

“基于共轭梯度法的维纳滤波矩阵求解

邱梁城¹, 严铮¹

摘要

本文对基于共轭梯度方法的阵列信号维纳滤波矩阵形成算法进行复现, 复现的主要算法为共轭 (CG) 这个算法来自于开始的阵列信号维纳滤波矩阵的求解过程, 我们将它与书本中给出的矩阵分解求解维纳滤波矩阵方法进行对比。给出两者的输入信噪比与输出信噪比的比值 G_W 以及最小均方误差 (MMSE) 作为性能指标的参考。通过仿真与讨论得出 CG 求解算法其优点在于简单直观, 并且在保证了原来的准确性的情况下大大减小了所耗时间。我们小组分工明确, 认真完成了复现工作, 具体分工在引言部分给出, 此外本文所提主要文献, 代码以及参考资料在附录中给出。

关键词

维纳滤波器 阵列信号处理 Krylov 子空间 共轭梯度法

¹ 西北工业大学

1. 引言

阵列信号处理技术在许多领域有所运用, 维纳滤波是最基础的阵列信号处理滤波器, 其最基本的方法是通过矩阵分解获得信号处理方程的解。而本文以麦克风阵列信号模型为例通过更加有效的解方程的 CG 算法 (共轭梯度法) 直接获得其维纳解。在保证其准确性的情况下算法耗时有大幅度减少。本文由以下部分构成: 第 2 节介绍信号模型和维纳滤波技术。第 3 节介绍经典维纳滤波矩阵求解方式。第 4 节介绍本文所要复现的共轭梯度的方法求解维纳滤波矩阵。第 5 章给出仿真实验结果, 最后给出一些结论。第 6 章为本次大作业的心得体会。

我们小组分工明确, 认真完成了复现工作, 具体的分工如下::

- 查阅讨论文献 (一起)
- 代码部分: 邱梁城
- 传统方法复现: 邱梁城, 严铮

• 报告

严铮: 摘要, 引言 (1), 信号模型 (2) 和经典求解维纳滤波器 (3.1), 算法仿真 (5) 和参考文献

邱梁城: 衡量指标 (3.2), 基于共轭梯度法的快速算法 (4), 仿真结果 (5), 参考文献和最终整理

2.1 信号模型

一个具有 N 个通道的麦克风阵列系统, 第 n 个通道在 t 时刻接收到的含噪声语音信号模型

$$y_n(t) = \mathbf{g}_n * \mathbf{s}(n) + \mathbf{v}_n(t) \quad (1)$$

$$= \mathbf{x}_n(t) + \mathbf{v}_n(t); \quad n = 1, 2, \dots, N \quad (2)$$

其中 g_n 表示从信号源到第 n 个麦克风系统传输函数, $*$ 表示卷积, $x_n(t)$ 和 $v_n(t)$ 分别表示麦克风接收到的信号成分及加性噪声, 且通常假设噪声与语言不相关。

语言增强的目的是由这 N 个通道收到的带噪声语音信号, 提取出目标的语音信号, 一般把 $x_1(t)$ 作为目标信号。多通道语言增强算法 [1] 主要基于线性滤波器, 即计算线性滤波器矩阵 H 使得通过滤波器后的信号估计为 $z = HY$ 其中 Y 为收到的含噪声语音信号。将其用向量形式表示如下

