求解¿性系ß

Solving Linear System

马龙 赵怡浩 许晟伟 张î 许靖 程超然

前沿ž践II

2019 年 4 月 2 日

Radiosity

î 题：

如何为静态场景渲染 I 照贴图？

例如下图中的有柱 P 和窗口的？ ô

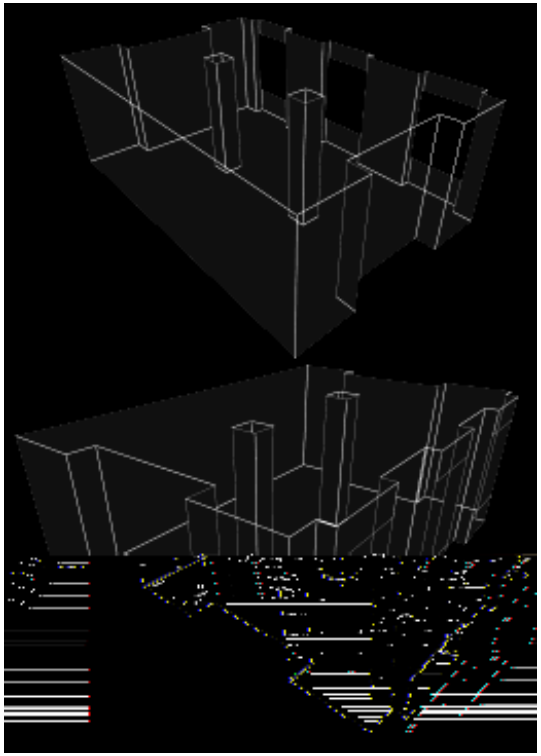


图 1: ？ ô 示意图

Radiosity

直观想法：Ï ¿ 踪！

图 2: 地5中ž时Ï ¿ 踪的应用

Radiosity

但是如果你并没有一块最新的RTX2080Ti的话...

们希望 ý 够 ~ 到一些更加“ Ï 济”的方法

不妨将 ? ô 分割 小贴片， 们来 ù 小贴片 行渲染

最简单的想法是， I 照到的小贴片是亮的， v 他是暗的

Radiosity

但是如果你并没有一块最新的RTX2080Ti的话...

们希望ý够~到一些更加“Ï济”的方法

不妨将？ô分割 小贴片， 们来ù小贴片 行渲染

最简单的想法是，Ï照到的小贴片是亮的，v他是暗的

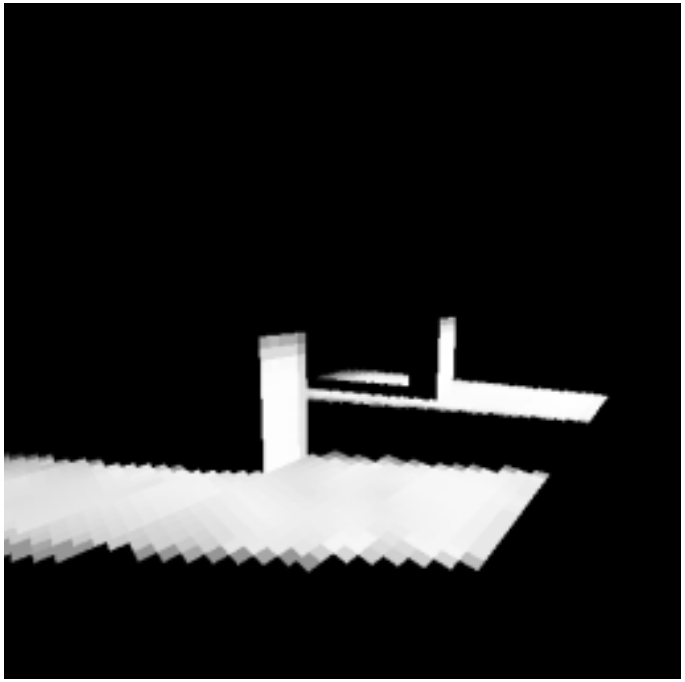
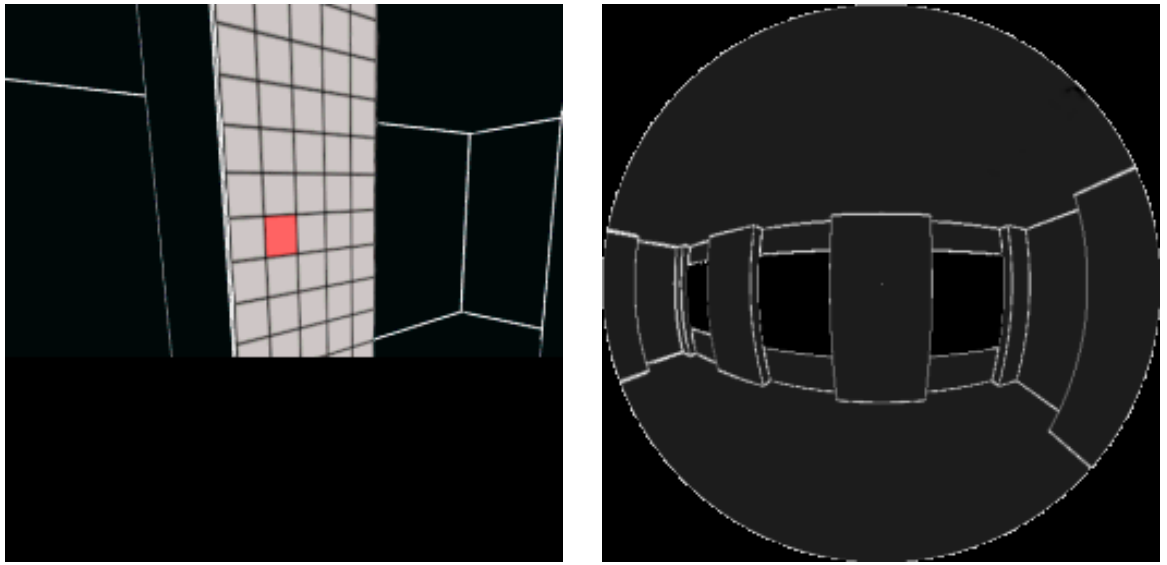


图 3: 效果不太好

Radiosity

î 题œ在了 + 反射上， 们想办法来处理 + 反射
不妨 H 从一个贴片的角度来观ß



(a) pic1. (b) pic2.

