

# 分块坐标下降法

文再文

北京大学北京国际数学研究中心

教材《最优化：建模、算法与理论》配套电子教案

<http://bicmr.pku.edu.cn/~wenzw/optbook.html>

致谢：本教案由朱楨源协助准备

- 1 分块坐标下降法
- 2 应用举例
- 3 收敛性分析
- 4 HOGWILD! 异步SGD
- 5 CYCLADES

## 问题形式

考虑具有如下形式的问题：

$$\min_{x \in \mathcal{X}} F(x_1, x_2, \dots, x_s) = f(x_1, x_2, \dots, x_s) + \sum_{i=1}^s r_i(x_i),$$

- $\mathcal{X}$  是函数的可行域，自变量  $x$  拆分成  $s$  个变量块  $x_1, x_2, \dots, x_s$ ，每个变量块  $x_i \in \mathbb{R}^{n_i}$ 。
- 函数  $f$  是关于  $x$  的可微函数，每个  $r_i(x_i)$  关于  $x_i$  是适当的闭凸函数，但不一定可微。
- 目标函数  $F$  的性质体现在  $f$ ，每个  $r_i$  以及自变量的分块上。通常情况下， $f$  对于所有变量块  $x_i$  不可分，但单独考虑每一块自变量时， $f$  有简单结构； $r_i$  只和第  $i$  个自变量块有关，因此  $r_i$  在目标函数中是一个可分项。
- 求解该问题的难点在于如何利用分块结构处理不可分的函数  $f$ 。

## 问题形式

- 分组LASSO模型：参数 $x = (x_1, x_2, \dots, x_G) \in \mathbb{R}^p$ 可以分成 $G$ 组，且 $\{x_i\}_{i=1}^G$ 中只有少数的非零向量。

$$\min_x \frac{1}{2n} \|b - Ax\|_2^2 + \lambda \sum_{i=1}^G \sqrt{p_i} \|x_i\|_2.$$

- $K$ -均值聚类问题的等价形式：

$$\begin{aligned} \min_{\Phi, H} \quad & \|A - \Phi H\|_F^2, \\ \text{s.t.} \quad & \Phi \in \mathbb{R}^{n \times k}, \text{ 每一行只有一个元素为1, 其余为0,} \\ & H \in \mathbb{R}^{k \times p}. \end{aligned}$$

- 低秩矩阵恢复：设 $b \in \mathbb{R}^m$ 是已知的观测向量， $\mathcal{A}$ 是线性映射。

$$\min_{X, Y} \frac{1}{2} \|\mathcal{A}(XY) - b\|_2^2 + \alpha \|X\|_F^2 + \beta \|Y\|_F^2,$$

其中 $\alpha, \beta > 0$ 为正则化参数。

## 问题形式

- 非负矩阵分解：设 $\mathcal{M}$ 是已知张量，考虑求解如下极小化问题：

$$\min_{A_1, A_2, \dots, A_N \geq 0} \frac{1}{2} \|\mathcal{M} - A_1 \circ A_2 \circ \dots \circ A_N\|_F^2 + \sum_{i=1}^N \lambda_i r_i(A_i),$$

其中“ $\circ$ ”表示张量的外积运算。

- 字典学习：设 $A \in \mathbb{R}^{m \times n}$ 为 $n$ 个观测，每个观测的信号维数是 $m$ ，现在我们要从 $A$ 中学习出一个字典 $D \in \mathbb{R}^{m \times k}$ 和系数矩阵 $X \in \mathbb{R}^{k \times n}$ ：

$$\begin{aligned} \min_{D, X} \quad & \frac{1}{2n} \|DX - A\|_F^2 + \lambda \|X\|_1, \\ \text{s.t.} \quad & \|D\|_F \leq 1. \end{aligned}$$

在这里自变量有两块，分别为 $D$ 和 $X$ ，此外对 $D$ 还存在球约束 $\|D\|_F \leq 1$ 。

# 挑战和难点

- 函数 $f$ 关于变量全体一般是非凸的，这使得问题求解具有挑战性
- 应用在非凸问题上的算法收敛性不易分析，很多针对凸问题设计的算法通常会失效
- 目标函数的整体结构十分复杂，变量的更新需要很大计算量
- **目标**: 发展一种更新方式简单且有全局收敛性（收敛到稳定点）的有效算法

# 变量划分

- 分块坐标下降法更新方式：按照 $x_1, x_2, \dots, x_s$ 的次序依次固定其他 $(s-1)$ 块变量极小化 $F$ ，完成一块变量的极小化后，它的值便立即被更新到变量空间中，更新下一块变量时将使用每个变量最新的值。
- 变量划分

$$\mathcal{X}_i^k = \{x \in \mathbb{R}^{n_i} \mid (x_1^k, \dots, x_{i-1}^k, x, x_{i+1}^{k-1}, \dots, x_s^{k-1}) \in \mathcal{X}\}.$$

- 辅助函数

$$f_i^k(x_i) = f(x_1^k, \dots, x_{i-1}^k, x_i, x_{i+1}^{k-1}, \dots, x_s^{k-1}),$$

其中 $x_j^k$ 表示在第 $k$ 次迭代中第 $j$ 块自变量的值，函数 $f_i^k$ 表示在第 $k$ 次迭代更新第 $i$ 块变量时所需要考虑的目标函数的光滑部分。

## 变量更新方式

在每一步更新中，通常使用以下三种更新格式之一：

$$x_i^k = \operatorname{argmin}_{x_i \in \mathcal{X}_i^k} \{f_i^k(x_i) + r_i(x_i)\}, \quad (1)$$

$$x_i^k = \operatorname{argmin}_{x_i \in \mathcal{X}_i^k} \left\{ f_i^k(x_i) + \frac{L_i^{k-1}}{2} \|x_i - x_i^{k-1}\|_2^2 + r_i(x_i) \right\}, \quad (2)$$

$$x_i^k = \operatorname{argmin}_{x_i \in \mathcal{X}_i^k} \left\{ \langle \hat{g}_i^k, x_i - \hat{x}_i^{k-1} \rangle + \frac{L_i^{k-1}}{2} \|x_i - \hat{x}_i^{k-1}\|_2^2 + r_i(x_i) \right\}, \quad (3)$$

- $L_i^k > 0$  为常数
- 在更新格式(3)中， $\hat{x}_i^{k-1}$  采用外推定义：

$$\hat{x}_i^{k-1} = x_i^{k-1} + \omega_i^{k-1} (x_i^{k-1} - x_i^{k-2}), \quad (4)$$

其中  $\omega_i^k \geq 0$  为外推的权重， $\hat{g}_i^k \stackrel{\text{def}}{=} \nabla f_i^k(\hat{x}_i^{k-1})$  为外推点处的梯度。



---

## Algorithm 1 分块坐标下降法

---

```
1: 初始化: 选择两组初始点  $(x_1^{-1}, x_2^{-1}, \dots, x_s^{-1}) = (x_1^0, x_2^0, \dots, x_s^0)$ .
2: for  $k = 1, 2, \dots$  do
3:   for  $i = 1, 2, \dots$  do
4:     使用格式(1) 或(2) 或(3) 更新  $x_i^k$ .
5:   end for
6:   if 满足停机条件 then
7:     返回  $(x_1^k, x_2^k, \dots, x_s^k)$ , 算法终止.
8:   end if
9: end for
```

---

- 三种格式都有其适用的问题，特别是子问题是否可写出显式解
- 在每一步更新中，三种迭代格式对不同自变量块可以混合使用，不必仅仅局限于一种。

# 算法格式

- BCD算法的子问题可采用三种不同的更新格式，这三种格式可能会产生不同的迭代序列，可能会收敛到不同的解，坐标下降算法的数值表现也不相同。
- 格式(1)是最直接的更新方式，它严格保证了整个迭代过程的目标函数值是下降的。然而由于 $f$ 的形式复杂，子问题求解难度较大。在收敛性方面，格式(1)在强凸问题上可保证目标函数收敛到极小值，但在非凸问题上不一定收敛。
- 格式(2) (3) 则是对格式(1)的修正，不保证迭代过程目标函数的单调性，但可以改善收敛性结果。使用格式(2)可使得算法收敛性在函数 $F$ 为非严格凸时有所改善。
- 格式(3)实质上为目标函数的一阶泰勒展开近似，在一些测试问题上有更好的表现，可能的原因是使用一阶近似可以避免一些局部极小值点。此外，格式(3)的计算量很小，比较容易实现。

## 例子：二元二次函数

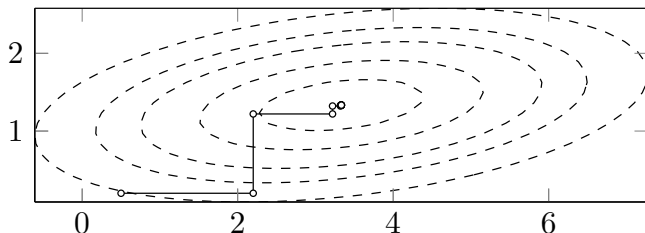
考虑二元二次函数的优化问题

$$\min f(x, y) = x^2 - 2xy + 10y^2 - 4x - 20y.$$

故采用格式(1)的分块坐标下降法为

$$x^{k+1} = 2 + y^k, \quad y^{k+1} = 1 + \frac{x^{k+1}}{10}.$$

下图描绘了当初始点为  $(x, y) = (0.5, 0.2)$  时的迭代点轨迹，可以看到在进行了7次迭代后迭代点与最优解已充分接近。



## 不收敛反例

值得注意的是，对于非凸函数 $f(x)$ ，分块坐标下降法可能失效。Powell在1973年就给出了一个使用格式(1)但不收敛的例子：

$$F(x_1, x_2, x_3) = -x_1x_2 - x_2x_3 - x_3x_1 + \sum_{i=1}^3 [(x_i - 1)_+^2 + (-x_i - 1)_+^2],$$

其中 $(x_i - 1)_+^2$ 的含义为先对 $(x_i - 1)$ 取正部再平方。设 $\varepsilon > 0$ ，初始点取为

$$x^0 = \left(-1 - \varepsilon, 1 + \frac{\varepsilon}{2}, -1 - \frac{\varepsilon}{4}\right),$$

容易验证迭代序列满足

$$x^k = (-1)^k \cdot (-1, 1, -1) + \left(-\frac{1}{8}\right)^k \cdot \left(-\varepsilon, \frac{\varepsilon}{2}, -\frac{\varepsilon}{4}\right),$$

这个迭代序列有两个聚点 $(-1, 1, -1)$ 与 $(1, -1, 1)$ ，但这两个点都不是 $F$ 的稳定点。

- 1 分块坐标下降法
- 2 应用举例**
- 3 收敛性分析
- 4 HOGWILD! 异步SGD
- 5 CYCLADES

# LASSO 问题求解

下面介绍如何使用分块坐标下降法来求解LASSO 问题

$$\min_x \mu \|x\|_1 + \frac{1}{2} \|Ax - b\|^2.$$

将自变量 $x$ 记为 $x = [x_i, \bar{x}_i^\top]^\top$ , 其中 $\bar{x}_i$ 为 $x$ 去掉第 $i$ 个分量而形成的列向量. 而相应地, 矩阵 $A$ 在第 $i$ 块的更新记为 $A = [a_i \quad \bar{A}_i]$ , 其中 $\bar{A}_i$ 为矩阵 $A$ 去掉第 $i$ 列而形成的矩阵.