$$y_n(k) = x_n(k) + v_n(k), \quad n = 1, 2, \dots, N \quad (3)$$

$$y_n(k) = [y_n(k) y_n(k-1) \dots y_n(k-L+1)]^T \quad (4)$$

$$x_n(k) = [x_n(k) x_n(k-1) \dots x_n(k-L+1)]^T \quad (5)$$

$$v_n(k) = [v_n(k) v_n(k-1) \dots v_n(k-L+1)]^T \quad (6)$$

$x_n(k)$ 和 $v_n(k)$ 分别表示麦克风接收到的语言信号和加性噪声, L 为滤波阶数, 令 H 为线性滤波器则第一道语言估计为

$$z(k) = HY(k) \quad (7)$$

$$z(k) = HY(k) = H[X(k) + V(k)] \quad (8)$$

2. 信号模型与维纳滤波

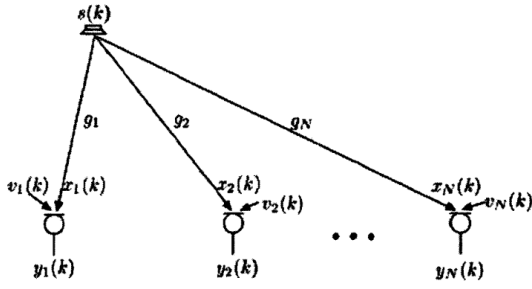


图 1. 信号模型

其中:

$$Y(k) = [y_1^T \ y_2^T \ \dots \ y_n^T]^T \quad (9)$$

$$X(k) = [x_1^T \ x_2^T \ \dots \ x_n^T]^T \quad (10)$$

$$V(k) = [v_1^T \ v_2^T \ \dots \ v_n^T]^T \quad (11)$$

$$H = [h_1^T \ h_2^T \ \dots \ h_n^T] \quad (12)$$

$h_n (n=1,2,\dots,N)$ 为 $L \times L$ 阶滤波矩阵, 我们定义估计的信号误差为:

$$e(k) = z(k) - x_1(k) \quad (13)$$

$$= (H - Q)X(k) + HV(k) \quad (14)$$

$$= e_x(k) + e_v(k) \quad (15)$$

其中

$$Q = [I_{L \times L} \ 0_{L \times L} \ \dots \ 0_{L \times L}]$$

是一个 $N \times NL$ 且 $I_{L \times L}$ 是 $L \times L$ 的单位矩阵

2.2 维纳滤波器

根据上节信号模型, 则估计的信号最小二乘误差 (MSE) 的表达式

$$J(W) = \text{tr} E[e(k)e(k)^T] \quad (16)$$

$$= E[x_1(k)x_1(k)^T] + \text{tr}[HR_{yy}H^T] - 2\text{tr}[HR_{yx_1}] \quad (17)$$

其中

$$R_{yy} = E[Y(k)Y^T(k)] \quad (18)$$

$$R_{yx_1} = E[Y(k)x_1^T(k)] \quad (19)$$

将上式 $J(H)$ 对 H 求偏导, 使估计信号误差最小

$$\frac{\partial J(H)}{\partial H} = 0 \quad (20)$$

得到维纳滤波矩阵

$$H_{wiener}^T = R_{yy}^{-1} R_{yx_1} \quad (21)$$

由于 $x_1(k)$ 无法观测但可知 $R_{yx_1} = R_{yy} - R_{vv}$ [5] 其中

$$R_{vv} = E[V(k)V^T(k)] \quad (22)$$

由此 R_{yy} 和 R_{vv} 为噪声自相矩阵都可以得到将其带入公式即可求得维纳滤波矩阵。

3. 经典求解维纳滤波矩阵

这一节中我们将介绍所复现的论文中所运用的求解维纳滤波矩阵算法, 同时给出评价滤波器性能的指标, 包括 ISNR, OSNR, 在陈景东教授的书中有很好的介绍。

3.1 求解维纳滤波矩阵

由于 R_{vv} 是一个满秩矩阵, 对称矩阵 R_{vv} 与 R_{xx} 可联合对角化

$$T^T R_{xx} T = \Lambda \quad (23)$$

$$T^T R_{vv} T = I_{NL} \quad (24)$$

$\Lambda = \text{diag}[\lambda_1 \ \lambda_2 \ \dots \ \lambda_{NL}]$ 由矩阵 $R_{vv}^{-1} R_{xx}$ 的特征值组成且 $\lambda_1 \geq \lambda_2 \geq \dots \lambda_{NL} \geq 0$, T 为特征向量 $t_1 \ t_2 \ \dots \ t_{NL}$ 对应组成的矩阵其是一个满秩矩阵, 则

