# 一种求解线性方程组的算法*

韩　峰
(西北核技术研究所五室　西安 710024)
(西安电子科技大学智能信息处理研究所　西安 710071)
王铁良
(西北核技术研究所五室　西安 710024)
焦李成
(西安电子科技大学智能信息处理研究所　西安 710071)

摘　要

把求解线性方程组的问题转化为一个二次函数的优化问题后，给出了一个降低该优化问题求解空间维数的方法. 使用这种方法把求解空间的维数降低到一维后，二次函数的最小值将很容易求得，从而得到线性方程组的解.

**关键词**:　线性代数方程组，二次函数，极值

# ALGORITHM FOR SOLVING SYSTEMS OF LINEAR EQUATIONS

Han Feng
(*Northwest Institute of Nuclear Technology, Xi'an*, 710024, *China*)
(*Institute of Intelligence Information Processing, Xidian Univ., Xi'an* 710071 *China*)
Wang Tieliang
(*Northwest Institute of Nuclear Technology, Xi'an*, 710024, *China*)
Jiao Licheng
(*Institute of Intelligence Information Processing, Xidian Univ., Xi'an* 710071 *China*)

**Abstract**

The problem of solving systems of linear equations is converted into calculating the minimum of a quadratic function. We presented a method of reducing the dimension of the optimizing problem for calculating the minimum. The dimension of the optimizing problem is degraded from N to one and the minimum can be obtained easily.

**Key words:** linear equations, quadratic function, extremum

* 2003 年 6 月 10 日收到.

## §1. 引 言

众所周知，许多实际问题最后常常归结为解一个线性代数方程组，因而对线性代数方程组的求解就成为一个重要问题. 当线性代数方程组系数矩阵的阶数不大时，可以用直接法求解，但当系数矩阵的阶数很大时，直接解法 "均告无效"[1]. 对高阶线性代数方程组的求解，一般使用迭代方法 [1,2]. 共轭梯度法是一种常用的迭代方法，它从理论上证明了，对于一个系数矩阵是 $n$ 阶对称正定矩阵的线性代数方程组，经过 $n$ 步迭代可以求得它的精确解. 在实际应用中，我们发现，当系数矩阵的阶数较高、条件数较大时，直接应用共轭梯度法求解线性代数方程组，经过 $n$ 步迭代，并不能求得精确解，而需要继续进行迭代. 本文借鉴了共轭梯度法的一些思想 [2,3]，提出了一个新的方法. 在把线性代数方程组的求解问题转化为一个二次函数的优化问题后，提出了一种依次降低求解空间维数的方法 —N 次降维法. 对于一个 $n$ 维问题，通过 $n-1$ 次降维，把问题化简成一个一维问题，从而可以简单地求得问题的解. 本文结构安排如下：第 2 节介绍把线性代数方程组求解问题转化为一个二次函数优化问题的方法；第 3 节用一个简单的例子对比说明共轭梯度法和本文方法搜索步骤的异同点，然后给出 N 次降维法的具体算法；第 4 节给出了算法的理论推导和证明，最后，把本文方法与共轭梯度法进行了比较，验证了本文方法的有效性.

## §2. 问题转化

考虑线性代数方程组

$$Ax = f. \tag{1}$$

其中 $A$ 是 $n$ 阶可逆的实矩阵， $x$，$f \in R^n$. 不妨设 $A$ 对称正定，不然考虑

$$A^TAx = A^Tf. \tag{2}$$

显然方程组 (1) 和 (2) 同解， $A^TA$ 对称正定. 设 $x^*$ 是方程组 (1) 的解，则

$$Ax^* = f. \tag{3}$$

构造二次函数

$$\begin{aligned} H(x) &= (A(x-x^*)\,,\,(x-x^*)) \\ &= (Ax\,,\,x) - 2(f\,,\,x) + (Ax^*\,,\,x^*). \end{aligned} \tag{4}$$

由于 $A$ 对称正定，显然当 $x = x^*$ 时， $H(x)$ 取极小值零. 令

$$F(x) = \frac{1}{2}(Ax, x) - (f, x). \tag{5}$$

显然当 $x = x^*$ 时， $F(x)$ 也取极小值.