在第 $i$ 块的更新中考虑格式(1)。做替换 $c_i = b - \bar{A}_i \bar{x}_i$ , 原问题等价于

$$\min_{x_i} f_i(x_i) \stackrel{\text{def}}{=} \mu |x_i| + \frac{1}{2} \|a_i\|^2 x_i^2 - a_i^\top c_i x_i.$$

可直接写出它的最小值点

$$x_i^k = \operatorname{argmin}_{x_i} f_i(x_i) = \begin{cases} \frac{a_i^\top c_i - \mu_i}{\|a_i\|^2}, & a_i^\top c_i > \mu, \\ \frac{a_i^\top c_i + \mu_i}{\|a_i\|^2}, & a_i^\top c_i < -\mu, \\ 0, & \text{其他.} \end{cases}$$

# K-均值聚类算法

- 当固定 $H$ 时, 设 $\Phi$ 的每一行为 $\phi_i^T$ , 那么根据矩阵分块乘法,

$$A - \Phi H = \begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_n^T \end{bmatrix} - \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_n^T \end{bmatrix} H = \begin{bmatrix} a_1^T - \phi_1^T H \\ a_2^T - \phi_2^T H \\ \vdots \\ a_n^T - \phi_n^T H \end{bmatrix}.$$

注意到 $\phi_i$ 只有一个分量为1, 其余分量为0, 不妨设其第 $j$ 个分量为1, 此时 $\phi_i^T H$ 相当于将 $H$ 的第 $j$ 行取出, 因此 $\|a_i^T - \phi_i^T H\|$ 为 $a_i^T$ 与 $H$ 的第 $j$ 个行向量的距离. 我们的最终目的是极小化 $\|A - \Phi H\|_F^2$ , 所以 $j$ 应该选矩阵 $H$ 中距离 $a_i^T$ 最近的那一行, 即

$$\Phi_{ij} = \begin{cases} 1, & j = \operatorname{argmin}_l \|a_i - h_l\|, \\ 0, & \text{其他.} \end{cases}$$

其中 $h_l^T$ 表示矩阵 $H$ 的第 $l$ 行.

## K-均值聚类算法

- 当固定 $\Phi$ 时，此时考虑 $H$ 的每一行 $h_j^T$ ，根据目标函数的等价性有

$$\|A - \Phi H\|_F^2 = \sum_{j=1}^k \sum_{a \in S_j} \|a - h_j\|^2,$$

因此只需要对每个 $h_j$ 求最小即可。设 $\bar{a}_j$ 是目前第 $j$ 类所有点的均值，则

$$\begin{aligned} \sum_{a \in S_j} \|a - h_j\|^2 &= \sum_{a \in S_j} \|a - \bar{a}_j + \bar{a}_j - h_j\|^2 \\ &= \sum_{a \in S_j} (\|a - \bar{a}_j\|^2 + \|\bar{a}_j - h_j\|^2 + 2 \langle a - \bar{a}_j, \bar{a}_j - h_j \rangle) \\ &= \sum_{a \in S_j} (\|a - \bar{a}_j\|^2 + \|\bar{a}_j - h_j\|^2), \end{aligned}$$

这里利用了交叉项 $\sum_{a \in S_j} \langle a - \bar{a}_j, \bar{a}_j - h_j \rangle = 0$ 的事实。因此容易看出，此时 $h_j$ 直接取为 $\bar{a}_j$ 即可达到最小值



# 非负矩阵分解

考虑最基本的非负矩阵分解问题

$$\min_{X, Y \geq 0} f(X, Y) = \frac{1}{2} \|XY - M\|_F^2.$$

可以计算梯度

$$\frac{\partial f}{\partial X} = (XY - M)Y^T, \quad \frac{\partial f}{\partial Y} = X^T(XY - M).$$

注意到在格式(3)中，当 $r_i(X)$ 为凸集示性函数时即是求解到该集合的投影，因此得到分块坐标下降法如下：

$$\begin{aligned} X^{k+1} &= \max\{X^k - t_k^x(X^k Y^k - M)(Y^k)^T, 0\}, \\ Y^{k+1} &= \max\{Y^k - t_k^y(X^k)^T(X^k Y^k - M), 0\}, \end{aligned}$$

其中 $t_k^x, t_k^y$ 是步长，

$$\min_{D, X} \frac{1}{2n} \|DX - A\|_F^2 + \lambda \|X\|_1 + \frac{\mu}{2} \|D\|_F^2.$$

- 当固定变量 $D$ 时，考虑函数

$$f_D(X) = \frac{1}{2n} \|DX - A\|_F^2 + \lambda \|X\|_1.$$

使用格式(3). 通过直接计算可得 $f_D(X)$ 中光滑部分的梯度为

$$G = \frac{1}{n} D^T (DX - A),$$

因此格式(3)等价于

$$X^{k+1} = \text{prox}_{t_k \lambda \|\cdot\|_1} \left( X^k - \frac{t_k}{n} (D^k)^T (D^k X^k - A) \right),$$

其中 $t_k$ 为步长.

$$\min_{D, X} \frac{1}{2n} \|DX - A\|_F^2 + \lambda \|X\|_1 + \frac{\mu}{2} \|D\|_F^2.$$

- 当固定变量 $X$ 时，考虑函数

$$f_X(D) = \frac{1}{2n} \|DX - A\|_F^2 + \frac{\mu}{2} \|D\|_F^2.$$

使用格式(1). 计算关于 $D^T$ 的梯度为

$$\nabla_{D^T} f_X(D) = \frac{1}{n} X(X^T D^T - A^T) + \mu D^T,$$

令梯度为零向量，可得

$$D = AX^T (XX^T + n\mu I)^{-1}.$$

因为 $X \in \mathbb{R}^{k \times n}$ ，其中 $k \ll n$ ，所以 $XX^T$ 是一个比较小的矩阵，可以方便地求出它的逆. 故格式(1)等价于

$$D^{k+1} = A(X^{k+1})^T (X^{k+1} (X^{k+1})^T + n\mu I)^{-1}.$$

# 最大割问题的非凸松弛

## 最大割问题

$$\begin{aligned} \text{(半定松弛)} \quad \min \quad & \langle C, X \rangle, \\ \text{s.t.} \quad & X_{ii} = 1, \quad i = 1, 2, \dots, n, \\ & X \succeq 0. \end{aligned}$$

$$\begin{aligned} \text{(非凸松弛)} \quad \min \quad & \langle C, V^T V \rangle, \\ \text{s.t.} \quad & v_i \in \mathbb{R}^p, \quad \|v_i\| = 1, \quad i = 1, 2, \dots, n, \\ & V = [v_1, v_2, \dots, v_n]. \end{aligned}$$

- 比较两种松弛方式可知，非凸松弛通过引入分解 $X = V^T V$ 并限制 $V$ 的每一列的 $l_2$ 范数为1，将半定松弛中的 $X$ 对角线元素为1以及 $X$ 半正定的约束消去了。
- 这两个问题一般不等价，当 $p$ 充分大时二者等价。实际计算中通常选取一个较小的 $p$ 。

## 最大割问题的非凸松弛

矩阵 $V$ 是按列分成 $n$ 块的，考虑格式(1)为例，取定 $i$ ，固定其余 $v_j$

$$\text{Tr} \left( \begin{bmatrix} C_{11} & \cdots & C_{1i} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ C_{i1} & \cdots & C_{ii} & \cdots & C_{in} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{ni} & \cdots & C_{nn} \end{bmatrix} \begin{bmatrix} v_1^T v_1 & \cdots & v_1^T v_i & \cdots & v_1^T v_n \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_i^T v_1 & \cdots & v_i^T v_i & \cdots & v_i^T v_n \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ v_n^T v_1 & \cdots & v_n^T v_i & \cdots & v_n^T v_n \end{bmatrix} \right),$$

根据以上矩阵分块示意图可知和 $v_i$ 有关的部分为

$$C_{ii} v_i^T v_i + \sum_{j \neq i} (C_{ij} + C_{ji}) v_i^T v_j.$$

由于约束 $\|v_i\| = 1$ ，上式中第一项是常数。最终在第 $i$ 步子问题是：

$$\min f_i(v_i) = \left( \sum_{j \neq i} C_{ji} v_j^T \right) v_i, \text{ s.t. } \|v_i\| = 1.$$

$$\text{其解为: } v_i = - \left( \sum_{j \neq i} C_{ji} v_j \right) / \left\| \sum_{j \neq i} C_{ji} v_j \right\|.$$

- 1 分块坐标下降法
- 2 应用举例
- 3 收敛性分析**
- 4 HOGWILD! 异步SGD
- 5 CYCLADES

## 交替线性化方法

- 我们对格式(3) 在  $s = 2$  且非凸的情况下进行收敛性分析. 定义:

$$\min \Psi(x, y) \stackrel{\text{def}}{=} f(x) + g(y) + H(x, y), \quad (x, y) \in \mathbb{R}^n \times \mathbb{R}^m$$

其中  $f$  和  $g$  为适当闭函数,  $H$  为其定义域上的连续可微函数.