$$t_i^T x(t) = 0, i = n+1 \ n+2 \ \dots \ NL \quad (25)$$

$$R_{vv}^{-1} - 1 = \sum_{i=1}^{NL} t_i t_i^T \quad (26)$$

$$R_{vv}^{-1} R_{xx} T = T \Lambda \quad (27)$$

$$T^T R_{yy} T = I_{NL} + \Lambda \quad (28)$$

我们能够将滤波矩阵 H 化为 $H = AT^T$

根据上一节定义估计值与真实值误差为 $e(k) = z(k) - x_1(k) = AT^T y(t) - x_1(t)$, 我们推断 MSE 准则变为:

$$J(A) = \text{tr}[R_{x_1} - 2AT^T R_{xx} T + A \Lambda A^T] \quad (29)$$

对 A 求偏导使 MSE 最小化得到 $A_w = I_i T^{-T} \Lambda (\Lambda + I_{NL})^{-1}$ 而 $A_w T^T = H_w$ 其中 H_w 即为维纳滤波矩阵, 至于 T 的求解, 我们可以联合利用 Cholesky[9] 分解与 Schur[10] 分解, 求得方程组中的 T 与 Λ , 进而求取滤波矩阵 H_w 最终得到

$$H_w = I_i R_{xx} \sum_{i=1}^{NL} \frac{t_i t_i^T}{1 + \lambda_i} = I_i R_{vv} \sum_{i=1}^{NL} \frac{\lambda_i}{1 + \lambda_i} t_i t_i^T \quad (30)$$

3.2 衡量指标

在本小节中，我们定义了一些基本度量 [13]，这些度量很好地描述阵列传感器和线性滤波矩阵的优劣。我们假设传感器 1 是参考传感器，那么我们定义：

输入的 SNR(信噪比) 为：

$$iSNR = \frac{tr(R_{x1})}{tr(R_{v1})} \quad (31)$$

其中 $\mathbf{R}_{x1} = E[\mathbf{x}_1(t)\mathbf{x}_1^T(t)]$ 和 $\mathbf{R}_{v1} = E[\mathbf{v}_1(t)\mathbf{v}_1^T(t)]$ 是 $x_1(t)$ 和 $y_1(t)$ 的自相关矩阵，我们知道由于 $x_1(t)$ 不能直接通过传感器获得，我们将直接获得输入信噪比依据 $\mathbf{y}_1(t)$ ，其中 $\mathbf{R}_{y1} = \mathbf{R}_{x1} + \mathbf{R}_{v1}$ 我们根据公式8能够得知：

$$\begin{aligned} R_z &= E[z(t)z^T(t)] \\ &= HR_{yy}H^T \\ &= HR_{xx}H^T + HR_{vv}H^T \\ &= R_{xfd} + R_{vrd} \end{aligned} \quad (32)$$

此时输出信噪比便可以很容易的获得：

$$\begin{aligned} oSNR(H) &= \frac{tr(R_{xfd})}{tr(R_{vfn})} \\ &= \frac{tr(HR_{xx}H^T)}{tr(HR_{vv}H^T)} \end{aligned} \quad (33)$$

信噪比增益为：

$$G_W = \frac{oSNR}{iSNR} \quad (34)$$

到此已经可以很好的描述滤波矩阵以及线性阵列的优劣问题，但是我們希望能够获得维纳滤波矩阵对于噪声以及信号单独的作用效果，我们定义降噪系数如下：

$$\xi_n(H) = \frac{tr(R_{v1})}{tr(R_{vrd})} \quad (35)$$

由于期望的信号可能被滤波矩阵失真，我们定义所需信号衰减因子为：

$$\xi_n(H) = \frac{tr(R_{x1})}{tr(R_{xfd})} \quad (36)$$

4. 基于共轭梯度法的快速算法

求解大型线性矩阵已经成为了当前工业所要解决的巨大问题，其中已经有很多成熟的算法，包括由 kryov 子空间引出的一系列算法，速度快、精度高使得他们被多次运用在大型矩阵的运算，其中共轭梯度法 [8] 具有内存小、收敛速度极快，可通过误差大小控制解的精确度，这一节中，我们将探讨其在维纳滤波矩阵求解过程中的运用。