图 4: 从贴片的角度

Radiosity

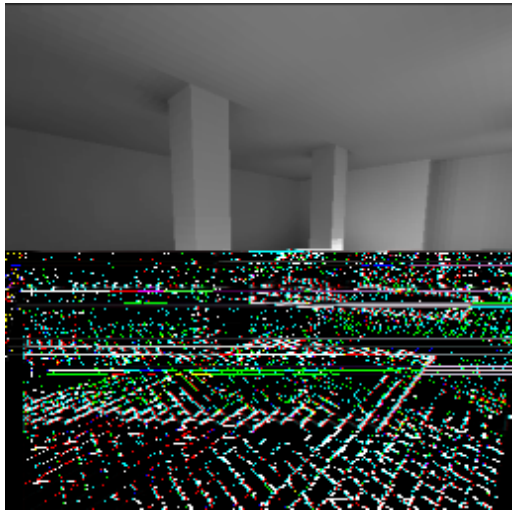
可以简单的认为，　们从贴片的角度看到的 v 他贴片会和当期贴片直接　行 + 反射
因此　们应当提œ一种计算 + 反射的算法，来解‡　个î题
　里　们介Í的是Radiosity（　射度算法）
一个贴片的　射度和他的　射 I 强（自身发 I ）　收到的 I 强和反射率有 s
v 中收到的 I 强可以简化为为是 v 他@有贴片　射度的¿性Ä合
因此，　们可以用贴片ô的一个邻接矩阵来衡量两两贴片ô的传递系数

$$B_i = E_i + R_i \sum_{j=1}^{N} B_j F_{ji}$$

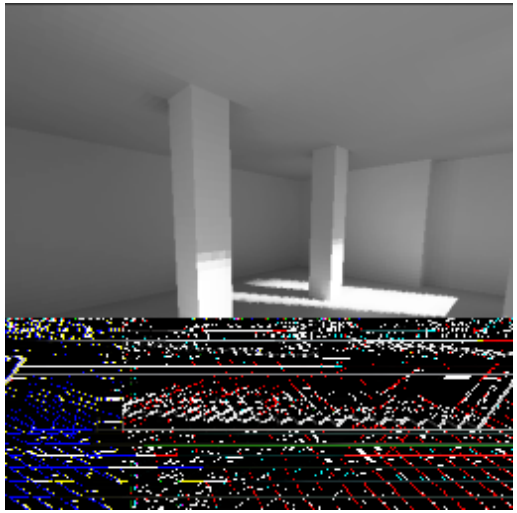
Radiosity

可以　代求解

(a) Itr.2.



(b) Itr.4.



(c) Itr.16.

也可以求解¿性方程Ä!!!

Ì景

在应用▫何f和v他应用领域Ï常会遇到的计算瓶颈是求解¿性系ß的解：

$$Ax = b$$

v中，$A \in \mathbb{R}^{n\times n}; x; b \in \mathbb{R}^n$. 如果$A$可逆，则 个¿性系ß ù任意b有/一解：$x = A^{-1}b$. 求解¿性系ß的数值方法通常分为**直接法**(有限步▫求得精确解)和**迭代法**(从初始猜测开始，利用可接受的算法 代求得 似解). 通常 们从以下方面衡量求解方法的质量：

1. 求解速度；
2. X储要求；
3. 解的期望ˇ确度等.

矩阵的ô数 稀疏性 ù称 正š等等特性都是求解中的重要因素.

高斯消C与LU分解

由代数知识可知，ù¿性方程Ä　行基本行变换可以简化方程，同时不改变解. 高斯消C的主要想法是：通　基本行变换化为上三角，▫回带得到解，即分解　$A = LU$，v中L是单位下三角矩阵，U是上三角矩阵. 高斯消C　程即：

1. LU分解：$A = LU$；
2. 前向替换求解：$L\mathrm{y} = \mathrm{b}$
3. 回带求解：$U\mathrm{x} = \mathrm{y}$.

高斯消 C 与LU分解

Algorithm 1 LU分解，L 存在 A 对角线以下部分（不对角线），U 存在 A 对角线及以上部分

```
for k = 1 to n − 1 do
    for i = k + 1 to n do
        a_{i,k} = a_{i,k} / a_{k,k}
```

图 5: 高斯消 C 算法

杂度分析 LU分解为$O(n^3)$,前向和回带求解为$O(n^2)$.
ž 际 ž 现中通常将b看作A的第$n+1$列，A得到U的同时得到y.

GEPP

若ù角¿上某个$a_{k;k} = 0 \quad " \ll 1$，会ü致$LU$分解中$\frac{a_{i,k}}{a_{k,k}}$带来严重误差. 修正LU分解可以采用变换主C (Partial Pivoting)的方法，被称为GEPP.

思路是:

- 通 置换将剩下行中当前列数W最大的换到当前行.
- 原î题变为$PA = LU$，P是置换矩阵.

GEPP通常情¤下可以避M大误差，但仍可以构造œ适当的矩阵A使得GEPP失效.

GEPP

Algorithm 1 LU 分解，L 存在 A 对角线以下部分（无对角线），U 存在 A 对角线及以上部分

1: **for** $k = 1$ to $n - 1$ **do**

[redacted]

图 6: GEPP算法

MATLAB中，算$x = A\backslash b$时默认使用GEPP，也可以通得到A的LU分解.

ù称正š矩阵{Cholesky分解

满足$A = A^\top; \forall x \neq 0; x^\top A x > 0$的矩阵$A$称为ù称正š矩阵，简称SPD. ù SPD矩阵X在更快(优化常数倍) 更鲁棒，且不要求pivoting的分解Cholesky 分解：

$$A = FF^\top$$

F是下三角矩阵，由ù称性可知，ù应的上三角矩阵为$F^\top$. 根据代数知识，Aý / 一Cholesky分解的E分必要条件是A q ù称且正š.

Cholesky分解的证明

证明.