- 对该问题, 格式化为如下基本形式:

$$\begin{aligned} x^{k+1} &\in \text{prox}_{c_k f} (x^k - c_k \nabla_x H(x^k, y^k)) \\ y^{k+1} &\in \text{prox}_{d_k g} (y^k - d_k \nabla_y H(x^{k+1}, y^k)) \end{aligned}$$

其中  $c_k, d_k$  为步长参数. 由于  $f$  和  $g$  不是凸函数, 相应地  $\text{prox}_f$  和  $\text{prox}_g$  是集合函数, 在迭代过程中只要求  $x_{k+1}$  和  $y_{k+1}$  是相应集合中的一个元素即可. 由于自变量只有两块, 对光滑部分  $H$  我们采用的是线性化处理, 因此该格式又称为近似点交替线性化方法.

- 为了保证  $\text{prox}_f$  和  $\text{prox}_g$  是良定义的, 还需要对  $f$  和  $g$  提出下界有限的假设.

## 非凸函数的邻近算子

(适当闭函数的邻近算子) 设  $h$  是适当闭函数(可以非凸), 且具有有限的下界, 即满足  $\inf_{x \in \text{dom} h} h(x) > -\infty$ , 定义  $h$  的邻近算子为

$$\text{prox}_h(x) = \arg \min_{u \in \text{dom} h} \left\{ h(u) + \frac{1}{2} \|u - x\|^2 \right\}.$$

### 定理

设  $h$  是适当闭函数且  $\inf_{x \in \text{dom} h} h(x) > -\infty$ , 则  $\forall x \in \text{dom} h$ ,  $\text{prox}_h(x)$  是  $\mathbb{R}^n$  上的非空紧集.

### Proof.

定义  $g(u) = h(u) + \frac{1}{2} \|u - x\|^2$ , 设  $\inf_{x \in \text{dom} h} h(x) = l$ .

取  $u_0 \in \text{dom} h$ , 由于  $\frac{1}{2} \|u - x\|^2$  无上界, 故  $\exists R > 0$ , 对  $\forall$  满足  $\|u - x\| > R$  的  $u$ , 成立  $\frac{1}{2} \|u - x\|^2 > g(u_0) - l$ , 即  $g(u) > g(u_0)$ .

这说明下水平集  $\{u \mid g(u) \leq g(u_0)\}$  含于球  $\|u - x\| \leq R$  内, 即  $g$  有一个非空有界下水平集. 显然  $g(u)$  是闭函数, 由 Weierstrass 定理可知,  $g(u)$  的最小值点集合  $\text{prox}_h(x)$  是非空紧集.  $\square$



# 非光滑非凸问题函数的次微分

前面介绍了闭凸函数的邻近算子与次梯度的关系，而对于非凸函数有类似的结论。首先回顾一下非光滑非凸函数的次微分。

## 次微分

设  $f: \mathbb{R}^n \rightarrow (-\infty, +\infty]$  是适当下半连续函数。

- 对给定的  $x \in \mathbf{dom} f$ ，满足如下条件的所有向量  $u \in \mathbb{R}^n$  的集合定义为  $f$  在点  $x$  处的 *Fréchet* 次微分：

$$\liminf_{y \rightarrow x, y \neq x} \frac{f(y) - f(x) - \langle u, y - x \rangle}{\|y - x\|} \geq 0,$$

记为  $\hat{\partial}f(x)$ 。当  $x \notin \mathbf{dom} f$  时，将  $\hat{\partial}f(x)$  定义为空集  $\emptyset$ 。

- $f$  在点  $x \in \mathbb{R}^n$  处的极限次微分(或简称为次微分)定义为

$$\partial f(x) = \{u \in \mathbb{R}^n : \exists x^k \rightarrow x, f(x^k) \rightarrow f(x), u^k \in \hat{\partial}f(x^k) \rightarrow u\}.$$

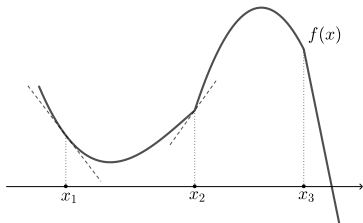
极限次微分通过对  $x$  附近的点处的 *Fréchet* 次微分取极限得到。

- $\hat{\partial}f(x) \subseteq \partial f(x)$ , 前者是闭凸集, 后者是闭集. 并非在所有的  $x \in \text{dom } f$  处都存在 *Fréchet* 次微分.
- 凸函数的次梯度要求不等式

$$f(y) \geq f(x) + \langle g, y - x \rangle, \quad g \in \partial f(x)$$

在定义域内全局成立, 而非凸函数只要求在极限意义下成立.

- 当  $f$  是可微函数时, *Fréchet* 次微分和次微分都退化成梯度.



如图,  $f(x)$  在  $x_3$  处不存在 *Fréchet* 次微分, 但存在次微分

## 定理

设  $h$  是适当闭函数(可非凸)且有下界,  $u \in \text{prox}_h(x)$ , 则  $x - u \in \partial h(u)$

## 假设条件

- (1)  $f: \mathbb{R}^n \rightarrow (-\infty, +\infty]$ ,  $g: \mathbb{R}^m \rightarrow (-\infty, +\infty]$  均为适当下半连续函数,  $\inf_{\mathbb{R}^n \times \mathbb{R}^m} \Psi > -\infty$ ,  $\inf_{\mathbb{R}^n} f > -\infty$ , 以及  $\inf_{\mathbb{R}^m} g > -\infty$
- (2)  $H: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$  是连续可微函数, 且  $\nabla H$  在有界集上是联合利普希茨连续的. 即对于任意的  $B_1 \times B_2 \subset \mathbb{R}^n \times \mathbb{R}^m$ , 存在  $L > 0$  使得对于任意的  $(x_i, y_i) \in B_1 \times B_2, i = 1, 2$  有

$$\begin{aligned} & \|(\nabla_x H(x_1, y_1) - \nabla_x H(x_2, y_2), \nabla_y H(x_1, y_1) - \nabla_y H(x_2, y_2))\| \\ & \leq L \|(x_1 - x_2, y_1 - y_2)\|. \end{aligned}$$

## 假设条件

- 根据假设，在有界集上 $H$ 关于每个分量都是梯度 $L$ -利普希茨连续的，且参数与另一分量无关。即

$$\begin{aligned}\|\nabla_x H(x_1, y) - \nabla_x H(x_2, y)\| &\leq L \|x_1 - x_2\|, \\ \|\nabla_y H(x, y_1) - \nabla_y H(x, y_2)\| &\leq L \|y_1 - y_2\|.\end{aligned}$$

- 可以直接写出 $\Psi(x, y)$ 的次微分：

$$\partial\Psi(x, y) = (\nabla_x H(x, y) + \partial f(x), \nabla_y H(x, y) + \partial g(y))$$

其中“+”表示为集合间的加法。

# 证明梗概

- **充分下降**：找到一个正常数 $\rho_1$  使得

$$\rho_1 \|z^{k+1} - z^k\|^2 \leq \Psi(z^k) - \Psi(z^{k+1})$$

- **次梯度上界**：假设算法产生的迭代序列有界，找到另一个常数 $\rho_2$ ，使得次梯度有一个上界估计：

$$\|w^{k+1}\| \leq \rho_2 \|z^{k+1} - z^k\|, \quad w^k \in \partial\Psi(z^k)$$

- **利用KL性质证明全序列收敛**：假设 $\Psi$  是一个KL函数，证明迭代序列 $\{z^k\}_{k \in \mathbb{N}}$  是一个柯西列。

注：前两个步骤是证明多数算法的基本步骤，当这两个性质成立时，对任意的算法产生的迭代序列的聚点集合都为非空连通紧集，且这些聚点都是 $\Psi$  的临界点。

## 近似点交替线性化方法下降量

设  $h: \mathbb{R}^d \rightarrow \mathbb{R}$  是连续可微函数, 梯度  $\nabla h$  是利普希茨连续的, 相应的常数为  $L_h$ ,  $\sigma: \mathbb{R}^d \rightarrow (-\infty, +\infty]$  是适当下半连续函数且  $\inf_{\mathbb{R}^d} \sigma > -\infty$ . 固定  $t < \frac{1}{L_h}$ , 则对任意的  $u \in \text{dom } \sigma$  和  $\tilde{u} \in \text{prox}_{t\sigma}(u - t\nabla h(u))$ , 有

$$h(\tilde{u}) + \sigma(\tilde{u}) \leq h(u) + \sigma(u) - \frac{1}{2} \left( \frac{1}{t} - L_h \right) \|\tilde{u} - u\|^2.$$

证明: 首先根据  $\sigma$  的假设,  $\tilde{u}$  是良定义的. 根据  $\tilde{u}$  的最优性, 有

$$\langle \tilde{u} - u, \nabla h(u) \rangle + \frac{1}{2t} \|\tilde{u} - u\|^2 + \sigma(\tilde{u}) \leq \sigma(u).$$

再结合二次上界, 有

$$\begin{aligned} h(\tilde{u}) + \sigma(\tilde{u}) &\leq h(u) + \langle \tilde{u} - u, \nabla h(u) \rangle + \frac{L_h}{2} \|\tilde{u} - u\|^2 + \sigma(\tilde{u}) \\ &\leq h(u) + \frac{L_h}{2} \|\tilde{u} - u\|^2 + \sigma(u) - \frac{1}{2t} \|\tilde{u} - u\|^2 \\ &= h(u) + \sigma(u) - \frac{1}{2} \left( \frac{1}{t} - L_h \right) \|\tilde{u} - u\|^2. \end{aligned}$$

## 充分下降定理

在假设条件下，设 $\{z^k\} = \{(x^k, y^k)\}$ 为迭代格式产生的迭代序列，且假设 $z^k$ 有界。取步长 $c_k = d_k = \frac{1}{\gamma L}$ ，其中 $\gamma > 1$ 是常数， $L$ 为 $\nabla H$ 的利普希茨系数，则以下结论成立：

(1) 迭代点处的函数值序列 $\{\Psi(z^k)\}$ 是单调下降的，且

$$\frac{\rho_1}{2} \|z^{k+1} - z^k\|^2 \leq \Psi(z^k) - \Psi(z^{k+1}), \quad \forall k \geq 0,$$

其中 $\rho_1 = (\gamma - 1)L$ ；

(2) 序列 $\{\|z^{k+1} - z^k\|\}_{k=1}^{\infty}$ 平方可和，即

$$\sum_{k=1}^{\infty} (\|x^{k+1} - x^k\|^2 + \|y^{k+1} - y^k\|^2) = \sum_{k=1}^{\infty} \|z^{k+1} - z^k\|^2 < +\infty,$$

