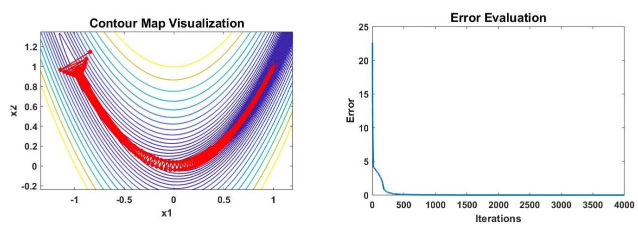
Ensure: Iteration Data: x_k, f_k, $error_k$ $f_0 \leftarrow f(x_0), g_0 \leftarrow \nabla f(x_0)$

while $k < max_iter$ & $error < tolerance$ **do**
    $\alpha \leftarrow \alpha_0$, $f_k \leftarrow f(x_k)$, $p_k = -\nabla f_k$
    while $f(x_k + \alpha p_k) > f(x_k) + c\alpha \nabla f_k^T p_k$ **do**
        $\alpha \leftarrow \rho\alpha$
    end while
    $error \leftarrow f(x_k + \alpha p_k)$
    $x_{k+1} \leftarrow x_k + \alpha p_k$
end while

We set $\alpha_0 = 1.0$, $\rho = 0.9$, $c = 10^{-4}$, and get the following results for steepest descent with respective initial x_0:



