Math 273 Midterm II Solutions

1. The statement inside the first `for` loop takes 3 flops per repetition, and is repeated $n-1$ times for a total of $3n-3$ flops. The division in the statement preceding the loop is 1 more, giving a total of $3n-2$ flops. The second `for` loop and its preceding statement have identical structure, and take another $3n-2$ flops.

The grand total for the tridiagonal Cholesky algorithm, then, counting in the flops for `CholTrid`, is $(3n-2)+(3n-2)+4n+\mathcal{O}(1) = 10n+\mathcal{O}(1)$ flops. Hence it is less efficient than the tridiagonal algorithm of §6.2, which requires (page 218) only $8n$ flops.

2. To perform Gaussian elimination, we subtract $-5/12$ times row 1 from row 2. This is achieved, as we learned from §§6.2-3 and the in-class demonstration `tdludemo.m`, by left-multiplying the system by the multiplier matrix

$$M = \begin{bmatrix} 1 & 0 \\ 5/12 & 1 \end{bmatrix}$$

The resulting system $MAx = Mb$ is

$$\begin{bmatrix} 12 & -4 \\ 0 & 13/3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 8 \\ 13/3 \end{bmatrix}$$

so $MA \equiv U$ is upper triangular and you compute the solution by back-substitution: $x_2 = (13/3)/(13/3) = 1$ and $x_1 = (8-(-4)(1))/12 = 1$.

Since $MA = U$, it follows that $A = M^{-1}U$. Now, U is upper triangular, and because M is unit lower triangular, so is M^{-1} and the unit lower triangular factor is given by

$$L = M^{-1} = \begin{bmatrix} 1 & 0 \\ -5/12 & 1 \end{bmatrix}$$

3. Think of a vector of data capture times `t=linspace(0,5,101)`; a vector of approximate derivatives of current at times `t(2:100)` can be computed by central differences with

```
>> dt = 0.05;
>> didt = (i(3:101)-i(1:99))/(2*dt);
```

(Forward differences could also be used, with some loss of accuracy.) Then the least squares computation of L and R is

```
>> A = [didt i(2:100)];
>> c = A \ E(2:100);
>> L = c(1); R = c(2);
```

4. The gradient of f is

$$\nabla f(x_1, x_2) = \begin{bmatrix} 4x_1 - 2x_2 + 1 \\ 2(x_2 - x_1) \end{bmatrix}$$

so the steepest descent vector at $\mathbf{x}_0 = (1,1)^T$ is $\mathbf{p} = -\nabla f(\mathbf{x}_0) = (-3,0)^T$.

The minimizing λ can be computed by substituting

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathbf{x}_0 + \lambda \mathbf{p} = \begin{bmatrix} 1 - 3\lambda \\ 1 \end{bmatrix}$$

into f and minimizing the resulting quadratic function of λ; the solution is easier, however, if you recall from the in-class demonstration `SDQ.m` that the gradient at the minimizing point is perpendicular to the search direction:

$$\nabla f(\mathbf{x}_0 + \lambda \mathbf{p}) \cdot \mathbf{p} = \begin{bmatrix} 4(1-3\lambda) - 2(1) + 1 \\ 2(1 - (1 - 3\lambda)) \end{bmatrix} \cdot \begin{bmatrix} -3 \\ 0 \end{bmatrix} = 0.$$

This equation says $-3(3 - 12\lambda) = 0$, so $\lambda = 1/4$.

5. Kepler's Equation can be rewritten

$$E = t + e \sin E,$$

so a simple starting guess for E would be $E_0 = t = 0.8$, since e is small and $\sin E < 1$. You get a rather better starting guess by putting t in for E on the right hand side, and taking $E_0 = t + e \sin t = 0.8 + .2056 \sin(0.8) = 0.9475$.

Writing the equation in the form

$$g(E) = E - t - e \sin E = 0$$

we have $g'(E) = 1 - e \cos E$ and the Newton correction is

$$E_1 = E_0 - \frac{g(E_0)}{g'(E_0)} = \frac{.9475 - .2056 \sin(.9475)}{1 - .2056 \cos(.9475)} = .9696.$$

(The starting guess $E_0 = 0.8$ leads to $E_1 = .9721$.)