并由此推出 $\lim_{k \rightarrow \infty} \|z^{k+1} - z^k\| = 0$ 。

- (1) 根据假设条件的(2),  $H(x, y)$ 关于每个分量都是利普希茨连续的, 由第30页的结论可得到每一步关于 $x^k$ 和 $y^k$ 的下降量估计:

$$\begin{aligned} & H(x^{k+1}, y^k) + f(x^{k+1}) \\ & \leq H(x^k, y^k) + f(x^k) - \frac{1}{2} \left( \frac{1}{c_k} - L \right) \|x^{k+1} - x^k\|^2 \\ & = H(x^k, y^k) + f(x^k) - \frac{1}{2}(\gamma - 1)L\|x^{k+1} - x^k\|^2, \end{aligned}$$

以及

$$\begin{aligned} & H(x^{k+1}, y^{k+1}) + g(y^{k+1}) \\ & \leq H(x^{k+1}, y^k) + g(y^k) - \frac{1}{2} \left( \frac{1}{d_k} - L \right) \|y^{k+1} - y^k\|^2 \\ & = H(x^{k+1}, y^k) + g(y^k) - \frac{1}{2}(\gamma - 1)L\|y^{k+1} - y^k\|^2. \end{aligned}$$



## 证明

将上述两个不等式相加，消去 $H(x^{k+1}, y^k)$ ，得到

$$\begin{aligned} & \Psi(z^k) - \Psi(z^{k+1}) \\ &= H(x^k, y^k) + f(x^k) + g(y^k) - H(x^{k+1}, y^{k+1}) - f(x^{k+1}) - g(y^{k+1}) \\ &\geq \frac{1}{2}(\gamma - 1)L (\|x^{k+1} - x^k\|^2 + \|y^{k+1} - y^k\|^2). \end{aligned}$$

由此立即可得

$$\frac{\rho_1}{2} \|z^{k+1} - z^k\|^2 \leq \Psi(z^k) - \Psi(z^{k+1}). \quad (5)$$

此外，容易得知迭代点处的函数值 $\{\Psi(z^k)\}$ 关于 $k$ 是单调递减的。根据假设 $\inf \Psi > -\infty$ 可知 $\Psi(z^k)$ 单调下降收敛到一个有限的数 $\Psi^*$ 。

# 证明

(2) 设 $N$ 为任意的整数，在(5)式中对 $k$ 求和，得

$$\sum_{k=0}^{N-1} \|z^{k+1} - z^k\|^2 \leq \frac{2}{\rho_1} (\Psi(z^0) - \Psi(z^N)) \leq \frac{2}{\rho_1} (\Psi(z^0) - \Psi^*).$$

令 $N \rightarrow \infty$ 即可得 $\sum_{k=0}^{\infty} \|z^{k+1} - z^k\|^2 < +\infty$ ，从而

$$\lim_{k \rightarrow \infty} \|z^{k+1} - z^k\| = 0$$

注：定理表明进行一轮近似点交替线性化迭代后，函数值下降量的下界可被相邻迭代点之间的距离控制。几乎所有下降类的算法在一定条件下都满足这个性质。到此我们完成了收敛性分析的第一个步骤。

# 次梯度上界

- 在上一步中我们证明了迭代点处的函数值 $\psi^k$ 最终会收敛到某个值
- 但是这个值和局部最优解的关系还没有明确说明
- 序列 $\{z^k\}$ 的收敛性质在上面的定理中也没有体现
- 在这一部分我们将讨论序列 $\{z^k\}$ 是否会趋于某个临界点，这是收敛性框架中的第二个步骤

## 次梯度上界

在假设条件下, 设 $\{z^k\}$ 是迭代格式产生的有界序列, 对任意的整数 $k$ , 定义

$$A_x^k = \frac{1}{c_{k-1}}(x^{k-1} - x^k) + \nabla_x H(x^k, y^k) - \nabla_x H(x^{k-1}, y^{k-1}),$$

以及

$$A_y^k = \frac{1}{d_{k-1}}(y^{k-1} - y^k) + \nabla_y H(x^k, y^k) - \nabla_y H(x^k, y^{k-1}).$$

则有 $(A_x^k, A_y^k) \in \partial\Psi(x^k, y^k)$  且

$$\|(A_x^k, A_y^k)\| \leq \|A_x^k\| + \|A_y^k\| \leq \rho_2 \|z^k - z^{k-1}\|,$$

其中 $\rho_2 = (2\gamma + 3)L$ .

## 证明

由迭代格式中更新 $x^k$ 的一阶最优性条件可知

$$\nabla_x H(x^{k-1}, y^{k-1}) + \frac{1}{c_{k-1}}(x^k - x^{k-1}) + u^k = 0,$$

其中 $u^k \in \partial f(x^k)$ 为 $f$ 的一个次梯度. 因此我们有

$$\nabla_x H(x^{k-1}, y^{k-1}) + u^k = \frac{1}{c_{k-1}}(x^{k-1} - x^k).$$

同理, 由迭代格式中关于 $y^k$ 的更新可知

$$\nabla_y H(x^k, y^{k-1}) + v^k = \frac{1}{d_{k-1}}(y^{k-1} - y^k),$$

其中 $v^k \in \partial g(y^k)$ 为 $g$ 的一个次梯度. 由 $A_x^k, A_y^k$ 的定义和 $\partial\Psi$ 的表达式可得

$$A_x^k = \nabla_x H(x^k, y^k) + u^k \in \partial_x \Psi(x^k, y^k),$$

$$A_y^k = \nabla_y H(x^k, y^k) + v^k \in \partial_y \Psi(x^k, y^k).$$

即有 $(A_x^k, A_y^k) \in \partial\Psi(x^k, y^k)$ , 我们需要证明的第一个结论因此成立.

## 证明

下面估计 $A_x^k$ 和 $A_y^k$ 的模长. 这里需要借助假设的(2), 即 $\nabla H$ 在有界集上关于 $(x, y)$ 是联合利普希茨连续的. 因此对 $\|A_x^k\|$ 我们有

$$\begin{aligned}\|A_x^k\| &\leq \frac{1}{c_{k-1}} \|x^{k-1} - x^k\| + \|\nabla_x H(x^k, y^k) - \nabla_x H(x^{k-1}, y^{k-1})\| \\ &\leq \frac{1}{c_{k-1}} \|x^{k-1} - x^k\| + L(\|x^{k-1} - x^k\| + \|y^{k-1} - y^k\|) \\ &= \left(L + \frac{1}{c_{k-1}}\right) \|x^{k-1} - x^k\| + L\|y^{k-1} - y^k\| \\ &= (\gamma + 1)L\|x^{k-1} - x^k\| + L\|y^{k-1} - y^k\| \\ &\leq (\gamma + 2)L\|z^{k-1} - z^k\|.\end{aligned}$$

其中, 第二个不等式是根据 $\nabla H$ 的利普希茨连续性, 最后一个不等式是将 $\|x^{k-1} - x^k\|$ 和 $\|y^{k-1} - y^k\|$  统一放大为 $\|z^{k-1} - z^k\|$ .

另一方面，对 $\|A_y^k\|$ 的估计只需要用到 $\nabla H$ 关于 $y$ 的利普希茨连续性：

$$\begin{aligned}
 \|A_y^k\| &\leq \frac{1}{d_{k-1}} \|y^k - y^{k-1}\| + \|\nabla_y H(x^k, y^k) - \nabla_y H(x^k, y^{k-1})\| \\
 &\leq \frac{1}{d_{k-1}} \|y^k - y^{k-1}\| + L \|y^k - y^{k-1}\| \\
 &= \left( \frac{1}{d_{k-1}} + L \right) \|y^k - y^{k-1}\| \\
 &\leq (\gamma + 1)L \|z^k - z^{k-1}\|.
 \end{aligned}$$

结合这两个估计我们最终得到

$$\|(A_x^k, A_y^k)\| \leq \|A_x^k\| + \|A_y^k\| \leq (2\gamma + 3)L \|z^k - z^{k-1}\| = \rho_2 \|z^k - z^{k-1}\|.$$

# 子列收敛性

- 上面的分析表明,  $\partial\Psi(z^k)$  将会包含一个模长不断趋于0的向量, 这暗示着某种收敛性. 由于有界序列  $\{z^k\}$  一定有收敛的子列, 因此猜想  $\{z^k\}$  的极限点应该和  $\Psi$  的临界点有一定的关系. 我们有:
- 定义  $\omega(z^0)$  为近似点交替线性化方法从点  $z^0$  出发产生迭代序列的所有极限点集, 且  $\{z^k\}$  是有界序列, 则以下结论成立:
  - (1)  $\emptyset \neq \omega(z^0) \subset \text{crit } \Psi$ , 其中  $\text{crit } \Psi$  定义为  $\Psi$  所有的临界点;
  - (2)  $z^k$  与集合  $\omega(z^0)$  的距离趋于0, 即

$$\lim_{k \rightarrow \infty} \text{dist}(z^k, \omega(z^0)) = 0;$$

- (3)  $\omega(z^0)$  是非空的连通紧集;
- (4)  $\Psi$  在  $\omega(z^0)$  上是一个有限的常数.



# 子列收敛性

- 上面的结论表明从点 $z^0$ 出发产生的点列 $\{z^k\}$ 的极限点都是 $\Psi$ 的临界点（次梯度集含有零向量）。
- 至此我们已经得到了迭代序列 $\{z^k\}$ 的子列收敛性，这至少保证了算法在迭代过程中与临界点越来越接近。
- 一个自然的问题就是： $\{z^k\}$ 全序列在何种条件下收敛？
- 这就要进入理论分析的第三个步骤：利用函数的KL性质。

# KL 性质

- 定义  $\Phi_\eta$  是凹连续函数  $\varphi: [0, \eta) \rightarrow \mathbb{R}_+$  的集合且满足如下条件: (i)  $\varphi(0) = 0$ ; (ii)  $\varphi$  在  $(0, \eta)$  内连续可微, 在点 0 处连续; (iii) 对任意的  $s \in (0, \eta)$ , 都有  $\varphi'(s) > 0$ .
- 设  $\sigma: \mathbb{R}^d \rightarrow (-\infty, +\infty]$  是适当下半连续函数.
  - (1) 称函数  $\sigma$  在给定点  $\bar{u} \in \mathbf{dom} \partial\sigma \stackrel{\text{def}}{=} \{u \mid \partial\sigma(u) \neq \emptyset\}$  处具有 KL 性质, 若存在  $\eta \in (0, +\infty]$  和  $\bar{u}$  的一个邻域  $U$  以及函数  $\varphi \in \Phi_\eta$ , 使得

$$\forall u \in U \cap [\sigma(\bar{u}) < \sigma < \sigma(\bar{u}) + \eta],$$

以下不等式成立:

$$\varphi'(\sigma(u) - \sigma(\bar{u})) \cdot \text{dist}(0, \partial\sigma(u)) \geq 1,$$

其中  $\text{dist}(x, S)$  表示点  $x$  到集合  $S$  的距离.