4.1 维纳滤波线性方程分析

对于求解滤波器方程组： $H = R_{yy}^{-1}R_{yx1}$ ，考虑到我们无法直接观测到 $X1$ ，所以我们需要对上式进行处理。我们可以很容易地根据公式22得出：

$$R_{yx1} = (R_{yy} - R_{vv})Q^T \quad (37)$$

其中 R_{vv} 为 $NL * NL$ 的矩阵，它可以在静音段利用 VDA 技术 [5] 估计得到。于是 H 的求解只取决于 R_{yy} 和 R_{vv} ，可以表示为：

$$H = Q(I_{NL} - R_{yy}^{-1}R_{vv})^T \quad (38)$$

我们考虑使用 CG 算法进行优化。根据公式23、24、28我们可以很清楚地知道 R_{yy} 是一个对称正定矩阵(特征值均大于 0)。我们构建如下方程组：

$$R_{yy}H = (R_{yy} - R_{vv})Q^T \quad (39)$$

其中 H 是一个 $NL * L$ 的矩阵，设 $M = (R_{yy} - R_{vv})Q^T$ ，是一个 $NL * L$ 的矩阵。那么，此方程组便可以按照如下方式求解：

$$R_{yy} * [h_1 \ h_2 \ h_3 \dots h_L] = [m_1 \ m_2 \ m_3 \dots m_L] \quad (40)$$

4.2 CG 算法求解

接下来我们将要对式进行求解，考虑到 CG 算法是对线性方程组求解的算法我们将分别对 $R_{yy} * h_i = m_i$ 进行求解。构建一组彼此关于 R_{yy} 共轭的正交基 $P_1, P_2, P_3, \dots, P_n$ ，则目标向量可以表示成这组基的线性组合：

$$H_i = \sum_{j=1}^n a_j P_j \quad (41)$$

即：

$$R_{yy} * h_i = m_i = \sum_{j=1}^n a_j R_{yy} P_j \quad (42)$$

$$P_k^T m_i = a_k P_k^T R_{yy} P_k \quad (43)$$

其中：

$$a_k = \frac{(P_k, m_i)}{(P_k, P_k) R_{yy}} = \frac{P_k^T m_i}{P_k^T R_{yy} P_k} \quad (44)$$

问题变为求解 P ，便可以求出 a_k 和 h_i 。

我们假设有一组线性无关的向量组 y_0, y_1, \dots, y_k ，尝试构造一组 A 共轭的向量组 P_0, P_1, \dots, P_k ，使其满足：

$$\text{span}\{y_0, y_1, \dots, y_k\} = \text{span}\{P_0, P_1, \dots, P_k\} \quad (45)$$

我们这里应用施密特正交化思想，使用迭代法实现 P 的选取。算法步骤如下：其中 (x_k) 代表第 k 步的迭代向量

步骤一：给定初值 $x_0 = 0$ ，计算出初始残差：

$$r_0 = m_i - R_{yy}x_0 \quad (46)$$

步骤二：取 $P_0 = r_0$ ，计算得到 x_1

$$x_1 = x_0 + a_0 P_0, \quad a_0 = \frac{(r_0, P_0)}{(R_{yy}P_0, P_0)} \quad (47)$$

步骤三：由 x_1 得到新的残差 $r_1 = m_i - R_{yy}x_1$ ，并以 r_1 为基础，对 P_0 做 A 的投影得到新的前进方向 P_1

$$P_1 = r_1 - \beta_1 P_0, \quad (48)$$

$$\beta_1 = \frac{(R_{yy}P_0, r_1)}{(R_{yy}P_0, P_0)} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \quad (49)$$

步骤四：重复步骤三得到新的迭代向量与残差

$$x_{k+1} = x_k + a_k P_k, \quad a_k = \frac{(r_k, P_k)}{(R_{yy}P_k, P_k)} = \frac{r_k^T r_k}{p_k^T R_{yy} P_k} \quad (50)$$

$$r_{k+1} \rightarrow P_{k+1} \rightarrow x_{k+2}$$

算法伪代码如下：

算法 1 共轭梯度法

输入： R_{yy} ：一个 $NL \times NL$ 的对称正定矩阵； m_i ：一个 $NL \times 1$ 的向量； x_0 ：一个 $NL \times 1$ 的向量，用于初始化

