

无约束优化算法比较及其极值点研究

毛 巍¹, 兰恒友²

(1. 四川理工学院理学院, 四川 自贡 643000; 2. 企业信息化与物联网测控技术四川省高校重点实验室, 四川 自贡 643000)

摘 要:求解无约束优化问题是数值计算方面的重要研究内容, 求解无约束优化问题的方法较多, 选择一种较为快速且复杂度较小的方法具有重要意义。介绍无约束优化问题中 7 种算法的基本思想和具体步骤, 并结合 MATLAB 软件编程仿真, 依据定量分析对仿真结果进行对比分析, 对这 7 种算法的优缺点和极限点的收敛情况进行对比研究, 并且根据其收敛迭代次数和数值计算结果精确度确定一个相对有效的算法。

关键词:无约束优化算法; 算法分析; 迭代收敛; 极限点比较

中图分类号: O224

文献标志码: A

引 言

随着计算机近 30 年的发展, 最优化方法发展成为数学应用领域一个重要的分支。至于“最优”的解释, 就是将某一事件发展为最好的状态。如今, 无论人们从事各种活动, 都希望能使所要从事的活动达到自己想要的理想状态。在现实生活中广泛存在着最优化问题。而将最优化问题转化为比较容易解决的数学类问题, 并快速找出最优解的数学方法就是最优化方法。求解目标函数极大值极小值问题是数学上的一类问题, 并且这也是最优化问题。因此求解最优化问题的最优化方法是一种数学类方法, 而不是工程类方法, 其重要意义体现在不仅在经济管理学、运筹学和系统工程等数学领域, 而且也日益广泛地应用在其他领域(如工程设计)^[1]。

非线性最优化, 或者说非线性规划, 是至少有一个含有自变量的非线性函数存在于要求解的最优化问题的约束条件和目标函数中。所以相对应的要用非线性规划方法求解非线性规划问题。而且相比求解线性规划问题, 其求解难度更大, 因为其不同于解线性规划问

题有单纯形算法这一通用方法。解非线性规划问题到目前为止, 还没有建立起适用于各种问题的一个通用算法对非线性规划问题是普遍有效的。各个方法都有自己特定的适用范围, 因而这是需要人们深入研究的一个领域。但由于很多问题需要进一步精确化, 以及计算机的发展, 使非线性规划在近 30 年得到迅速的发展, 开始形成最优化问题的一个重要分支, 有广泛的应用, 特别是为最优化设计提供数学理论基础^[2]。

对于这些方法, 可以大致归类为三大类方法: 最速下降法、牛顿法及拟牛顿法。根据这三大类方法, 不同研究方向的研究者都有很多研究结论。Fvieve J^[3] 等于 2000 年对最速下降法进行了分析和完善, 并结合众多学者的研究, 总结出最速下降法收敛速度慢以及通常在计算过程前期迭代或者期间插步骤适用^[4]。牛顿法是一种求解无约束最优化问题的经典方法, 但其存在的不足(例如 Hesse 矩阵的可逆性)也令其进行不断完善, 比如 Joseph W^[5] 等研究出牛顿法和最速下降法的组合方法。拟牛顿法, 类似于最速下降法, 具有牛顿法的快速收敛性, 是一种较为有效的求解最优化问题的方法, 并且依据其算法思想, 研究者将拟牛顿法细分为几种算法

收稿日期: 2015-06-05

基金项目: 四川理工学院科研项目(2015RC07); 企业信息化与物联网测控技术四川省高校重点实验室开放基金项目(2013WZJ01)

作者简介: 毛 巍(1991-), 女, 四川眉山人, 硕士生, 主要从事最优化方法方面的研究, (E-mail) mwsuse@126.com

(BFGS 和 SR1 等),王宜举等^[6]在非线形规划与算法中详细讨论了几种常见的拟牛顿法。

本文考虑如下无约束优化问题:

$$\underset{x \in R^n}{\text{Min}} f(x)$$

求解这类无约束优化问题的算法很多,包括使用导数的算法以及不使用导数的算法。本文主要研究的算法为最速下降法、阻尼牛顿法、修正牛顿法、对称秩 1 算法(SR1)、BFGS 算法、DFP 算法和 Broyden 族算法,运用 MATLAB 软件对算法进行编程,详细介绍了 7 种算法的算法思想和算法步骤,并对各种算法的计算过程和结果进行分析,总结各类算法的优缺点,并比较算法优劣,最终得到相对有效的求解算法^[7]。此外,在相同的初始条件下,计算和分析 7 种无约束优化算法对同一算例产生的迭代点能否同时收敛到同一个点的可行性。