令 $x = x^* + s\Delta x$, 其中 $\Delta x$ 是一个非零向量， $s \in R$, 则

$$F(x) = \frac{1}{2}(Ax\,,\,x) - (f\,,\,x) = F(x^* + s\Delta x). \tag{6}$$

对每一个给定的 $s$ 是一个等量面，它是一个过点 $x^* + s\Delta x$ 的 $n$ 维椭球面，当 $s = 0$ 时，

椭球面退化成一点 $x^*$. 由

$$\begin{aligned} F(x^*+s\Delta x) &= \frac{1}{2}(A(x^*+s\Delta x)\ ,\ x^*+s\Delta x)-(f\ ,\ x^*+s\Delta x) \\ &= F(x^*)+\frac{s^2}{2}(A\Delta x\ ,\ \Delta x) \end{aligned} \tag{7}$$

及 $A$ 对称正定使 $(A\Delta x.\Delta x)>0$. 可知 $F(x^*+s\Delta x)$ 是 $s^2$ 的单调增函数, 因此, 当 $s^2$ 变大时, 由 (6) 确定的椭球面一个包一个, 越向外, $F(x)$ 越大, 而它们都以 $x^*$ 为中心. 下面讨论如何求出由 (6) 确定的椭球面的中心, 也就是使 $F(x)$ 取极小值的 $x^*$.

## §3. N 次降维法

### 3.1　与共轭梯度法搜索过程的比较

共轭梯度法是一种常用的迭代方法 [1-3], 下面用一个简单的例子来说明它的搜索步骤. 考虑一个由 (5) 式确定的二次函数, 其中 $A$ 是 2 阶对称正定矩阵, $x,f\in R^2$. 由前节可知, 函数 $F(x)$ 的等值线是一族共心椭圆, $F(x)$ 在椭圆的中心取得极小值. 如图 1 所示, 按照共轭梯度法的步骤, 第一步, 从任意初始点 $x_1$ 出发, 取搜索方向 $p_1$ 为 $x_1$ 点的负梯度方向 $-g_1$ 进行一维搜索, 得到使 $F(x)$ 取到局部极小值的点 $x_2$. 第二步, 从点 $x_2$ 出发, 沿方向 $p_2$ 进行一维搜索, $p_2$ 是由 $x_2$ 点的负梯度方向 $-g_2$ 和 $p_1$ 确定的, 使得 $p_2$ 和 $p_1$ 相互 $A$ 共轭. 此时, 由点 $x_2$ 出发沿 $p_2$ 方向的直线 $l_2$ 是椭圆的弦 $l_1$ 的共轭直径, 所以直线 $l_2$ 通过椭圆的中心, 因此从点 $x_2$ 出发沿方向 $p_2$ 进行一维搜索, 就可以搜索到椭圆中心 $x^*$. 通过上面的搜索过程可以看出, 共轭梯度法的搜索过程是: 从任意取定的一点出发, 依次找到了 $n$ 个相互共轭的方向, 并在这 $n$ 个方向上进行搜索.

观察共轭梯度法的搜索过程发现, 只要能够找到一条通过椭球 (对于二维函数就是椭圆) 中心的直径, 然后在这条直径上进行搜索, 就能得到椭球中心. 基于这样的认识, 我们提出了一种寻找椭球直径的方法, 为叙述方便, 仍以上面二维例子介绍我们的方法.