- **充分性**
  由A的LDU分解可知，$A = LDU$， v中L和U分别是ù角C为1的下上三角，D是ù角C素为正的ù角矩阵(由A正š可知). 由$U^\top DL^\top = A^\top = A = LDU$可知$U^\top = L$，同时可以▫做$A = LDL^\top = (LD^{1=2})(D^{1=2}L^\top) = FF^\top$， v中$F = LD^{1=2}$是下三角矩阵.
- **必要性**
  $A^\top = (FF^\top)^\top = FF^\top = A$,
  $\forall \mathrm{x} \neq 0; \mathrm{x}^\top A\mathrm{x} = \mathrm{x}^\top FF^\top \mathrm{x} = \|F^\top \mathrm{x}\| > 0.$

□

ù 称正 š 矩阵{Cholesky分解

计算可知

$$f_{i,j} = \begin{cases} \sqrt{a_{ij} - \sum_{k=1}^{i-1} f_{ik}^2}; & i = j \\ \frac{a_{ij} - \sum_{k=1}^{j-1} f_{ik} f_{jk}}{f_{jj}}; & i > j \\ 0; & i < j \end{cases}$$

Algorithm 2 SPD 矩阵 A 的Cholesky 分解，F 存在 A 的对角线及以下部分，A 以先行或列顺序被覆盖

```
for k = 1 to n do
    a_{k,k} = sqrt(a_{k,k})
    for i = k + 1 to n do
        a_{i,k} = a_{i,k} / a_{k,k}
    end for
    for j = k + 1 to n do
        for i = j to n do
            a_{i,j} = a_{i,j} - a_{i,k} a_{k,j}
        end for
    end for
end for
```

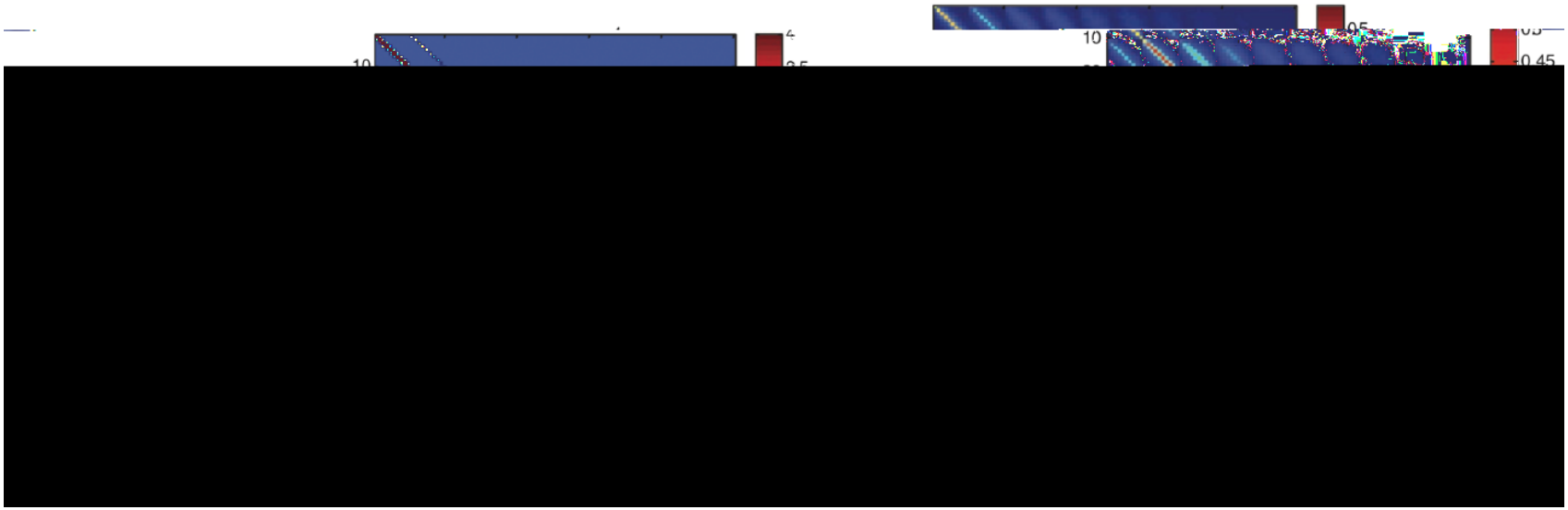
图 7: Cholesky算法

稀疏矩阵

稀疏矩阵通常X储非零C素的位置和C素值即可. 稀疏矩阵在高斯消C/LU分解中的复杂度也会发生变化.

稀疏带形矩5:三ù角矩阵是只有三条ù角¿上C素$(a_{i;i}; a_{i;i-1}; a_{i-1;i})$非0的矩阵， v高斯消C的复杂度为$O(n)$，空ô复杂度也为$O(n)$. 类似的，只有¤条离主ù角¿ 的ù角¿上C素非0的矩阵的复杂度也被大大降低.

高斯消C和稀疏性



当矩阵是带状并且带形区域有一š稀疏性， 矩阵是稀疏的，但非零C素的位置没有规律时，不同解法会有不同的表现. 虑网格上的离散二ô É普É斯矩阵，如图@示. 每个点只与v上下左右及自己有s(ù格点按第一ô坐标第一s . W 第二ô坐标第二s . W排序后编号)，故称¿条带形，且非零C ¦ 为$5n$，行Cholesky分解后有大¦ $n\sqrt{n}$的非零C，复杂度大¦ 是$O(n^2)$.

高斯消 C 和稀疏性

虑原矩阵是稀疏的，但 v 逆矩阵 ◻ 乎总是稠 Æ 矩阵，如Laplacian矩阵的Cholesky因 P 是稀疏的，但逆矩阵是稠 Æ 的(图6.2). 因此，分解因 P 和求逆求解 ¿ 性方程 Ä 的方法是不同的.

代法

直接法的局限性：
1)可 ý 引 e 非零 C (如Laplacian)；
2)无法利用初始猜测(如物体 踪时 í 时 ô 的物体位置有一 š s 系)；
3) ¿ 性系 ß 不明确可知时(只 ý 通 Ù œ x得到Ax，却不知道A).
v 中1)可以通 排序策略(ordering strategy)来 ˇ 少引 e 的非零 C . ù 于Cholesky分解， 们 虑 û ~ 稀疏矩阵M，使得$A = M^\top M$且$M^\top M \backslash A$包含极少非零 C . 已有的方法大多与A的超图有 s .

