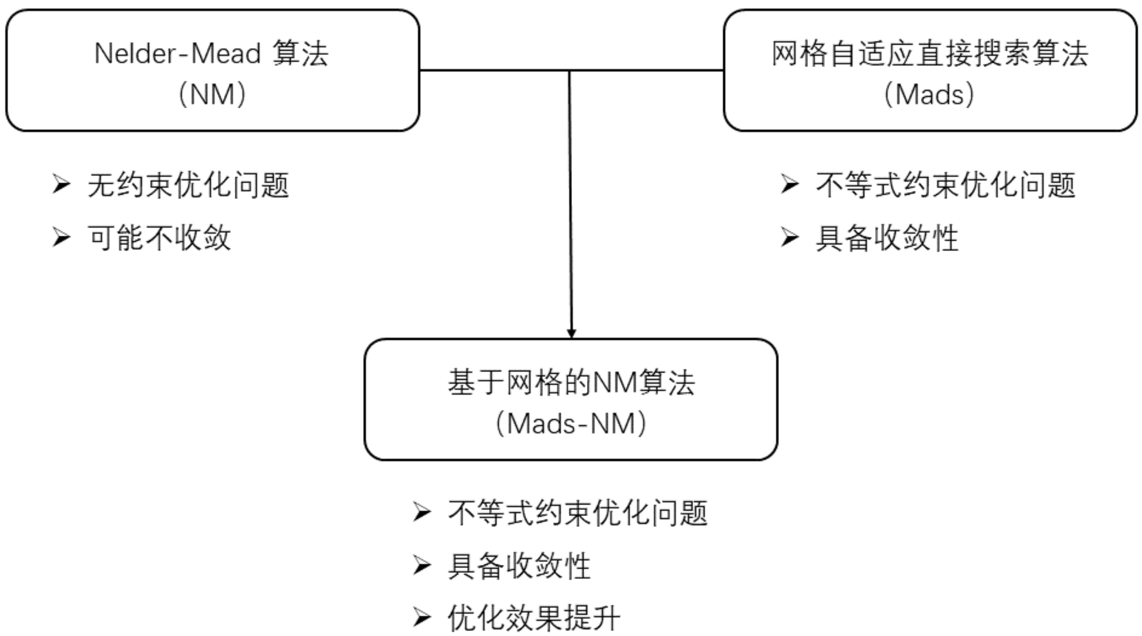
基于网格的NM不等式约束优化算法[1]总结报告

侯力广 521070910043

1.引言

在黑盒优化问题中：Nelder-Mead(NM)算法是一种处理无约束优化问题的算法，通过不断替换单纯性中最差的点来实现下降，但无法保证算法的收敛性；网格自适应方向搜索(Mads)算法是一种常用于处理不等式约束优化问题的算法，该方法有严谨的收敛性证明。在文献[1]中，作者将NM算法与Mads算法进行融合，提出的Mads-NM算法不仅可以解决不等式约束优化问题，具备严谨的收敛性证明，而且相较于NM和Mads均有明显的数值效果提升（文章脉络示意图如下）。



本次报告将分为四部分：NM无约束优化算法、Mads不等式约束优化算法、Mads-NM不等式约束优化算法，以及三种方法的效果比较。

2.NM无约束优化算法

2.1 目标问题

考虑无约束优化问题

$$\min_{x \in \mathrm{R}^n} f(x)$$

其中$f:\mathbb{R}^n \mapsto \mathbb{R}$为目标函数。

2.2 基本概念

♠ **定义2.1 (控制)：** $x,y\in\mathbb{R}^n$，若$f(x)<f(y)$，称x控制y，记作$x\prec y$.

