

# BÁO CÁO MÔN HỌC

## Sử dụng phương pháp lặp Seidel và Gauss-Seidel (cho hệ chéo trội) để giải hệ phương trình đại số tuyến tính

Giảng viên hướng dẫn: Lê Thị Ngọc Yến

Ngày 2 tháng 3 năm 2016



Nhóm báo cáo:

1. Phan Huy Hoàng ( nhóm trưởng ) - Toán Tin 2-K59
2. Trần Thị Thanh Tâm - Toán Tin 2-K59
3. Nguyễn Văn Thành Dũng - Toán Tin 2-K59

## Mục lục

1	Lời mở đầu	2
2	Bài toán đặt ra	3
3	Phương pháp lặp Seidel	5
4	Phương pháp lặp Gauss - Seidel	10
5	Chạy chương trình trên máy	12
6	Nhóm báo cáo	17
7	Kết luận	18

# 1 Lời mở đầu

- Sau khi học các môn học cơ sở ngành như: Tin học đại cương, Kỹ Thuật Lập Trình và Cơ sở Giải Tích Hàm, dưới sự hướng dẫn của giáo viên bộ môn Giải Tích Số, cô Hà Thị Ngọc Yến, chúng em đã tiến hành tìm hiểu nội dung đề tài để viết nên báo cáo này. Đề tài của nhóm em là: **Tìm hiểu phương pháp lặp Seidel và Gauss-Seidel để giải hệ phương trình đại số tuyến tính.**

- Nội dung chính của báo cáo bao gồm những phần cơ bản như: lý thuyết cơ bản, các định nghĩa và định lý trong đề tài, code chương trình và chạy chương trình trên máy,....

- Như chúng ta đã biết, để giải hệ phương trình đại số tuyến tính ( ĐSTT ), chúng ta có 2 phương pháp chính cơ bản như sau: phương pháp giải đúng ( như Gauss, Gauss-Jordan,...) và phương pháp giải gần đúng ( như lặp đơn, Seidel, Gauss-Seidel). Trong báo cáo này ta sẽ đề cập tới phương pháp giải gần đúng Seidel và Gauss-Seidel cho hệ phương trình ĐSTT, những ưu điểm, hạn chế của phương pháp so với phương pháp lặp đơn, khi nào và tại sao sử dụng phương pháp này,...

- Vì cũng là lần đầu tìm hiểu đề tài, thời gian tìm hiểu tương đối ngắn, hơn nữa cũng là 1 trong 2 nhóm thuyết trình đầu tiên nên báo cáo do nhóm thực hiện không tránh khỏi những thiếu sót ( cả về mặt trình bày và nội dung báo cáo ). Với khả năng có hạn nên có thể trong báo cáo chưa trình bày được đầy đủ nội dung chương trình, cũng mong giáo viên bộ môn góp ý và lượng thứ.

— Tài liệu được viết bằng LATEX, 24/02/2016 —

## 2 Bài toán đặt ra

### Chuẩn vectơ và sự hội tụ của dãy vectơ

- Mục đích của phương pháp tìm nghiệm gần đúng của hệ đại số tuyến tính là: xuất phát từ vectơ  $X^{(0)}$  gọi là xấp xỉ đầu, tìm cách hiệu chỉnh dần sau một số bước ta thu được dãy vectơ  $\{X^{(k)}\}$  mà vectơ  $X^{(0)}$  gần đúng với nghiệm đúng của hệ. Vì vậy cần thiết phải đưa vào khái niệm về sự hội tụ của dãy vectơ. Khái niệm đơn giản và dễ hình dung nhất là **sự hội tụ theo tọa độ**.

- **Định nghĩa 1:** Dãy vectơ  $X^{(k)} = (x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)})^t$  với  $k = 1, 2, \dots$  trong không gian  $n$  chiều, hội tụ tới vectơ  $X = (x_1, \dots, x_n)^t \in R^n$  nếu:

$$\left| x_i^{(k)} - x_i \right| \rightarrow 0 \text{ khi } k \rightarrow \infty \quad \forall i = \overline{1, n}$$

- Và kí hiệu  $\lim_{k \rightarrow \infty} X^{(k)} = X$ .

- Sự hội tụ như vậy gọi là **hội tụ theo tọa độ**.

- Tuy vậy khái niệm này khó sử dụng, ta sẽ sử dụng sự hội tụ theo nghĩa khác mà hệ quả của nó cũng là sự hội tụ theo tọa độ. Để mô tả sự hội tụ này trước hết ta gán cho mỗi vectơ  $X$   $n$  chiều một số thực không âm  $N(X)$  mà ta gọi là chuẩn của vectơ  $X$ .

- **Định nghĩa 2:** Ta gọi **chuẩn của vectơ  $X$** , ký hiệu  $\|X\| = N(X)$  là một số không âm thỏa mãn điều kiện sau:

1.  $\|X\| = N(X) \geq 0, \|X\| = 0 \Leftrightarrow X = 0$
2.  $\|kX\| = N(kX) = |k|\|X\|, \quad k - \text{const}$
3.  $\|X + Y\| = N(X + Y) \leq \|X\| + \|Y\|$

( $X, Y$  là những vectơ  $n$  chiều)

- Như vậy,  $\|X\| = N(X)$  có thể xem như độ dài của vectơ  $X$  và các điều kiện 1, 2, 3 của nó giống tính chất độ dài hình học của vectơ hình học.