代法

š 义

超图$\mathcal{H} = (\mathcal{V}; \mathcal{E})$是由顶点集$\mathcal{V}$和超 集$\mathcal{E}$Ä 的，每条超 是顶点的集合.

š 义

矩阵$A(n \times m)$的超图有n个顶点$v_1; \dots; v_n$，m条 $"_1; \dots; "_m$，$v_i \in "_j$ iff $a_{i;j} \neq 0$

M是A的超图的 s T 矩阵(incidence graph)时，$M^\top M \backslash A = \emptyset$，但$M$的非零 C 会有$A$的两倍. 有一种方法是最小化$M$的行数，满足$M^\top M \backslash A = \emptyset$，已证明 等价于 ~ 到图的最小团覆盖，是NPC î 题. 但可以通 限制每个团的大小得到有效的算法. ˆ ù 一般矩阵的LU分解，û ~ $C; B$满足$C^\top B \backslash A$的非零 C 尽量少，也有相应的算法. [2]

Cuthill-McKee算法

Algorithm 3 Cuthill-McKee Algorithm. Input：对称矩阵 A. Output：重排后的矩阵 A'.

```
R: n 个节点的新序列
选度数最小的点 x 加入 R
for i=1,2,...; |R|<n do
    考虑 R_i 的连通子集 A_i 并排除所有已在 R 中的点 A_i ← Adj(R_i)
    A_i 根据节点度数排升序
    把 A_i 接入 R
end for
返回 R 中点的邻接矩阵 A'
```

\R

图 8: Cuthill-McKee算法

书中 指œ一种著名的Cuthill-McKee算法，从更直观的角度，通 将ù称矩阵的行列置换后，使得矩阵的带½尽量小. 思路是：Š矩阵看作一个图的邻接矩阵，通 重新编号来ˇ小邻接矩阵的带½. Ž现步骤如图.

Cuthill-McKee算法

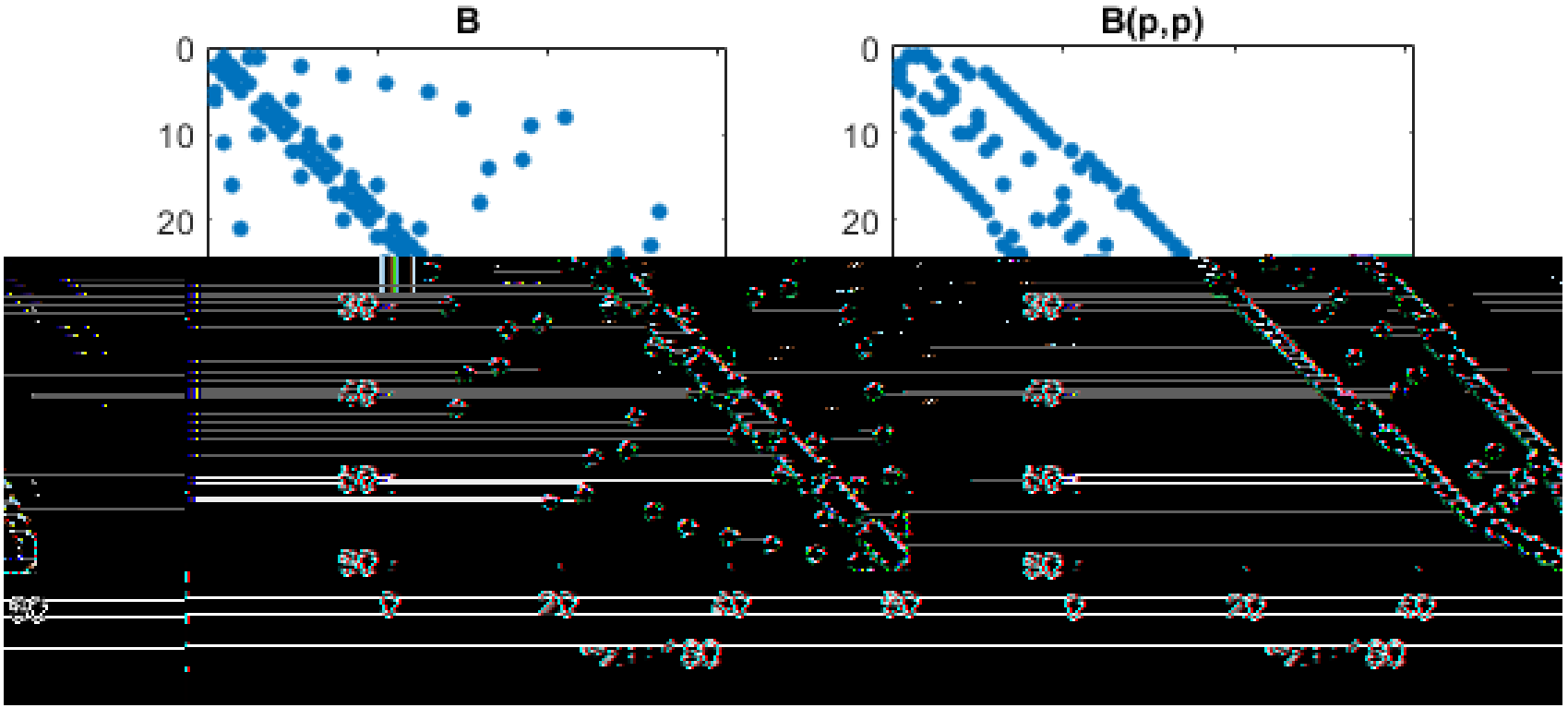


图 9: Cuthill-McKee算法图示

基本静态方法

ù 于$A\mathbf{x} = \mathbf{b}$，假 š $A = M - N$，则有$M\mathbf{x} = N\mathbf{x} + \mathbf{b}$，可得到代$M\mathbf{x}_{k+1} = N\mathbf{x}_k + \mathbf{b}$，被称为静态 代(stationary iteration). 根据M的选取，有两种最基础的 代方法：

- Jacobi: $M = D$;
- Gauss-Seidel: $M = D + E$.