♠ **定义2.2 (更旧)：** 若算法中x在y之前生成，则称x比y旧。定义函数$\mathrm{Older}:\mathbb{R}^n\times\mathbb{R}^n\longmapsto\mathbb{R}$

$$\mathrm{Older}(x,y)=\begin{cases} x & \text{若 } x \text{ 在 } y \text{ 之前生成} \\ y & \text{其他} \end{cases}$$

$\mathrm{Older}(x,y)$会输出x,y中更旧的那个。

♠ **定义2.3 (更好)：** 定义函数$\mathrm{Best}:\mathbb{R}^n\times\mathbb{R}^n\longmapsto\mathbb{R}$

$$\mathrm{Best}(x,y)=\begin{cases} x & \text{若 } x\prec y \\ y & \text{若 } y\prec x \\ \mathrm{Older}(x,y) & \text{若 } f(x)=f(y) \end{cases}$$

易证Best具有传递性。若$x=\mathrm{Best}(x,y)$，则称x比y更好。

♠ **定义2.4 (有序单纯形)：** 将$\mathbb{R}^n$中$n+1$个仿射无关的点排序为$y^0,y^1,\cdots y^n$，使得$y^{i-1}\prec y^i(i=1,2,\cdots,n)$，则集合

$$\mathbb{Y}=\{y^0,y^1,\cdots,y^n\}$$

称为一个有序单纯形。

♠ **定义2.5 (区域)：** 记$\mathbb{Y}=\{y^0,y^1,\cdots,y^n\}$为$\mathbb{R}^n$上的有一个有序单纯形，测试点$x\in\mathbb{R}^n$

- 若$y^n\prec x$，则x属于内缩区域；
- 其余情况，若$x\prec y^0$，则x属于扩张区域；
- 其余情况，若x至少掌控了$\mathbb{Y}$中的两个点，则x属于反射区域；
- 其余情况，若x掌控了$\mathbb{Y}$中的零或一个点，则x属于外缩区域。

♠ **注释I：** 定义2.1-2.5仅针对无约束优化的目标问题2.1，对于不等式约束优化的目标问题3.1和4.1，我们会在下文给出新的定义。

2.3 NM算法思路

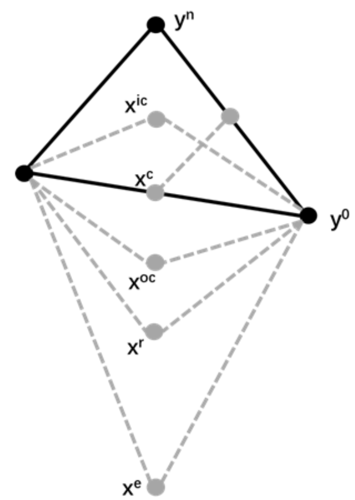
♣ **整体思路**

NM算法于1965年被首次提出，其核心思路是：初始化一个有序单纯形并进行迭代，每一步迭代都要用一个更好的点来替换当前单纯形的最差点【详见NM-1】。随着迭代次数的增加，单纯形会变得越来越好，单纯形的最优点可能会收敛到目标问题的最优解【详见NM-2】。当迭代满足一定条件（例如相邻两次迭代的最优点函数值之差小于某个容忍值）时，迭代停止，将最终得到的单纯形的最优点作为目标问题的最优解。

♣ NM-1. 替换最差点的原则

NM算法的每一步迭代都会将有序单纯形$\mathbb{Y}$中最差的点y^n替换为一个更好的点。现做定义如下：

$x^c = \frac{1}{n}\sum_{i=0}^{n-1} y^i$	中心
$x^r = x^c + (x^c - y^n)$	反射点
$x^e = x^c + 2(x^c - y^n)$	延长点
$x^{oc} = x^c + \frac{1}{2}(x^c - y^n)$	外缩点
$x^{ic} = x^c - \frac{1}{2}(x^c - y^n)$	内缩点
$\gamma = \frac{1}{2}$	收缩参数