- Thông thường, ta có 3 loại chuẩn như sau:

( **Chuẩn cực đại** )

$$\|X\|_{(\infty)} = \max_i |x_i|$$

( **Chuẩn tuyệt đối** )

$$\|X\|_{(1)} = \sum_{i=1}^n |x_i|$$

( **Chuẩn Ôcolit** )

$$\|X\|_{(2)} = \left[ \sum_{i=1}^n |x_i|^2 \right]^{\frac{1}{2}}$$

- Dễ dàng chỉ ra rằng các chuẩn trên thỏa mãn các điều kiện 1, 2, 3.

- **Định nghĩa 3:** Ta nói dãy vectơ  $\{X^{(k)}\} = \{(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)})^t\}$  hội tụ tới vectơ  $X = (x_1, x_2, \dots, x_n)^t$ , nếu theo chuẩn nào đó ta có  $\|X^{(k)} - X\| \rightarrow 0$  khi  $k \rightarrow \infty$ .

- Với 3 loại chuẩn đã xét thì dễ dàng chứng minh được định nghĩa 1 và định nghĩa 3 tương đương.

- Bây giờ xét ma trận  $A = [a_{ij}]$  là ma trận vuông cấp  $n$ , ta cũng có:

- **Định nghĩa 4:** Ta gọi **chuẩn của ma trận  $A$** , ký hiệu  $\|A\|$  (đọc là chuẩn của ma trận  $\|A\|$ ) là số không âm thỏa mãn các điều kiện sau:

1.  $\|A\| \geq 0$  " = 0"  $\Leftrightarrow A = 0$  (ma trận không)
2.  $\|kA\| = |k|\|A\|, k = \text{const}$
3.  $\|A + B\| \leq \|A\| + \|B\|$

(A, B là những ma trận vuông cấp  $n$ )

- Các chuẩn của ma trận  $A$  thường được chọn:

(theo hàng)

$$\|A\|_{(\infty)} = \max_i \sum_{j=1}^n a_{ij}$$

(theo cột)

$$\|A\|_{(1)} = \max_i \sum_{j=1}^n a_{ij}$$

(chuẩn Ocolit)

$$\|A\|_{(2)} = \left[ \sum_{i=1}^n \sum_{j=1}^n a_{ij}^2 \right]^{\frac{1}{2}}$$

- Dễ dàng chỉ ra rằng 3 dạng chuẩn trên thỏa mãn các điều kiện về chuẩn; đồng thời ta cũng có:

$$\|AB\| \leq \|A\| \|B\| \Rightarrow \|A^m\| \leq \|A\|^m$$

- **Định nghĩa 5:** Dãy ma trận  $\{A^{(k)}\} = \{(a_{ij}^{(k)})_{i,j=\overline{1,n}}\}$

-  $k = 1, 2, \dots$  hội tụ tới ma trận  $A = (a_{ij})_n$  khi  $k \rightarrow \infty$  nếu  $\lim_{k \rightarrow \infty} a_{ij}^{(k)} = a_{ij}, \forall i, j$  (hội tụ theo dãy các phần tử

hoặc  $\|A^{(k)} - A\| \rightarrow 0$  khi  $k \rightarrow \infty$  (hội tụ theo chuẩn)

- Ký hiệu  $\lim_{k \rightarrow \infty} A^{(k)} = A$

- Trong bài toán phần lớn thường có sự tham gia của cả ma trận và vectơ:

$$AX = \left[ \sum_{j=1}^n a_{ij} \cdot x_j \right]_{i=\overline{1,n}}$$

(là vectơ cột)

### 3 Phương pháp lặp Seidel

- Phương pháp lặp Seidel đặt ra nhằm mục đích **tiết kiệm ô nhớ trong máy tính**, đồng thời cũng là **tránh thủ tối đa lượng thông tin đã có trong quá trình tính**.

- Xét hệ phương trình đại số tuyến tính (ĐSTT):

$$X = \alpha X + \beta \quad (1)$$

- Ta phân tích ma trận  $\alpha$  thành:  $\alpha = \alpha^{(1)} + \alpha^{(2)}$ :

$$\alpha = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & 0 & \vdots \\ a_{n1} & \dots & 0 \end{pmatrix} + \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \dots & a_{nn} \end{pmatrix}$$

- Từ (1) ta có:

$$X = \alpha X + \beta = [\alpha^{(1)} + \alpha^{(2)}]X + \beta$$

$$\text{hay : } X = \alpha^{(1)}X + \alpha^{(2)}X + \beta \quad (2)$$

• **Thuật toán:** Chọn xấp xỉ đầu là  $X^{(0)}$  bất kỳ, tính:

$$\begin{cases} x_1^{(1)} = \sum_{j=1}^n \alpha_{1j} \cdot x_j^{(0)} + \beta_1 \\ x_2^{(1)} = \alpha_{21} \cdot x_1^{(1)} + \sum_{j=2}^n \alpha_{2j} \cdot x_j^{(0)} + \beta_2 \\ \dots\dots\dots \\ x_n^{(1)} = \sum_{j=1}^{i-1} \alpha_{ij} \cdot x_j^{(1)} + \sum_{j=i}^n \alpha_{ij} \cdot x_j^{(0)} + \beta_j \\ \dots\dots\dots \\ x_n^{(1)} = \sum_{j=1}^{n-1} \alpha_{nj} \cdot x_j^{(1)} + \alpha_{nn} \cdot x_n^{(0)} + \beta_n \end{cases}$$

hay:

$$x_i^{(1)} = \sum_{j=1}^{i-1} \alpha_{ij} \cdot x_j^{(1)} + \sum_{j=i}^n \alpha_{ij} \cdot x_j^{(0)} + \beta_j \quad i = 1, 2, \dots, n$$