- (2) 若  $\sigma$  在  $\mathbf{dom} \partial\sigma$  上处处满足 KL 性质, 则称  $\sigma$  是一个 KL 函数.

# KL性质的解释

- 一大类函数都具有KL性质，该性质刻画了函数本身在给定点 $\bar{u}$ 处的某种行为。
- 如果点 $\bar{u}$ 不是函数 $\sigma$ 的临界点，那么KL性质在点 $\bar{u}$ 处自然成立。因此KL性质成立的不平凡情形是 $\bar{u}$ 是 $\sigma$ 的临界点，即 $0 \in \partial\sigma(\bar{u})$ 。
- 这种情况下KL性质保证了“函数 $\sigma$ 可被锐化”。直观上来说，令

$$\tilde{\varphi}(u) = \varphi(\sigma(u) - \sigma(\bar{u})),$$

KL性质在某种条件下可以改写成

$$\text{dist}(0, \partial\tilde{\varphi}(u)) \geq 1,$$

其中 $u$ 的取法需要保证 $\sigma(u) > \sigma(\bar{u})$ 。

- 以上性质表明，无论 $u$ 多么接近临界点 $\bar{u}$ ， $\tilde{\varphi}(u)$ 的次梯度的模长均大于1。所以KL性质也被称为是函数 $\sigma$ 在重参数化子 $\varphi$ 下的一个锐化，这种几何性质在分析一阶算法的收敛性时起到关键作用。

半代数, 次分析以及对数指数函数是KL函数

- $\mathbb{R}^d$  的子集  $S$  是一个半代数集, 如果存在有限个实多项式函数  $g_{ij}, h_{ij} : \mathbb{R}^d \rightarrow \mathbb{R}$  使得

$$S = \bigcup_{j=1}^p \bigcap_{i=1}^q \{u \in \mathbb{R}^d : g_{ij}(u) = 0, h_{ij}(u) < 0\}$$

- 函数  $h : \mathbb{R}^d \rightarrow (-\infty, +\infty]$  称为半代数的, 如果它的图  $\{(u, t) \in \mathbb{R}^{d+1} : h(u) = t\}$  是  $\mathbb{R}^{d+1}$  的半代数子集
- 设  $\sigma(u) : \mathbb{R}^d \rightarrow (-\infty, +\infty)$  是下半连续的恰当函数. 若  $\sigma$  是半代数的, 则它在 **dom**  $\sigma$  中任一点处满足KL性质.

例子：

- 实多项式函数.
- 半代数集的指示函数.
- 半代数函数的有限和与有限乘积.
- 半代数函数的复合.
- 上极限/下极限类函数. 例如, 当 $g$ 是半代数函数并且 $C$ 是半代数集时,  $\sup\{g(u, v) : v \in C\}$ 是半代数的.
- 半正定矩阵锥, **Stiefel**流形以及恒秩矩阵都是半代数集.
- $S$ 是 $\mathbb{R}^d$ 中的非空半代数子集, 则函数 $x \rightarrow \text{dist}(x, S)^2$ 是半代数的.
- $\|\cdot\|_0, \|\cdot\|_p$ 是半代数函数, 其中 $p$ 是有理数.

# 一致KL性质

由于非凸问题有多个临界点，有时单个点 $\bar{u}$ 处的KL性质是不够的，我们需要引入一致KL性质：

- 设 $\Omega$ 是紧集， $\sigma: \mathbb{R}^d \rightarrow (-\infty, +\infty]$ 是适当下半连续函数，在 $\Omega$ 上为常数且在 $\Omega$ 的每个点处都满足KL性质，则存在 $\varepsilon > 0, \eta > 0, \varphi \in \Phi_\eta$ 使得对任意 $\bar{u} \in \Omega$ 和所有满足以下条件的 $u$ ：

$$\{u \in \mathbb{R}^d : \text{dist}(u, \Omega) < \varepsilon\} \cap [\sigma(\bar{u}) < \sigma < \sigma(\bar{u}) + \eta],$$

有

$$\varphi'(\sigma(u) - \sigma(\bar{u}))\text{dist}(0, \partial\sigma(u)) \geq 1.$$

## 证明

- 因为 $\mathbb{R}^d$ 上的紧集可以由有限多个开集覆盖，因此该问题可在有限个点上进行讨论。设 $\mu$ 是 $\sigma$ 在 $\Omega$ 上的取值。由于 $\Omega$ 是紧集，根据有限覆盖定理，存在有限多个开球 $B(u_i, \varepsilon_i)$ （其中 $u_i \in \Omega, i = 1, 2, \dots, p$ ）使得 $\Omega \subset \bigcup_{i=1}^p B(u_i, \varepsilon_i)$ 。
- 现在考虑这些点 $u_i$ 。在点 $u_i$ 上KL性质成立，设 $\varphi_i: [0, \eta_i) \rightarrow \mathbb{R}_+$ 是对应的重参数化子，则对任意 $u \in B(u_i, \varepsilon_i) \cap [\mu < \sigma < \mu + \eta_i]$ ，有逐点的KL性质：

$$\varphi_i'(\sigma(u) - \mu) \text{dist}(0, \partial\sigma(u)) \geq 1.$$

取充分小的 $\varepsilon > 0$ 使得

$$U_\varepsilon \stackrel{\text{def}}{=} \{u \in \mathbb{R}^d \mid \text{dist}(u, \Omega) \leq \varepsilon\} \subset \bigcup_{i=1}^p B(u_i, \varepsilon_i).$$

# 证明

- 取  $\eta = \min_i \eta_i$ , 以及

$$\varphi(s) = \int_0^s \max_i \varphi'_i(t) dt, \quad s \in [0, \eta].$$

容易验证  $\varphi \in \Phi_\eta$ .

- 对任意的  $u \in U_\varepsilon \cap [\mu < \sigma < \mu + \eta]$ ,  $u$  必定落在某个球  $B(u_{i_0}, \varepsilon_{i_0})$  中, 我们有

$$\begin{aligned} \varphi'(\sigma(u) - \mu) \text{dist}(0, \partial\sigma(u)) &= \max_i \varphi'_i(\sigma(u) - \mu) \text{dist}(0, \partial\sigma(u)) \\ &\geq \varphi'_{i_0}(\sigma(u) - \mu) \text{dist}(0, \partial\sigma(u)) \geq 1. \end{aligned}$$

即一致KL性质成立.



# 有限长度性质

设 $\Psi$ 是KL函数，且满足假设条件，则以下结论成立：

① 序列 $\{z^k\}$ 的长度有限，即

$$\sum_{k=1}^{\infty} \|z^{k+1} - z^k\| < +\infty.$$

② 序列 $\{z^k\}$ 收敛到 $\Psi$ 的一个临界点 $z^* = (x^*, y^*)$ .

注：上述定理的(1)有别于第一个步骤中充分下降定理的(2)：后者只得到了 $\|z^{k+1} - z^k\|$ 平方可和的结论，而前者则说明从 $z^0$ 出发，迭代序列的轨迹长度是有限的。这个结论显然比充分下降定理要强，也是推导全序列收敛的关键。

# 证明

- 由于 $\{z^k\}$ 是有界序列，存在收敛子列 $\{z^{k_q}\} \rightarrow \bar{z}, q \rightarrow \infty$ 。和之前的推导类似，不管全序列 $\{z^k\}$ 收敛性如何，对应的函数值列 $\{\Psi(z^k)\}$ 总是收敛的，且

$$\lim_{k \rightarrow \infty} \Psi(z^k) = \Psi(\bar{z}). \quad (6)$$

以下不妨设 $\Psi(\bar{z}) < \Psi(z^k)$ 。这是因为若存在 $\bar{k}$ 使得 $\Psi(z^{\bar{k}}) = \Psi(\bar{z})$ ，由充分下降性可知 $z^{\bar{k}+1} = z^{\bar{k}}$ ，进而有 $z^k = z^{\bar{k}}, \forall k > \bar{k}$ 。结论自然成立。

- 由极限(6)和极限点集 $\omega(z^0)$ 的性质 $\lim_{k \rightarrow \infty} \text{dist}(z^k, \omega(z^0)) = 0$ 可知对任意的 $\varepsilon, \eta > 0$ ，存在充分大的正整数 $l$ ，使得对任意的 $k > l$ ，

$$\Psi(z^k) < \Psi(\bar{z}) + \eta, \quad \text{dist}(z^k, \omega(z^0)) < \varepsilon.$$

- 以上的分析说明当 $k$ 充分大时，迭代点序列最终会满足一致 KL 性质的前提。下面就在这个结论下分别证明定理的两个结论。

## 证明

- (1) 根据临界点的性质,  $\omega(z^0)$  是非空紧集, 且  $\Psi$  在  $\omega(z^0)$  上是常数. 在一致KL性质中令  $\Omega = \omega(z^0)$ , 对任意的  $k > l$ ,

$$\varphi'(\Psi(z^k) - \Psi(\bar{z})) \text{dist}(0, \partial\Psi(z^k)) \geq 1.$$

根据第二个步骤中次梯度上界的引理可知

$$\text{dist}(0, \partial\Psi(z^k)) \leq \|(A_x^k, A_y^k)\| \leq \rho_2 \|z^k - z^{k-1}\|.$$

代入KL性质有

$$\varphi'(\Psi(z^k) - \Psi(\bar{z})) \geq \frac{1}{\rho_2} \|z^k - z^{k-1}\|^{-1}. \quad (7)$$

另外, 由  $\varphi$  的凹性, 有

$$\begin{aligned} & \varphi(\Psi(z^k) - \Psi(\bar{z})) - \varphi(\Psi(z^{k+1}) - \Psi(\bar{z})) \\ & \geq \varphi'(\Psi(z^k) - \Psi(\bar{z}))(\Psi(z^k) - \Psi(z^{k+1})). \end{aligned} \quad (8)$$

