运筹学与控制论

DOI:10.11721/cqnuj20140403

约束不定二次规划的一个快速收敛算法*

蔡 剑
(南京航空航天大学 金城学院,南京 211156)

摘要:对不定二次规划,本文提出了一种线性化技术,将其近似地转化为一个线性规划问题;然后,结合后者的线性约束条件,提出了一个缩减子超矩形算法,该算法的主要思想是对于违犯线性约束条件的变量,从箱约束条件中先行删除,再利用分枝算法求最优值点。本文证明了算法的全局收敛性。数值算例表明,对于大规模的二次规划问题,仍能快速求出结果。

关键词:不定二次规划;线性化技术;子超矩形;全局优化
中图分类号:O221.2 **文献标志码:**A **文章编号:**1672-6693(2014)04-0012-04

在优化问题中,经常遇到不定二次规划。本文考虑带箱约束的不定二次规划问题。对非凸二次规划问题,Tao、Gao 和 Liu 提出了分枝算法[1-3],通过逐步缩小自变量的范围来求最优解;Horst 和 Rober 提出了外部逼近算法[4],Zhang 等在 FR 方法的基础上构造了 FR 型谱共轭梯度法[5],此后 Du 和 Lu 等又对文献[5]提出的 FR 型谱共轭梯度法进行了补充和修正[6-7],获得了很好的全局收敛性和数值结果。

这些算法都是通过修正搜索方向和步长达到求解目的,而搜索方向和步长公式比较复杂难懂,且目标函数是非线性的,使得编程复杂且运行速度变慢,尤其对大规模二次规划问题不适用。本文考虑到非凸二次规划函数自身的特点,它可以转化为两个凸函数的差,并利用线性化技术,找到凸函数的线性上下界,将不定二次规划问题转化为松弛线性规划问题,使得算法简单易懂,并且运行速度加快,同时考虑到不定二次规划自身所带的一般约束条件,提出了缩减子超矩形算法,不断缩小约束条件的范围,该算法和分枝算法结合,能使不定二次规划问题并不因所带约束条件增多而速度减慢,并且从理论上证明了算法的全局收敛性。

考虑不定二次规划问题

$$\begin{cases}\min & f(y)=\dfrac{1}{2}y^{\mathrm{T}}\boldsymbol{A}y+b^{\mathrm{T}}y\\ \text{s. t.} & \boldsymbol{B}y\leqslant c,\ y\in Y^0=\{y_i:0<y_i^l\leqslant y_i\leqslant y_i^u<\infty\}\end{cases}\tag{1}$$

其中 $\boldsymbol{A}=(a_{ij})_{n\times n}$,$\boldsymbol{B}=(b_{ij})_{n\times n}$ 为 n 阶实对称矩阵,b、y 和 c 均为 n 维列向量。

$$f(y)=\frac{1}{2}y^{\mathrm{T}}\boldsymbol{A}y+b^{\mathrm{T}}y=\frac{1}{2}y^{\mathrm{T}}(\boldsymbol{A}+\rho I)y+b^{\mathrm{T}}y-\frac{\rho}{2}y^{\mathrm{T}}y=\Phi(y)-\frac{\rho}{2}\Psi(y)$$

其中 $\Phi(y)=\dfrac{1}{2}y^{\mathrm{T}}(\boldsymbol{A}+\rho I)y+b^{\mathrm{T}}y$,$\Psi(y)=\dfrac{\rho}{2}y^{\mathrm{T}}y$,只要 ρ 取的足够大,可以使 $\boldsymbol{A}+\rho\boldsymbol{I}$ 为正定矩阵,所以 $\Phi(y)$,$\Psi(y)$ 为可微的凸函数,则 $f(y)$ 表示为两个凸函数 $\Phi(y)$ 与 $\dfrac{\rho}{2}\Psi(y)$ 的差。因为 $\Phi(y)$ 是凸函数,则 $\Phi(y)\geqslant\Phi(y^k)+\langle\nabla\Phi(y^k),y-y^k\rangle$,$\nabla\Phi(y^k)$ 为 $\Phi(y)$ 的一阶偏微分,$\langle\nabla\Phi(y^k),y-y^k\rangle$ 表示内积。