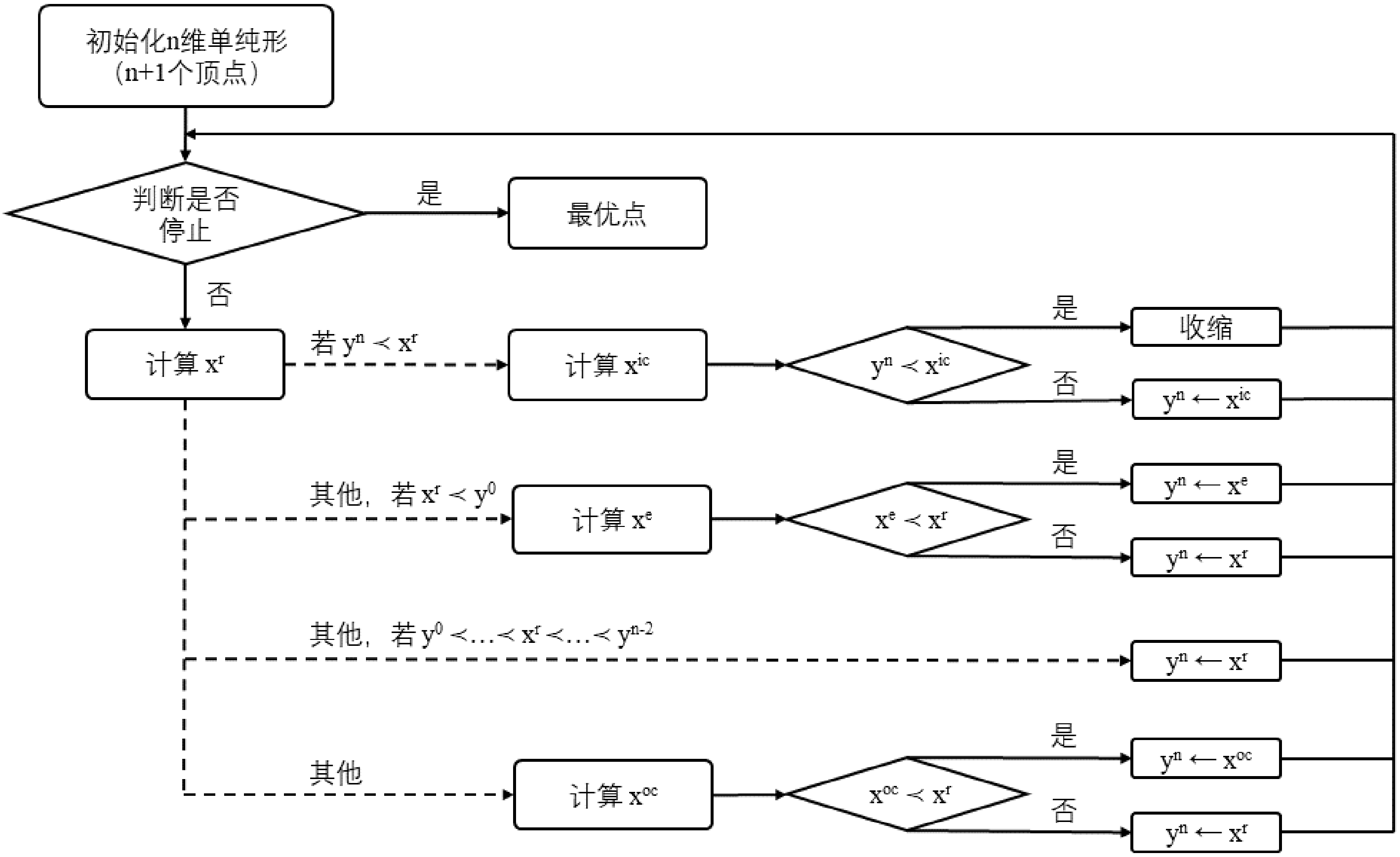
每一步迭代有五种替换方案，要么用x^r, x^e, x^{oc}, x^{ic}中的一个点替换y^n，要么将$y^1, y^2, \cdots, y^n$向y^0的方向收缩，即：

$$\mathbb{Y} \longleftarrow \{y^0, y^0 + \gamma(y^1 - y^0), \cdots, y^0 + \gamma(y^n - y^0)\}. \tag{1}$$

针对不同的情况，NM会选取不同的替换点：

- 若x^r属于内缩区域且x^{ic}属于内缩区域，用公式(1)收缩；
- 若x^r属于内缩区域且x^{ic}不属于内缩区域，用x^{ic}替换y^n；
- 除上述情况，若x^r属于扩张区域，用$\mathrm{Best}(x^r, x^e)$替换y^n；
- 除上述情况，若x^r属于反射区域，用x^r替换y^n；
- 除上述情况，即x^r属于外缩区域，用$\mathrm{Best}(x^r, x^{oc})$替换$y^n$；

