L. Vandenberghe 11/10/14
EE133A

Midterm solutions

Problem 1. The *trace* of a square matrix is the sum of its diagonal elements. What is the complexity (number of flops for large n) of computing the following quantities?

1. The trace of AB, where A and B are $n \times n$ matrices.
2. The trace of C^{-1}, where the $n \times n$ matrix C is lower triangular and nonsingular.

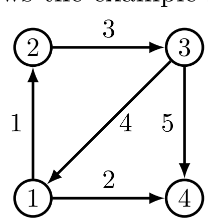
Solution.

1. $2n^2$ flops. The trace of AB is $\sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} B_{ji}$. This requires n^2 multiplications and $n^2 - 1$ additions.
2. $2n$ flops. The diagonal elements of C^{-1} are $1/C_{ii}$. Computing the trace of C^{-1} requires n divisions and $n - 1$ additions.

Problem 2. The node-arc incidence matrix of a directed graph with n nodes and m arcs is the $n \times m$ matrix A with elements

$$A_{ij} = \begin{cases} -1 & \text{if arc } j \text{ leaves node } i \\ 1 & \text{if arc } j \text{ enters node } i \\ 0 & \text{otherwise.} \end{cases}$$

The figure shows the example from the lecture. (Note this is only an *example*.)



$$A = \begin{bmatrix} -1 & -1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Which of the following statements is correct? Choose one and explain your answer.

1. All node-arc incidence matrices are right invertible.
2. No node-arc incidence matrices are right invertible.
3. Some node-arc incidence matrices are right invertible, some are not. It depends on the graph.

Solution. The rows of a node-arc incidence matrix are linearly dependent. Each colum contains one element equal to 1, one element equal to -1, and the other elements are zero, so the sum of the rows are zero ($\mathbf{1}^T A = 0$). A node-arc incidence matrix is therefore never right invertible (lecture 4, page 26).

To see this directly, if a right inverse X existed, we would have the following contradiction. On the one hand, $\mathbf{1}^T AX = 0$ because $\mathbf{1}^T A = 0$. On the other hand, $\mathbf{1}^T AX = \mathbf{1}^T$ because $AX = I$.

Problem 3. Define a sequence of matrices A_k for $k = 0, 1, 2 \ldots$, as follows: $A_0 = 1$ and for $k \geq 1$,

$$A_k = \begin{bmatrix} A_{k-1} & A_{k-1} \\ -A_{k-1} & A_{k-1} \end{bmatrix}.$$

Therefore A_k is a matrix of size $2^k \times 2^k$. For $k = 1, 2, 3$ we have

$$A_1 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \end{bmatrix}.$$

Show that all matrices A_k in the sequence (not just the first four) are invertible. What is the inverse of A_k?

Solution. The columns of A_k are mutually orthogonal and have norm $\sqrt{n}$ where $n = 2^k$:

$$A_k^T A_k = nI.$$

Therefore A_k is invertible with $A_k^{-1} = (1/n)A_k^T$.

This can be shown by induction. For $k = 0$, we have $A_0^T A_0 = 1$. Suppose $A_{k-1}^T A_{k-1} = 2^{k-1}I$. Then

$$\begin{aligned} A_k^T A_k &= \begin{bmatrix} A_{k-1} & A_{k-1} \\ -A_{k-1} & A_{k-1} \end{bmatrix}^T \begin{bmatrix} A_{k-1} & A_{k-1} \\ -A_{k-1} & A_{k-1} \end{bmatrix} \\ &= \begin{bmatrix} 2A_{k-1}^T A_{k-1} & 0 \\ 0 & 2A_{k-1}^T A_{k-1} \end{bmatrix} \\ &= \begin{bmatrix} 2^k I & 0 \\ 0 & 2^k I \end{bmatrix}. \end{aligned}$$

Problem 4. Let Q be an $n \times n$ orthogonal matrix, partitioned as

$$Q = \left[\begin{array}{cc} Q_1 & Q_2 \end{array} \right]$$

where Q_1 has size $n \times m$ and Q_2 has size $n \times (n-m)$, with $0 < m < n$. Consider the $n \times n$ matrix

$$A = Q_1 Q_1^T - Q_2 Q_2^T.$$

1. Show that A can also be written in the following two forms:

$$A = 2Q_1 Q_1^T - I, \qquad A = I - 2Q_2 Q_2^T.$$

2. Show that A is orthogonal.

3. Describe an efficient method for solving $Ax = b$ and give the complexity of the method (the dominant term(s) in the flop count, including the coefficient). If you know several methods, give the method that has the lowest complexity when $m < n/2$.