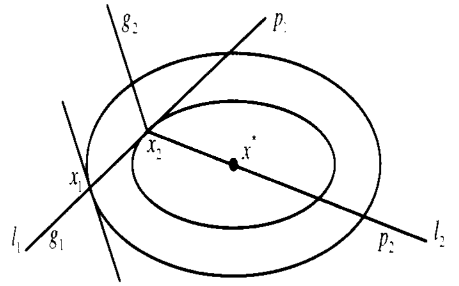


图 1　两变量二次函数的共轭梯度搜索

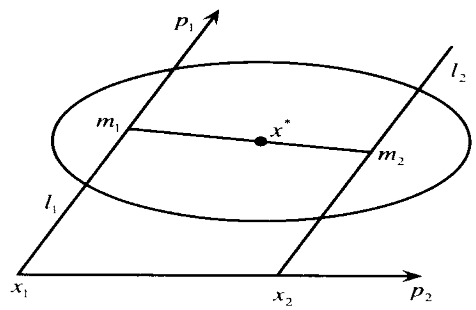


图 2　N =2 时 N 次降维法的搜索

如图 2 所示, 任取初始点 $x_1$ 以及两线性无关向量 $p_1$ 和 $p_2$. 第一步, 从 $x_1$ 点出发沿方向 $p_1$ 进行一维搜索, 得到使函数 $F(x)$ 在直线 $l_1$ 上取局部极小值的点 $m_1$, 再在从 $x_1$ 点出发沿方向 $p_2$ 的直线上任取一点 $x_2$, 然后从 $x_2$ 点出发沿方向 $p_1$ 进行一维搜索, 得到使函数 $F(x)$ 在直线 $l_2$ 上取局部极小值的点 $m_2$. 第二步, 在由 $m_1$ 和 $m_2$ 确定的直线上进行搜

索，就可求得椭圆中心 $x^*$. 事实上，通过第一步的搜索，确定了与两条直线 $l_1, l_2$ 相交的某个椭圆的两条弦的中点，而椭圆两条平行弦中点的连线通过椭圆的中心，因此通过第一步的搜索，我们确定了一条直线，它通过椭圆中心. 在第二步中，只要在第一步中确定的直线上搜索就可以求得椭圆的中心，这相当于通过第一步的搜索，问题的搜索空间被降了一维. 类似地，对于一个 $n$ 维椭球，只要通过 $n-1$ 次搜索、降维就可以把问题化简为一个一维问题，从而简单地获得问题的解. 从上面的两个例子可以看到，共轭梯度法和本文方法的相同之处是都利用了共轭方向的概念，不同之处是共轭梯度法是依次在 $n$ 个相互共轭的方向进行搜索，而本文方法是每进行一次搜索，就把搜索空间降低了一维，直至搜索空间变为一条直线，再在其上搜索，就找到了全局极小点. 下面给出 N 次降维法的具体算法以及理论证明.

### 3.2 算法描述

取 $p_i$ 的初值为 $n$ 维空间中的一组基

$$p_i = [\overbrace{0\ 0\cdots0\ \underset{i}{1}\ \ 0\ \ \cdots\ \ 0}^{n}]^T, \quad i = 1, 2, \cdots n.$$

算法步骤:

1. 任取一初始点 $x_0$;
2. for j = 1 to n

   for i = j to n

   $x_0 = x_0 + p_i$

   $x_i = x_0 + \dfrac{(f - Ax_0, p_j)}{(Ap_j, p_j)} p_j$

   end

   for i = j+1 to n

   $p_i = x_i - x_j$

   end

   $x_0 = x_j$

   end
3. $x^* = x_0$

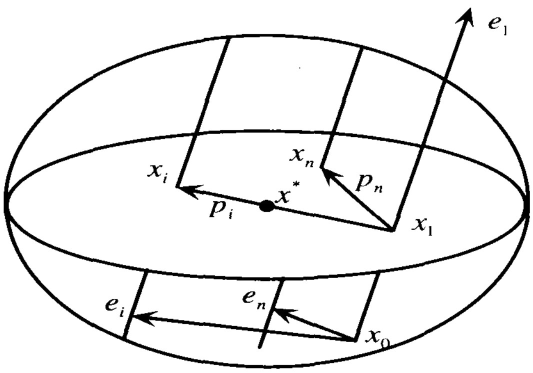


图 3 $e_1$ 方向的共轭平面

在上面的算法中，当 j=1 时有 $p_j = e_1$，所以这时求 $p_j$ 的共轭平面，就是求 $e_1$ 的共轭平面，要求 $e_1$ 的共轭平面只要求出 $x_1, x_2\cdots, x_n$，然后令 $p_i = x_i - x_1, i = 2, 3, \cdots. n$. 则由点 $x_1$ 和向量 $p_2, p_3, \cdots, p_n$ 确定的平面就是 $e_1$ 的共轭平面. 图 3 是 $e_1$ 共轭平面的示意图.