NM算法流程图：



♣ NM-2. 收敛性及相关研究

正如引言中所指出，NM算法未必能收敛至问题最优解。1998年McKinnon提出了一个$\mathbb{R}$上的光滑的严格凸函数[2]，在该函数上，NM算法产生的实验点序列无法接近唯一的最小值。1998-2012年，人们采用多种方法改进了NM算法，使其在某些限制条件下具备一定的收敛性[1]。本文[1] (2018)则是将NM算法融入了Mads算法，利用Mads的收敛性分析来确保算法收敛。

3.Mads不等式约束优化算法

3.1 目标问题

考虑不等式约束优化问题

$$\min_{x \in X \subset \mathrm{R}^n} f(x)$$
$$s.t. \quad c(x) \leq 0$$

其中$f : X \mapsto \mathrm{R} \cup \infty$，$c : X \mapsto (\mathrm{R} \cup \infty)^m$，$c = (c_1, \cdots, c_m)^T$，$X$是$\mathbb{R}^n$的子集，$c$可松弛，$f$与$c$的平滑性无要求。

3.2 基本概念

♠ **定义3.1 (正张成集)：** 给定向量空间V，若集合S的元素的非负线性组合可以张成V，则称S为V的正张成集。

♠ **定义3.2 (网格)：** 设$G \in \mathbb{R}^{n\times n}$为可逆矩阵，$Z \in \mathbb{Z}^{n\times p}$满足$Z$的列向量构成一个$\mathbb{R}^n$的正张成集，设$D = GZ$。

对于当前最优解$x^k \in \mathbb{R}^n$，以x^k为中心，尺寸为δ^k的网格定义为

$$M^k = \{x^k + \delta^k Dy \quad y \in \mathbb{N}^p\} \subset \mathbb{R}^n$$

其中δ^k称为网格尺寸参数。

♠ **定义3.3 (框架)：** 设$G \in \mathbb{R}^{n\times n}$为可逆矩阵，$Z \in \mathbb{Z}^{n\times p}$满足$Z$的列向量构成一个$\mathbb{R}^n$的正张成集，设$D = GZ$。

对于网格尺寸参数δ^k，取Δ^k满足$\Delta^k \geq \delta^k$，则以x^k为中心，尺寸为Δ^k的框架定义为

$$F^k = \{x \in M^k \quad ||x - x^k||_\infty \leq \Delta^k b\}$$

其中$b = max\{||d'||_\infty : d' \in \mathbb{D}\}$，$\mathbb{D}$表示矩阵$D$的列向量构成的集合；$\Delta^k$称为框架尺寸参数。

♠ **定义3.4 (违约函数)：** 设约束条件的索引集$J = \{1, 2, \cdots, m\}$，定义违约函数

$$h(x) = \begin{cases} \sum_{j\in J} \left(\max\{c_j(x), 0\}\right)^2 & \text{若 } x \in X \\ \infty & \text{其他} \end{cases}$$

$h(x)$总是非负的，$h(x) = 0 \iff x$在问题的可行域中，即$x \in \Omega = \{x \in X : c(x) \leq 0\}$.

♠ **定义3.5 (控制)：** $x, y \in \mathbb{R}^n$，若

- x, y均可行，且$f(x) < f(y)$；或
- x, y均不可行，且$f(x) \leq f(y)$，$h(x) \leq h(y)$，且至少有一个是严格不等号

则称x控制y，记作$x \prec y$.