1 算法分析

1.1 最速下降法

1.1.1 算法思想

最速下降法又称梯度法,它是极小化算法中一种下降方向为负梯度方向的算法,虽然目前不再具有实用性,但其是研究其他算法的基础。最速下降法的关键就是选取最速下降的方向,且负梯度方向就是最速下降方向,进行一维搜索,直至满足精度要求,停止计算^[8]。

1.1.2 算法步骤

步 0 设定初始点 $x_0 \in R^n$, 容许误差 $0 \leq \varepsilon \ll 1$, 设 $k := 1$ 。

步 1 计算 $g_k = \nabla f(x_k)$ 。若 $\|g_k\| \leq \varepsilon$, 停止计算, 并将 x_k 作为最优解的近似解。

步 2 取方向 $d_k = -g_k$ 。

步 3 依据线搜索技术来选取步长因子 a_k 。

步 4 令 $x_{k+1} := x_k + a_k d_k, k := k + 1$, 转步 1。

1.2 拟牛顿法

1.2.1 算法思想

拟牛顿法的基本算法思想是利用目标函数值与一阶导数信息,构造目标函数近似曲率,并且同时具有类似牛顿法的快速收敛性的优点。拟牛顿法不用求逆矩阵,而是用 Hesse 矩阵的某个近似矩阵取代,是一种较为有效的求解无约束问题的算法^[9]。

1.2.2 算法步骤

(1) SR1 算法步骤

步 0 设定初始点 $x_0 \in R^n$, 终止误差 $0 \leq \varepsilon \ll 1$, 对称正定矩阵 H_0 (通常取单位矩阵 I_n), 令 $k := 0$ 。

步 1 若 $g_k \leq \varepsilon$, 停止计算, 并将 x_k 作为极小点的近似点。

步 2 计算搜索方向 $d_k = -H_k g_k$ 。

步 3 依据线搜索技术来选取步长因子 a_k 。

步 4 令 $x_{k+1} := x_k + a_k d_k$, 确定对称正定矩阵 H_{k+1} ,

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^T}{(s_k - H_k y_k)^T y_k}$$

步 5 令 $k := k + 1$, 转步 1。

(2) BFGS 算法步骤^[2,10]

步 0 给定参数 $\delta \in (0, 1), \sigma \in (0, 0.5)$, 初始点 $x_0 \in R^n$, 终止误差 $0 \leq \varepsilon \ll 1$, 对称正定矩阵 B_0 (通常取为 $G(x_0)$ 或单位矩阵 I_n), 令 $k := 0$ 。

步 1 计算 $g_k = \nabla f(x_k)$ 。若 $g_k \leq \varepsilon$, 停止计算, 并将 x_k 作为极小点的近似点。

步 2 解线性方程组, 得解 $d_k, B d_k = -g_k$ 。

步 3 设 m_k 满足如下不等式中的非负最小整数 m :

$$f(x_k + \delta^m d_k) \leq f(x_k) + \sigma \delta^m g_k^T d_k$$

令

$$a_k = \delta^{m_k}, x_{k+1} = x_k + a_k d_k$$

步 4 由校正公式确定 B_{k+1} ,

$$B_{k+1} = \begin{cases} B_k & y_k^T s_k \leq 0 \\ B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k} & y_k^T s_k > 0 \end{cases}$$

步 5 令 $k := k + 1$, 转步 1。

(3) DFP 算法步骤^[7]

步 0 给定参数 $\delta \in (0, 1), \sigma \in (0, 0.5)$, 初始点 $x_0 \in R^n$, 终止误差 $0 \leq \varepsilon < 1$ 。初始对称正定矩阵 B_0 (通常取为 $G^{-1}(x_0)$ 或单位矩阵 I_n), 令 $k := 0$ 。

步 1 计算 $g_k = \nabla f(x_k)$ 。若 $g_k \leq \varepsilon$, 停算, 并将 x_k 作为极小点的近似点。

步 2 计算搜索方向 $d_k = -H_k g_k$ 。

步 3 设 m_k 满足如下不等式中的非负最小整数 m :

$$f(x_k + \delta^m d_k) \leq f(x_k) + \sigma \delta^m g_k^T d_k$$

令

$$a_k = \delta^{m_k}, x_{k+1} = x_k + a_k d_k$$

步 4 由校正公式确定 H_{k+1} ,

$$H_{k+1} = \begin{cases} H_k & s_k^T y_k \leq 0 \\ H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k} & s_k^T y_k > 0 \end{cases}$$