## §4. 理论证明

对于由形如 (6) 式确定的椭球，我们有下面的定理:

**定理 1.** 设 $n$ 维椭球方程为

$$F(x) = \frac{1}{2}(Ax\ .\ x) - (f\ ,\ x) = F(a),$$

其中 $A$ 是对称正定矩阵， $x\ .\ f\ .\ a \in R^n$. $p$ 为任意一个给定的方向，那么存在一个 $n-1$ 维平面，把椭球沿方向 $p$ 的所有的弦平分，且椭球的中心在这个平面上.

证明. 椭球方程为

$$F(x) = \frac{1}{2}(Ax\ .\ x) - (f\ ,\ x) = F(a), \tag{8}$$

其中 $x\ .\ f\ .\ a \in R^n$. $p$ 任一给定向量. 任取一点 $x_1$，再取向量 $p_1, p_2, \cdots, p_{n-1}$. 使 $p. p_1. p_2. \cdots. p_{n-1}$ 组成空间的一组基，考虑超平面

$$x = x_1 + \alpha_1 p_1 + \alpha_2 p_2 + \cdots + \alpha_{n-1} p_{n-1} + \lambda p, \tag{9}$$

其中 $\alpha_1. \alpha_2. \cdots. \alpha_{n-1}. \lambda \in R$. 当 $\alpha_i\ .\ i = 1, 2, \cdots, n-1$ 固定时， (9) 式确定了一条过点 $x_1 + \sum_{i=1}^{n-1} \alpha_i p_i$ 沿方向 $p$ 的直线，联立 (8) 和 (9), 可以求得这条直线和椭球的交点，即

$$\begin{aligned}
&F\Big(x_1 + \lambda p + \sum_{i=1}^{n-1} \alpha_i p_i\Big) - F(a) = 0, \\
&\frac{1}{2}(Ap. p)\lambda^2 - \lambda\Big(f - A\Big(x_1 + \sum_{i=1}^{n-1} \alpha_i p_i\Big)\ ,\ p\Big) + \frac{1}{2}\Big(A\sum_{i=1}^{n-1} \alpha_i p_i, \sum_{i=1}^{n-1} \alpha_i p_i\Big) \\
&-\Big(f - Ax_1\ ,\ \sum_{i=1}^{n-1} \alpha_i p_i\Big) + F(x_1) - F(a) = 0.
\end{aligned} \tag{10}$$

方程 (10) 的根 $\lambda_1$ 和 $\lambda_2$ 对应于两个交点，而与两个交点的中点对应的 $\lambda'$ 值为

$$\lambda' = \frac{1}{2}(\lambda_1 + \lambda_2) = \frac{\Big(f - A\Big(x_1 + \sum_{i=1}^{n-1} \alpha_i p_i\Big), p\Big)}{(Ap, p)}, \tag{11}$$

因此过点 $x_1 + \sum_{i=1}^{n-1} \alpha_i p_i$ 沿方向 $p$ 的直线与椭球相交的弦的中点为

$$x = x_1 + \alpha_1 p_1 + \alpha_2 p_2 + \cdots + \alpha_{n-1} p_{n-1} + \lambda' p. \tag{12}$$

当 $\alpha_i\ ,\ i = 1, 2, \cdots, n-1$ 变动时，由 (12) 式确定的 $n-1$ 维平面就把椭球沿方向 $p$ 的所有的弦平分. 显然平面 (12) 把椭球沿方向 $p$ 的直径平分，因此椭球的中心在平面 (12) 上. 证毕.

