运筹学与控制论

DOI:10.3969/J. ISSN. 1672-6693.2010.01.003

二次罚函数的可分化方法*

赫振华[1,2],白富生[1]
(1. 重庆师范大学 数学与计算机科学学院,重庆 400047;2. 重庆市忠县中学,重庆 忠县 404303)

摘要:可分方法用于将一个复杂的大规模优化问题分解成各个子问题进行求解。本文对可分优化问题给出两种可分方法,即分别将辅助问题原理(APP)方法和分块协调下降(BCD)方法应用于二次罚函数方法(QPM),并提出相应的 QPM + APP 算法和 QPM + BCD 算法,使得在求解可分优化问题时仅需要修正罚因子。最后给出了两个算例,通过与文献[1]中的 ALR + APP 和 ALR + BCD 算法作比较来求解,所得的计算结果说明本文给出的两种算法是具有有效性的。

关键词:可分优化问题;可分化方法;二次罚函数方法;辅助问题原理方法;非线性高斯-赛德尔方法
中图分类号:O221.2 文献标识码:A 文章编号:1672-6693(2010)01-0011-05

对于求解约束最优化问题,可以利用罚函数方法把它无约束化,然后通过求解无约束罚问题的全局极小而得到原约束最优化问题的全局极小。当问题的规模很大并具有一定的可分结构时,可以利用可分化方法进行分解。可分化方法又称为分解方法,可应用于多区域电力系统分析、网络设计、价格决策管理等模型中,早在上世纪 60 年代就已提出,如 Dantzig-Wolfe 分解和 Bender 分解[2]。后来有不少相关的研究成果问世[2-12]。2002 年,文献[1]从理论和实践两个方面比较了辅助问题原理(APP,Auxiliary problem principle)方法[13]和分块协调下降(BCD,Block coordinate descent)方法[14],两种分离技巧应用于增广拉格朗日松弛(ALR,Augmented lagrangian relaxation)方法时所得的两种可分方法。有时 BCD 方法也称为非线性高斯-赛德尔(Gauss-Seidel)方法。文献[15]对其定义如下。

假定 X 是笛卡尔乘积空间,$X = \prod_{j=1}^{n} X_j$,且设当前点为 $x^k = (x_1^k, x_2^k, \cdots, x_n^k)$,则非线性 Gauss-Seidel 算法定义为

$$\min_{x_i \in X_i} F(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \cdots, x_n^k), i = 1, 2, \cdots, n \tag{1}$$

本文将 APP 方法和 BCD 方法应用于二次罚函数方法,用于求解可分优化问题。二次罚函数方法(QPM,Quadratic penalty method)通过构造二次惩罚项函数,并把它加到目标函数中去,将原约束规划问题转化为一系列无约束最优化问题进行求解。在应用 ALR 方法求解时,不仅需要修正罚因子,而且要进行对偶变量的迭代,而通过二次罚函数方法求解原约束规划问题时,仅需要修正罚因子,因此在计算上更加简单。

1 预备知识

定义 1[2] (可分函数)假定空间 X 是可分的,即为乘积空间 $X_1 \times X_2 \times \cdots \times X_p$,任意 $x \in X$ 可以写为 $x = (x_1, x_2, \cdots, x_p)$,$x_j \in X_j$,$j = 1, 2, \cdots, p$,称函数 f 在 X 上是可分函数,如果存在 p 个定义在 X_j 上的函数 f_j,使得 $f(x) = \sum_{j=1}^{p} f_j(x_j)$。

命题 1[2] 假定每个函数 f_j 在 S_j 上是连续且有界的,这里 S_j 是 X_j 的一个子集合,则

$$\inf\{\sum_{j=1}^{p} f_j(x_j), x_j \in S_j, j = 1, 2, \cdots, p\} = \sum_{j=1}^{p} \inf\{f_j(x_j), x_j \in S_j\}$$

* 收稿日期:2009-06-08
资助项目:国家自然科学基金(No. 10171118)
作者简介:赫振华,女,硕士研究生,研究方向为最优化理论与算法;通讯作者:白富生,E-mail:fsbai@cqnu.edu.cn

2 QPM + APP 方法

为简单起见,考虑如下带有一个等式的约束的可分优化问题[P$_1$]

$$\min f(x) = \sum_{j=1}^{L} f_j(x_j)$$
$$\text{s.t. } h(x) = \sum_{j=1}^{L} h_j(x_j) = 0, x \in X, x_j \in X_j \tag{2}$$