输出： x ：一个 $NL \times 1$ 的向量，表示方程 $R_{yy}h_i = m_i$ 的解

1: 初始化：

2: $r_0 = m_i - R_{yy}x_0$

3: $p_0 = r_0$

4: $k = 0$

5: **while** 未收敛 **do**

6: $\alpha_k = \frac{r_k^T r_k}{p_k^T R_{yy} p_k}$

7: $x_{k+1} = x_k + \alpha_k p_k$

8: $r_{k+1} = r_k - \alpha_k R_{yy} p_k$

9: $\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$

10: $p_{k+1} = r_{k+1} + \beta_k p_k$

11: $k = k + 1$

12: **end while**

13: 输出 x

由此得到的满足误差条件的 x_k 便为我们所需的 h_i ，能够证明由此算法产生的向量序列 r_l, P_l 满足：

$$l r_l \perp \text{span}\{r_0, r_1, \dots, r_{l-1}\} \quad (51)$$

$$P_l \perp \text{span}\{P_0, P_1, \dots, P_{l-1}\} \quad (52)$$

$$\begin{aligned} & \text{span}\{r_0, r_1, \dots, r_{l-1}\} \\ &= \text{span}\{P_0, P_1, \dots, P_{l-1}\} \\ &= \text{span}\{r_0, A r_0, \dots, A^{l-1} r_0\} \end{aligned} \quad (53)$$

其中 $\text{span}\{r_0, A r_0, \dots, A^{l-1} r_0\}$ 为 krylov 子空间 [14] 我们对每一个 h_i 进行迭代求解，最终可以得到滤波器 H 的输出。

5. 算法仿真对比

书中 4.21 给出了一个例子，本文从该例子着手进行算法的仿真与对比假定一个随机信号 $x(t) = A \cos(2\pi f_0 t + \phi)$ 其中 A 与 f_0 给定 $\phi \sim U(0, 2\pi)$ 假定干扰信号 $u(t)$ 为高斯白噪声即 $u(t) \sim \mathcal{N}(0, \sigma^2 u)$ 且信号与 $x(t)$ 不相关，传感器包含热高斯噪声 $w_m(t) \sim \mathcal{N}(0, \sigma^2 w)$ 。收到的信号为 $y_m(t) = x_m(t) + v_m(t), m = 1, 2, \dots, M$ ，其中 $v_m(t) = w_m(t) + u_m(t), m = 1, 2, \dots, M$ $v_m(t)$ 为加性噪声。

$$[R_{x_1}]_{i,j} = \frac{1}{2} A^2 \cos[2\pi f_0(i-j)] \quad (54)$$

$$i\text{SNR} = 10 \log \frac{A^2}{\sigma^2 u + \sigma^2 w} (\text{dB}) \quad (55)$$

给出条件 $A = 0.5, f_0 = 0.1, \sigma^2 u = 0.5, \sigma^2 w = 0.01 \sigma^2 u$ 则 $i\text{SNR} = 6.06 \text{dB}$ 选择上文提到的 $G(H)$ 与 $J(H)/L$ 作为衡量指标，以 CG 算法求解维纳滤波矩阵进行仿真实验。并通过改变滤波阶数与传感器个数验证两者对于还原信号的影响。结果如图2、3所示：

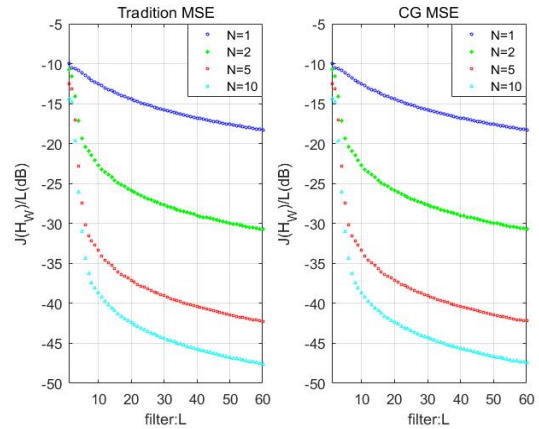


图 2. 不同算法下维纳滤波后的误差

能够发现 CG 算法的运用并不会降低滤波器求解矩阵的精确度，其效果与传统算法相当，并且能够从结果图中看到随着滤波阶数与传感器个数的增多，我们对于信号的 G_w 能够有效的提高，并且滤波后的信号与期望信号的误差会随着两者的增加逐渐