**定义 1.** 称能够把椭球沿任意给定方向的所有弦平分的椭球截面为该方向的共轭平面.

**引理 1.** 设椭球方程为

$$F(x)=\frac{1}{2}(Ax\ ,\ x)-(f\ ,\ x)=F(a). \tag{13}$$

其中 $A$ 是对称正定矩阵, $x$ , $f$ , $a\in R^n$, $p$ 为任意一个给定的方向, $l_1,l_2$ 为椭球沿方向 $p$ 的任意两条弦, $x_1,x_2$ 是这两条弦的中点, 那么有

$$(Ap,x_1-x_2)=0. \tag{14}$$

证明. 在 $l_1,l_2$ 上分别任取两点 $m_1,m_2$, 则

$$\begin{aligned} x_1&=m_1+\frac{(f-Am_1,p)}{(Ap,p)}p,\\ x_2&=m_2+\frac{(f-Am_2,p)}{(Ap,p)}p. \end{aligned} \tag{15}$$

$$\begin{aligned}(Ap,x_1-x_2)&=(Ap,m_1-m_2+\frac{(f-Am_1-f+Am_2,p)}{(Ap,p)}p\\ &=(Ap,m_1-m_2)+\frac{(f-Am_1-f+Am_2,p)}{(Ap,p)}(Ap,p)\\ &=(Ap,m_1-m_2)-(A(m_1-m_2),p)=0.\end{aligned} \tag{16}$$

证毕.

**定理 2.** 设椭球方程为

$$F(x)=\frac{1}{2}(Ax\ ,\ x)-(f\ ,\ x)=F(a). \tag{17}$$

其中 $A$ 是对称正定矩阵, $x$ , $f$ , $a\in R^n$. $p$ 为任意一个给定的方向, 那么存在椭球沿方向 $p$ 的 $n$ 条弦 $l_1,l_2,\cdots,l_n$, $x_1,x_2,\cdots,x_n$ 是这 $n$ 条弦的中点, 使得由 $x_1,x_2,\cdots,x_n$ 这 $n$ 个点确定的平面是椭球 $p$ 方向的共轭平面.

证明. 在空间中取 $n-1$ 个向量 $q_1,q_2,\cdots,q_{n-1}$ 使得 $q_1,q_2,\cdots,q_{n-1}$ 和 $p$ 组成一组基, 然后正交化这组基得到 $e_1,e_2,\cdots,e_n$, 使 $e_n$ 的方向与向量 $p$ 相同.

任取空间中的一点 $x_0$, 我们知道通过点 $x_0+\lambda_i e_i$ 沿方向 $e_n$ 的直线为

$$x=x_0+\lambda_i e_i+te_n,\quad \lambda_i\neq 0,\quad t\in R,\quad i=1,2,\cdots,n. \tag{18}$$

这 $n$ 条直线与椭球相交所得弦的中点可表示为

$$x_i=x_0+\lambda_i e_i+t_i e_n.$$

令

$$p_{i-1}=x_i-x_1,\quad i=2,3,\cdots,n. \tag{19}$$

则

$$p_{i-1}=x_i-x_1=\lambda_i e_i-\lambda_1 e_1+(t_i-t_1)e_n\ ,\ i=2,3,\cdots,n. \tag{20}$$

显然 $p_i$, $i=1,2,\cdots,n-1$ 线性无关. 事实上, 若

$$\alpha_1 p_1+\alpha_2 p_2+\cdots+\alpha_{n-1}p_{n-1}=0, \tag{21}$$

分别用 $e_1, e_2, \cdots, e_{n-1}$ 对 (21) 式两边做内积得到

$$\begin{aligned}&\alpha_1+\alpha_2+\cdots+\alpha_{n-1}=0,\\&\alpha_1(e_2,e_2)=0,\\&\alpha_2(e_3,e_3)=0,\\&\cdots\cdots\\&\alpha_{n-2}(e_{n-1},e_{n-1})=0,\end{aligned} \tag{22}$$

即 $\alpha_1, \alpha_2, \cdots, \alpha_{n-1}$ 全部为零, 故 $p_i, i=1,2,\cdots,n-1$ 线性无关. 由 $x_1, x_2, \cdots, x_n$ 这 $n$ 个点确定的平面为

$$x = x_1 + \alpha_1 p_1 + \alpha_2 p_2 + \cdots + \alpha_{n-1} p_{n-1}. \tag{23}$$

下面证明椭球沿 $p$ 方向的任意一条弦的中点在平面 (23) 上.