步 5 令 $k := k + 1$, 转步 1。

(4) Broyden 族算法步骤^[11-12]

步 0 给定参数 $\delta \in (0, 1)$, $\sigma \in (0, 0.5)$, 初始点 $x_0 \in R^n$, 终止误差 $0 \leq \varepsilon \ll 1$ 。初始对称正定矩阵 B_0 (通常取为 $G^{-1}(x_0)$ 或单位矩阵 I_n), 令 $k := 0$ 。

步 1 计算 $g_k = \nabla f(x_k)$ 。若 $g_k \leq \varepsilon$, 停止计算, 并将 x_k 作为极小点的近似点。

步 2 计算搜索方向 $d_k = -H_k g_k$ 。

步 3 设 m_k 满足如下不等式中的非负最小整数 m :

$$f(x_k + \delta^m d_k) \leq f(x_k) + \sigma \delta^m g_k^T d_k$$

令

$$a_k = \delta^{m_k}, x_{k+1} = x_k + a_k d_k$$

步 4 由校正公式确定 H_{k+1} ,

$$H_{k+1} = \begin{cases} H_k & s_k^T y_k \leq 0 \\ H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k} + \varphi v_k v_k^T & s_k^T y_k > 0 \end{cases}$$

步 5 令 $k := k + 1$, 转步 1。

1.3 牛顿法

1.3.1 算法思想

将在迭代点 x_k 处的目标函数 $f(x)$ 进行二次泰勒展开, 且将其展开式作为目标模型函数, 并利用目标函数梯度和海森矩阵逆矩阵得到后续迭代点 x_{k+1} 。在适当条件下, 序列 x_k 收敛, 依据该二次目标模型函数极小点序列而近似求得相应的目标函数极小点。

1.3.2 算法步骤

(1) 阻尼牛顿法

步 0 设定参数 $0 \leq \varepsilon \ll 1$, $\delta \in (0, 1)$, $\sigma \in (0, 0.5)$, 初始点 $x_0 \in R^n$, 且 $k := 0$ 。

步 1 计算 $g_k = \nabla f(x_k)$ 。若 $g_k \leq \varepsilon$, 停算, 输出

$$x^* \approx x_k。$$

步 2 计算 $G_k = \nabla^2 f(x_k)$, 求解 $d_k, G_k d = -g_k$ 。

步 3 记 m_k 满足如下不等式中的非负最小整数 m ,

$$f(x_k + \delta^m d_k) \leq f(x_k) + \sigma \delta^m g_k^T d_k$$

步 4 令

$$a_k = \delta^{m_k}, x_{k+1} := x_k + a_k d_k, k := k + 1,$$

转步 1。

(2) 修正牛顿法

步 0 给定参数 $0 \leq \varepsilon < 1$, $\delta \in (0, 1)$,

$\sigma \in (0, 0.5)$, 初始点 $x_0 \in R^n$, 令 $k := 0$ 。

步 1 计算 $g_k = \nabla f(x_k)$, $\mu^k = g_k^{1+\tau}$ 。若 $g_k \leq \varepsilon$, 停止计算, 并将 x_k 为极小点的近似点。

步 2 计算 Hesse 矩阵 $G_k = \nabla^2 f(x_k)$, 求解 $d_k, (G_k + \mu_k I) d = -g_k$ 。

步 3 记 m_k 满足如下不等式中的非负最小整数 m ,

$$f(x_k + \delta^m d_k) \leq f(x_k) + \sigma \delta^m g_k^T d_k$$

令

$$a_k = \delta^{m_k}, x_{k+1} := x_k + a_k d_k$$

步 4 $k := k + 1$, 转步 1。

2 仿真结果以及分析

2.1 数值计算结果

运用 MATLAB 对 7 个算法进行仿真求解无约束最优化问题:

$$\min f(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2$$

并得出结果(表 1)。

表 1 仿真运行结果

算法	最优点	最优解	迭代次数 k	运行时间/s
最速下降法	$(1, 1)^T$	1.1630e - 10	1159	0.4343
SRI	$(1, 1)^T$	7.0304e - 19	22	0.0286
BFGS	$(1, 1)^T$	2.2005e - 11	20	0.0278
DFP	$(1, 1)^T$	7.0983e - 15	27	0.0735
Broyden 族	$(1, 1)^T$	2.1465e - 17	23	0.0291
阻尼牛顿法	$(0.8033, 0.6436)^T$	0.0390	100	0.0574
修正牛顿法	$(0.7946, 0.6293)^T$	0.0426	150	0.0834