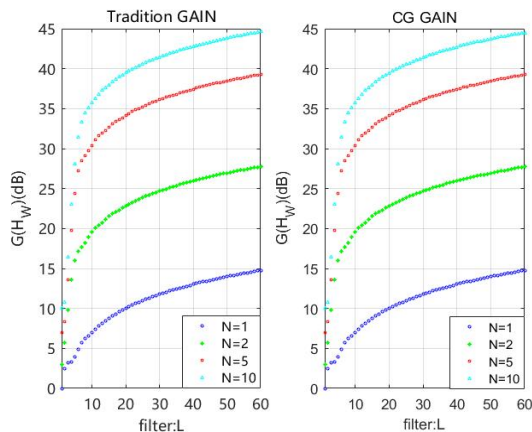


图 3. 不同算法下维纳滤波矩阵对于信号的增益

减小，反映了多阵列传感器能够对语音信号有更好的增强。

接着我们在保证滤波阶数不变的时候改变传感器个数，以此获得两种算法在不同传感器个数下的运行时间：

表 1. 算法运行时间

	N=1	N=5	N=10	N=50
CG	0.0036	0.0029	0.0063	2.9311
tradition	0.0053	0.0059	0.1272	10.3927

我们能够发现 CG 算法能够明显的改善求解维纳滤波矩阵的运算复杂度，同时也能够保持较高的精度，为了进一步验证维纳滤波器分别对于信号与噪声的作用，我们改变输入信噪比的大小，并且保证滤波阶数 L 为 30，求得传感器个数对于降噪系数与信号衰减因子的影响。