任取空间中一点 $x_0$. 通过点 $x_0$ 沿方向 $p$ 的直线为

$$x = x_0 + tp. \tag{24}$$

它与椭球相交所得弦的中点为

$$x' = x_0 + \frac{(f - Ax_0, p)}{(Ap, p)} p. \tag{25}$$

又 $p_1, p_2, \cdots, p_{n-1}, p$ 是空间的一组基, 所以存在一组系数 $\mu_1, \mu_2, \cdots, \mu_{n-1}, \mu$ 使得

$$x_0 = x_1 + \sum_{i=1}^{n-1} \mu_i p_i + \mu p. \tag{26}$$

把 (26) 代入 (25) 式得到

$$\begin{aligned}x' &= x_1 + \sum_{i=1}^{n-1}\mu_i p_i + \mu p + \frac{(f - A(x_1 + \sum\limits_{i=1}^{n-1}\mu_i p_i + \mu p), p)}{(Ap,p)} p\\&= x_1 + \sum_{i=1}^{n-1}\mu_i p_i + \mu p + \frac{(f - Ax_1, p) - \mu (Ap,p)}{(Ap,p)} p\\&= x_1 + \frac{(f - Ax_1, p)}{(Ap,p)} p + \sum_{i=1}^{n-1}\mu_i p_i\\&= x_1 + \sum_{i=1}^{n-1}\mu_i p_i.\end{aligned} \tag{27}$$

(27) 式表明 $x'$ 在平面 (23) 上, 即平面 (23) 是椭球 $p$ 方向的共轭平面. 证毕.

定理 1 表明, 椭球的中心在任一给定方向的共轭平面上, 而任一方向的共轭平面的维数是 $n-1$. 这就启发我们, 通过 $n-1$ 次降维, 就可以把问题化简为一个一维问题, 然后只需要在一条直线上寻找椭球的中心. 定理 2 给出了求任一给定方向的共轭平面的方法.

## §5. 方法验证

为了验证本文算法的有效性, 我们构造了一个线性代数方程组. 对系数矩阵, 我们的取

法是任意取一个 $n$ 阶随机矩阵 $G$, 然后令 $A = G^TG$, 任取一个随机向量作为初值 $x_0$, 令 $f = A * [n, n-1, \cdots, 1]^T$, 显然精确解为 $x^* = [n, n-1, \cdots, 1]^T$.

我们把上面的算法与共轭梯度法进行了比较, 结果如表 1 所示. 所进行的计算都是在 Matlab 下编程计算的. 所使用计算机的 CPU 为奔腾 4, 主频 2G, 内存 256M. 表 1 是本文方法和共轭梯度法都进行 $n$ 步计算所得解的精度对比, 显然共轭梯度法所求解的精度明显大大低于本文方法所求解的精度. 从计算时间来看, 当 $n$ 逐渐增大时, 本文方法的计算时间约是共轭梯度法的两倍. 表 2 是 $n = 50$ 时两种方法迭代 $n$ 步所求解的对比, 本文方法求得了精确解, 而共轭梯度法所求得的解有较大的偏差.

**表 1 共轭梯度法与本文方法的对比**

| | | $error = \|Ax^* - f\|_2$ | | 计算时间 (秒) | | |
|---|---|---|---|---|---|---|
| 维数 | 条件数 | 共轭梯度法 | 本文方法 | 共轭梯度法 $t_1$ | 本文方法 $t_2$ | $t_2/t_1$ |
| 50 | 1.7825e+008 | 0.0494 | 1.3541e-010 | 0.01 | 0.04 | 4 |
| 200 | 7.6792e+008 | 1.2844 | 1.7178e-007 | 0.09 | 0.9510 | 10.6 |
| 500 | 1.5887e+010 | 1.7766 | 4.5603e-005 | 1.572 | 10.085 | 6.4 |
| 800 | 1.3827e+009 | 1.0626 | 2.0519e-004 | 12.5580 | 40.3380 | 3.2 |
| 1000 | 8.3181e+010 | 1.3305 | 2.9135e-004 | 26.8790 | 75.6180 | 2.8 |