2.2 数值分析

最速下降法求得的最优解为 $(1, 1)^T$, 迭代次数为 1159, 最优值为 1.1630e - 10, 迭代时间为 0.4343 s。从仿真结果知, 虽然最速下降法精准求得最优点, 但是其迭代次数太多, 所以效率不高。最速下降法程序设计简单, 存储小, 对初始点没有特定的要求, 即使从一个不理想的初始点出发也收敛到局部极小点。综合上述分析结果, 得出以下结论: (1) 在远离极小点的迭代点, 目标

函数每一次迭代都会导致其函数值下降幅度较大; (2) 在接近极小点的迭代点, 由于牛顿法锯齿现象, 会导致目标函数每一次前进的距离逐渐缩短, 从而使目标函数收敛速度较慢^[13]。

阻尼牛顿法和修正牛顿法求得的最优解为 $(0.8033, 0.6436)^T$ 和 $(0.7946, 0.6293)^T$, 迭代次数为 100 和 150, 最优值为 0.0390 和 0.0426, 运行时间为 0.0574 s 和 0.0834 s。相比较最速下降法, 虽然没有精

准求得目标值,但是这两种算法的迭代次数减少了许多,相对应的运行时间也会减少很多。而且发现在程序运行过程中,牛顿法由于求得 Hesse 矩阵的逆矩阵,而在计算过程中并不能确切保证 Hesse 矩阵一直是可逆的。所以,当选取初始点为一定值时,就可能出现程序不能运行的问题,这也是牛顿法最大的缺陷^[14]。

拟牛顿法中 SR1、BFGS、DFP、Broyden 族 4 种方法的最优解均为 $(1, 1)^T$, 迭代次数分别为 22、20、27、23, 最优值为 $7.0304e-19$ 、 $2.2005e-11$ 、 $7.0983e-15$ 、 $2.1465e-17$, 运行时间为 0.0286 s、0.0278 s、0.0735 s、0.0291 s。比较最速下降法和牛顿法,可以明显地发现不论是优点还是迭代次数,均比上述两种方法有较大的提升。再比较拟牛顿法中 4 种算法的运行结果,迭代次数以及相对应的运行时间都相差不大,但是 BFGS 算法在迭代次数以及相对应的运行时间上均有优势。所以,只要精度足够高,该方法很理想^[15]。

3 极限点比较

3.1 计算收敛结果

由于算法的仿真结果在一定程度上会受计算机编写相关程序的影响。因此在编写程序中出现的微小细节都会使某一个算法的性能以及结果产生比较大的差异性,如初始步长和一维搜索策略的不同,因此为了消除影响,在编程时对这 7 种无约束最优化算法运用定量分析,即这 7 种算法具有相同的一维搜索策略、初始步长以及终止条件,尽最大可能使算法本身成为判断算法有效性唯一依据^[16]。因为研究的问题都是连续且可导的,并且依据无约束最优化问题的条件,迭代点趋于最优的快慢可以通过梯度趋于 0 的快慢近似地反映出来。本文终止条件为梯度,通过仿真结果分析评估这 7 种无约束最优化算法的有效性。

无约束最优化问题为:

$$\min f(x) = (x_1^3 - x_2)^2 + (x_2 - x_1)^4$$

显然 $(0, 0)$, $(1, 1)$, $(-1, -1)$ 均是该问题的最优解,对于初始点分别为 $(1, 3)$ 、 $(0.5, 2)$ 、 $(-1, -3)$ 和 $(-0.5, -2)$ 时,7 种无约束优化算法的计算结果见表 2~表 5。

表 2 初始点为 $(1, 3)$ 的计算结果

算 法	终止点 x^*	$f(x^*)$	迭代次数 k	极限点
最速下降法	(1.0077, 1.0232)	5.8392e-08	8330	(1, 1)
阻尼牛顿法	(0.0774, 0.0578)	1.4711e-11	37	(0, 0)
修正牛顿法	(0.0774, 0.0578)	1.4753e-11	89	(0, 0)
SR1	(0.0774, 0.0577)	1.4712e-11	14	(0, 0)
BFGS	(1.0009, 1.0028)	1.3085e-11	26	(1, 1)
DFP	(0.9984, 0.9952)	1.1017e-10	118	(1, 1)
Broyden 族	(0.9978, 0.9935)	3.6066e-10	35	(1, 1)

