上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

数学科学学院
SJTU

混合变量优化的网格自适应直接搜索算法

Abramson, M. A., Audet, C., Chrissis, J. W., Walston, J. G. (2009).
Mesh Adaptive Direct Search Algorithms for Mixed Variable Optimization.
Optimization Letters

林元莘

2023 年 12 月 31 日

目录

混合变量优化的困难

混合变量优化问题的特征便是有**连续性变量和分类变量的组合**。分类变量也可以是提前定义的类别，类别变量也不一定可以数值化。
混合变量优化问题会遇到以下困难：

- 混合变量优化问题会具有**十分复杂的可行域和解空间**，在算法设计上和计算复杂度上带来巨大挑战。
- 传统的优化算法，例如分支定界法，就不能直接运用在此类问题上。
- 在目标函数或者限制函数是类似黑箱的函数时，在一些情况下，变量的重新表述不可行。

现有的相关算法

- 广义模式搜索算法 (generalized pattern search，GPS) 算法: [1]
该算法系统地评估位于当前迭代点周围的网格或晶格上的试验点。该网格由一组有限的正向跨度方向定义，这些方向使得空间中的任何向量都可以表示为该集合中方向的非负线性组合。
- 但是，GPS 算法在处理非线性的优化问题时，需要生成无限条切线锥的正生成方向，这是不可行的。后来有人提出用增广拉格朗日逼近和过滤 GPS 算法来解决这个问题，但算法局限性依然存在。
- 本文作者基于 filter-GPS 算法上提出了一种新的算法，即**网格自适应直接搜索算法** (mesh adaptive directional search，MADS)。这个算法用较好的普适性，并且有较强的收敛性。

1 研究背景

2 **算法内容**

3 收敛性证明

4 总结

问题介绍

上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

混合变量优化

我们记可行域为 Ω，目标函数为 $f:\Omega \to R\cup\{\infty\}$，x 为变量。我们给出连续型变量和离散型变量的最大维数 n^c, n^d，令变量 $x=(x^c, x^d)$ 分为连续和类别的两部分，那么我们就有 $x^c\in\Omega^c\subseteq\mathbb{R}^{n^c}, x^d\in\Omega^d\subseteq\mathbb{Z}^{n^d}$。则优化问题为

$$\min_{x\in\Omega} f(x)$$

$$\Omega=\bigcup_{x^d\in\Omega^d}\left(\Omega^c(x^d)\times\{x^d\}\right)$$

在真实算法过程中，我们利用障碍法得到新的障碍目标函数 $f_\Omega = f+\psi_\Omega$，其中 ψ_Ω 在 Ω 处取 0，其他处取 ∞。[2]

三个重要步骤

使用 MADS 算法进行每一步迭代都要经历三个步骤

- 选择搜索步骤 (optional search step)
- 局部轮询步骤 (local poll step)
- 扩展轮询步骤 (extended poll step)

选择搜索步骤

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

网格 M_k

每次迭代都要生成一个新的网格 M_k

$$M_k = \bigcup_{i=1}^{i_{max}} M_k^i \times \Omega^d$$

$$M_k^i = \bigcup_{x \in V_k} \{x^c + \Delta_k^m D^i z : z \in \mathbb{N}^{|D^i|}\}$$

其中网格大小参数 $\Delta_k^m > 0$ 以及 V_k 表示所有在第 k 步迭代已经被评估过的测试点。

- 搜索步骤允许在任意有限的网格点集合下，评估 f_Ω 的大小，搜索策略不影响收敛性。
- 好的搜索策略（例如，利用到替代法）[3] 可以提高算法的表现。

局部轮询步骤

MADS 框架

在每一步迭代时，MADS 框架如下定义：

$$P_k(x) = \{(x^c + \Delta_k^m d, x^d) : d \in D_k(x)\} \subset M_k$$

其中 $D_k(x) \subseteq D^{i_0}$ 是关于第 i_0 个离散变量的正向生成方向的集合。

- 局部轮询主要评估的便是 $P_k(x_k) \bigcup N(x_k)$ 上的点。
- 如果局部轮询没有找到更小的目标函数值，那么由扩展轮询步骤来查找部分有希望的值的周边值。

网格与框架的可视化

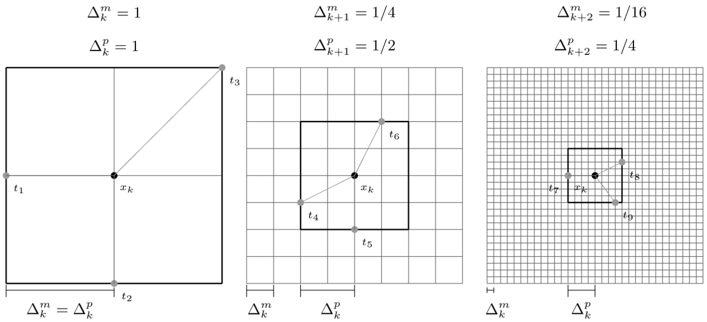

图: 当 MADS 框架 $P_k = \{x_k + \Delta_k^m d : d \in D_k\} = \{t^1, t^2, t^3\}$ 并且网格大小参数 Δ_k^m 和轮询大小参数 Δ_k^p 不同时的例子