其中 f_j :$\mathbf{R}^{n_j} \to \mathbf{R}$ 和 h_j :$\mathbf{R}^{n_j} \to \mathbf{R}$,$j = 1,2,\cdots,L$ 连续可微,$X = \prod_{j=1}^{L} X_j$,每个 X_j 是 $\mathbf{R}^{n_j}(j = 1,2,\cdots,L)$ 的非空紧子集,这里 $n_1 + n_2 + \cdots n_L = n(L \leqslant n)$。对于多于一个等式约束的问题可类似考虑。

问题[P$_1$]的二次罚函数为

$$F_c(x) = F_c(x_1, x_2, \cdots, x_L) = \sum_{j=1}^{L} f_j(x_j) + \frac{c}{2}(h(x))^2 \tag{3}$$

相对应的无约束罚问题为

$$[\mathrm{QP}_1] \min_{x \in X} F_c(x) = F_c(x_1, x_2, \cdots, x_L) = \sum_{j=1}^{L} f_j(x_j) + \frac{c}{2}(h(x))^2 \tag{4}$$

在问题(4)中,二次罚项 $(h(x))^2$ 破坏了可分性,因此可用 APP 方法[1] 将二次罚项在当前点 $x^k = (x_1^k, x_2^k, \cdots, x_L^k)$ 线性化,再添加一个二次可分项,从而可将问题[QP$_1$] 分解成几个子问题。

为了更清楚地解释,令 $L = 2$,即考虑如下问题[P$_2$]

$$[\mathrm{P}_2] \min f(x) = \sum_{j=1}^{2} f_j(x_j)$$
$$\text{s.t. } h(x) = \sum_{j=1}^{2} h_j(x_j) = 0, x \in X \tag{5}$$

其中 f_j :$\mathbf{R}^{n_j} \to \mathbf{R}$ 和 h_j :$\mathbf{R}^{n_j} \to \mathbf{R}$,$j = 1,2$,是连续可微,$X = \prod_{j=1}^{2} X_j$,每个 X_j 是 $\mathbf{R}^{n_j}(j = 1,2)$ 的非空紧子集,$n_1 + n_2 = n(2 \leqslant n)$。

问题[P$_2$]相对应的罚问题为

$$[\mathrm{QP}_2] \min_{x \in X} F_c(x) = \sum_{j=1}^{2} f_j(x_j) + \frac{c}{2}(h(x))^2 \tag{6}$$

应用 APP 方法,将二次罚项$(h(x))^2$ 在当前点 $x^k = (x_1^k, x_2^k)$线性化,再添加一个二次可分项

$$2c(\|x_1 - x_1^k\|^2 + \|x_2 - x_2^k\|^2)$$

则上述问题[QP$_2$]近似等价于下面的问题(7)式

$$\min_{x_1 \in X_1, x_2 \in X_2} \sum_{j=1}^{2} f_j(x_j) + c\sum_{j=1}^{2} A_j \big|_{x = x^k} \cdot x_j + 2c(\|x_1 - x_1^k\|^2 + \|x_2 - x_2^k\|^2) \tag{7}$$

即
$$\min_{x_1 \in X_1, x_2 \in X_2} \sum_{j=1}^{2} (f_j(x_j) + cA_j \big|_{x = x^k} \cdot x_j + 2c\|x_j - x_j^k\|^2) \tag{8}$$

这里 $A_j = h(x)\dfrac{\partial h_j}{\partial x_j}, j = 1,2$,则问题[P$_2$] 对应的罚函数 F_c 的近似极小可分解为下面的两个子问题

$$\min_{x_j \in X_j} f_j(x_j) + cA_j \big|_{x = x^k} \cdot x_j + 2c\|x_j - x_j^k\|^2, j = 1,2 \tag{9}$$

下面给出问题[P$_2$]对应的 QPM + APP 算法。

步 0 初始化。选取初始点(可以是非可行点)$x^k = (x_1^k, x_2^k) \in X$,选取初始罚因子 $c_0 > 0$,控制精度 $\varepsilon > 0$,令 $k = 0, N > 1$;

步 1 求解 2 个子问题
$$x_1^{k+1} = \arg\min_{x_1 \in X_1} f_1(x_1) + cA_1 \big|_{x = x^k} \cdot x_1 + 2c\|x_1 - x_1^k\|^2$$
$$x_2^{k+1} = \arg\min_{x_2 \in X_2} f_2(x_2) + cA_2 \big|_{x = x^k} \cdot x_2 + 2c\|x_2 - x_2^k\|^2$$