表 3 初始点为 $(0.5, 2)$ 的计算结果

算 法	终止点 x^*	$f(x^*)$	迭代次数 k	极限点
最速下降法	(1.0077, 1.0232)	5.8429e-08	8299	(1, 1)
阻尼牛顿法	(0.0174, 0.0147)	1.4711e-11	22	(0, 0)
修正牛顿法	(0.0093, 0.0000)	7.4806e-09	16	(0, 0)
SR1	(1.0013, 1.0040)	5.4021e-11	23	(1, 1)
BFGS	(1.0041, 1.0125)	4.8384e-09	19	(1, 1)
DFP	(0.0073, 0.0077)	1.1473e-09	1695	(0, 0)
Broyden 族	(0.0131, 0.0000)	2.9684e-08	28	(0, 0)

表 4 初始点为 $(-1, -3)$ 的计算结果

算 法	终止点 x^*	$f(x^*)$	迭代次数 k	极限点
最速下降法	(-1.0077, -1.0232)	5.8392e-08	8330	(-1, -1)
阻尼牛顿法	(-0.0774, -0.0578)	1.4711e-11	37	(0, 0)
修正牛顿法	(-0.0774, -0.0578)	1.4753e-11	89	(0, 0)
SR1	(-0.0774, -0.0577)	1.4712e-11	14	(0, 0)
BFGS	(-1.0009, -1.0028)	1.3085e-11	26	(-1, -1)
DFP	(-0.9984, -0.9952)	1.1017e-10	118	(-1, -1)
Broyden 族	(-0.9978, -0.9935)	3.6066e-10	35	(-1, -1)

表 5 初始点为 $(-0.5, -2)$ 的计算结果

算 法	终止点 x^*	$f(x^*)$	迭代次数 k	极限点
最速下降法	$(-1.0077, -1.0232)$	$5.8429e-08$	8299	$(-1, -1)$
阻尼牛顿法	$(-0.0174, -0.0147)$	$1.4711e-11$	22	$(0,0)$
修正牛顿法	$(-0.0093, -0.0000)$	$7.4806e-09$	16	$(0,0)$
SR1	$(-1.0013, -1.0040)$	$5.4021e-11$	23	$(-1, -1)$
BFGS	$(-1.0041, -1.0125)$	$4.8384e-09$	19	$(-1, -1)$
DFP	$(-0.0073, -0.0077)$	$1.1473e-09$	1695	$(0,0)$
Broyden 族	$(-0.0131, -0.0000)$	$2.9684e-08$	28	$(0,0)$

3.2 收敛结果分析

由表 2 ~ 表 5 可知,最速下降法的迭代次数始终是最多的,为 8330 次和 8299 次,在 4 个不同的初始点中,当初始点设定为 $(1,3)$ 、 $(0.5,2)$ 时,极限点为 $(1,1)$;当初始点设定为 $(-1, -3)$ 、 $(-0.5, -2)$ 时,极限点为 $(-1, -1)$ 。牛顿法算法求函数极值时,4 个不同初始点产生的点列均收敛到点 $(0,0)$,而且收敛速度较快。拟牛顿法迭代次数相对更快,其中 SR1 算法在初始点设定为 $(1,3)$ 、 $(-1, -3)$ 时,均收敛到点 $(0,0)$,而 BFGS、DFP 和 Broyden 族算法在初始点为 $(1,3)$ 、 $(-1, -3)$ 时,分别收敛到 $(1,1)$ 和 $(-1, -1)$,但初始点为 $(0.5, 2)$ 、 $(-0.5, -2)$ 时,SR1 算法和 BFGS 算法则分别收敛到点 $(1,1)$ 和 $(-1, -1)$,而 DFP 和 Broyden 族算法均收敛到 $(0,0)$ 。因此可知阻尼牛顿法、修正牛顿法和 SR1 算法产生的极限点具有变化的不稳定性,而另外 4 种算法均产生同一趋势。

