数值最优化方法大作业

21231024 曹致瑾

1. 实验代码说明

该实验的全部代码由matlab2016b编写，对代码文件中各函数与脚本的说明如下表所示：

表一 各函数与脚本的说明

|  |  |
| --- | --- |
| 脚本/函数名称 | 脚本/函数功能 |
| get\_problem.m | 脚本，加载所需要求解的问题函数及其初值 |
| wolf.m | 函数，利用wolf准则进行线搜索，  返回满足wolf准则的线搜索步长 |
| powerful\_wolf.m | 函数，利用强wolf准则进行线搜索，  返回满足强wolf准则的线搜索步长 |
| gll.m | 函数，利用gll准则进行线搜索，  返回满足gll准则的线搜索步长 |
| gill\_murry\_correction.m | 函数，对当前x的hesse矩阵进行gill-murry修正，  返回gill\_murry修正后的当前点的hesse矩阵 |
| damped\_newton.m | 阻尼newton主函数，  对其决定求解的参数说明如下：  problem的值为more-testing中的问题序号  line\_search\_method的值赋1，进行wolf线搜索  line\_search\_method的值赋2，进行强wolf线搜索  line\_search\_method的值赋3，进行gll线搜索  use\_gm\_correction的值为1，进行gill-murry修正  use\_gm\_correction的值为0，不进行gill-murry修正  epsilon = 10^(-8)为迭代精度参数  flag为算法终止标志，为1则终止迭代跳出循环  k为迭代次数  x(:，k)为第k次迭代中函数自变量值 |
| hybrid\_newton.m | 混合newton主函数，参数同上 |

1. 单调线搜索与非单调线搜索的比较

2.1问题(2)——Freudenstein and Roth function

2.1.1问题描述



2.1.2实验环境设置

强wolfe搜索准则参数：alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %wolfe算法参数

sigma = 0.9; %wolfe算法参数

gll搜索参数：alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %gll算法参数

终止条件：

2.1.3数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | (11.4128,-0.8968) | (11.4128,-0.8968) | (11.4128,-0.8968) | (11.4128,-0.8968) |
| f(x\*) | 48.9843 | 48.9843 | 48.9843 | 48.9843 |
| 迭代次数 | 31 | 31 | 31 | 31 |
| 函数调用次数 | 124 | 269 | 389 | 484 |
| 程序运行状况说明 | 收敛到局部极小值点 | 收敛点局部极小值点 | 收敛到局部极小值点 | 收敛点局部极小值点 |

2.1.4结果分析

在两种方法都能对问题进行求解的情况下，用强wolfe准则进行线搜索求解问题的迭代次数与函数调用次数小于gll准则。而在都是使用gll准则的情况下，m值越大，求解问题的迭代次数与函数调用次数就越大。

2.2问题(21)——Extended Rosenbrock function

2.2.1问题描述



2.2.2实验环境设置

强wolfe搜索准则参数： alpha = 3;%初始步长

index = 0.5; %缩减步长系数

rho = 0.0001; %wolfe算法参数

sigma = 0.9; %wolfe算法参数

gll搜索参数：alpha = 3;%初始步长

index = 0.5; %缩减步长系数

rho = 0.0001; %gll算法参数

终止条件：

2.2.3数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | / | (1,...,1) | (1,...,1) | (1,...,1) |
| f(x\*) | / | 6.5260e-21 | 3.7421e-20 | 3.4483e-22 |
| 迭代次数 | / | 71 | 83 | 101 |
| 函数调用次数 | / | 650 | 1145 | 1847 |
| 程序运行状况说明 | 当迭代至xt点处，无法找到满足强wolfe准则的步长，求解失败 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 |

（注：此处的xt为（-1.0990，1.2458，-1.0990，1.2458，-1.0990，1.2458，-1.0990，1.2458，-1.0990，1.2458））

2.2.4结果分析

一般而言，gll准则相比强wolfe准则更有可能求解成功，因为强wolfe准则比gll准则条件更严苛，所以强wolfe准则更有可能出现找不到满足条件的线搜索步长的情况，此时迭代便无法进行，造成求解失败。

同时，该算例也可说明2.1中结论，在都是使用gll准则的情况下，m值越大，求解问题的迭代次数与函数调用次数就越大。

2.3问题(32)——Linear function-full rank

2.3.1问题描述



2.3.2实验环境设置

强wolfe搜索准则参数： alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %wolfe算法参数

sigma = 0.9; %wolfe算法参数

gll搜索参数：alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %gll算法参数

终止条件：

2.3.3数值结果

1. n = 3， m = 4时数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | (-1,-1,-1) | (-1,-1,-1) | (-1,-1,-1) | (-1,-1,-1) |
| f(x\*) | 1 | 1 | 1 | 1 |
| 迭代次数 | 24 | 24 | 24 | 24 |
| 函数调用次数 | 96 | 206 | 291 | 351 |
| 程序运行状况说明 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 |