- Sau bước một, ta được xấp xỉ thứ nhất  $X^{(1)}$ :

$$X^{(1)} = (x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)})^t$$

-Lặp lại quá trình trên đối với  $X^{(1)}$ , ta được  $X^{(2)}$ , ....cứ như vậy đến phép lặp thứ  $k$ , ta có:

$$x_i^{(k)} = \sum_{j=1}^{i-1} \alpha_{ij} \cdot x_j^{(k)} + \sum_{j=i}^n \alpha_{ij} \cdot x_j^{(k-1)} + \beta_j \quad i = \overline{1, n} \quad k = 1, 2, \dots \quad (3)$$

hay:

$$X^{(k+1)} = \alpha^{(1)}X^{(k+1)} + \alpha^{(2)}X^{(k)} + \beta \quad k = 0, 1, 2, \dots \quad (4)$$

- **Điều kiện hội tụ và sai số:** Nếu quá trình lặp (4) hội tụ thì:  $\lim_{x \rightarrow \infty} x^{(k)} = \bar{X}$  thì  $\bar{X} = X^*$  là nghiệm đúng của hệ (1) hay (2)

- Thực vậy, từ (4), do  $X^{(k)} \rightarrow \bar{X}$  khi  $X \rightarrow \infty$  nên:

$$\bar{X} = \alpha^{(1)}\bar{X} + \alpha^{(2)}\bar{X} + \beta = (\alpha^{(1)} + \alpha^{(2)})\bar{X} + \beta$$

hay:

$$\bar{X} = \alpha\bar{X} + \beta$$

vậy:

$$\bar{X} = X^*$$

- Điều kiện để quá trình lặp (4) hội tụ và sai số.

**Định lí 1:** Nếu một chuẩn nào đó của ma trận  $\alpha$  thỏa mãn điều kiện:

$$\|\alpha\|_{(p)} < 1 \quad (p = \infty, 1)$$

thì phương pháp lặp Seidel hội tụ tới nghiệm đúng  $X^*$  của hệ phương trình (1) với nghiệm xấp xỉ đầu  $X^{(0)}$  bất kỳ.

- Đồng thời ta có các công thức đánh giá sai số:

**a) Nếu  $\|\alpha\|_{\infty} < 1$  thì:**

$$\|X^{(k)} - X^*\|_{\infty} < \frac{\lambda}{1 - \lambda} \|X^{(k)} - X^{(k-1)}\|_{\infty} \quad (5)$$

trong đó:

$$\lambda = \max_i \frac{\sum_{j=i}^n |\alpha_{ij}|}{1 - \sum_{j=i}^{i-1} |\alpha_{ij}|} \leq \|\alpha\|_{\infty} < 1$$

hoặc:

$$\|X^{(k)} - X^*\|_{\infty} < \frac{\|\alpha\|^k}{1 - \|\alpha\|} \|X^{(1)} - X^0\|_{\infty} \quad (6)$$

**b) Nếu  $\|\alpha\|_{(1)} < 1$  thì:**

$$\|X^{(k)} - X^*\|_1 < \frac{\zeta^k}{(1 - S)(1 - \zeta)} \|X^{(1)} - X^0\|_{(1)} \quad (7)$$

trong đó:

$$S = \max_i \sum_{j=1}^n |\alpha_{ij}|; \quad \zeta = \max_j \frac{\sum_{i=1}^j |\alpha_{ij}|}{1 - \sum_{i=j+1}^n |\alpha_{ij}|} \leq \|\alpha\|_{(1)} < 1$$

**CHỨNG MINH:** Khi  $\|\alpha\|_{(\infty)} < 1$  - theo (3), ta có:

$$x_i^{(k)} = \sum_{j=1}^{i-1} \alpha_{ij} \cdot x_j^{(k)} + \sum_{j=i}^n \alpha_{ij} \cdot x_j^{(k-1)} + \beta_j \quad i = \overline{1n}; k = 1, 2, \dots$$

- Nếu  $X^*$  là nghiệm đúng của phương trình (1) thì:

$$x_i^* = \sum_{j=1}^n \alpha_{ij} \cdot x_j^* + \beta_j \quad i = \overline{1n}$$

$$\Rightarrow x_i^{(k)} - x_i^* = \sum_{j=1}^{i-1} |\alpha_{ij}| (x_j^{(k)} - x_j^*) + \sum_{j=i}^n |\alpha_{ij}| (x_j^{(k-1)} - x_j^*)$$

$$\Rightarrow |x_i^{(k)} - x_i^*| \leq \sum_{j=1}^{i-1} |\alpha_{ij}| |x_j^{(k)} - x_j^*| + \sum_{j=i}^n |\alpha_{ij}| |x_j^{(k-1)} - x_j^*| \quad i = \overline{1n} \quad (8)$$

- Đặt  $P_i = \sum_{j=1}^{i-1} |\alpha_{ij}|$ ;  $q_i = \sum_{j=i}^n |\alpha_{ij}|$ ;

mà  $\forall j$  :  $|x_i^{(k)} - x_i^*| \leq \|X^{(k)} - X^*\|_{\infty}$  nên từ (9):

$$|x_i^{(k)} - x_i^*| \leq P_i \|X^{(k)} - X^*\|_{\infty} + q_i \|X^{(k-1)} - X^*\|_{\infty} \quad (9)$$

giả sử với vectơ  $X^{(k)}$ , có chỉ số  $s$  nào đó mà:

$$|x_s^{(k)} - x_s^*| = \max_i |x_i^{(k)} - x_i^*| = \|X^{(k)} - X^*\|_{\infty}$$

thì từ (9) ta có:

$$\|X^{(k)} - X^*\|_{\infty} \leq p_s \|X^{(k)} - X^*\|_{\infty} + q_s \|X^{(k-1)} - X^*\|_{\infty}$$

hay

$$\|X^{(k)} - X^*\|_{\infty} \leq \frac{q_s}{1 - p_s} \|X^{(k-1)} - X^*\|_{\infty}$$