当极小化目标函数采用最速下降法时,彼此相邻两个搜索方向存在正交性,因而产生的迭代点列路径表现为“之”字路线。最速下降法有两个特点:局部收敛很快和全局收敛很慢,尤其是快要到最优点时,收敛速度最慢。牛顿法与拟牛顿法是按照固定的设定坐标方向依次搜索,其搜索方向具有固定性。牛顿法有较快的收敛速度,但在初始点离极小点比较远的时候,是否有下降的方向不能确定。拟牛顿法和最速下降法一样只要求每一步迭代时知道目标函数的梯度。相比较最速下降法,这类方法具有巨大的优势,尤其针对较为困难的问题。

4 结束语

无约束最优化问题的计算方法是数值计算领域的重要研究课题,快速求解无约束最优化问题具有重要意义。本文针对这个问题,比较了其中的 7 种算法,通过单最优点和多最优点两个数值例子的 MATLAB 分析,可以发现每一种方法的优点和其缺陷。通过分析仿真结果,相比于最速下降法和牛顿法,可以明显地发现拟牛

顿法不论是最优点还是迭代次数,均比上述两种方法有较大的提升。

综合而言,主要是因为这 7 种无约束优化算法算法具有差异的搜索方法构造,所以在相同的初始条件情况下,这 7 种算法的迭代点列就会使其对应的极限点不相同。因此,当求解含有多个最优解的优化问题时,要根据问题需要测验多个初始点,最终达到寻求最优解多解的目的。

参 考 文 献:

- [1] 李志军.非线性最优化方法研究[J].数学学习与研究,2013(5):61-63.
- [2] 吴祈宗,侯福均.运筹学与最优化方法[M].2 版.北京:机械工业出版社,2013.
- [3] Fliege J,Svaiter B F.Steepest descent methods for multi-criteria optimization[J].Mathematical Methods of Operational Research,2000,51(3):479-494.
- [4] 李欣.求解无约束优化问题的算法研究[D].西安:电子科技大学,2009.
- [5] So J,Wu J,Zou X.A reaction - diffusion model for a single species with age structure.I Traveling wavefronts on unbounded domains [J].Royal Society of London Proceedings,2001,457(2012):1841.
- [6] 王宜举,修乃华.非线性规划理论与算法[M].西安:陕西科学技术出版社,2004.
- [7] 经红霞.无约束最优化问题的算法研究与实现[D].北京:北京邮电大学,2013.
- [8] 刘盎然.线性方程组的迭代和最速下降法[J].赤峰学院学报:自然科学版,2014(2):10-13.
- [9] 屈晓军.非凸无约束优化问题的修正拟牛顿算法 [D].长沙:湖南大学,2014.
- [10] 蒋华杰.BFGS 算法的最优化问题及在 MATLAB 中的实现[J].科技创新导报,2014(19):88.
- [11] 王娜.求解奇异问题几种迭代格式的收敛性[D].哈

- 尔滨:哈尔滨理工大学,2014.
- [12] 景书杰,赵建卫,韩学锋.一种改进的逆 Broyden 算法[J].吉林师范大学学报:自然科学版,2015,36(1):66-68.
- [13] 高蒙.求解无约束最优化问题算法比较[J].市场周刊:理论研究,2014(5):155-156.
- [14] 王越.牛顿法的一种改进[J].邢台学院学报,2014(4):163-164.
- [15] 陈姗.求解无约束最优化问题的一个新的拟牛顿方法[D].南京:南京理工大学,2013.
- [16] 经红霞.无约束最优化问题的算法研究与实现[D].北京:北京邮电大学,2013.

The Comparison of Unconstrained Optimization Algorithms and Their Extreme Point Study

MAO Wei¹, LAN Hengyou²

(1. Institute of Nonlinear Science and Engineering Computing, Sichuan University of Science & Engineering, Zigong 643000, China; 2. Enterprise Informatization and Network Measurement and Control Technology Sichuan Provincial Key Laboratory of University, Sichuan University of Science & Engineering, Zigong 643000, China)

Abstract: Solving unconstrained optimization problems is an important research problem in numerical calculations. At present, there are many ways to solve the unconstrained optimization problem, so choosing a rapid method that with small complexity is significant. Firstly, the basic ideas and concrete steps of seven algorithms in unconstrained optimization problems are introduced. Then combined with the MATLAB software programming simulation, according to the quantitative analysis, the simulation results are contrasted and analyzed. Finally, the advantages and disadvantages, the convergence situations of extreme points of seven algorithms are contrasted and studied, and a relatively efficient algorithm is determined according to the convergence iterations and the accuracy of numerical computing results.

Key words: unconstrained optimization algorithm; algorithm analysis; the iterative convergence; comparing the limit point