



THE UNIVERSITY OF TEXAS
AT AUSTIN

CS383C NUMERICAL ANALYSIS

Homework 03

Edited by L^AT_EX

Department of Computer Science

STUDENT

Jimmy Lin

xl5224

COURSE COORDINATOR

Robert A. van de Geijn

UNIQUE NUMBER

53180

RELEASE DATE

Sep. 18 2014

DUE DATE

May. 23 2014

TIME SPENT

10 hours

September 21, 2014

Exercise 2. Show that if H is a reflector, then

2.1 $HH = I$

Since H is a reflector, we have

$$H = I - 2uu^H \quad (1)$$

where u is unit vector ($\|u\|_2^2 = 1$).

$$\begin{aligned} HH &= (I - 2uu^H)(I - 2uu^H) \\ &= I \cdot I - 2uu^H - 2uu^H + 4(uu^H)(uu^H) \\ &= I - 4uu^H + 4\|u\|_2^2 uu^H \\ &= I - 4uu^H + 4uu^H \\ &= I \end{aligned} \quad (2)$$

Lemma 1. For arbitrary vector $a \in \mathbb{R}^n$, $b \in \mathbb{R}^n$, $c \in \mathbb{R}^n$, $ab^T c = (b^T c)a$.

Proof.

$$ab^T c = \begin{pmatrix} a_0 b^T \\ a_1 b^T \\ \vdots \\ a_{n-1} b^T \end{pmatrix} c = \begin{pmatrix} a_0 b^T c \\ a_1 b^T c \\ \vdots \\ a_{n-1} b^T c \end{pmatrix} = \begin{pmatrix} (b^T c) a_0 \\ (b^T c) a_1 \\ \vdots \\ (b^T c) a_{n-1} \end{pmatrix} = (b^T c) \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} = (b^T c) a \quad (3)$$

□

Lemma 2. $(uu^H)(uu^H) = \|u\|_2^2 uu^H$

Proof.

$$\begin{aligned} (uu^H)(uu^H) &= (uu^H u) u^H \\ &= ((u^H u) u) u^H \\ &= (\|u\|_2^2 u) u^H \\ &= \|u\|_2^2 uu^H \end{aligned} \quad (4)$$

□

2.2 $H = H^H$

$$\begin{aligned} H^H &= (I - 2uu^H)^H \\ &= I^H - (2uu^H)^H \\ &= I - 2(u^H)^H u^H \\ &= I - 2uu^H \\ &= H \end{aligned} \quad (5)$$

2.3 $HH^H = I$

In terms of (5), multiply both sides with H and then reuse conclusion in (2)

$$H^H H = HH = I \quad (6)$$

Exercise 4. Show that if $x \in \mathbb{R}^n$, $v = x \mp \|x\|_2 e_0$ and $\tau = v^T v / 2$, then