(a) $x_0 = (1.2, 1.2)^T$



(b) $x_0 = (-1.2, 1)^T$

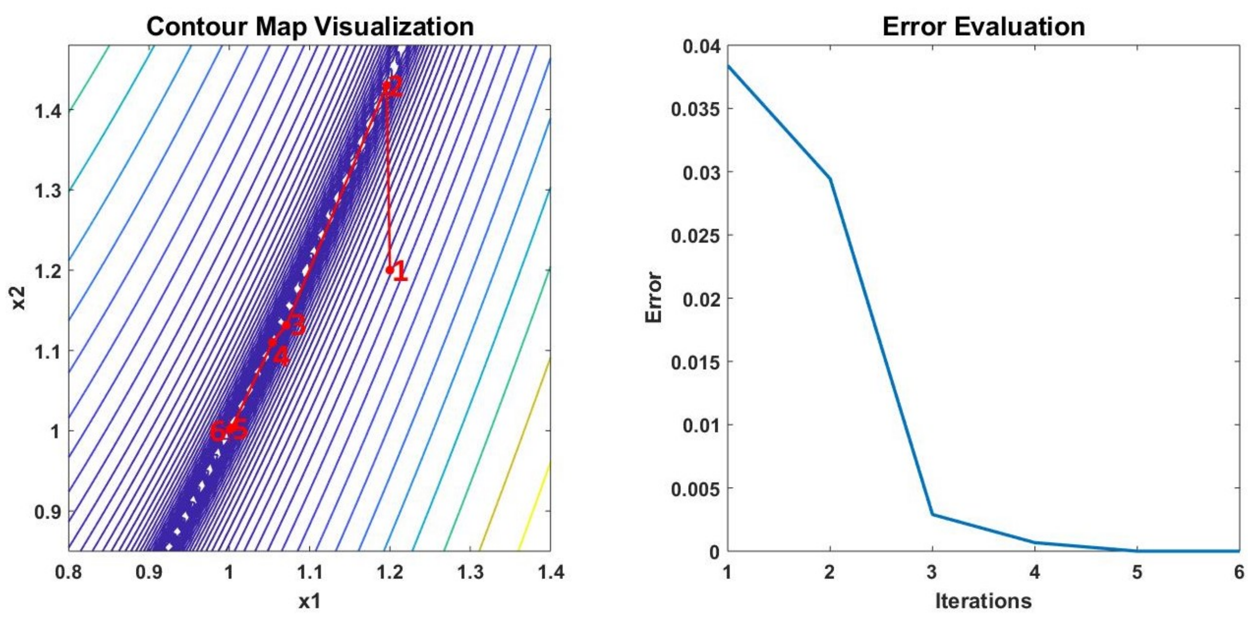
Figure 1: BLS visualization

iter	$alpha$	x_k	$error$	iter	α	x_k	$error$
0	1.0000	[1.2000, 1.2000]	5.8000000	0	1.0000	[-1.2000, 1]	24.20000000
1	0.1853	[1.0289, 1.2711]	4.51635320	1	0.3874	[-0.8413, 1.1464]	22.62797763
2	0.1668	[1.1788, 1.1981]	3.70091699	2	0.3487	[-1.1390, 0.9649]	15.62969072
3	0.1501	[1.0406, 1.2565]	3.01985241	3	0.2824	[-0.8793, 1.0758]	12.69272661
......	......	......	......	......	......	......	......
3712	0.0006	[1.0074, 1.0141]	0.00010015	4058	0.0008	[0.9923, 0.9837]	0.00013942
3713	0.0006	[1.0068, 1.0144]	0.00010009	4059	0.0007	[0.9916, 0.9840]	0.00011278
3714	0.0006	[1.0074, 1.0141]	0.00009999	4060	0.0006	[0.9922, 0.9838]	0.00009730

If we let $p_k = -\nabla^2 f_k^{-1} \nabla f_k$. We set $\alpha_0 = 1.0$, $\rho = 0.8$, $c = 10^{-7}$.

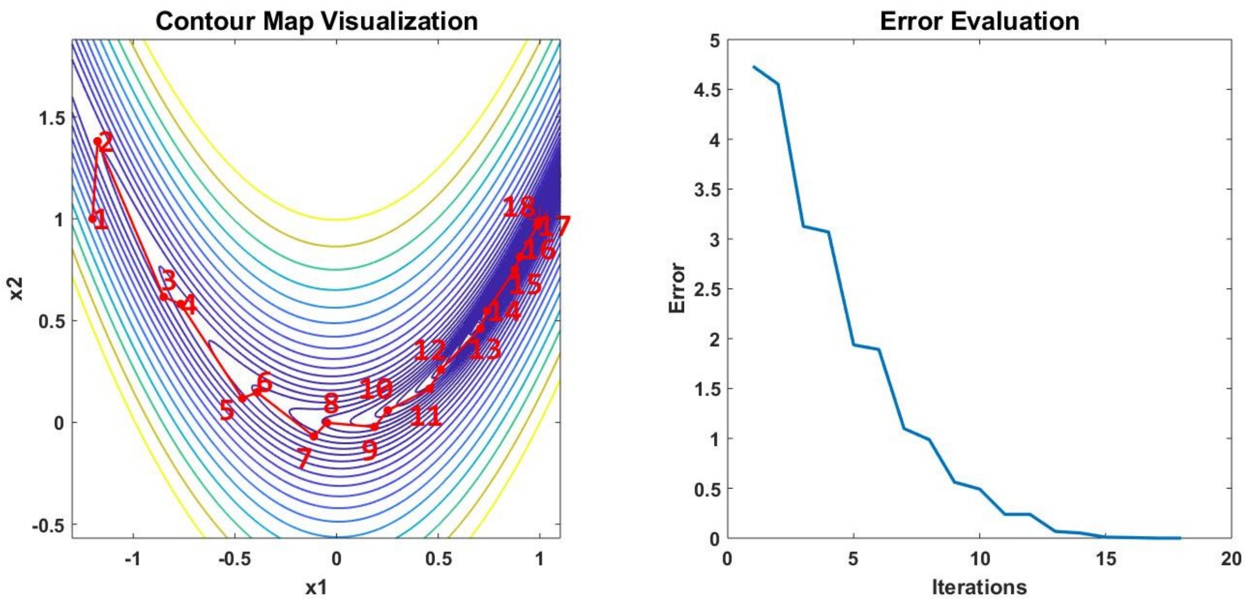
iter	α	x_k	$error$
0	1.0000	[-1.2000, 1]	24.20000000
1	1.0000	[-1.1753, 1.3807]	4.73188433
2	0.1678	[-0.8501, 0.6164]	4.55208328
3	1.0000	[-0.7669, 0.5813]	3.12684375
4	0.4096	[-0.4631, 0.1181]	3.06958455
5	1.0000	[-0.3910, 0.1477]	1.93751642
6	0.4096	[-0.1119, -0.0684]	1.89172141
7	1.0000	[-0.0472, -0.0020]	1.09842124
8	0.4096	[0.1863, -0.0223]	0.98712462
9	1.0000	[0.2519, 0.0592]	0.56146051
10	0.5120	[0.4578, 0.1651]	0.49175979
11	1.0000	[0.5126, 0.2597]	0.23848904
12	0.6400	[0.7075, 0.4614]	0.23814901
13	1.0000	[0.7407, 0.5475]	0.06738375
14	0.6400	[0.8767, 0.7496]	0.05091149
15	1.0000	[0.9025, 0.8138]	0.00955744
16	1.0000	[0.9885, 0.9698]	0.00561861
17	1.0000	[0.9932, 0.9863]	0.00004692
18	1.0000	[1.0000, 0.9999]	0.00000022