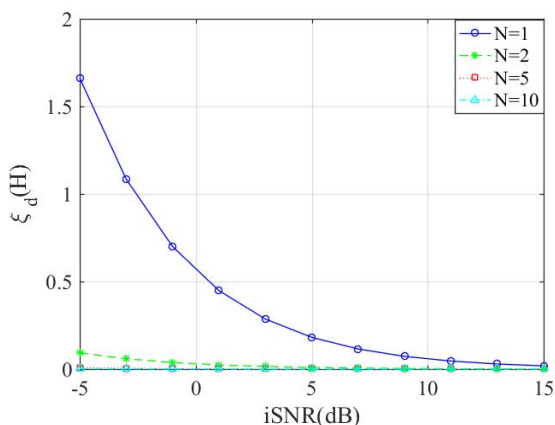


图 4. 传感器个数对于信号衰减因子的影响

结果如图4, 图5所示，能够发现随着传感器个数

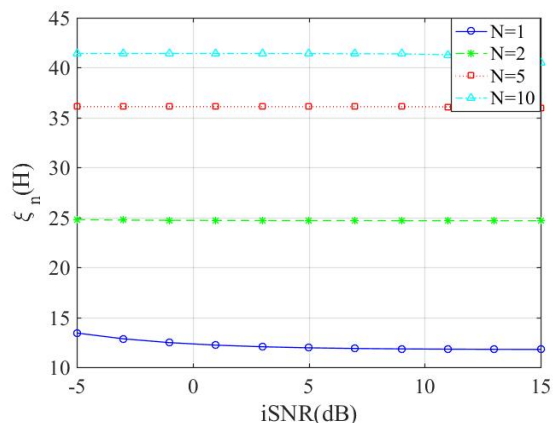


图 5. 传感器个数对降噪系数的影响

的增多，维纳滤波器对噪声的抑制效果会逐渐变好，并且对于信号的恢复也有更令人满意的效果。

6. 心得体会

6.1 心得体会一/邱梁城

写到这里计算方法这门课也告一段落了。

回顾起这学期课堂，从一开始接触计算方法的兴奋，到遇见晦涩难懂公式时的沉闷，再到能够实际应用这些方法时的宽慰，庆幸自己坚持了下来。

张老师的这门课让我学习到了很多概念和算法知识，各类的插值法、海森伯格矩阵、CG 算法、krylov space 等等，之前自己对深度学习有着很大兴趣，但是却没有深入的了解里面的算法原理、如何实现这门课的学习让我能对原理开始进行剖析，再懵懂的工程实践中让我能够有一把自己的匕首攻克难题。再深度算法的学习过程中，共轭梯度算法是一个很重要的内容，但是之前一直一知半解，借着这次大作业的机会，通过查找文献，动手实践，终于能够将其解决。

对于我来说，这门课最大的收获并不在于算法知识的进展和积累，张老师再课堂上不断提起的创新意识、思考意识才是让我受益终身的思想，通过课堂上的问题深入，让我们能够有着自己思考的机会和过程，十分难得宝贵。

数学是思考者的天堂，这句话现在越发感觉正确，对于以前来说，没有考虑过在工程中该如何解决大型线性方程组的求解问题，似乎只会暴力求解，但是无论是这学期所学的 LU 分解、特征值求解、cholesy 分解等等均是快速高效的算法，但是时常质问自己，换做我又是否能想出这些算法或是在这些算法上更近一步，能思考、会思考才是他们与我

们的不同，这门课在学习思想上的教育远远大于这门课本身。

此外，在最后的大作业中，调研，仿真，撰写报告的流程，其实和我们做研究的过程很像，在一次的报错中进步，学习才是科研本身的魅力所在。

由衷的感谢张老师在这一学期中对我的教诲！

6.2 心得体会二/严铮

我觉得这节课最重要的就是让我明白可以用迭代的方法求出工程问题的数值解，为解决问题提供了一些新思路。对于本次大作业，一开始有些许害怕，因为大作业有许多部分组成，但通过查阅资料，交流与讨论，最终豁然开朗。对于之前只用过 word 文档的我来说，latex 的使用还是很令人头疼，好在通过学习网上学习资料，也算是成为了入门新手。总之我觉得本次课程让我受益匪浅，而在完成作业的同时也学习到了很多别的东西，相信一定会对以后的学习有所帮助。

参考文献

- [1] 胡曙辉. 麦克风阵列语音增强算法研究. PhD thesis, 西安电子科技大学.
- [2] 赫靓. 基于 *Krylov* 子空间的快速 *DOA* 估计. PhD thesis, 哈尔滨工业大学, 2020.
- [3] 罗金玉, 刘建平, and 张一闻. 麦克风阵列信号处理的研究现状与应用. 现代电子技术, 33(23):5, 2010.
- [4] 丁猛. 基于麦克风阵列的语音增强算法概述. 大众科技, (3):29-30, 2011.
- [5] 黄剑文. 多通道语音增强算法及应用研究. Master's thesis, 福州大学, 2014.
- [6] 杜军 and 桑胜举. 基于麦克风阵列的语音增强技术及应用. 计算机应用与软件, 26(10):3, 2009.
- [7] 姚胜伟. 几类共轭梯度算法的研究. PhD thesis, 华东理工大学.
- [8] 房喜明, 令锋, and 傅守忠. 求解线性方程组的一般共轭梯度法. 数学理论与应用, (2):10, 2019.
- [9] Higham N J. Cholesky factorization[J]. Wiley interdisciplinary reviews: computational statistics, 2009, 1(2): 251-254.
- [10] Zhang, Fuzhen, ed. The Schur complement and its applications. Vol. 4. Springer Science Business Media, 2006.
- [11] Perry A. A modified conjugate gradient algorithm[J]. Operations Research, 1978, 26(6): 1073-1078.
- [12] Tesch K, Gerkmann T. Nonlinear spatial filtering in multichannel speech enhancement[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021, 29: 1795-1805.
- [13] Benesty J, Cohen I, Chen J. Fundamentals of signal enhancement and array signal processing[M]. John Wiley Sons, 2017.
- [14] Burden R L, Faires J D, Burden A M. Numerical analysis[M]. Cengage learning, 2015.