其中 $A_j = h(x)\dfrac{\partial h_j}{\partial x_j}, j = 1,2$,得到一个新的迭代点 $x^{k+1} = (x_1^{k+1}, x_2^{k+1})$;

步 2　终止准则和罚参数的修正。若 $|h(x^{k+1})| \leqslant \varepsilon$ 则停止,x^{k+1} 是原问题[P_2] 的一个近似全局极小点。否则增大罚因子,即 $c_{k+1} = Nc_k, k = k+1$,转步 1。

注 1　利用文献[1]中 ALR + APP 方法求解原问题[P_2]的最优解,需要不断地修正罚因子和拉格朗日乘子两项,而通过算法 QPM + APP 求解时仅需修正罚因子 c。

3 QPM + BCD 方法

考虑问题(4)式,可利用 BCD 方法(非线性 Gauss-Seidel 方法)把问题(4)式分解成 L 个子问题进行求解。

命题 2[15]　(非线性 Gauss-Seidel 方法的收敛性)假定 $F: \mathbf{R}^n \mapsto \mathbf{R}$ 在集合 X 上连续可微且凸,同时假定当 x 的其他分量都给定时,对每个 i,F 是 x_i 的严格凸函数,若 x^k 是由非线性 Gauss-Seidel 得到的一个序列,则 x^k 的极限点可使 F 在 X 达到极小。

据命题 2 可知,当 x 的其他分量都给定,$F_c(x)$ 是关于 x_i 的严格凸函数时,$i = 1,2,\cdots,L$,非线性 Gauss-Seidel 方法的收敛性可以保证。

设当前点为 $x^k = (x_1^k, \cdots, x_i^k, \cdots, x_L^k)$,则根据非线性 Gauss-Seidel 方法得知第 i 个子问题为

$$\min_{x_i \in X_i} F_c(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \cdots, x_L^k), i = 1,2,\cdots,L \tag{10}$$

这样就得到一个新的迭代点 $x^{k+1} = (x_1^{k+1}, \cdots, x_i^{k+1}, \cdots, x_L^{k+1})$。

下面给出 QPM + BCD 算法。

步 0　初始化。选取初始点(可以是非可行点)$x^k = (x_1^k, \cdots, x_L^k) \in X$,选取初始罚因子 $c_0 > 0$,控制精度 $\varepsilon > 0$,令 $k = 0, N > 1$;

步 1　求解 L 个子问题

$$x_i^{k+1} = \arg\min_{x_i \in X_i} F_c(x_1^{k+1}, \cdots, x_{i-1}^{k+1}, x_i, x_{i+1}^k, \cdots, x_L^k), i = 1,2,\cdots,n$$

得到一个新的迭代点 $x^{k+1} = (x_1^{k+1}, \cdots, x_i^{k+1}, \cdots, x_L^{k+1})$;

步 2　终止准则和罚参数的修正。若 $|h(x^{k+1})| \leqslant \varepsilon$ 则停止,x^{k+1} 是原问题[P_3] 的一个近似全局极小点,否则增大罚因子,即 $c_{k+1} = Nc_k, k = k+1$,转步 1。

4 数值例子

这里给出一些数值例子进一步说明本文提出的算法是有效的。

对于例 1 和例 2,分别通过算法 QPM + APP 与算法 ALR + APP[1]、算法 QPM + BCD 与算法 ALR + BCD[1] 做比较来求解,计算过程中取步 0 中 $\varepsilon = 10^{-5}$,数值结果记录在下面的表格中,其中 x_0 为任意选取的初始点;x^* 代表原问题的最优解;$f(x^*)$ 表示原问题的最优值。

例 1[16]

$$\min f(x) = x_1^2 + x_2^2 + x_3^2$$
$$\text{s.t.}\ \ h(x) = x_1 + x_2 + x_3 - 3 = 0$$

例 1 相对应的带有二次罚项的罚问题为

$$\min_{x \in X} F_c(x) = x_1^2 + x_2^2 + x_3^2 + \frac{c}{2}(x_1 + x_2 + x_3 - 3)^2 \tag{11}$$

通过算法 QPM + APP 可将问题(11)式分解为下面 3 个独立的子问题,假设当前点为 x^k,

$$\min_{x_i} x_i^2 + c(x_1^k + x_2^k + x_3^k - 3)x_i + 2c\|x_i - x_i^k\|^2, i = 1,2,3 \tag{12}$$