## 证明

为了表示方便, 定义

$$\Delta_{p,q} = \varphi(\Psi(z^p) - \Psi(\bar{z})) - \varphi(\Psi(z^q) - \Psi(\bar{z})),$$

其中  $p, q$  为任意正整数. 定义常数

$$C = \frac{2\rho_2}{\rho_1} > 0.$$

根据不等式(8), 使用(7)式和第一个步骤中的充分下降定理分别估计不等号右边的两项, 有

$$\begin{aligned}\Delta_{k,k+1} &\geq \varphi'(\Psi(z^k) - \Psi(\bar{z}))(\Psi(z^k) - \Psi(z^{k+1})) \\ &\geq \frac{1}{\rho_2} \|z^k - z^{k-1}\|^{-1} \cdot \frac{\rho_1}{2} \|z^{k+1} - z^k\|^2 \\ &= \frac{\|z^{k+1} - z^k\|^2}{C \|z^k - z^{k-1}\|},\end{aligned}$$

等价于

$$\|z^{k+1} - z^k\| \leq \sqrt{C \Delta_{k,k+1} \|z^k - z^{k-1}\|}.$$

# 证明

根据基本不等式  $2\sqrt{ab} \leq a + b, \forall a, b > 0$ , 我们取  $a = \|z^k - z^{k-1}\|$ ,  $b = C\Delta_{k,k+1}$ , 则

$$2\|z^{k+1} - z^k\| \leq \|z^k - z^{k-1}\| + C\Delta_{k,k+1}.$$

对任意的  $k > l$ , 在上式中把  $k$  替换成  $i$  并对  $i = l+1, l+2, \dots, k$  求和, 得

$$\begin{aligned} 2 \sum_{i=l+1}^k \|z^{i+1} - z^i\| &\leq \sum_{i=l+1}^k \|z^i - z^{i-1}\| + C \sum_{i=l+1}^k \Delta_{i,i+1} \\ &\leq \sum_{i=l+1}^k \|z^{i+1} - z^i\| + \|z^{l+1} - z^l\| + C\Delta_{l+1,k+1}. \end{aligned}$$

最后一个不等式是因为  $\Delta_{p,q} + \Delta_{q,r} = \Delta_{p,r}$ .

# 证明

注意到上式不等号右边刚好可以和左边部分抵消，我们有

$$\begin{aligned} & \sum_{i=l+1}^k \|z^{i+1} - z^i\| \\ & \leq \|z^{l+1} - z^l\| + C\left(\varphi(\Psi(z^{l+1}) - \Psi(\bar{z})) - \varphi(\Psi(z^{k+1}) - \Psi(\bar{z}))\right) \\ & \leq \|z^{l+1} - z^l\| + C\varphi(\Psi(z^{l+1}) - \Psi(\bar{z})). \end{aligned}$$

不等式右边是有界的数且与 $k$ 无关，由级数收敛的定义立即可得

$$\sum_{k=1}^{\infty} \|z^{k+1} - z^k\| < +\infty.$$

- (2) 在  $\sum_{k=1}^{\infty} \|z^{k+1} - z^k\| < +\infty$  的前提下  $\{z^k\}$  全序列收敛是显然的. 这等价于证明  $\{z^k\}$  是柯西列. 对任意  $q > p > l$ ,

$$z^q - z^p = \sum_{k=p}^{q-1} (z^{k+1} - z^k),$$

根据三角不等式,

$$\|z^q - z^p\| = \left\| \sum_{k=p}^{q-1} (z^{k+1} - z^k) \right\| \leq \sum_{k=p}^{q-1} \|z^{k+1} - z^k\|,$$

而  $\|z^{k+1} - z^k\|$  的可和性意味着  $\sum_{k=l+1}^{\infty} \|z^{k+1} - z^k\|$  趋于 0. 因此  $\{z^k\}$  是一个柯西列, 算法产生的迭代序列有全序列收敛性.

- 1 分块坐标下降法
- 2 应用举例
- 3 收敛性分析
- 4 HOGWILD! 异步SGD**
- 5 CYCLADES



# 随机梯度下降

考虑优化问题

$$\min_{x \in \mathbb{R}^n} f(x) = \sum_{e \in E} f_e(x_e)$$

- $e$  表示  $\{1, 2, \dots, n\}$  的一个子集.
- 在许多机器学习问题中,  $n$  和  $|E|$  都很大. 注意: “集合”  $E$  可以包含相同元素  $e$  的多个拷贝.
- $f_e$  只在  $x$  的几个分量上起作用, 即  $x_e$ .
- $f_e$  可简单地看作  $\mathbb{R}^n$  上的函数, 只需要忽略那些不在子集  $e$  中的分量.

# 问题定义

## 例子：机器学习中的应用

- 最小化经验风险

$$\min_x \frac{1}{n} \sum_{i=1}^n l_i(a_i^T x)$$

- $a_i$  表示第  $i$  个数据点,  $x$  是模型.  $l_i$  是一个损失函数.
- 逻辑回归, 最小二乘, SVM ...
- 如果每个  $a_i$  都是稀疏的, 那它就是我们讨论的问题.

# 问题定义

## 例子：一般最小化问题

- 最小化如下的问题

$$\min_{x_1, \dots, x_{m_1}} \min_{y_1, \dots, y_{m_2}} \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \phi_{i,j}(x_i, y_j)$$

- $\phi_{i,j}$  是凸的标量函数.  $x_i$  和  $y_j$  是向量.
- 矩阵补全和矩阵分解问题：

$$\phi_{i,j} = (A_{i,j} - x_i^T y_j)^2$$

- $n = m_1 m_2$  个函数，每个函数只涉及两个变量.
- 一个变量最多被  $m_1 + m_2$  个函数共享.

# 随机梯度下降

- 梯度法格式为

$$x_{k+1} = x_k - \gamma_k \nabla f(x_k)$$

其中  $\nabla f(x_k) = \sum_{e \in E} \nabla f_e(x_k)$ .

- 随机梯度下降(SGD)

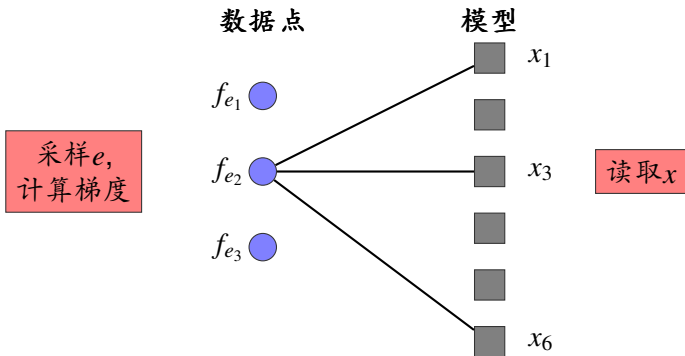
$$x_{k+1} = x_k - \gamma_k \nabla f_{s_k}(x_k)$$

其中  $s_k$  从  $E$  中随机采样得到.

- $\gamma_k$  是步长(或学习率). 它可以是一个常数或逐渐减小.
- SGD的想法: 对于很大的  $|E|$  和  $n$ , 计算全梯度  $\nabla f(x)$  的代价可能相当大. 是否可以只计算  $\nabla f(x)$  的一小部分同时保证算法收敛性?

# 随机梯度下降

一步SGD:



# SGD的优势

SGD 的流行有下列几个原因:

- 相对经典的梯度下降更少的计算量.
- 对噪声更加鲁棒.
- 实现简单.
- 接近最优的学习表现.
- 内存资源占用少.
- 稳定.

# 并行SGD

- 在大数据集上，SGD可能需要更新非常多步。
- 目标：SGD的并行版本。
- 如何并行化SGD？
  - 并行化一步更新 – 即使是深度网络，计算 $\nabla f_{s_k}(x_k)$ 也是廉价的。因此写可并行的计算代码可能并不值得。
  - 并行化更新 – SGD是串行的，所以并行化几乎不可能实现。
  - 我们是否可以并行化一个串行算法呢？

# 并行SGD

如何并行化一个串行算法？

- **No** – 几乎所有情形.
- **Almost yes** – 有稀疏结构的问题.
- 对于我们的问题:

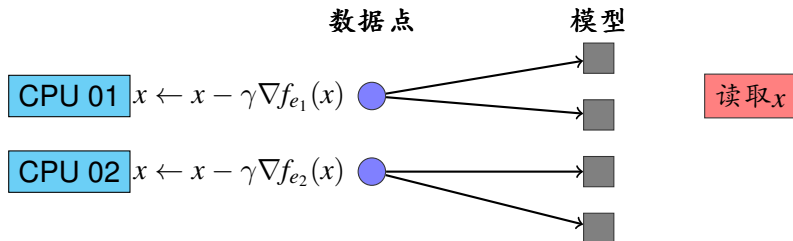
$$\min_{x \in \mathbb{R}^n} f(x) = \sum_{e \in E} f_e(x_e)$$

如果大多数 $f_e$ 都不涉及 $x$ 的相同分量, 那么我们可以利用稀疏性来并行化SGD.



# 并行SGD

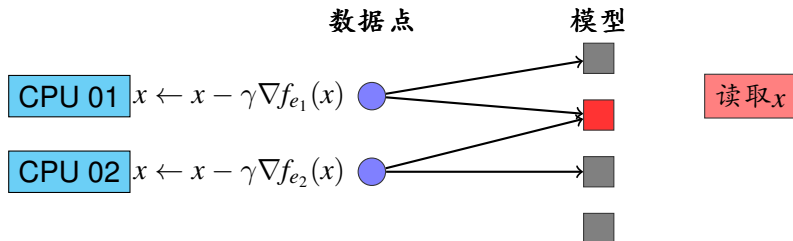
理想情形：：



- $f_{e_1}$  和  $f_{e_2}$  涉及的分量没有重叠.
- 两个并行和两个串行更新等价.
- **没有冲突** 意味着可以加速!

# 并行SGD

非理想情形：



- $f_{e_1}$  和  $f_{e_2}$  都涉及  $x_2$ .
- 冲突意味着更少的并行化.
- 如何解决这类冲突?

## 为什么冲突影响那么大？

- CPU 不能直接访问内存.
- 计算只能在CPU 缓存上完成.
- 数据从内存读取到CPU 缓存. 计算完成后, 计算结果又被传回内存.
- CPU 并不知道是否有其他CPU的本地更新尚未传回内存.