1. n = 6， m = 12时数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | (-1,...-1) | (-1,...-1) | (-1,...-1) | (-1,...-1) |
| f(x\*) | 6 | 6 | 6 | 6 |
| 迭代次数 | 24 | 24 | 24 | 24 |
| 函数调用次数 | 96 | 206 | 191 | 351 |
| 程序运行状况说明 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 |

1. n=10, m=10时数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | (-1,...-1) | (-1,...-1) | (-1,...-1) | (-1,...-1) |
| f(x\*) | 3.1691e-18 | 3.1691e-18 | 3.1691e-18 | 3.1691e-18 |
| 迭代次数 | 24 | 24 | 24 | 24 |
| 函数调用次数 | 96 | 206 | 191 | 351 |
| 程序运行状况说明 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 |

1. n = 10， m = 15时数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | (-1,...-1) | (-1,...-1) | (-1,...-1) | (-1,...-1) |
| f(x\*) | 5 | 5 | 5 | 5 |
| 迭代次数 | 24 | 24 | 24 | 24 |
| 函数调用次数 | 96 | 206 | 191 | 351 |
| 程序运行状况说明 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 |

1. n = 20， m = 26时数值结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 线搜索方法 | 强wolfe | gll(m=5) | gll(m=10) | gll(m=15) |
| 求得x\*值 | (-1,...-1) | (-1,...-1) | (-1,...-1) | (-1,...-1) |
| f(x\*) | 6 | 6 | 6 | 6 |
| 迭代次数 | 25 | 25 | 25 | 25 |
| 函数调用次数 | 100 | 215 | 305 | 370 |
| 程序运行状况说明 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 | 收敛到全局极小值点 |

2.3.4结果分析

这组函数的实验结果令人惊讶，对于选定的m、n，迭代的收敛点列在强wolfe、gll（m=5，m=10，m=15）的情况下实际上是完全相同的，而迭代次数因为gll需要计算先前点的函数值有不同。对于这个结果，我推测应该是这组函数的性质比较好，以至于每个点都恰好在第一个步长下就满足了线搜索指标（在强wolfe、gll（m=5，m=10，m=15）下都是这样），因此最终收敛到终止点的点列完全相同。

1. 两种 Newton 型方法有效性的比较

3.1问题(4)——Brown badly scaled function

3.1.1问题描述



3.1.2实验环境设置

强wolfe搜索准则参数： alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %wolfe算法参数

sigma = 0.9; %wolfe算法参数

Gill-murry修正参数： u = 1e-16%机器精度

终止条件：

3.1.3数值结果

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | / | 1.0e+06 \*（1.0000, 0.0000） |
| f(x\*) | / | 2.5714e-30 |
| 迭代次数 | / | 48 |
| 函数调用次数 | / | 334 |
| 程序运行状况说明 | 当迭代至xt点处，无法找到满足强wolfe准则的步长，求解失败 | 收敛到全局极小值点 |

（注：此处的xt为1.0e+05 \*（3.0000，0.0000））

3.1.4结果分析

对阻尼Newton法的点xt进行分析，此处g'\*d的值为8.2001e+11，显然，该方向不为下降方向，因此在这一方向上难以找到满足强wolfe准则的alpha。而Gill-murry修正牛顿法对不正定的矩阵进行了修正，使得求得的迭代方向d更可能为下降方向，解决了这一问题。

3.2问题(5)——Beale function

3.2.1问题描述



3.2.2实验环境设置

强wolfe搜索准则参数： alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %wolfe算法参数

sigma = 0.9; %wolfe算法参数

Gill-murry修正参数： u = 1e-16%机器精度

终止条件：

3.2.3数值结果

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | （0.0000，1.0000） | （3.0000，0.5000） |
| f(x\*) | 14.2031 | 1.8588e-18 |
| 迭代次数 | 25 | 23 |
| 函数调用次数 | 100 | 232 |
| 程序运行状况说明 | 收敛点处hesse矩阵为    矩阵特征值有正有负，说明收敛到鞍点，此时函数值为14.2031 | 收敛到全局极小值点 |