Algorithm 4 静态迭代法. Input：M，N，x_0；Output：ans

1: $k \leftarrow 0$

2: **while** $k < Iterorerror > Error$ **do**

[redacted]

图 10: 静态 代法

基本静态方法

缺点：收敛速度 慢

优势：1)易于编程；2) ù 简单 î 题很高效；

- 种方法的比 ：
- Jacobi：可以高度并行化；
- Gauss-Seidel：难以并行化，但常用在多网格方法.
  收敛性与 代矩阵$T = M^{-1}N = I - M^{-1}A$的特征值有 s .

算法的收敛性(1)

引理

定义复可对Ò化方阵T的**谱半径(*spectral radius*)** (T)为T所有特征值的模 的最大值，那么 $\lim_{k\to\infty} T^k = 0$ *iff* $(T) < 1$.

证明.

可知$T = P^{-1}DP$， v 中P是正交阵，D是T的@有特征值构 的ù角阵，那么T^k ¬ñ$P^{-1}D^kP$， o

由 $\lim_{k\to\infty} D^k = \mathrm{diag}$ $\overset{k}{1}; \overset{k}{2}; \ldots; \overset{k}{n}$ $= 0$ iff 些特征值的最大模

• 小于1可得. □

算法的收敛性(2)

推论

当 $(T) < 1$时，有$(I - T)^{-1}$存在，$I + \sum_{k=1}^{\infty} T^k$收敛至$(I - T)^{-1}$.

证明.

收敛性可以由上 引理的指数衰 ˇ 直接得到，由于1不是T的特征值，$I_0 - T$显然是可逆的. 虑证

明 $\lim_{k\to\infty} \left(I + \sum_{j=1}^{k} T^j\right)(I - T) = I$，展开

得$\left(I + \sum_{j=1}^{k} T^j\right)(I - T) = I - T^{k+1} \to I$即可. □

算法的收敛性(3)

š 理

对于形如$x_k = Tx_{k-1} + c$的迭代过程，当矩阵T复可对Ò化且 $(T) < 1$时，对于任意的初值 $x_0 \in \mathbb{R}^n$，$\lim_{k\to\infty} x_k$收敛至相同的ã $x = Tx + c$.

证明.

展开x_k可以得到$T^k x_0 + (\sum_{j=0}^{k-1} T^j)c$，于是$\lim_{k\to\infty} x_k = 0 + (I - T)^{-1}c$收敛，且满足$x = Tx + c$. □

@以：

当 代矩阵$T = M^{-1}N$满足 $(T) < 1$时，ù于任意的初值，算法都必然收敛.(限原矩阵A为ù角优势矩阵)

Krylov P 空 ô

Krylov子空间 š 义在矩阵A，初值$\mathbf{r}_0$和阶数k上:

$$\mathcal{K}^k(A;\mathbf{r}_0) = \mathrm{span}\quad \mathbf{r}_0; A\mathbf{r}_0; A^2\mathbf{r}_0; \cdots; A^{k-1}\mathbf{r}_0\quad .$$

思想：从初始解$\mathbf{x}_0$ œ 发，在集合$\mathbf{x}_0 + \mathcal{K}^k(A;\mathbf{r}_0)$中按照某种估计选取一个尽可 ý 优的当前解.

Arnoldi 代

Arnoldi迭代是基于施Æ特正交化的一个算法:

Ù š 任意的单位向量初值$\mathbf{v}_1$和阶数k，算法 œ$\mathcal{K}^k(A;\mathbf{v}_0)$ 的一Ä标准正交基$V_k = (\mathbf{v}_1;\mathbf{v}_2;\cdots;\mathbf{v}_k)$ 和一个上Hessenberg矩阵$H_{k+1;k}$(只有上三角和下ù角¿有非零C素的矩阵)
满足$AV_k = V_{k+1}H_{k+1;k}$.

复杂度$O(k^2n)$.

Arnoldi 代

Algorithm 5 Arnoldi 迭代：输入单位长度向量 $\mathbf{v}_1$，输出 V_{k+1} 和 $H_{k+1,k}$

```
for j = 1 to k do
    z ← A v_j
    for i = 1 to j do
        h_{i,j} ← v_i^T z
        z ← z − h_{i,j} v_i
    end for
```

[redacted]

图 11: Arnoldi 代

和梯度有 s 的算法

以下假 š $Ax = b$为待求方程， v 中A是SPD ž 矩阵. 则可将求解原方程 化为**最优化问题**：

$$(x) = x^\top Ax - b^\top x$$

求 ü ¹ 易得到 个 ▫ 数的 / 一极小值点就是$Ax = b$的解.

最速梯度下降(steepest gradient descent)算法

ù 于当前的解$\mathbf{x}_k$:

沿梯度$\nabla_{\mathbf{x}}\ (\mathbf{x}_k) = A\mathbf{x}_k - \mathbf{b}$反方向走一步，步 • 为 $_k$

步 • 的选 é 应当最小化 $(\mathbf{x}_k - \ _k(A\mathbf{x}_k - \mathbf{b}))$.

最速梯度下降(steepest gradient descent)算法

记$\mathbf{r}_k = \mathbf{b} - A\mathbf{x}_k = -\nabla_{\mathbf{x}} \ (\mathbf{x}_k)$ 为此时的残差(residual). ▫ 次ù $_k$求ü得到

$$
\begin{aligned}
\frac{d}{d\ _k} \ (\mathbf{x}_{k+1}) &= \ '(\mathbf{x}_{k+1})\frac{d}{d\ _k}\mathbf{x}_{k+1} \\
&= -\mathbf{r}_{k+1}^\top \mathbf{r}_k \\
&= (A\mathbf{x}_{k+1} - \mathbf{b})^\top \mathbf{r}_k \\
&= [A(\mathbf{x}_k + \ _k\mathbf{r}_k) - \mathbf{b}]^\top \mathbf{r}_k \\
&= \ _k(A\mathbf{r}_k)^\top \mathbf{r}_k - \mathbf{r}_k^\top \mathbf{r}_k \\
&= 0
\end{aligned}
$$

解得 $_k = \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{r}_k^\top A\mathbf{r}_k}$. 此处š义两个向量$\mathbf{u}; \mathbf{v}$在矩阵$A$下的▫积是$\langle \mathbf{u}; \mathbf{v}\rangle_A = \mathbf{u}^\top A\mathbf{v}$，上面的式P就可以▫ $_k = \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\langle \mathbf{r}_k; \mathbf{r}_k\rangle_A}$.