- Đặt

$$\lambda = \max_i \frac{q_i}{1 - p_i} = \max_i \frac{\sum_{j=i}^n |\alpha_{ij}|}{1 - \sum_{j=1}^{i-1} |\alpha_{ij}|} \quad (10)$$

$$\Rightarrow \|X^{(k)} - X^*\|_{\infty} \leq \lambda \|X^{(k-1)} - X^*\|_{\infty} \quad (11)$$

- Ta sẽ chứng minh:  $\lambda \leq \|\alpha\|_{\infty} < 1$



- Ta có:  $p_i + q_i = \sum_{j=1}^n |\alpha_{ij}| \leq \|\alpha\|_\infty < 1$  theo giả thiết ta đặt ra ban đầu nên  $q_i \leq \|\alpha\|_\infty - p_i \Rightarrow \forall i = \overline{1, n}$  ta có:

$$\frac{q_i}{1 - p_i} < \frac{\|\alpha\|_\infty - p_i}{1 - p_i} \leq \frac{\|\alpha\|_\infty - p_i \|\alpha\|_\infty}{1 - p_i} = \|\alpha\|_\infty < 1$$

hay

$$\lambda = \max_i \frac{q_i}{1 - p_i} \leq \|\alpha\|_\infty < 1$$

- Với biểu thức (12), lặp k lần ta được:

$$\begin{aligned} \|X^{(k)} - X^*\|_\infty &\leq \lambda \|X^{(k-1)} - X^*\|_\infty \leq \dots \leq \lambda^k \|X^{(0)} - X^*\|_\infty \\ &\longrightarrow \|X^{(k)} - X^*\|_\infty \rightarrow 0 \quad \text{ khi } \quad k \rightarrow \infty \end{aligned}$$

hay  $\lim_{k \rightarrow \infty} X^{(k)} = X^*$  là điều phải chứng minh.

- Quay trở lại, ta sẽ chứng minh 2 công thức sai số (5) và (6)

- Xét phép lặp liên tiếp của  $X^{(k+1)}$  và  $X^{(k)}$ , tương tự như trên, ta có:

$$\|X^{(k+1)} - X^{(k)}\|_\infty \leq \lambda \|X^{(k)} - X^{(k-1)}\|_\infty \quad (12)$$

- Với p nguyên dương bất kì, ta có:

$$\begin{aligned} \|X^{(k+p)} - X^{(k)}\|_\infty &\leq \|X^{(k+p)} - X^{(k+p-1)}\|_\infty + \dots + \|X^{(k+1)} - X^{(k)}\|_\infty \\ &\leq (\lambda^p + \lambda^{p-1} + \dots + \lambda) \|X^{(k)} - X^{(k-1)}\|_\infty \\ &\leq \lambda \frac{1 - \lambda^p}{1 - \lambda} \|X^{(k)} - X^{(k-1)}\|_\infty \end{aligned} \quad (13)$$

- Cố định k, cho  $p \rightarrow \infty$  thì  $X^{k+p} \rightarrow X^*$ . Từ (13) có:

$$\|X^* - X^{(k)}\|_\infty \leq \frac{\lambda}{1 - \lambda} \|X^{(k)} - X^{(k-1)}\|_\infty$$

$\Rightarrow$  BĐT (5) được chứng minh.

- Với BĐT (6), ta lấy BĐT (12) với k là  $k + p - 1$

$$\|X^{(k+p)} - X^{(k+p-1)}\|_\infty \leq \lambda \|X^{(k+p-1)} - X^{(k+p-2)}\|_\infty$$

- Lặp p lần, ta được:

$$\|X^{(k+p)} - X^{(k+p-1)}\|_\infty \leq \lambda^p \|X^{(k)} - X^{(k-1)}\|_\infty \quad (14)$$

- Trong BĐT (14) chọn k = 1:

$$\|X^{(1+p)} - X^{(p)}\|_\infty \leq \lambda^p \|X^{(1)} - X^{(0)}\|_\infty$$

tiếp tục thay  $p = k - 1$  ta được:

$$\|X^{(k)} - X^{(k-1)}\|_{\infty} \leq \lambda^{k-1} \|X^{(1)} - X^{(0)}\|_{\infty}$$

thay vào BĐT (5), ta có:

$$\|X^{(k)} - X^{(*)}\|_{\infty} \leq \frac{\lambda}{1-\lambda} \|X^{(k)} - X^{(k-1)}\|_{\infty} \leq \frac{\lambda^k}{1-\lambda} \|X^{(1)} - X^{(0)}\|_{\infty}$$

$\Rightarrow$  BĐT (6) được chứng minh.

### • NHẬN XÉT CHUNG:

- Phương pháp Seidel có **độ chính xác cao** và **tốc độ hội tụ nhanh hơn** phương pháp lặp đơn (vì  $\lambda < \|\alpha\| < 1$ )

- Phương pháp Seidel **tiết kiệm bộ nhớ hơn** phương pháp lặp đơn vì:

▽ Phương pháp lặp đơn sử dụng  $2n$  ô nhớ cho  $X^{(k+1)}$  và  $X^{(k)}$ .