2005 年,Shen 提出了广义几何规划的线性化方法[8],将此线性化技术应用于凸函数 $\Psi(y)=y^{\mathrm{T}}y$,得到 $\Psi^l(y;Y,\sigma)\leqslant\Psi(y)\leqslant\Psi^u(y;Y,\sigma)$,其中 $\sigma=(\sigma_1,\sigma_2,\cdots,\sigma_n)^{\mathrm{T}}$ 是 n 维向量,且 $\sigma_i=0$ 或者 $1(i=1,2,\cdots,n)$,$e_i=(0,0,\cdots,\underset{(i)}{1},0,\cdots,0)^{\mathrm{T}}$,$z(\sigma)=z^l+\sum\limits_{i=1}^{n}\sigma_i(z_i^u-z_i^l)e_i$,$z^l=2y^l$,$z^u=2y^u$,$\Psi^l(y;Y,\sigma)=z(\sigma)^{\mathrm{T}}y+\Psi(y(\sigma))-z(\sigma)^{\mathrm{T}}y(\sigma)$,$\Psi^u(y;Y,\sigma)=z(1-\sigma)^{\mathrm{T}}y+\Psi(y(\sigma))-z(1-\sigma)^{\mathrm{T}}y(\sigma)$。

* **收稿日期:**2013-09-18 **修回日期:**2013-12-25 **网络出版时间:**2014-7-3 23:03
作者简介:蔡剑,女,讲师,研究方向为运筹学与控制论,E-mail:caijian4215806@163.com
网络出版地址:http://www.cnki.net/kcms/detail/50.1165.N.20140703.2303.003.html

从而可以将不定二次规划问题(1)近似地转化为如下松弛线性规划问题

$$\begin{cases}\min \quad g(y;Y,\sigma)\\ \text{s. t.} \quad \boldsymbol{B}y\leqslant c, y\in Y^0=\{y_i:0<y_i^l\leqslant y\leqslant y_i^u<\infty\}\end{cases} \tag{2}$$

其中 $g(y;Y,\sigma)=\Phi(y^k)+\langle\nabla\Phi(y^k),y-y^k\rangle-\frac{\rho}{2}\Psi^u(y;Y,\sigma)$。

1 缩减子超矩形算法

问题(1)中 $\boldsymbol{B}y\leqslant c$ 即为 $\sum_{j=1}^{n}b_{ij}y_j\leqslant c_i, i=1,\cdots,n$,令矩形 $Y^k=\{y\in\mathbf{R}^n:l^k\leqslant y\leqslant u^k\}$ 为问题(1)的约束区间,很明显每一个 $b_{ij}y_j$ 在区间 $[l_j^k,u_j^k]$ 上的最大最小值都是在区间 $[l_j^k,u_j^k]$ 的极点取到,结合问题(1)的线性约束,给出如下缩减矩形的算法。

步 1,令 $I=\{1,2,\cdots,n\}$,$m_i=\sum_{j=1}^{n}\max\{b_{ij}l_j^k,b_{ij}u_j^k\}$,$n_i=\sum_{j=1}^{n}\min\{b_{ij}l_j^k,b_{ij}u_j^k\}$,$i\in I$;

步 2,计算 m_i,n_i,若 $n_i>c_i$,则算法停止,直接从子超矩形中删去 Y^k;若 $n_i\leqslant c_i$,直接转步 3;

步 3,比较 m_i 与 c_i,若 $m_i<c_i$,则转步 2,继续迭代,计算 m_{i+1},n_{i+1};若 $m_i\geqslant c_i$,直接转步 4;

步 4,令 $j=1,2,\cdots,n$,若 $b_{ij}>0$,修改矩形 Y^k 的端点,令 $u_j^k=\min\left\{u_j^k,\frac{c_i-rL_i+\min\{b_{ij}l_j^k,b_{ij}u_j^k\}}{b_{ij}}\right\}$;若 $b_{ij}\leqslant 0$,令 $l_j^k=\max\left\{l_j^k,\frac{c_i-rU_i+\max\{b_{ij}l_j^k,b_{ij}u_j^k\}}{b_{ij}}\right\}$;