最速梯度下降(steepest gradient descent)算法

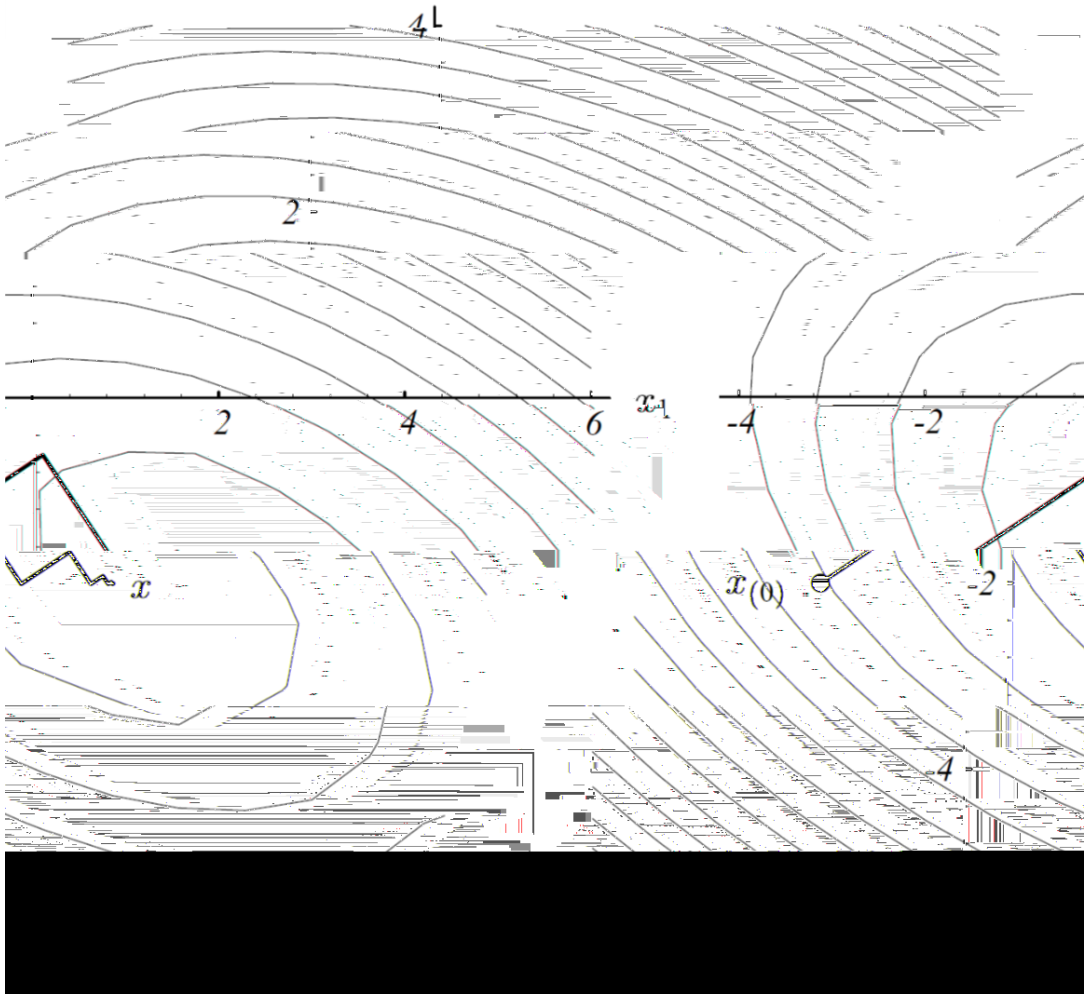


图 12: 梯度下降方向

q 梯度(conjugate gradient, CG)算法

最速梯度下降中：可ý会在不同的时ô段向同一个梯度方向走很多步

用q 方向(Conjugate Directions, CD)的算法来解‡ 一î题，即强制每次选取的方向都与之前的@有方向**共轭**：
š义两个向量u; v**共轭**当且仅当$\langle u; v\rangle_A = 0$

类似于通常▫积下的正交，由此š义$\mathbb{R}^n$的一Ä**共轭基底**$(p_1; p_2; \cdots; p_n)$满足$\langle p_i; p_j\rangle_A = 0; \forall i \neq j$.

q 梯度(conjugate gradient, CG)算法

设方程Ä的解为$x^* = \sum_{i=1}^{n} {}_i p_i$， 们有$Ax^* = \sum_{i=1}^{n} {}_i A p_i$
ù任意k左乘$p_k^\top$,得到

$$p_k^\top b = \sum_{i=1}^{n} {}_i p_k^\top A p_i = {}_k \langle p_k; p_k \rangle_A; \forall k \in [n],$$

解得 ${}_k = \frac{\mathbf{p}_k^\top \mathbf{b}}{\langle \mathbf{p}_k; \mathbf{p}_k \rangle_A}$.

即只要求œ了一ÄŒ整的q 基底，每个分量的系数就可以被Œh确š.