Solution.

1. Since Q is orthogonal we have $Q^T Q = QQ^T = I$. Expanding $QQ^T = I$ gives

$$QQ^T = \left[\begin{array}{cc} Q_1 & Q_2 \end{array} \right] \left[\begin{array}{cc} Q_1 & Q_2 \end{array} \right]^T = Q_1 Q_1^T + Q_2 Q_2^T = I.$$

Therefore

$$A = Q_1 Q_1^T - Q_2 Q_2^T = Q_1 Q_1^T - (I - Q_1 Q_1^T) = 2Q_1 Q_1^T - I$$

and

$$A = Q_1 Q_1^T - Q_2 Q_2^T = (I - Q_2 Q_2^T) - Q_2 Q_2^T = I - 2Q_2 Q_2^T.$$

2. We have to verify that $A^T A = I$, using any of the three expressions. For example, if we use the third expression,

$$\begin{aligned} A^T A &= (I - 2Q_2 Q_2^T)(I - 2Q_2 Q_2^T) \\ &= I - 2Q_2 Q_2^T - 2Q_2 Q_2^T + 4Q_2 Q_2^T Q_2 Q_2^T \\ &= I - 2Q_2 Q_2^T - 2Q_2 Q_2^T + 4Q_2 Q_2^T \\ &= I. \end{aligned}$$

On line 3 we used the property that Q_2 has orthonormal columns ($Q_2^T Q_2 = I$).

3. Since A is orthogonal and symmetric, we have $A^{-1} = A^T = A$ and we can solve the equation by making the product $x = Ab$, using one of the three expressions. If $m < n - m$ the best method is to use $A = 2Q_1 Q_1^T - I$ because this involves products with Q_1 and its transpose. The other expressions involve products with Q_2 and its transpose, and since Q_2 has more columns than Q_1, this will be more expensive.

We compute the matrix-vector product

$$x = (2Q_1 Q_1^T - I)b = 2Q_1(Q_1^T b) - b$$

in three steps.

- Compute the product $y = Q_1^T b$. $2mn$ flops.
- Compute the product $z = Q_1 y$. $2mn$ flops.
- Compute the vector $x = 2z - b$. $2n$ flops.

The total is $4mn$ flops.

Problem 5. Explain how you can solve the following problem using an LU factorization of A. Given a nonsingular $n \times n$ matrix A and two n-vectors b and c, find an n-vector x and a scalar y such that

$$Ax + yb = c \quad \text{and} \quad \|x\|^2 = 1.$$

We assume that $b \neq 0$ and $\|A^{-1}c\| < 1$. Clearly state every step in your algorithm. How many solutions (x, y) are there?

Solution. We express x as $x = A^{-1}c - yA^{-1}b$ and determine y by solving the equation

$$\|A^{-1}c - yA^{-1}b\|^2 = 1.$$

Expanding the norm gives a quadratic equation in y:

$$\|A^{-1}c\|^2 - 2y(A^{-1}c)^T(A^{-1}b) + y^2\|A^{-1}b\|^2 = 1.$$

This quadratic equation has two roots because $\|A^{-1}c\| < 1$ and $\|A^{-1}b\|^2 > 0$.

The algorithm is as follows.

- Compute the LU factorization $A = PLU$.
- Compute $u = A^{-1}b$ using the standard method:
  - Compute $w = P^T b$.
  - Solve $Lz = w$ for z by forward substitution.
  - Solve $Uu = z$ for u by back substitution.
- Compute $v = A^{-1}c$ in the same manner.
- Compute the three scalars

  $$\alpha = \|u\|^2, \qquad \beta = -2u^T v, \qquad \gamma = \|v\|^2 - 1,$$

  find a root of the quadratic equation $\alpha y^2 + \beta y + \gamma = 0$, and take $x = v - yu$.