扩展轮询步骤

扩展轮询点

$$X_k(\xi_k) = \bigcup_{y_k \in N_k^{\xi_k}} \bigcup_{j=1}^{J_k} P_k(y_k^j)$$

其中 $N_k^{\xi_k} = \{y \in N(x_k) : f_\Omega(x_k) \le f_\Omega(y) \le f_\Omega(x_k) + \xi_k\}$

扩展轮询步骤在有限点集合里进行轮询，终止点 $z_k = y_k^{J_k}$ 出现时要满足存在 $d \in D_k(z_k)$ 使得 $f_\Omega(z_k^c + \Delta_k^m d, z_k^d) < f_\Omega(x_k)$ 或者对任意 $d \in D_k(z_k)$ 有 $d \in D_k(z_k)$ 使得 $f_\Omega(z_k^c + \Delta_k^m d, z_k^d) \ge f_\Omega(x_k)$。

以上三个步骤的任意一个找到了更好的网格点，那么迭代停止，形成新的网格中心点 x_{k+1}。如果没有找到，则细化网格点。

网格参数更新

我们设置一个确定的有理数 $\tau > 1$ 和两个整数 $w^- \leq -1, \quad w^+ \geq 0$，网格大小参数 Δ_k^m 如下更新：

网格大小参数更新

$$\Delta_{k+1}^m = \tau^{w_k} \Delta_k^m$$

$$w_k \in \begin{cases} \{0, 1, \ldots, w^+\} & \text{若找到了更优的网格点} \\ \{w^-, w^- + 1, \ldots, -1\} & \text{其他情况} \end{cases}$$

找到更优点时，我们更新网格中心点并且放松网格大小，防止过多的搜索；如果没有找到新的更优点，我们保留网格中心点，通过缩小网格大小来尝试找到更优的点。

算法 1: 网络自适应方向搜索算法 (MADS)[2]

Input: 目标函数 f，可行域 Ω，算法参数 ξ, Δ_0^m，起始点 x_0
Output: 最优点 x^*

设置参数 $\xi_0 \geq \xi$，$\Delta_0^p \geq \Delta_0^m$。初始迭代 $k=0$;
while *True* **do**
  计算 M_k，在 M_k 上进行**搜索步骤**;
  if 找到更优点 **then**
    更新中心点和参数，进入下一步循环;
  else
    计算 $P_k(x_k)$，在 $P_k(x_k) \bigcup N(x_k)$ 上进行**局部轮询步骤**;
    if 找到更优点 **then**
      更新中心点和参数，进入下一步循环;
    else
      计算 $X(\xi_k)$，在 $X(\xi_k)$ 上进行**扩展轮询步骤**;
      if 找到更优点 **then**
        更新中心点和参数，进入下一步循环;
      else
        保留中心点，更新网格参数;
      end
    end
  end
  if 迭代次数达到上限或者中心点无法继续更新 **then**
    break
  end
  $k \leftarrow k+1$;
end
return x_{k+1};

MADS 通过三重遍历动作来进行最优点的迭代，通过参数更新来达到网格的细化或者粗化，达到**自适应网格**的效果。

前提假设

收敛性分析需要依靠一些标准的假设。

- 初始点 x_0 满足 $f_\Omega(x_0) < \infty$。
- 所有迭代点列 $\{x_k\}$ 都落在一个紧集内。
- 离散领域的集合 $N(x_k)$ 都在网格 M_k 中。
- $\liminf_{k\to\infty} \Delta_k^p = \liminf_{k\to\infty} \Delta_k^m = 0$
- 存在一个最小框架中心的改良子序列 $\{x_k\}$，存在极限点 $\hat{x} = \lim_{k\in K} x_k, \quad \hat{y} = \lim_{k\in K} y_k$。对于每次扩展轮询步骤从 y_k 开始生成的终止点 z_k 并且 $\lim_{k\in K} \Delta_k^p = 0$ 也有极限点 $\hat{z} = \lim_{k\in K} z_k$.
- $\hat{y} \in N(\hat{x})$

概念引入

超切向量

向量 $v \in R^{n^c}$ 是一个在点 x 上，对于 Ω 的连续变量的超切向量。如果其满足 $\exists\epsilon > 0$，使得对于所有 $y \in B_\epsilon(x^c)$，$w \in B_\epsilon(v)$ 和 $t \in (0, \epsilon)$，满足 $(y + tw, x^d) \in \Omega$。
Ω 在 x 点的所有超切向量形成的集合 $T_\Omega^H(x)$ 称为超切锥