步 5,转步 2,继续迭代,计算 m_{i+1},n_{i+1}。

从这个算法最终得到的超矩形令其为 Y^k,与进行这个算法之前的 Y^k 相比,区间被缩减了,这个算法删除了一些不合要求的子超矩形以及一些线性约束,这对整个算法而言缩减了计算量,最终提高了收敛速度。

2 全局收敛算法

先利用缩减子超矩形算法,得到缩减的区间 Y^k。在此基础上,利用分枝定界算法来解决问题(2)中的松弛线性规划。保证收敛到全局最优解的重要因素是合适的分块策略,本文选择了一个简单且是标准的双分规则,这种方法能充分的确保收敛性,因为它总是能使所有的区间趋于 0。下面给出了全局收敛算法。令 $\beta(Y^k)$ 表示问题(2)在区间 Y^k 上的最优解,$y^k=y(Y^k)$ 表示相应的最小值点。

步 0,初始化。1) 令 $k=0$,所以活动结点的集合为 $Q_0=\{Y^0\}$,上界 $\alpha=\infty$,可行点的集合 $F=\varnothing$;2) 给定参数 $\varepsilon>0$,求解问题(2)在 Y^0 上的最值,得到其在 $y^0=y(Y^0)$ 处的最优目标值为 $\beta_0=\beta(Y^0)$,若 y^0 对问题(1)是可行的,则更新 F 和 α,$F=F\cup\{y^0\}$,$\alpha=f_0(y^0)$,若 $\alpha\leqslant\beta_0+\varepsilon$,则算法停止,$y^0$ 是问题(1)的最优解,否则,执行步 1;3) 若 y^0 对问题(1)是不可行的,即不满足约束条件,则直接转步 1。

步 1,选择 Y^k 的中点 y_{mid},若 y_{mid} 对问题(1)可行,令 $F=F\cup\{y_{\text{mid}}\}$,定义上界 $\alpha=\min\limits_{y\in F}f_0(y)$,若 $F\neq\varnothing$,则最好的可行点为 $b=\arg\min\limits_{y\in F}f_0(y)$,转步 2;若 y_{mid} 对问题(1)不可行,则直接转步 2。

步 2,根据分枝规则,将区间 Y^k 分成两个子超矩形,称包含这新的分块矩形的集合为 S^k。1) 对每一个 $Y\in S^k$,利用第 2 部分的缩减子超矩形算法,得到的新的子超矩形令为 Y,计算 $\min\limits_{y\in Y}g(y;Y,\sigma)$,$j=0,1,\cdots,m$,若 $\min\limits_{y\in Y}g(y)>\alpha$,则从 S^k 中删去相应的 Y,令 $S^k=S^k\backslash Y$,考虑 S^k 中另一元素;2) 若 $S^k\neq\varnothing$,求解相应的问题(1)的最值,得到 $\beta(Y)$ 以及相应的 $y(Y)$,对任意 $Y\in S^k$。若 $\beta(Y)>\alpha$,令 $S^k=S^k\backslash Y$,否则如步 1 中更新 α,F 和 b 的值。

步 3,剩余的分块集为 $Q_k=(Q_k\backslash Y^k)\cup S^k$,给定一个新的下界 $\beta_k=\min\limits_{Y\in Q_k}\beta(Y)$。

步 4,令 $Q_{k+1}=Q_k\backslash\{Y:\alpha-\beta(Y)\leqslant\varepsilon,Y\in Q_k\}$,若 $Q_{k+1}=\varnothing$,则算法停止,α 为问题(1)的最优值,b 为其最优解,否则令 $k=k+1$,转入步 5。

步 5,选择一活动结点满足 $Y^k=\arg\min\limits_{Y\in Q_k}\beta(Y)$,$y^k=y(Y^k)$,返回步 1。

3 算法的收敛性

问题(2)是问题(1)的线性下界,问题(2)的最优解能否收敛于问题(1)的最优解,利用中值定理证明收敛性。

定理1　对问题(1)和问题(2),在算法执行的第k步迭代中,对任意$Y\in Q_k$,设$Y=\{y:l\leqslant y\leqslant u,l_i\leqslant y_i\leqslant u_i,i=1,\cdots,n\}\subseteq Y^0$,$\lambda=(\lambda_1,\cdots,\lambda_n)^{\mathrm{T}}$,其中$\lambda_i=u_i-l_i$,$i=1,\cdots,n$,又对任意$y\in Y$,记$D(y)=f(y)-g(y)$,则$D(y)$在$Y$上的最大值$D_{\max}(y)$满足$\lim\limits_{\max\lambda_i\to 0^+}D_{\max}(y)=0$。