♠ **注释II：** 这里针对不等式约束优化的目标问题3.1给出了新的“控制”定义。

3.3 Mads算法思路

♣ 整体思路

Mads的每一步迭代由两个主要步骤组成：搜索和轮询。

- 搜索步骤可以使用各种策略来探索变量空间，一个精心设计的搜索步骤可以让算法避免落入局部最优解。

- 轮询步骤是在框架尺寸参数Δ^k所限制区域内的变量空间进行局部搜索，轮询步骤保证了算法的收敛性。

下面我们用一个例子直观地阐述Mads的搜索-轮询过程。对于

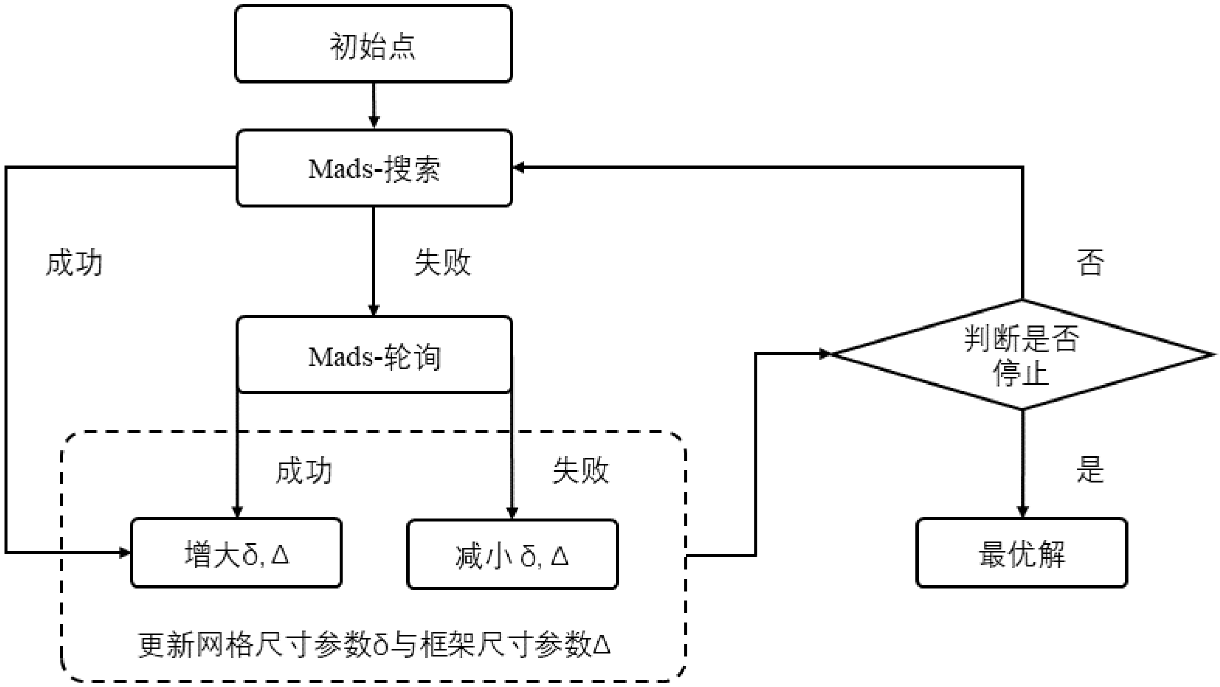
♣ Step2. 迭代：

Algorithm 2: The mesh adaptive direct search algorithm (Mads)

Given a user-defined set of starting point: $V^0 \subset \mathbb{R}^n$,
and initial mesh and poll size vectors: typically $\delta_i^0 = \Delta_i^0 = 1$ for $i = 1, 2, \ldots, n$.
Set the iteration counter: $k \leftarrow 0$.

1. Search step (optional):
   - Launch the simulation on a finite set S^k of mesh points.
   - If successful, go to 3.
2. Poll step:
   - Launch the simulation on the set P^k of poll points.
3. Updates:
   - Update the cache V^{k+1}, the incumbent x^{k+1}
   - and the mesh and poll size vectors δ^{k+1} and Δ^{k+1}.
   - Increase the iteration counter $k \leftarrow k + 1$ and go to 1.

3.3 Mads算法总结：



参考文献

[1]Audet, C., Tribes, C. Mesh-based Nelder–Mead algorithm for inequality constrained optimization. Comput Optim Appl 71, 331–352 (2018).

[2]McKinnon, K.I.M.: Convergence of the Nelder–Mead simplex method to a nonstationary point. SIAMJ. Optim. 9(1), 148–158 (1998)