iter	$alpha$	x_k	$error$
0	1	[1.2000, 1.2000]	5.8000000
1	1	[1.1959, 1.4302]	0.0383840
2	0.64	[1.0709, 1.1313]	0.0294439
3	1	[1.0537, 1.1100]	0.0028975
4	1	[1.0030, 1.0034]	0.0006720
5	1	[1.0010, 1.0020]	0.0000010
6	1	[1.0000, 1.0000]	0.0000000



(a) $x_0 = (1.2, 1.2)^T$

Figure 2: BLS visualization of Newton Method



(a) $x_0 = (-1.2, 1)^T$

Figure 3: BLS visualization of Newton Method

(P3.2): Show that if $0 < c_2 < c_1 < 1$, there may be no step lengths that satisfy the Wolfe conditions.

Proof:

$\because$ $\phi(\alpha) = f(x_k + \alpha p_k)$ has a lower bound,

$\therefore$ for $0 < c_1 < 1$, $l(\alpha) = f(x_k) + \alpha c_1 \nabla f_k^T p_k$ has at least one intersection with the curve $\phi(\alpha)$.

Let α' as one of the intersection(s), then we have $f(x_k + \alpha' p_k) = f(x_k) + \alpha' c_1 \nabla f_k^T p_k$

$\forall \alpha < \alpha'$, Armijo condition is satisfied.

Further, based on mean value theorem, $\exists\ \alpha^* \in (0, \alpha')$, $\phi'(\alpha^*) = \frac{\phi(\alpha')-\phi(0)}{\alpha'}$

Here we look at those $\phi(\alpha)$ with $\phi'(\alpha) < 0$, $\forall \alpha \in (0, \alpha')$ (Choose smallest α^*, if more than one).

$\exists\ 0 < c_2 < \frac{\max_{\alpha \in (\alpha^*, \alpha')} \phi'(\alpha)}{\phi'(\alpha^*)} p_k c_1 < c_1$, $s.t.\ \forall\ \alpha'' \in (0, \alpha')$,

$$\nabla f(x_k + \alpha'' p_k)^T p_k = \phi'(\alpha'') < \frac{c_2}{c_1}\phi'(\alpha^*) = \frac{c_2}{c_1 \alpha'}\alpha' c_1 \nabla f_k^T p_k = c_2 \nabla f_k^T p_k$$

$\therefore$ curve condition may not be satisfied when given $0 < c_2 < c_1 < 1$, neither do the entire Wolfe conditions. ■

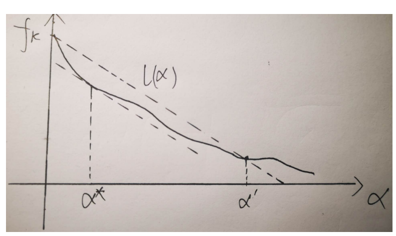


Figure 4: The sketch of proof process.