## 如何解决冲突问题？

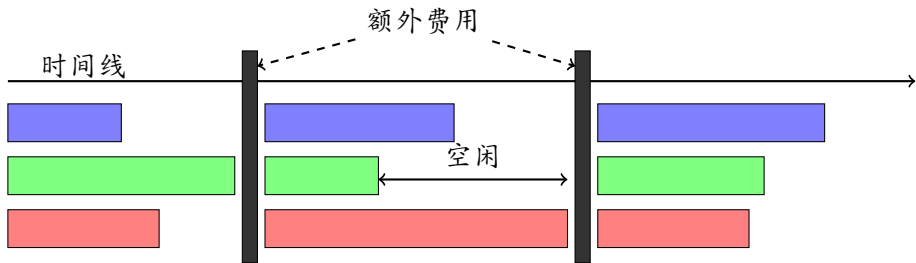
- 通过“坐标”或“内存锁定”的方法。
- 锁定类型：排他锁。
- 告诉其他CPU：我现在正在更新变量，在我完成之前请不要做任何修改。
- 确保执行并行SGD时的正确性。
- 仅能提供有限的加速。当冲突过于频繁时，情况会更糟——甚至比SGD的顺序版本还要慢！

# 异步更新

Q: 如何解决冲突问题?

A: 在异步编程中可以直接忽略它.

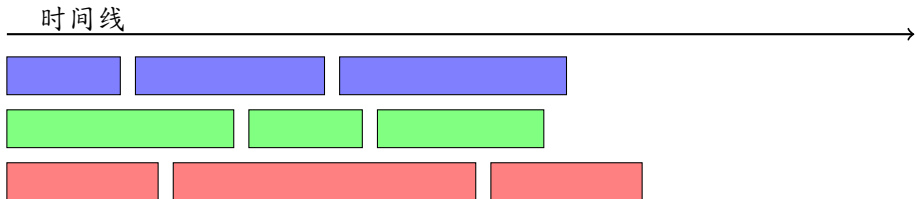
同步算法:



- 负载不均衡导致空闲.
- 正确但速度慢.

# 异步更新

异步算法:



- 处理器之间不需要同步.
- 没有空闲时间—每个处理器始终在工作.
- 高可扩展性.
- 有噪声但速度快.

# Hogwild! 异步SGD

如果我们删除并行SGD中的所有锁定和同步进程，就可以得到Hogwild! 算法。

---

## Algorithm 2 Hogwild! 异步SGD

---

- 1: 每个处理器异步做
  - 2: **loop**
  - 3:   从 $E$ 中随机采样 $e$ .
  - 4:   从内存中读取数据 $x_e$ .
  - 5:   计算 $G_e(x) := |E|\nabla f_e(x)$ .
  - 6:   **for**  $v \in e$  **do**
  - 7:      $x_v \leftarrow x_v - \gamma b_v^T G_e(x)$ .
  - 8:   **end for**
  - 9: **end loop**
-

# Hogwild! 异步SGD

Hogwild! 的一种变体:

---

## Algorithm 3 Hogwild! 异步SGD(变体1)

---

- 1: 每个处理器异步做
  - 2: **loop**
  - 3: 从 $E$  中随机采样 $e$ .
  - 4: 从内存中读取数据 $x_e$ .
  - 5: 计算 $G_e(x) := |E|\nabla f_e(x)$ .
  - 6: 采样 $v \in e$ , 然后 $x_v \leftarrow x_v - \gamma|e|b_v^T G_e(x)$ .
  - 7: **end loop**
- 

注意:

- 计算了 $f_e$  的全梯度, 但只更新一个分量.
- 因子 $|e|$  保证了 $\mathbb{E}[|e|b_v^T G_e(x)] = \nabla f(x)$ .
- 看起来浪费计算量, 但是更容易分析.



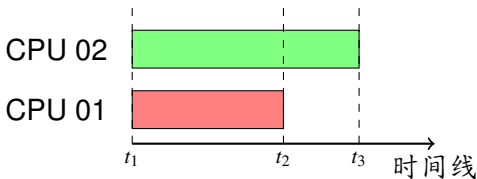
## 异步更新的问题

- CPU 缓存和内存的不一致导致结果不正确.
- 较旧的更新可能被较新的更新覆盖.
- 假设我们想用两个线程执行两次更新

$$\begin{aligned}x_3 &= x_2 - \gamma \nabla f_{e_2}(x_2) \\ &= x_1 - \gamma (\nabla f_{e_2}(x_2) + \nabla f_{e_1}(x_1))\end{aligned}$$

- $\nabla f_{e_1}(x_1)$  和  $\nabla f_{e_2}(x_2)$  的计算被分别分配到CPU1 和CPU2.

# 异步更新的问题



时间	$\nabla f$ 的 $x$		内存中的 $x$	缓存中的 $\nabla f$		执行的操作	
$t_1$	$x_1$	$x_1$	$x_1$	—	—	从内存中读取 $x_1$	
$t_2$	$x_2$	$x_1$	$x_1 - \gamma \nabla f_{e_1}(x_1)$	$\nabla f_{e_1}(x_1)$	—	更新内存	计算
$t_3$	$x_2$	$x_2$	$x_2 - \gamma \nabla f_{e_2}(x_1)$	$\nabla f_{e_1}(x_1)$	$\nabla f_{e_2}(x_1)$	空闲	更新内存

- 我们**希望**得到:  $x_1 - \gamma(\nabla f_{e_1}(x_1) + \nabla f_{e_2}(x_2))$ .
- 我们**实际**得到:  $x_1 - \gamma(\nabla f_{e_1}(x_1) + \nabla f_{e_2}(x_1))$ .

# Hogwild!的分析

## 假设条件

- $\nabla f$  是利普西茨连续的.

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$$

- $f$  是强凸函数. 每个  $f_e$  都是凸函数.

$$f(y) \geq f(x) + (y - x)^T \nabla f(x) + \frac{c}{2} \|y - x\|^2$$

- $f_e$  的梯度有界. 由于  $G_e(x) = |E| \nabla f_e(x)$ ,

$$\|G_e(x)\| \leq M, \forall x$$

- $\gamma c < 1$ , 否则即使是梯度下降法也有可能失败.
- 基于Hogwild!(变体1) 算法.

# 主要结果

## Proposition 4.1 (主要结果).

假设从计算梯度到在步骤 $j$ 中使用它之间的延迟, 即 $j - k(j)$ , 总是小于等于 $\tau$ , 并且 $\gamma$  定义为

$$\gamma = \frac{\theta \varepsilon c}{2LM^2\Omega(1 + 6\rho\tau + 4\tau^2\Omega\Delta^{1/2})} \quad (9)$$

其中 $\varepsilon > 0$  且 $\theta \in (0, 1)$ . 定义 $D_0 = \|x_0 - x_\star\|^2$  并设整数 $k$  满足

$$k \geq \frac{2LM^2\Omega(1 + 6\tau\rho + 6\tau^2\Omega\Delta^{1/2}) \log(LD_0/\varepsilon)}{c^2\theta\varepsilon} \quad (10)$$

那么在 $x$  的 $k$  步分量更新后, 我们有 $\mathbb{E}[f(x_k) - f_\star] \leq \varepsilon$ .

## Remarks on Prop 4.1

- 将Hogwild! 算法看作带有滞后的SGD.

$$x_{j+1} \leftarrow x_j - \gamma |e| \mathcal{P}_v G_e(x_{k(j)})$$

- 滞后的时间需要能被 $\tau$ 控制住. 也就是说, 在第 $j$ 步更新, 用于计算梯度的点在之前 $\tau$ 步内.
- $\tau$  正比于线程数.
- 步长 $\gamma$  应该是 $\mathcal{O}(\varepsilon)$  以确保收敛.
- 为了达到 $\varepsilon$  的精度, 我们需要进行至少 $\mathcal{O}(1/\varepsilon)$  步更新.

- 1 分块坐标下降法
- 2 应用举例
- 3 收敛性分析
- 4 HOGWILD! 异步SGD
- 5 CYCLADES**

# 问题定义

考虑优化问题

$$\min_{x \in \mathbb{R}^n} f(x) = \sum_{e \in E} f_e(x_e)$$

- $e$  表示  $\{1, 2, \dots, n\}$  的一个子集.
- 在许多机器学习问题中,  $n$  和  $|E|$  都很大. 注意: “集合”  $E$  可以包含相同元素  $e$  的多个拷贝.
- $f_e$  只在  $x$  的几个分量上起作用, 即  $x_e$ .
- $f_e$  可简单地看作  $\mathbb{R}^n$  上的函数, 只需要忽略那些不在子集  $e$  中的分量.

- HOGWILD! (异步SGD):

$$x_{k+1} \leftarrow x_k - \gamma_k \nabla f_{s_k}(x_k)$$

- 所有内核异步执行更新，没有内存锁定.
- 忽略冲突的变量.
- 在一些情形下没有收敛性保证(失去了SGD的一些性质)
- 复杂的理论分析|>\_<|



# 其他方式?

为了解决冲突:

- 传统并行

- 更新时锁定冲突变量.
- 保证更新的正确性, 但只能获得有限的加速.

- HOGWILD!

- 更新时忽略冲突.
- 更快的加速, 但有更高的风险不收敛.

- CYCLADES

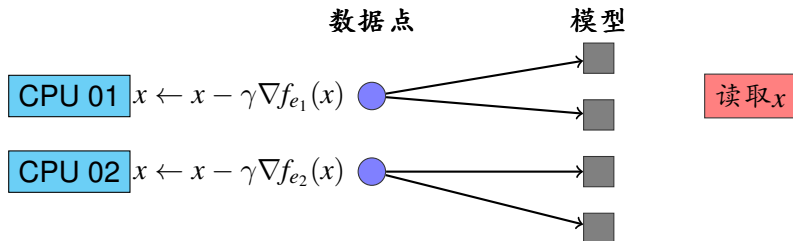
- 通过重新排列更新顺序来避免冲突.
- 更快的加速.
- 以高概率保留SGD的性质.

## 为什么冲突影响那么大？

- CPU 不能直接访问内存.
- 计算只能在CPU 缓存上完成.
- 数据从内存读取到CPU 缓存. 计算完成后, 计算结果又被传回内存.
- CPU 并不知道是否有其他CPU的本地更新尚未传回内存.