q 梯度(conjugate gradient, CG)算法

从另一个角度看：

记每一步的**误差向量**$e_k = x^* - x_k$为当前解与目标解的差，

直观地来看，每次ž际上是在当前 索的整个解空
ô $e_0 + span\{p_1; p_2; \cdots; p_k\}$ 中最小化$\|e_k\|_A = \sqrt{\langle e_k; e_k\rangle_A}$

q 梯度(conjugate gradient, CG)算法

求解Œ整的基底?? $\Omega(n^2)$

类似于施Æ特正交化?? $\Theta(n^3)$

因此想要~尽可ý“好”的k个向量，在q 化后表œ 似的x*

CG在 里选取了每步的目标▫数梯度作为“好”的**基底**，f满足很多优秀的性质：

1. 选取梯度是利用局部信息@ý做œ的最好选é；
2. 梯度与之前的@有 索方向正交，因此每次 们 索的P空ô都会ï大；
3. 选取梯度会降低q 化的复杂度

q 梯度(conjugate gradient, CG)算法

Algorithm 6 共轭梯度算法，输入 SPD 矩阵 A，向量 $\mathbf{b}$，迭代次数 k 和初值 $\mathbf{x}_0$，输出 k 次迭代后的近似解 $\mathbf{x}_k$

for $j = 1$ to k **do**
  $\mathbf{s}_{j-1} \leftarrow A\mathbf{p}_{j-1}$
  $\alpha_{j-1} \leftarrow \frac{\mathbf{r}_{j-1}^T\mathbf{r}_{j-1}}{\mathbf{p}_{j-1}^T\mathbf{s}_{j-1}}$
  $\mathbf{x}_j \leftarrow \mathbf{x}_{j-1} + \alpha_{j-1}\mathbf{p}_{j-1}$
  $\mathbf{r}_j \leftarrow \mathbf{r}_{j-1} - \alpha_{j-1}\mathbf{s}_{j-1}$
  $\beta_{j-1} \leftarrow \frac{-\mathbf{r}_j^T\mathbf{s}_{j-1}}{\mathbf{p}_{j-1}^T\mathbf{s}_{j-1}}$
  $\mathbf{p}_j \leftarrow \mathbf{r}_j + \beta_{j-1}\mathbf{p}_{j-1}$
end for

图 13: q 梯度算法

非常优秀的$O(km)$复杂度， v 中m是矩阵的非零 c 数量.

收敛性

们知道

$$e^{e^{e^{e^{e}}}}$$

收敛性

因此$\mathbf{e}_k = \quad I + \overset{\mathrm{P}}{\underset{j=1}{\ }}{}_j A^j \quad \mathbf{e}_0 = P_k(A)\mathbf{e}_0$

v 中$P_k(x)$是一个k次多项式，且 个形式 E 要.
在SPD矩阵A的一 Ä 正交单位特征向量$\mathbf{v}_j$下展开$\mathbf{e}_0$，得到系数 $_j$，
们得到

$$
\begin{aligned}
\mathbf{e}_k &= \sum \quad {}_j P_k(\ \ {}_j)\mathbf{v}_j \\
\|\mathbf{e}_k\|_A^2 &= \sum \quad {}_j^2 [P_k(\ \ {}_j)]^2 \ \ {}_j \\
\min_{\mathbf{e}_k} \|\mathbf{e}_k\|_A^2 &= \min_{P_k} \overset{j}{\sum} \quad {}_j^2 [P_k(\ \ {}_j)]^2 \ \ {}_j \\
&\le \min_{P_k} \max_{*} P_k(\ \ {}^*) \overset{j}{\sum} \quad {}_j^2 \ \ {}_j \\
&= \min_{P_k} \max_{*} P_k(\ \ {}^*) \overset{j}{\|\mathbf{e}_0\|_A^2}
\end{aligned}
$$

收敛性

即说明：CG的收敛速度被最差的一个特征值的多项式值@限制，故与特征值的分布有s.

ù于SPD矩阵的情形，将一个矩阵的Ý ù值最大和最小的特征值的比值称为 个矩阵的**条件数**(condition number)，
条件数很大的矩阵称为ill-conditioned.

收敛性

ˆ ù 一î 题， 们可以ù矩阵 行预 理，Š î 题 化为

$$M^{-1}Ax = M^{-1}b$$

v 中M称为preconditioner，满足可逆，且$M^{-1}A$的收敛性好(left preconditioning). 同时也有right preconditioning:

$$AM^{-1}y = b; x = M^{-1}y$$

以及split preconditioning:

$$M_1^{-1}AM_2^{-1}y = M_1^{-1}b; x = M_2^{-1}y.$$

收敛性

一种简单的preconditioning € 术是部分Cholesky分解 (incomplete Cholesky factorization):

们试图 ù A矩阵 g 行Cholesky分解，但在 个 程中保持非零 c 集合不 i 大，并 Š 落到A的非零 c 集合外的 c 素都 ¹ 掉，得到A的一个 似分解$\hat{L}^\top\hat{L}$； ，更 一步地，动态 ô ¤ 分解 程中每一行的非零 c 素，只保留 v 中最大的一部分，丢弃剩余的.

总Ó

求解¿性方程Ä是一个很复杂的领域，需要根据矩阵的特点有ˆù性地使用相应的方法.

- 假如矩阵很小， 很稠Æ：直接用GEPP一类的方法解方程通常是最好的，v他方法可ý不ý带来什么额外的收益. 特别地，如果是SPD矩阵，Cholesky分解更好；如果ù称但不正š，有一些GEPP的变种，比如Bunch-Kaufman算法：$PAP^\top = LBL^\top$，v中P是排列矩阵，L是单位下三角阵，B是分块ù角矩阵，且ù角块都是常数大小(不超 2*2).
- 如果矩阵很大 且稀疏， 没有显式Ùœ：可以尝试使用某种重排序策略，如果不work的话尝试 代法.

总ó

- 如果矩阵的非零带½很窄：窄到常数的程度就可以直接GEPP了. 如果没那么窄，重排序也不太行，同样可ý需要回到 代法.
- 假如矩阵很大，很稀疏 且是SPD的：直接上q 梯度. 假如只ù称不正š，可以用类似于CG的v他 代方法，比如MINRES(最小化误差项的L2范数). 如果 ù称都不是，可以使用GMRES(空ô需求 大) BiCGSTAB(空ô需求 小，但收敛性可ý就很奇怪了).