证明　由线性化技术,得到$g(y;Y,\sigma)=\Phi(y^k)+\langle\nabla\Phi(y^k),y-y^k\rangle-\frac{\rho}{2}\Psi^u(y;Y,\sigma)$,其中$\Psi^u(y;Y,\sigma)=z(1-\sigma)^{\mathrm{T}}y+\Psi(y(\sigma))-z(1-\sigma)^{\mathrm{T}}y(\sigma)$,所以

$$D(y)=f(y)-g(y)=\Phi(y)-\frac{\rho}{2}\Psi(y)-\Phi(y^k)-\langle\nabla\Phi(y^k),y-y^k\rangle+\frac{\rho}{2}\Psi^u(y;Y,\sigma)=$$

$$\Phi(y)-\Phi(y^k)-\langle\nabla\Phi(y^k),y-y^k\rangle+\frac{\rho}{2}[\Psi^u(y;Y,\sigma)-\Psi(y)]$$

则　$$D_{\max}(y)=\Phi(y)-\Phi(y^k)-\langle\nabla\Phi_j(y^k),y-y^k\rangle+\frac{\rho}{2}[z(1-\sigma)^{\mathrm{T}}y+\Psi(y(\sigma))-z(1-\sigma)^{\mathrm{T}}y(\sigma)-\Psi(y)]$$

由中值定理有$\Psi(y)-\Psi(y(\sigma))=\nabla\Psi(\xi)(y-y(\sigma))$,其中$\xi=\theta y+(1-\theta)y(\sigma)$,$0<\theta<1$,则有

$$D_{\max}(y)=\Phi(y)-\Phi(y^k)-\langle\nabla\Phi_j(y^k),y-y^k\rangle+\frac{\rho}{2}[z(1-\sigma)^{\mathrm{T}}-\nabla\Psi(\xi)][y-y(\sigma)]$$

又$\lambda_i=u_i-l_i$,当$\max\lambda_i\to 0$时,$y(\sigma)\to y$,则

$$\lim_{\max\lambda_i\to 0^+}\mathrm{D}_{\max}(y)=\lim_{\lambda\to 0^+}[\Phi(y)-\Phi(y^k)-\langle\nabla\Phi(y^k),y-y^k\rangle]+\lim_{\max\lambda_i\to 0^+}\frac{\rho}{2}[z(1-\sigma)^{\mathrm{T}}-\nabla\Psi(\xi)][y-y(\sigma)]=0$$

证毕

注　定理1说明了当子超矩形Y的宽度充分小时,问题(2)的解充分接近问题(1)的解。

定理2　设问题(1)的全局最优解存在,则算法或者在有限步内求得问题(1)的全局最优解,或者算法产生的迭代序列的极限点必为问题(1)的全局最优解。

证明　对于算法的每一次迭代,$k=0,1,\cdots$,假设$\beta_k\leqslant\min\limits_{y\in Y^0}f(y)$,$Y^k\in\arg\min\limits_{Y\in Q_k}\beta(Y)$,$y^k=y(Y^k)\in Y^k$为真。很显然$\{\beta_k\}$是一非下降序列,且有上界,因此$\{\beta_k\}$存在极限$\beta:=\lim\limits_{k\to\infty}\beta_k=\min\limits_{y\in D}f(y)$。又$\{y^k\}$是在紧集上的序列,因此它存在收敛子序列,对任意$\hat{y}\in Y^a$,存在序列$\{y^r\}\subset\{y^k\}$,且$\lim\limits_{r\to\infty}y^r=\hat{y}$,由给定的算法,得到算法第二步中的分块集的细分在$Y^0$上是穷举的,且被选择用来在第2步进行细分的集合的边界在逐步得到改进,同时问题(1)中使用的线性子函数g在Y^0上是严格一致的,因而存在一下降序列$\{Y^q\}\subset Y^r$,其中$Y^r\in Q_r$,且$y^q\in Y^q$,$\beta_q=\beta(Y^q)=g(y^q;Y^q)$,$\lim\limits_{q\to\infty}y^q=\{\hat{y}\}$,此外,因为$g(y)$在$Y^0$上是严格一致的,所以有$\lim\limits_{q\to\infty}\beta_q=\beta=f(\hat{y})$,又因为$Y^0$是闭集,因此$\hat{y}\in Y^0$,则$\hat{y}\in D$,所以$\beta=f(\hat{y})=\min\limits_{y\in Y^0}f(y)$,$\hat{y}\in Y^*$。　证毕