**表 2 $n = 50$ 时两种方法迭代 $n$ 步所求解的对比**

| 本文方法 | 共轭梯度法 | 本文方法 | 共轭梯度法 | 本文方法 | 共轭梯度法 |
|---|---|---|---|---|---|
| 50.0000 | 50.6828 | 33.0000 | 32.7050 | 16.0000 | 18.1703 |
| 49.0000 | 48.5661 | 32.0000 | 31.9720 | 15.0000 | 15.0462 |
| 48.0000 | 46.8294 | 31.0000 | 31.3957 | 14.0000 | 14.0761 |
| 47.0000 | 46.1612 | 30.0000 | 28.7411 | 13.0000 | 13.6444 |
| 46.0000 | 45.5002 | 29.0000 | 29.0044 | 12.0000 | 11.6433 |
| 45.0000 | 44.9502 | 28.0000 | 26.8187 | 11.0000 | 11.5459 |
| 44.0000 | 45.3843 | 27.0000 | 26.9482 | 10.0000 | 10.3793 |
| 43.0000 | 43.6470 | 26.0000 | 25.9573 | 9.0000 | 8.7549 |
| 42.0000 | 42.1919 | 25.0000 | 26.5778 | 8.0000 | 7.2305 |
| 41.0000 | 41.0440 | 24.0000 | 24.9140 | 7.0000 | 6.5536 |
| 40.0000 | 40.4366 | 23.0000 | 23.7109 | 6.0000 | 5.6751 |
| 39.0000 | 38.2897 | 22.0000 | 23.4966 | 5.0000 | 5.2762 |
| 38.0000 | 37.7859 | 21.0000 | 19.6294 | 4.0000 | 4.0184 |
| 37.0000 | 35.5099 | 20.0000 | 20.8049 | 3.0000 | 2.5226 |
| 36.0000 | 34.9973 | 19.0000 | 20.3974 | 2.0000 | 1.8339 |
| 35.0000 | 36.0634 | 18.0000 | 18.1699 | 1.0000 | -0.1408 |
| 34.0000 | 33.1826 | 17.0000 | 16.4539 | | |

## §6. 讨　论

本文方法把一个 $n$ 维问题, 通过 $n-1$ 次降维, 化简成了一个一维问题, 从而简单地

获得了问题的解. 上面的算例验证了本文方法的有效性. 从计算量来看，本文方法和共轭梯度法的计算量是同阶的. 当维数较大时，本文方法的计算量 (计算时间) 约是共轭梯度法的两倍. 从计算结果的精确性来看，本文方法和共轭梯度法都进行 $n$ 步计算，本文方法的计算结果明显优于共轭梯度法. 当 $n$ 较大时，共轭梯度法的解距精确解有较大的偏差，而本文方法仍然能够求得精确解. 当共轭梯度法进行 $n$ 步求得精确解的精度较低时，我们还对共轭梯度法进行了进一步的迭代计算. 计算结果表明，要使共轭梯度法求得的解的精度达到本文方法求得的解的精度，则共轭梯度法迭代所用的时间就要远大于本文方法所用的时间. 综上所述，本文方法较好地实现了经过 $n$ 步计算，求得精确解的理论结论. 另外，需要指出的一点是，若本文方法经过 $n$ 步计算后，所求解的精度不能满足要求，那么还可以把所求的解作为初始点，继续进行 $n$ 步计算.

## 参 考 文 献


[1] 蔡大用等，高等数值分析 [M]. 清华大学出版社，北京， 1997. 41-70.

[2] 胡家赣，线性代数方程组的迭代解法 [M]. 科学出版社，北京， 1997. 173-185.

[3] 张智星等. 神经 — 模糊和软计算 [M]. 西安交通大学出版社，西安， 2000, 89-135.