$$(I - \frac{1}{\tau} v v^T) = \pm \|x\|_2 e_0$$

We start from the reflector $I - \frac{1}{\tau} v v^T$ on x ,

$$\begin{aligned}
& (I - \frac{1}{\tau} v v^T) x \\
&= \left(I - \frac{2 v v^T}{v^T v} \right) x \\
&= \left(I - \frac{2(x \mp \|x\|_2 e_0)(x \mp \|x\|_2 e_0)^T}{(x \mp \|x\|_2 e_0)^T (x \mp \|x\|_2 e_0)} \right) x \\
&= \left(I - 2 \frac{x x^T \mp \|x\|_2 (x e_0^T + e_0 x^T) + \|x\|_2^2 e_0 e_0^T}{2\|x\|_2^2 \mp 2\|x\|_2 e_0^T x} \right) x \\
&= \left(I - \frac{x x^T \mp \|x\|_2 (x e_0^T + e_0 x^T) + \|x\|_2^2 e_0 e_0^T}{\|x\|_2^2 \mp \|x\|_2 e_0^T x} \right) x \\
&= \frac{\|x\|_2^2 \mp \|x\|_2 e_0^T x - x x^T \pm \|x\|_2 (x e_0^T + e_0 x^T) - \|x\|_2^2 e_0 e_0^T}{\|x\|_2^2 \mp \|x\|_2 e_0^T x} x \\
&= \frac{\|x\|_2^2 x \mp \|x\|_2 e_0^T x x - x x^T x \pm \|x\|_2 (x e_0^T + e_0 x^T) x - \|x\|_2^2 e_0 e_0^T x}{\|x\|_2^2 \mp \|x\|_2 e_0^T x} \\
&= \frac{(\|x\|_2^2 x - x x^T x) + (\mp \|x\|_2 e_0^T x x \pm \|x\|_2 x e_0^T x) \pm \|x\|_2 e_0 x^T x - \|x\|_2^2 e_0 e_0^T x}{\|x\|_2^2 \mp \|x\|_2 e_0^T x} \\
&= \frac{\pm \|x\|_2 e_0 x^T x - \|x\|_2^2 e_0 e_0^T x}{\|x\|_2^2 \mp \|x\|_2 e_0^T x} \\
&= \frac{\pm e_0 x^T x - \|x\|_2 e_0 e_0^T x}{\|x\|_2 \mp e_0^T x} \\
&= \frac{\pm (\|x\|_2^2 e_0 - \|x\|_2 (e_0^T x) e_0)}{\|x\|_2 \mp e_0^T x} \\
&= \frac{\pm (\|x\|_2 - (e_0^T x)) \|x\|_2 e_0}{\|x\|_2 \mp e_0^T x} \\
&= \pm \|x\|_2 e_0
\end{aligned} \tag{7}$$

Note that above derivation frequently makes use of the lemma 1.

Exercise 5. Complex

Exercise 6. Matrix Equivalence

We start from right hand side

$$\begin{aligned}
 RHS &= I - \frac{1}{\tau_1} \begin{pmatrix} 0 \\ 1 \\ u_2 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ u_2 \end{pmatrix}^H \\
 &= \left(I - \frac{1}{\tau_1} \begin{array}{c|c} 0 & 0 \\ \hline \begin{pmatrix} 1 \\ u_2 \end{pmatrix} & \begin{pmatrix} 1 \\ u_2 \end{pmatrix}^H \end{array} \right) \\
 &= \left(I - \begin{array}{c|c} 0 & 0 \\ \hline 0 & \frac{1}{\tau_1} \begin{pmatrix} 1 \\ u_2 \end{pmatrix} \begin{pmatrix} 1 \\ u_2 \end{pmatrix}^H \end{array} \right) \\
 &= \left(\begin{array}{c|c} I & 0 \\ \hline 0 & I - \frac{1}{\tau_1} \begin{pmatrix} 1 \\ u_2 \end{pmatrix} \begin{pmatrix} 1 \\ u_2 \end{pmatrix}^H \end{array} \right) \\
 &= LHS
 \end{aligned} \tag{8}$$

Exercise 11. Expensive Algorithm

As indicated by **Theorem 10** in the note, we have cost of the algorithm in Figure 6 for $A \in \mathbb{C}^{m \times n}$

$$C_{FormQ}(m, n) = 2mn^2 - \frac{2}{3}n^3 \tag{9}$$

For $m = n$, the cost can be simplified as

$$C_{FormQ}(A) = \frac{4}{3}n^3 = \mathbb{O}(n^3) \tag{10}$$

However, if we accumulate Q by using n householder transformation with

$$Q = (\dots((IH_0)H_1)\dots H_{n-1}) \tag{11}$$

Then the cost we have is at least

$$C_{accumulation}(A) = n^3 \cdot (n - 1) = \mathbb{O}(n^4) \tag{12}$$

where n^3 comes from each one matrix multiplication, and $n - 1$ comes from the total number of householder matrix H_i ($i \in \mathbb{Z}, i \in [0, n - 1]$).

Comparing formula (10) and (12), it is obvious that the accumulation method is much more expensive than algorithm in Figure 6.