▽ Trong khi đó, phương pháp Seidel **chỉ cần  $n$  ô nhớ cho  $X^{(k)} \rightarrow X^{(k+1)}$** . Vì giả sử sau khi tính  $X_1^{(k+1)}$  thì không cần sử dụng  $X^{(k)}$ , lúc đó bỏ số  $X_1^{(k+1)}$  vào ô nhớ của  $X^{(k)}$  để tính  $X_1^{(k+2)}$ , ....cứ như vậy...

### • VÍ DỤ: : Tìm nghiệm gần đúng của phương trình sau bằng phương pháp lặp Seidel:

$$\begin{cases} 1.02x_1 - 0.05x_2 - 0.1x_3 = 0.795 \\ -0.11x_1 + 1.03x_2 - 0.05x_3 = 0.849 \\ -0.11x_1 + 0.12x_2 - 0.04x_3 = 1.398 \end{cases}$$

**Giải:**

$$\text{- Hệ} \Leftrightarrow \begin{cases} x_1 = -0.02x_2 + 0.05x_3 + 0.795 \\ x_2 = 0.11x_1 - 0.03x_3 + 0.849 \\ x_3 = 0.11x_1 + 0.12x_2 - 0.04x_3 + 1.398 \end{cases} \Rightarrow \begin{pmatrix} -0.02 & 0.05 & 0.1 \\ 0.11 & -0.03 & 0.05 \\ 0.11 & 0.12 & -0.04 \end{pmatrix}$$

$$\text{- Với } B_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0.11 & 0 & 0 \\ 0.11 & 0.12 & 0 \end{pmatrix}; \quad B_2 = \begin{pmatrix} -0.02 & 0.05 & 0.1 \\ 0 & -0.03 & 0.05 \\ 0 & 0 & -0.04 \end{pmatrix}$$

- Xét phép lặp  $X_{(k+1)} = B_1 X_{(k+1)} + B_2 X_{(k)} + g$

$$\Leftrightarrow \begin{cases} x_1^{(k+1)} = (-0.02x_1^k + 0.05x_2^k + 0.1x_3^k) + 0.795 \\ x_2^{(k+1)} = 0.11x_1^{k+1} + (-0.03x_2^k + 0.05x_3^k) + 0.849 \\ x_3^{(k+1)} = 0.11x_1^{k+1} + 0.12x_2^{(k+1)} - 0.04x_3^{(k+1)} + 1.398 \end{cases}$$

$k$	$x_1^k$	$x_2^k$	$x_3^k$
0	0.8	0.85	1.4
1	0.9615	0.999265	1.56767
2	0.9825	1.00548	1.564025

.....

## 4 Phương pháp lặp Gauss - Seidel

- **Định nghĩa** : Là phương pháp Seidel áp dụng cho hệ phương trình đại số tuyến tính với **ma trận đường chéo trội**.

⇒ Vậy thế nào là ma trận đường chéo trội ???

- **Ma trận đường chéo trội**: **Ma trận đường chéo trội** là ma trận thỏa mãn:

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}| \quad i = \overline{1, n}$$

- Trường hợp cụ thể, phương trình  $AX = B$ , trong đó ma trận hệ số  $A$  có đường chéo trội thì quá trình lặp Seidel được tính như sau: chọn xấp xỉ đầu  $X^{(0)} = (x_1^{(0)}, x_1^{(1)}, \dots, x_1^{(n)})^t$  thì:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[ \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} + \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right] + \frac{a_{i,n+1}}{a_{ii}} \quad i = \overline{1, n} \quad k = 0, 1, 2, \dots \quad (15)$$

- **VÍ DỤ** : Tìm nghiệm gần đúng của phương trình sau bằng phương pháp lặp Seidel - Gauss Seidel:

$$\begin{cases} 4x_1 - 2.4x_2 - 0.8x_3 = 8 \\ 0.9x_1 + 3x_2 - 1.5x_3 = 9 \\ 0.4x_1 - 0.8x_2 + 4x_3 = 20 \end{cases}$$

**Giải** : Dễ thấy hệ phương trình đã cho có dạng đường chéo trội

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}| \quad i = \overline{1, n}$$

nên dễ dàng đưa được về dạng  $X = \alpha X + \beta$ , trong đó:

$$\alpha = \begin{bmatrix} 0 & -0.6 & 0.2 \\ -0.3 & 0 & 0.5 \\ -0.1 & 0.2 & 0 \end{bmatrix}; \quad \beta = \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix}$$

- Tính được  $\|\alpha\|_{\infty} = 0.8 < 1$  (hoặc  $\|\alpha\|_1 = 0.8 < 1$ ) nên quá trình lặp Seidel hội tụ.

- Chọn xấp xỉ đầu  $X^{(0)} = \beta = (2, 3, 5)^t$

- Quá trình lặp thực hiện theo công thức BDT (3) hoặc (15), kết quả được ghi trong bảng dưới sau:

$k$	$x_1^k$	$x_2^k$	$x_3^k$
0	2	35	5
1	1.92	3.1924	5.044548
2	1.9093489	3.194952	5.0448056
3	1.909199	3.1949643	5.0448073
4	-0.4498708972	6.267367706	6.298460631
5	-0.5007284974	6.299448865	6.309962623
6	-0.5176767979	6.31028435	6.31382455
7	-0.5234057	6.31393395	6.315127367

- Xem  $X^{(3)}$  là nghiệm gần đúng phải tìm, ta có thể đánh giá sai số của  $X^{(3)}$  theo công thức (5):

$$\|X^{(7)} - X^*\|_\infty \leq \frac{\lambda}{1 - \lambda} \|X^{(7)} - X^{(6)}\|_\infty$$

$$\lambda = \max_i \frac{\sum_{j=i}^n |\alpha_{ij}|}{1 - \sum_{j=i}^n |\alpha_{ij}|} = \max(0.8; 0.515463; 0) = 0.8$$