可以利用 QPM + BCD 算法把问题(11)式分解成如下 3 个子问题

$$x_i^{k+1} = \arg\min_{x_i} F_c(x_1^k, x_i, x_3^k), i = 1,2,3 \tag{13}$$

表1记录了利用不同算法、取不同的初始点得到的例1的数值结果。由表1可知,算法QPM+APP与QPM+BCD的计算效果和算法ALR+APP与ALR+BCD的计算效果是基本一致的。事实上,在文献[16]中,一个已知的全局极小点为$x^*=(1,1,1)$,相应的全局极小值是$f^*=3$。

表1 例1的数值结果

Tab.1 The numeric resules of example 1

算法	x^0	x^*	$f(x^*)$	$h(x^*)$	算法	x^0	x^*	$f(x^*)$	$h(x^*)$
QPM+APP	$\begin{pmatrix}0\\0\\0\end{pmatrix}$	$\begin{pmatrix}1.0000\\1.0000\\1.0000\end{pmatrix}$	3.0000	-20.9700×10^{-6}	QPM+BCD	$\begin{pmatrix}0\\0\\0\end{pmatrix}$	$\begin{pmatrix}1.0601\\0.9977\\0.9421\end{pmatrix}$	3.0070	-9.0509×10^{-6}
ALR+APP	$\begin{pmatrix}0\\0\\0\end{pmatrix}$	$\begin{pmatrix}1\\1\\1\end{pmatrix}$	3.0000	-9.3582×10^{-7}	ALR+BCD	$\begin{pmatrix}0\\0\\0\end{pmatrix}$	$\begin{pmatrix}1\\1\\1\end{pmatrix}$	3.0000	-1.1187×10^{-7}
QPM+APP	$\begin{pmatrix}0.5\\0.5\\0.5\end{pmatrix}$	$\begin{pmatrix}1.0000\\1.0000\\1.0000\end{pmatrix}$	3.0000	-6.1003×10^{-6}	QPM+BCD	$\begin{pmatrix}0.5\\0.5\\0.5\end{pmatrix}$	$\begin{pmatrix}1.0601\\0.9977\\0.9421\end{pmatrix}$	3.0070	-9.0509×10^{-6}
QPM+APP	$\begin{pmatrix}0.5\\0.5\\0.5\end{pmatrix}$	$\begin{pmatrix}1\\1\\1\end{pmatrix}$	3.0000	-6.8474×10^{-7}	QPM+BCD	$\begin{pmatrix}0.5\\0.5\\0.5\end{pmatrix}$	$\begin{pmatrix}1\\1\\1\end{pmatrix}$	3.0000	-1.1187×10^{-7}

例2[10]

$$\min f(x) = x_1^2 + x_2^2$$
$$\text{s.t. } h(x) = x_1 + x_2 - 10 = 0$$

表2记录了利用不同算法、取不同的初始点得到的例2的数值结果,由表2可知,算法QPM+APP与QPM+BCD的计算效果和算法ALR+APP与ALR+BCD的计算效果是基本一致的。事实上,在文献[10]中,根据拉格朗日松弛方法,求得例2的一个全局极小点为$x^*=(5,5)$,全局极小值为$f=50$。

表2 例2的数值结果

Tab.1 The numeric resules of example 2

算法	x^0	x^*	$f(x^*)$	$h(x^*)$	算法	x^0	x^*	$f(x^*)$	$h(x^*)$
QPM+APP	$\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}5.0000\\5.0000\end{pmatrix}$	50.0000	-7.9736×10^{-6}	QPM+BCD	$\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}5.2182\\4.7818\end{pmatrix}$	50.0952	-8.6156×10^{-6}
ALR+APP	$\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}5.0004\\5.0004\end{pmatrix}$	50.0083	8.3286×10^{-4}	ALR+BCD	$\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}5.0001\\4.9999\end{pmatrix}$	50.0000	3.5578×10^{-7}
QPM+APP	$\begin{pmatrix}4\\4\end{pmatrix}$	$\begin{pmatrix}5.0000\\5.0000\end{pmatrix}$	50.0000	-8.21097×10^{-6}	QPM+BCD	$\begin{pmatrix}4\\4\end{pmatrix}$	$\begin{pmatrix}5.2182\\4.7818\end{pmatrix}$	50.0952	-8.6156×10^{-6}
ALR+APP	$\begin{pmatrix}4\\4\end{pmatrix}$	$\begin{pmatrix}5.0004\\5.0004\end{pmatrix}$	50.0083	8.3286×10^{-7}	ALR+BCD	$\begin{pmatrix}4\\4\end{pmatrix}$	$\begin{pmatrix}5.0001\\4.9999\end{pmatrix}$	50.0000	3.5578×10^{-7}