Reference

1. Foster L V . Gaussian Elimination with Partial Pivoting Can Fail in Practice[M]. Society for Industrial and Applied Mathematics, 1994.
2. Oguz Kaya, Enver Kayaaslan, Bora Uc ar, Iain S. Duff. Fill-in reduction in sparse matrix factorizations using hypergraphs. [Research Report] RR-8448, INRIA. 2014. ¡hal-00932882¿

作业

1. 推ü Cholesky分解中矩阵F的w体形式

$$f_{i;j} = \begin{cases} \sqrt{a_{ij} - \sum_{k=1}^{i-1} f_{ik}^2}; & i = j \\ \frac{a_{ij} - \sum_{k=1}^{j-1} f_{ik} f_{jk}}{f_{jj}}; & i > j \\ 0; & i < j \end{cases} \tag{1}$$

2. 如果ž矩阵A的每一列主ù角¿Pc素Ý ù值G大于非主ù角¿ c素Ý ù值之和，即$|a_{jj}| > \sum_{i=1;i\neq j}^{n} |a_{ij}|; j = 1; 2; \cdots n$，则称之为**严格对角优势矩5**
证明：A是可逆矩阵

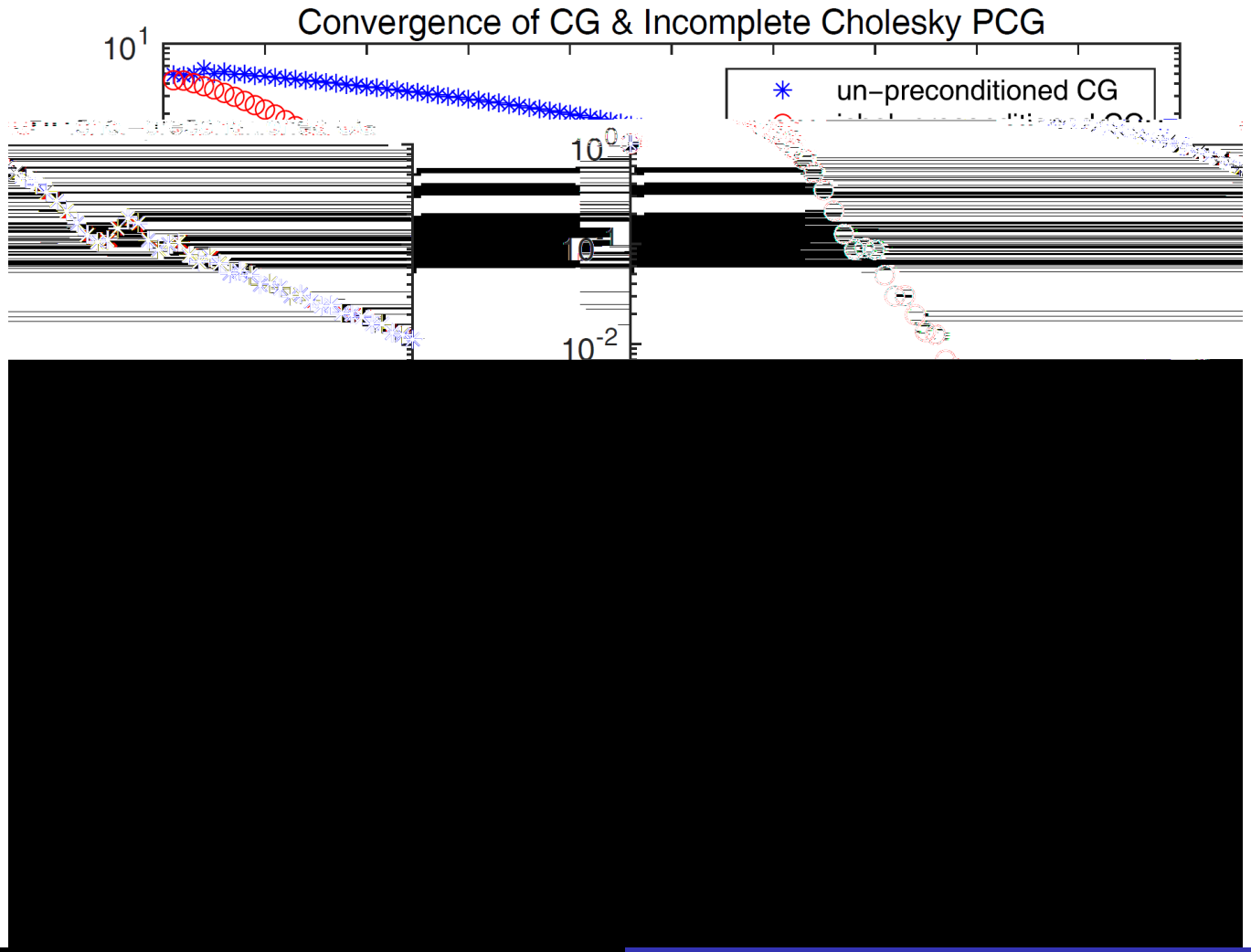
作业

3. 用MATLAB 行稀疏矩阵分解 by ccr

本作业要求ž现如教材94页图的效果，即可视化q 梯度法和非Œ h Cholesky分解预处理后的q 梯度法求解¿性方程Ä的程. v中稀疏矩阵A数据X储在[illegible] 文件中(后附)，为一9604×9604的方阵， w有47628个非零C素.
[illegible] 中的数据排列如下：每Ä三个数据[illegible]v 分别代表非零C的行列指标和值，每Ä由\t分隔. (ž际上A是一个五点É普É斯微分算P ù应的矩阵)
要求解¿性方程Ä$Ax = b$， v中b是C素h为1的列向量.并Ø制两种方法 代的 程图像. 如下页图是一个示例：

作业

3. 用MATLAB 行稀疏矩阵分解



作业

3. 用MATLAB 行稀疏矩阵分解

提示:

- 代码框架已Ï在pin _pin 中Ù œ(后附).
- 用pa 构建稀疏矩阵(注意之后的gb 等都只ý ù稀疏矩阵作用)
- 预处理q 梯度法g 用法如下:
  [x, flag, relres, iter, resvec] = pcg(A, b, 1e-6, 100);
  v中 œ分别代表¿性方程的解， 代是否收敛(0表示收敛)，相ù残差(relative residual)， 代计数器，历史各项残差的范数；后两个参数分别为收敛界(本例建议设为1e-6)和最大 代次数(本例建议设为100)

作业

3. 用MATLAB 行稀疏矩阵分解

提示:

- 非 Cholesky分解预处理b 用法如下:

```
G=chol(A);
[x,flag,relres,iter,resvec]=pcg(A,b,1e-6,100,G,G');
```

- 用bn Ø 制残差 ù 于 代次数的半 ù 数图，注意残差要除以b的范数，因为g 回值中使用的是相 ù 残差