- Còn

$$\|X^{(7)} - X^{(6)}\|_\infty = \max_i \|x_i^{(7)} - x_i^{(6)}\|_\infty = 0.0057289021$$

- Vậy

$$\|X^{(7)} - X^*\|_0 \leq \frac{0.8}{1 - 0.8} \cdot 0,0057289021 = 0.0229156084$$

## 5 Chạy chương trình trên máy

### ■ Nội dung chương trình bao gồm:

- ▽ Code chương trình
- ▽ File huong\_dan.txt
- ▽ File input\_A.txt
- ▽ File input\_B.txt
- ▽ File input\_x0.txt
- ▽ File output.txt

### ■ I/O

- ▽ Input:
  - Ma trận vuông về trái A ( hay là các hệ số về trái của hệ phương trình ĐSTT ).
  - Ma trận về phải B( hay hệ số về phải của hệ phương trình ĐSTT ).
  - Nghiệm xấp xỉ đầu  $X^{(0)}$ .
- ▽ Output:
  - Hiển thị trên màn hình giá trị x sau các vòng lặp.
  - Ghi giá trị nghiệm thỏa mãn điều kiện sai số vào file.

### ■ Hướng dẫn:

- ▽ Nhập ma trận vuông A ( về trái ) hay hệ số về trái của HPT vào file "input\_A.txt"
- ▽ Nhập ma trận vuông B ( về phải ) hay hệ số về phải của HPT vào file "input\_B.txt"
- ▽ Nhập giá trị xấp xỉ đầu của  $X^{(0)}$  vào file "input\_x0.txt"
- ▽ Kết quả cuối cùng của hệ phương trình ĐSTT sẽ được in ra màn hình console và file "output.txt"

### ■ Code:

```
#include<conio.h>
#include<stdio.h>
#include<math.h>
#include<stdlib.h>
#include<ctype.h>

int main()
{
    int n, i, j , dem=0;
    int traloi;
    printf("Ban da doc huong dan .(0 khi chua doc hoac 1 khi da doc) ?\n");
    scanf("\t%d",&traloi);
    if(traloi == 0)
    {
        printf("\tThe end ^-^");
        getch();
    }
}
```

```

    return 0;
}
printf("Cho biet cap cua ma tran da nhap n=");
scanf("%d",&n);
float a[n][n], b[n], x[n], y[n], d ,t, epsi;

// *****Lay du lieu tu file*****
// *****Lay he so ma tran vuong A*****
FILE *filein_A;
filein_A = fopen("input_A.txt", "r");
for ( i = 0; i < n; i++)
    for( j = 0; j < n; j++)
    {
        fscanf(filein_A, "%f", &a[i][j]);
    }
fclose(filein_A);

//*****Lay he so ma tran B*****
FILE *filein_B;
filein_B = fopen("input_B.txt","r");
for ( i = 0; i < n; i++)
{
    fscanf(filein_B, "%f", &b[i]);
}
fclose(filein_B);

//*****Lay gia tri ban dau x*****
FILE *filein_x0;
filein_x0=fopen("input_x0.txt","r");
for ( i=0;i<n;i++)
{
    fscanf(filein_x0,"%f",&x[i]);
}
fclose(filein_x0);

//*****In ra 2 ma tran*****
printf("\nMa tran ve trai ban da nhap A ");
printf("\n");
for( i = 0; i < n; i++)
{
    for ( j = 0; j < n; j++)
    {
        printf("%10.2f", a[i][j]);
    }
    printf("\n");
}
printf("\nMa tran ve phai ban da nhap B: \n");
for ( i = 0; i < n; i++)
{
    printf("%10.2f", b[i]);
    printf("\n");
}
printf("\n");

```

```
//*****In ra gia tri ban dau *****
printf("\nGia tri ban dau cua x ban da nhap: \n");
for ( i = 0; i < n; i++)
{
    printf("%10.2f", x[i]);
    printf("\n");
}
printf("\n");

//*****Nhap sai so epsilon *****
printf("\nNhap so epsilon:");
scanf("%f", &epsi);

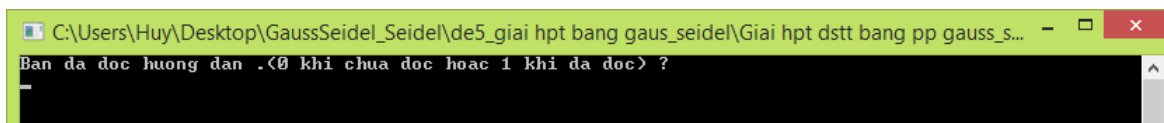
//*****GIAI HE PHUONG TRINH*****
FILE *fileout;
fileout = fopen("output.txt", "w");
printf("\nGia tri sau cac vong lap:\n");
do
{
    dem += 1;
    printf("Vong lap %d\n", dem);
    for (i = 0; i < n; i++)
    {
        y[i] = (b[i] / a[i][i]);
        for (j = 0; j < n; j++)
        {
            if (j == i)
                continue;
            y[i] = y[i] - ((a[i][j] / a[i][i]) * x[j]);
            x[i] = y[i];
        }
        printf("          x%d = %12.8f  ", i + 1, y[i]);
        t = fabs( d - x[i] );
        d = x[i];
    }
    printf("\n");
    // Danh gia sai so theo cong thuc 19 SGK trang 71
}while((t > epsi*fabs(x[dem])));

fprintf(fileout, "Gia tri vong lap cuoi cung la\n");
for (i = 0; i < n; i++)

//*****Ghi ket qua vao file*****
fprintf(fileout, "x%d = %10.8f ", i + 1, y[i]);
fclose(fileout);
getch();
return 0;
}
```