Clarke 切向量

向量 $v \in R^{n^c}$ 是一个在点 x 上，对于 Ω 的连续变量的 Clarke 切向量。如果任意向量列 $\{y_k\}$ 收敛到 x^c，任意正实数列 $\{t_k\}$ 收敛到 0，存在收敛到 v 的向量列 $\{w_k\}$ 满足 $(y_k + t_k w_k, x^d) \in \Omega$。
所有 Clarke 切向量集合 $T_\Omega^{Cl}(x)$ 称为 Clarke 切锥。
记 x 点的切锥记作 $T_\Omega^{C_0}(x)$，如果 $T_\Omega^{Cl}(x) = T_\Omega^{C_0}(x)$，那么称 Ω 在 x 点正则。

广义求导

上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

此时我们的函数性质并不确定，要在这里使用最优化的相关理论，则需要引入广义方向导数和广义梯度的概念。

广义方向导数

$$f^\circ(x;v) = \limsup_{y\to x^c,(y,x^d)\in\Omega,t\downarrow 0,(y+tv,x^d)\in\Omega} \frac{f(y+tv,x^d)-f(y,x^d)}{t}$$

广义梯度

$$\partial f(x) = \{s\in\mathbb{R}^{n^c} : f^\circ(x;v)\geq v^T s, \forall v\in\mathbb{R}^{n^c}\}$$

收敛性证明

上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

我们给出 Clarke 稳定点的定义，在证明中在混合变量的意义下利用基于 Clarke 理论的稳定点

Clarke 稳定点

让 f 在 x^* 附近 Lipschitz 连续。如果对于任意方向 v 属于 Clarke 切锥或者切锥，满足 $f^\circ(x^*;v) \geq 0$。那么我们就称 x^* 为 Clarke 或者切稳定点。
若该点负梯度存在，并且属于 Clarke 切锥或者切锥的极线，就称为 Clarke 或切 KKT 稳定点

MADS 算法会产生一列改良点列，我们希望这个点列可以收敛到稳定点。

定理

令 $\hat{w}$ 表示迭代中的改良点子列的极限点。如果 f 在 $\hat{w}$ 上 Lipschitz 连续，则对于超切锥 $T_\Omega^H(\hat{w})$ 里的任意改良方向 v，满足 $f^\circ(\hat{w};v) \geq 0$

收敛性证明

定理

$\hat{x}$ 是所有框架的最小中心点的极限点。假设 $T_\Omega^H \neq \emptyset$，并且改良方向在其中渐近致密。

- 如果 f 在 $\hat{x}$ 附近 Lipschitz 连续，那么 $\hat{x}$ 是一个 Clarke 稳定点。
- 如果 f 在 $\hat{x}$ 处严格可导，那么 $\hat{x}$ 是一个 clarke KKT 稳定点。

若 Ω 在 $\hat{x}$ 上正则，

- 如果 f 在 $\hat{x}$ 附近 Lipschitz 连续，那么 $\hat{x}$ 是一个切稳定点。
- 如果 f 在 $\hat{x}$ 处严格可导，那么 $\hat{x}$ 是一个切 KKT 稳定点。

对于扩展轮询步骤的终止点，也有类似如上结果，说明了该方法有收敛性，是可靠实用的。

总结

- 本文主要基于混合变量优化问题，提出了 MADS 网格自适应直接搜索算法。
- 解释了 MADS 算法组成的三个步骤，并且给出算法流程。
- 基于 Clarke 稳定的理论，证明算法具有收敛性

优点：较先前算法扩展了方法，可以处理混合变量问题，并且给出收敛性证明。
缺点：网格搜索计算开销巨大，需要较大存储空间和运算量。
展望：对 MADS 这一普适性算法继续改进，并且可以将其扩展到随机噪声或者多目标问题上。[4] [5]

参考文献

[1] AUDET C, DENNIS JR J E. Analysis of generalized pattern searches[J]. SIAM Journal on optimization, 2002, 13(3): 889-903.

[2] AUDET C, DENNIS JR J E. Mesh adaptive direct search algorithms for constrained optimization[J]. SIAM Journal on optimization, 2006, 17(1): 188-217.

[3] AUDET C, DENNIS JR J E, LE DIGABEL S. Globalization strategies for mesh adaptive direct search[J]. Computational Optimization and Applications, 2010, 46(2): 193-215.

[4] ABRAMSON M A, AUDET C, CHRISSIS J W, et al. Mesh adaptive direct search algorithms for mixed variable optimization[J]. Optimization Letters, 2009, 3: 35-47.

[5] AUDET C, SAVARD G, ZGHAL W. A mesh adaptive direct search algorithm for multiobjective optimization[J]. European Journal of Operational Research,

谢谢

林元莘 · 混合变量优化的网格自适应直接搜索算法