3.2.4结果分析

阻尼Newton法无法保证hesse矩阵的正定性，可能会造成结果收敛到鞍点的情况，而Gill-murry修正牛顿法对不正定的矩阵进行了修正，使得函数更有可能收敛到全局最小值。

3.3问题(26)——Trigonometric function

3.3.1问题描述



3.3.2实验环境设置

强wolfe搜索准则参数： alpha = 1;%初始步长

index = 0.6; %缩减步长系数

rho = 0.0001; %wolfe算法参数

sigma = 0.9; %wolfe算法参数

gll搜索参数：alpha = 1;%初始步长

index = 0.5; %缩减步长系数

rho = 0.0001; %gll算法参数

Gill-murry修正参数： u = 1e-16%机器精度

终止条件：

3.3.3数值结果

1. n = 2， m = 2时数值结果

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | / | （0.2431，0.6127） |
| f(x\*) | / | 2.0316e-17 |
| 迭代次数 | / | 19 |
| 函数调用次数 | / | 214 |
| 程序运行状况说明 | 当迭代至xt点处，无法找到满足强wolfe准则的步长，求解失败 | 收敛到全局极小值点 |

（注：此处的xt为（0.5788，0.5095），此时g'\*d=0.0043>0,说明迭代方向不是下降方向）

1. n = 3， m = 3时数值结果

由于强wolfe搜索易出现失败，这里换成m=5的gll搜索进行对比

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | / | （0.1387，0.1524，0.4678） |
| f(x\*) | / | 1.1537e-17 |
| 迭代次数 | / | 25 |
| 函数调用次数 | / | 266 |
| 程序运行状况说明 | 当迭代至xt点处，无法找到满足强wolfe准则的步长，求解失败 | 收敛到全局极小值点 |

（注：此处的xt为（0.3333，0.3333，0.3333），此时g'\*d=0.1433>0,说明迭代方向不是下降方向）

1. n = 4， m = 4时数值结果

由于强wolfe搜索易出现失败，这里换成m=5的gll搜索进行对比

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | （0.1455，0.1602，  0.4250，0.2168） | （0.0892，0.0941，  0.1003，0.3809） |
| f(x\*) | 3.0282e-04 | 1.6659e-17 |
| 迭代次数 | 29 | 26 |
| 函数调用次数 | 252 | 275 |
| 程序运行状况说明 | 收敛点处hesse矩阵为    矩阵特征值全正，说明收敛到局部极小值点 | 收敛到全局极小值点 |

1. n =8， m = 8时数值结果

由于强wolfe搜索易出现失败，这里换成m=5的gll搜索进行对比

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | / | / |
| f(x\*) | / | / |
| 迭代次数 | / | / |
| 函数调用次数 | / | / |
| 程序运行状况说明 | 当迭代至xt1点处，无法找到满足强wolfe准则的步长，求解失败 | 当迭代至xt2点处，无法找到满足强wolfe准则的步长，求解失败 |

（注：此处的xt1为（0.125，...,0.125），此时g'\*d=0.0729>0,说明迭代方向不是下降方向，此处的xt2为（0.1104，0.1026，0.0723，0.1194，0.1085，0.1308，0.1269，0.1258）此时g'\*d=-1.4450e+13>0,迭代方向是下降方向，但由于线搜索的限制仍没找到满足的步长）

1. n = 10， m = 10时数值结果

|  |  |  |
| --- | --- | --- |
| 线搜索方法 | 阻尼Newton法 | Gill-murry修正Newton法 |
| 求得x\*值 | / | / |
| f(x\*) | / | / |
| 迭代次数 | / | / |
| 函数调用次数 | / | / |
| 程序运行状况说明 | 当迭代至xt1点处，无法找到满足强wolfe准则的步长，求解失败 | 当迭代至xt2点处，无法找到满足强wolfe准则的步长，求解失败 |

（注：此处的xt1为（0.0749，0.0779，0.0816，0.2535，0.0919，0.0996，0.1079，0.1102，0.1050，0.0990），此时g'\*d=1.4314e-05>0,说明迭代方向不是下降方向，此处的xt2为（0.1，...，0.1），此时g'\*d=2.3438e+13>0,说明迭代方向不是下降方向）

3.3.4结果分析

该实验中的前三组同样可以说明实验3.1中结论。同时，后两组实验可说明，即使使用了Gill-murry修正，在未结合负曲率算法等的情况下，仍然无法保证由newton方法求得的方向d一定为下降方向。并且，该算法在遇到接近奇异点时由于求逆可能出现误差，也会对求得的方向d造成影响