4 数值例子

考虑带一般约束的大规模不定二次规划问题

$$\begin{cases}\min\quad f(y)=\dfrac{1}{2}y^{\mathrm{T}}\boldsymbol{A}y+b^{\mathrm{T}}\mathrm{y}\\ \mathrm{s.\,t.}\quad \boldsymbol{B}y\leqslant c,0\leqslant y_i\leqslant 10,i=1,\cdots,n\end{cases}$$

其中,矩阵$\boldsymbol{A}$,$\boldsymbol{B}$,向量b和c中的元素都是由0和1之间的数随机产生的。由C++语言编程运行,结果见表1,其中n为所有矩阵的阶数及所有向量的维数,$t(s)$为本算法运行时间。从表1可以看出,随着矩阵阶数的增大,运行时间也在增长,但算法是可行的,说明此算法对运行大规模不定二次规划问题有效。

表1　带一般约束的大规模不定二次规划运行问题的结果

Tab.1　The result of large-scale DC programming with general bounds

n	10	16	18	26	30	40	64	80	100
$t(s)$	0.8	3.51	2.72	10.34	176.2	81.26	476.28	312.3	972.6

参考文献:

[1] Tao P D. A combined DC optimization-ellipsoidal branch and bound algorithm for dolving nonconvex quadratic pro-

gramming problems[J]. Joural of Combinatorial Optimization,1998,2(1):9-28.

[2] Gao Y L,Shang Y L,Zhang L S. A branch and reduce approach for solving nonconvex quadratic programming problems with quadratic constraints[J].Operations Research Transactions,2005,9(2):23-36.

[3] Liu G S,Zhang J Z. A new branch and bound algorithm for solving quadratic programs with linear compleme Ntarity constraints[J].Journal of computational and Applied Mathematica,2002,146(1):77-87.

[4] Horst R,Rober U. Convergent outer approximation algorithm for solving unary problems[J]. Journal of Global Optimization,1998,13(1):123-149.

[5] Zhang L,Zhou W,Li D. Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search[J]. Numer Math,2006,104(4):561-572.

[6] Du S Q,Chen Y Y. Global convergence of a modified spectral FR conjugate gradient method[J]. Applied Mathematics and Computation,2008,202(2):766-770.

[7] Lu A Q,Liu H M,Zheng X Y,et al. A variant spectral-type FR conjugate gradient method and its global convergence [J]. Applied Mathematics and Computation, 2011, 217(12):5547-5552.

[8] Shen P P. Linearization method of global optimization for generalized geometric programming[J]. Applied Mathematics and Computation,2005,162(1):353-370.

Operations Research and Cybernetics

Fast Convergence Algorithm for Indefinite Quadratic Programming Problems with the General Bound

CAI Jian

(College of Jincheng, Nanjing University of Aeronautics and Astronautics, Nanjing 211156, China)

Abstract: A linear transformation method is presented to solve indefinite quadratic programming in this paper. First, the method is to translate indefinite quadratic programming into a relaxed linear programming. Then, according o the linear constraints the reduced sub-super-rectangle algorithm is proposed. The variable which unsatisfied linear constraints is deleted from box constraints by using the algorithm. After that, the results of optimal point are calculated by using the branching algorithm. The global convergence of the algorithm is proved in this paper. In order to verify the validity of the algorithm, a large scale indefinite quadratic programming is increased. The new algorithm is still applicable. The numerical example shows that the algorithm can quickly calculate the results.

Key words: indefinite quadratic programming; linear technology; sub-super-rectangle; global optimization

(责任编辑　黄　颖)