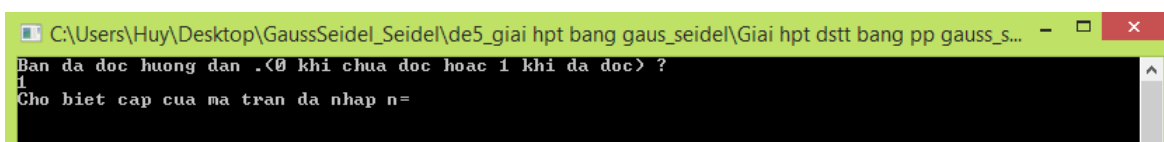
## ■ Phân tích chương trình:

- ▽ Khi chạy chương trình, màn hình console hiện lên dòng chữ: "Bạn đã đọc hướng dẫn chưa?". Nhấn 1 để tiếp tục chương trình, nhấn 0 để dừng chương trình.



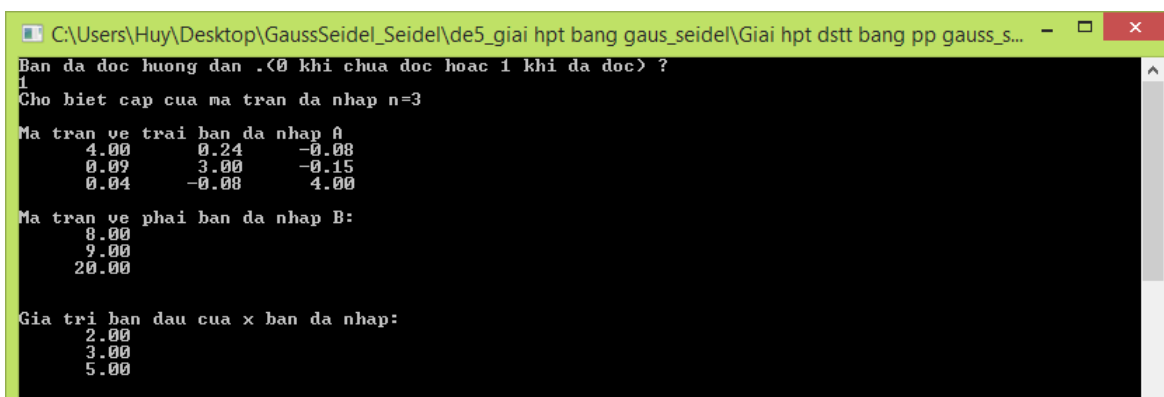
Hình 1: Yêu cầu người dùng đọc hướng dẫn trước khi chạy chương trình.

- ▽ Nhập vào cấp ma trận A tương ứng với cấp ma trận đã nhập trong file "input\_A.txt".



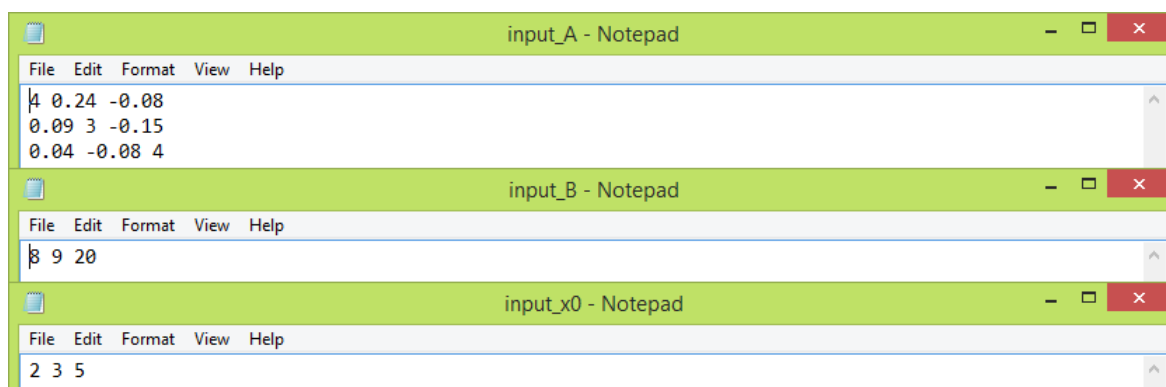
Hình 2: Nhập cấp ma trận.

- ▽ Sau đó, màn hình console sẽ hiện ra giá trị hệ số trong 2 ma trận A, B và các giá trị của nghiệm xấp xỉ đầu  $X^{(0)}$  bằng cách input dữ liệu từ các file.txt tương ứng.