# 关于冲突

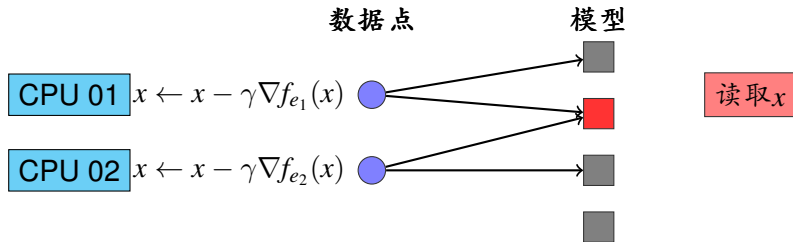
理想情形：：



- $f_{e_1}$  和  $f_{e_2}$  涉及的分量没有重叠.
- 两个并行和两个串行更新等价.
- **没有冲突** 意味着可以加速!

# 关于冲突

非理想情形：



- $f_{e_1}$  和  $f_{e_2}$  都涉及  $x_2$  .
- 冲突意味着更少的并行化.
- 如何解决这类冲突? HOGWILD! 忽略了冲突. CYCLADES 避免了冲突.

# 更新冲突图

## Definition 1 (二部更新变量图).

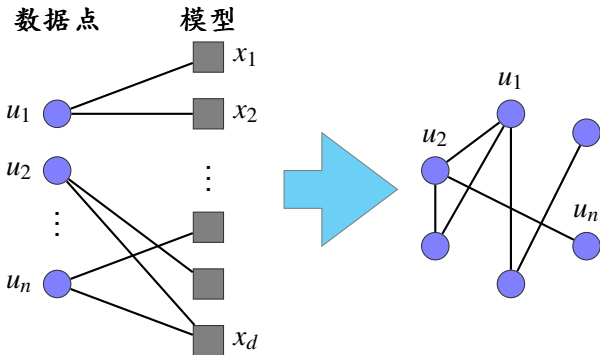
定义  $G_u$  为更新  $u_1, \dots, u_n$  与  $d$  个变量之间的二部更新变量图 (*bipartite update-variable graph*). 如果  $u_i$  需要读取和写入  $x_j$ , 则在  $G_u$  中将更新  $u_i$  链接到变量  $x_j$ .  $E_u$  表示二部图中的边数.  $\Delta_L$  表示  $G_u$  的左侧最大顶点度.  $\bar{\Delta}_L$  表示左侧平均度.

## Definition 2 (冲突图).

定义  $G_c$  为  $n$  个顶点上的冲突图, 每个顶点对应一个更新  $u_i$ .  $G_c$  的两个顶点与一条边相连, 当且仅当相应的更新在二部更新图  $G_u$  中共享至少一个变量.  $\Delta$  表示  $G_c$  的最大顶点度.

# 更新冲突图

从二部更新变量图到冲突图

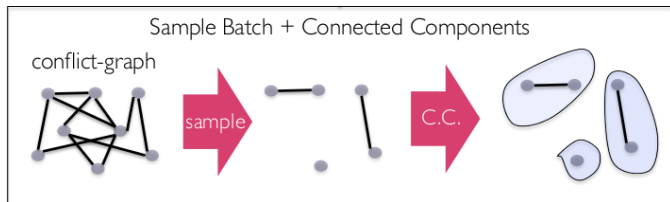


## Remarks

- 在冲突图中，如果两个顶点由一条边相连，那么它们对应的更新相互冲突.
- $G_c$  的作用是用于分析CYCLADES 算法，实际中并不会构建它.
- 假设相连的元素有 $N_{cc}$  块，那么可以使用 $N_{cc}$  个处理器来进行异步更新，避免冲突发生. 但是, 在大多数问题中 $N_{cc}$  等于1!

# CYCLADES 的想法

- 需要更多的连接块来实现并行.
- 只采样部分的更新.



如果进行采样，可以获得多少连接块？



# 连接块的个数

## Theorem 1.

$G$  是由  $n$  个顶点构成的图，最大顶点度是  $\Delta$ 。以概率  $p = \frac{1-\varepsilon}{\Delta}$  独立采样每个顶点，并定义  $G'$  为采样的顶点诱导的子图。那么在大概率意义上， $G'$  的最大连接块个数至多为  $\frac{4}{\varepsilon^2} \log n$  个。

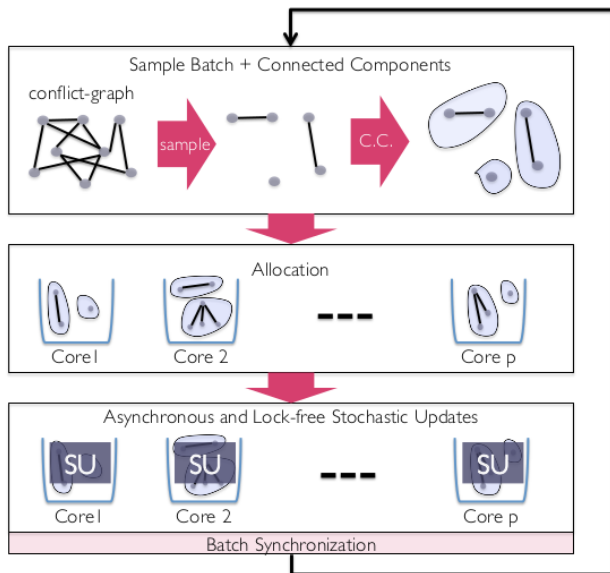
## Theorem 2.

$G$  是由  $n$  个顶点构成的图，最大顶点度是  $\Delta$ 。以概率  $B = (1-\varepsilon)\frac{n}{\Delta}$  有放回或无放回地进行采样，并定义  $G'$  为采样的顶点诱导的子图。那么在大概率的意义上， $G'$  的最大连接块个数至多为  $\mathcal{O}\left(\frac{\log n}{\varepsilon^2}\right)$  个。

## Remarks

- 定理2 可以看作是定理1 的一个推论（并不显然）。之后将给出定理1的简要证明。
- 根据定理2，如果采样  $B = (1 - \epsilon) \frac{n}{\Delta}$  个顶点，那么至少会有  $\mathcal{O}(\epsilon^2 B / \log n)$  个连接块，每块的大小最大为  $\mathcal{O}(\log n / \epsilon^2)$ 。
- 连接块的个数增加了很多—适合并行化

# CYCLADES 的想法



# CYCLADES 的想法

- 采样后，使用某个算法并行查找所有连接块（待续）。
- 确定所有连接块后，将它们分配给处理器（考虑负载均衡）。
- 每个处理器异步、独立地执行随机更新，不会出现冲突或错误共享问题。
- 重复这三个阶段直到完成。

---

## Algorithm 4 CYCLADES

---

```
1: Input:  $G_u, T, B$ .  
2: 从  $G_u$  中采样  $n_b = T/B$  个子图.  
3: for  $i = 1 : n_b$  do  
4:   将子图中的连接块分配到  $P$  个核上.  
5:   for 每个核(异步) do  
6:     在分配的连接块上执行本地更新.  
7:   end for  
8: end for
```

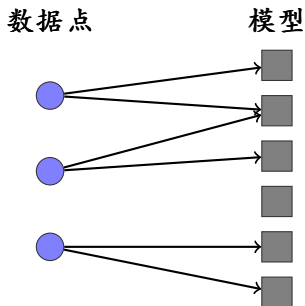
---

- $T$  表示需要运行的所有更新个数.
- 每批结束时进行一次同步.
- $G_u$  由问题的结构决定.

## 并行计算批中的连接块

- 如果我们使用冲突图 $G_c$ 来计算连接块...
- 计算代价将正比于采样边的个数.
- 然而, 构建 $G_c$ 需要 $n^2$ 次计算. 计算代价高.
- 所以只使用二部更新变量图来计算连接块!

# 并行计算批中的连接块



## 简单的信息传输想法

- 数据点发送标签.
- 变量计算min并发回.
- 数据点计算min.
- 迭代直至结束.

# 并行计算批中的连接块

## Remarks

- 所需的迭代次数可以被最长最短路径的长度（即 $G_c$ 的直径）控制住.
- 总体复杂度： $\mathcal{O}(E_u \log n / P)$ ， $P$ 是核数. 注意：当 $P$ 很大时很有用.
- 只需要二部图 $G_u$ ，不需要冲突图 $G_c$ .



## 分配连接块

计算得到批中的连接块后，就可以将这些连接块分配给 $P$ 个核了。

- $w_i$  表示用第 $i$ 个数据点进行更新的代价。
- $W_{C(i)} = \sum_{j \in C(i)} w_j$  是执行第 $i$ 个连接块中更新的代价。
- 对每个批处理，我们需要

$$\min \max W_{C(i)}$$

来获取最好的负载均衡。

- 这是一个NP困难问题。可以利用近似算法来获得次优方案。

# 分配连接块

---

## Algorithm 5 贪婪分配

---

- 1: 估计每个连接块的计算代价  $W_1, \dots, W_m$ .
  - 2: 按降序对  $W_i$  进行排序.
  - 3: **for**  $i = 1 : m$  **do**
  - 4:     选择当前最大的  $W_i$ .
  - 5:     将  $W_i$  添加到当前代价总和最少的核中.
  - 6: **end for**
- 

- 4/3-近似算法.
- $w_i$  与该更新的out-degree 成正比.

# 主要结果

## Theorem 3 (主要结果).

假设给定的更新变量图 $G_u$ 的左侧平均度和最大度分别为 $\bar{\Delta}_L$ 和 $\Delta_L$ , 并且满足 $\Delta_L/\bar{\Delta}_L \leq \sqrt{n}$ , 诱导的最大冲突度为 $\Delta$ . 那么, 在 $P = \mathcal{O}(n/\Delta\bar{\Delta}_L)$ 个核上, 取批大小为 $B = (1 - \varepsilon)\frac{n}{\Delta}$ , 对任意的 $c > 1$ , **CYCLADES**算法执行 $T = cn$ 次更新的时间大概率为

$$\mathcal{O}\left(\frac{E_u \cdot \kappa}{P} \log^2 n\right)$$

- $\kappa$  是更新 $E_u$ 中一条边的代价.
- **CYCLADES** 可以以大概率获得与串行算法相同的结果. 因此, 它需要目标函数 $f$ 及 $f_e$ 的相似属性.

## Remarks of Thm 3

- 每批的采样可以有放回或无放回.
- 核心数是有限的. 因为如果 $P$ 太大, 就很难均衡地分配连接块.
- 批大小是有限的. 较小的 $B$ 可以诱导带有很多连接块的子图.
- 定理1是证明这个主要结果的关键基础.