5 结论

本文提出两种二次罚函数可分化方法,给出了相应的算法QPM+APP和QPM+BCD,并用两个简单例子来说明这两种方法的计算结果.算法QPM+APP的收敛性目前仍在考虑中。

参考文献:

[1] Reltran C,Heredia F J. Unit commitment by augmented Lagrangian relaxation:testing two decomposition approaches [J]. Journal of Optimization Theory and Applications,

2002,112(2):295-314.

[2] Mahey. Decomposition methods for mathematical programming[M]//Pardalos P,Resende M G C. Handbook of applied optimization. Oxford:Oxford University Press,2002.

[3] Engelmann B. Convexification and decomposition of separable problems[J]. Opti mization,1992,26:61-82.

[4] Chen G,Teboulle M. A proximal-based decomposition method for convex minimization problems[J]. Mathematical Programming,1993,64:81-101.

[5] Mahey P. Separarable augmented Lagrangians for the decomposition of large convex programs[J]. Investigation Operativa,1995,5:1-26.

[6] Ruszczynski A. On convergence of an augmented lagrangian decomposition method for sparse convex optimization[J]. Mathematics of Operations Research,1995,20(3):634-656.

[7] Gabay D,Mercier B. A dual algorithm for the solution of nonliear variational problems via finite-element approximations[J]. Computprs and Mathematics with Applications 1976,2:17-40.

[8] Glowinski,R. Marrocco A. Sur 1 approximation par elements finis dordre un. et la resolution par penalisation-dualited'une classe de problemes de Dirichlet nonlieaires[J]. Revue Francaise d'Automatigue Informatique et Recherche perationelle 1975,2:41-76.

[9] Bertsekas D P,Tsitsiklis J N. Parallel and distributed computation:numerical merhodss[M]. New Jersey:Prentice-Hall, 1989.

[10] Antonio J C,Roberto M,Enriqne C,et al. Decomposition techniques in mathematical programming[M]. New York: Engineering and Science Application,2006.

[11] 刘茜. 广义半无限极大极小规划的一个新的最优性条件[J]. 重庆师范大学学报(自然科学版),2009,26(2): 12-17.

[12] 杨晓琪. 约束最优化问题的非线性无约束方法[J]. 重庆师范大学学报(自然科学版),2004,21(2):1-3.

[13] Cohen G. Auxiliary problem principle and decomposition of optimization problems[J]. Journal of Optimization Theory and Applications,1980,32:277-305.

[14] Bertsekas D P. Nonlinear programming[M]. Nashua,NH: Athena Scientific,1995.

[15] Bertsekas D P,Tsitsiklis J N. Parallel and distributed computation:numerical methods[M]. Nashua,NH:Athena Scientific,1997.

[16] Mokhtar S B,Hanif D S,Shetty C M. Nonlinear programming theory and algorithms[M]. New Jersey:John wiley & sons inc,1993.

Decomposition Methods in Quadratic Penalty Function

HE Zhen-hua[1,2], *BAI Fu-shen*[1]

(1. College of Mathematics and Computer Sciences, Chongqing Normal University, Chongqing 400047;
2. Chongqing Zhongxian Senior Middle School, Zhongxian Chongqing 404303, China)

Abstract: The decomposition methods are used to solve large-scale optimization problems by decomposing them into sub-problems. In this paper we present two decomposition methods for solving separable optimization problems. We apply the Auxiliary Problem Principle (APP) method and the Block Coordinate Descent (BCD) method to the Quadratic Penalty Method (QPM) respectively and also present the corresponding QPM + APP Algorithm and QPM + BCD Algorithm. Meanwhile, In Ref. 1, for a separable problem the authors apply the APP and BCD method to the Augmented Lagrangian Relaxation (ALR) method and solve the problem, so both the dual variable and the penalty parameter must be updated. But we only update the penalty parameter by the present methods. Two numerical examples are given to show the usefulness of the presented methods by comparing with the ALR + BCD and the ALR + BCD Algorithm in Ref. 1.

Key words: separable optimization; decomposition methods; quadratic penalty method; auxiliary problem principle method; nonlinear Gauss-Seidel method

(责任编辑 黄 颖)