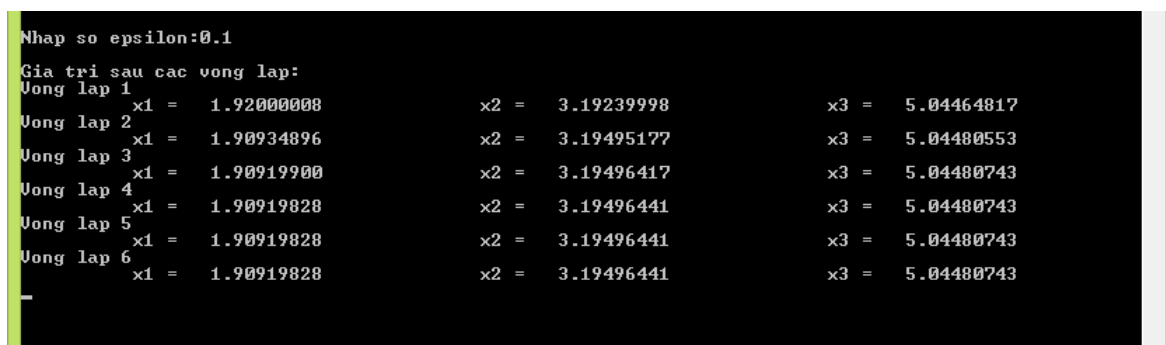
Hình 3: Hiện thị ma trận A, B và nghiệm xấp xỉ đầu  $X^{(0)}$





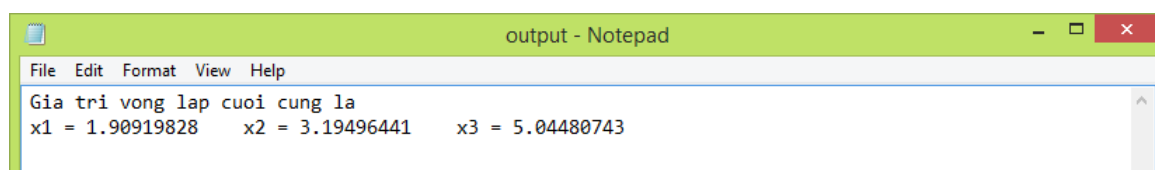
Hình 4: File "input\_A.txt", "input\_B.txt", "input\_x0.txt"

- ▽ Nhập sai số epsilon. Sai số nhập càng nhỏ thì kết quả trả về của các ẩn càng chính xác. Sau đó, màn hình console sẽ hiện ra tất cả các kết quả của từng vòng lặp tương ứng. Số vòng lặp nhiều hay ít phụ thuộc vào sai số epsilon vừa nhập từ bàn phím.



Hình 5: Nhập sai số epsilon và hiển thị kết quả

- ▽ Kết quả cuối cùng sẽ được lưu vào file "output.txt".



Hình 6: Lưu kết quả vào file "output.txt"

## 6 Nhóm báo cáo

- Nhóm báo cáo gồm 3 thành viên:

1. Phan Huy Hoàng ( nhóm trưởng ) - Tìm hiểu đề tài, viết báo cáo.
2. Trần Thị Thanh Tâm - Tìm hiểu đề tài, thuyết trình.
3. Nguyễn Văn Thành Dũng - Viết code chương trình.

- Các công việc mà nhóm đã thực hiện khi tìm hiểu đề tài:

### ■ Đọc - hiểu đề tài :

- ▽ - Nhóm đã dựa vào những kiến thức có sẵn trong giáo trình Giải tích số, kết hợp với đó là tìm hiểu thêm các thông tin ở trên mạng phục vụ cho bài thuyết trình và viết chương trình trên máy.
- ▽ - Bên cạnh đó, thì trong 1 tuần tìm hiểu, nhóm cũng đã họp để thống nhất đề tài, bổ sung những phần lí thuyết còn thiếu, phân chia đầu việc chuẩn bị cho bài thuyết trình trên lớp.

### ■ Thuyết trình - Chạy code chương trình:

- ▽ Nhóm cũng đã chuẩn bị phần slide và code để trình chiếu, tuy còn chưa được hoàn chỉnh. Dưới sự góp ý của giáo viên bộ môn, nhóm cũng đã rút kinh nghiệm và chỉnh sửa lại trong phần báo cáo lần này.

### ■ Viết báo cáo: Trong báo cáo bao gồm những phần chính như sau:

- ▽ Lý thuyết chung của phương pháp lặp Seidel và Gauss - Seidel.
- ▽ Chạy chương trình trên máy.
- ▽ Hoạt động của nhóm, phân chia đầu việc từng thành viên.
- ▽ Tổng kết, ưu nhược điểm của phương pháp.
- ▽ Code chương trình.

## 7 Kết luận

- Qua quá trình tìm hiểu đề tài, các thành viên trong nhóm cũng đã có kiến thức cơ bản về phương pháp lặp Seidel và Gauss-Seidel.

- Ưu điểm:

- ▽ Đối với phương pháp Gauss-Seidel hay còn được gọi là phương pháp lặp cải tiến có **độ chính xác cao** và **tốc độ hội tụ nhanh hơn** phương pháp lặp đơn (vì  $\lambda < \|\alpha\| < 1$ )
- ▽ Phương pháp Seidel **tiết kiệm bộ nhớ hơn** phương pháp lặp đơn vì chỉ sử dụng  $n$  ô nhớ thay vì  $2n$  ô nhớ cho  $X^{(k+1)}$  và  $X^{(k)}$ .
- ▽ **Dễ lập trình.** Chuẩn  $\|\alpha\|$  càng gần 1 thì tốc độ hội tụ càng nhanh, nghiệm càng chính xác hơn.

- Nhược điểm:

- ▽ Không phải tất cả các phương trình có nghiệm đều hội tụ nên việc sử dụng phương pháp lặp Seidel và Gauss - Seidel là không khả thi. Điều kiện đủ là chuẩn  $\|\alpha\|$  nằm trong khoảng  $(0, 1)$
- ▽ Nếu hệ số hội tụ quá bé ( $\|\alpha\| \ll 1$ , gần 0 hơn) thì ma trận sẽ lâu hội tụ về kết quả. Khi đó việc sử dụng phương pháp lặp đơn để giải hệ phương trình ĐSTT là tối ưu hơn.

—HẾT—