Homework 5

Elizabeth Hunt

November 1, 2023

1 Question One

See LIZFCM → Matrix Routines → `lu decomp` & `bsubst`.
The test `UTEST(matrix, lu_decomp)` is a unit test for the `lu_decomp` routine, and `UTEST(matrix, bsubst)` verifies back substitution on an upper triangular 3×3 matrix with a known solution that can be verified manually.
Both can be found in `tests/matrix.t.c`.

2 Question Two

Unless the following are met, the resulting solution will be garbage.

1. The matrix U must be not be singular.
2. U must be square (or it will fail the `assert`).
3. The system created by $Ux = b$ must be consistent.
4. U is (quite obviously) in upper-triangular form.

Thus, the actual calculation performing the LU decomposition (in `lu_decomp`) does a sanity check for 1-3 will fail an assert, should a point along the diagonal (pivot) be zero, or the matrix be non-factorable.

3 Question Three

See LIZFCM → Matrix Routines → `fsubst`.
`UTEST(matrix, fsubst)` verifies forward substitution on a lower triangular 3×3 matrix with a known solution that can be verified manually.

4 Question Four

See LIZFCM → Matrix Routines → `gaussian_elimination` and `solve_gaussian_elimination`.

5 Question Five

See LIZFCM → Matrix Routines → `m_dot_v`, and the `UTEST(matrix, m_dot_v)` in `tests/matrix.t.c`.

6 Question Six

See `UTEST(matrix, solve_gaussian_elimination)` in `tests/matrix.t.c`, which generates a diagonally dominant 10×10 matrix and shows that the solution is consistent with the initial matrix, according to the steps given. Then, we do a dot product between each row of the diagonally dominant matrix and the solution vector to ensure it is near equivalent to the input vector.

7 Question Seven

See `UTEST(matrix, solve_matrix_lu_bsubst)` which does the same test in Question Six with the solution according to `solve_matrix_lu_bsubst` as shown in the Software Manual.

8 Question Eight

No, since the time complexity for Gaussian Elimination is always less than that of the LU factorization solution by $O(n^2)$ operations (in LU factorization we perform both backwards and forwards substitutions proceeding the LU decomp, in Gaussian Elimination we only need back substitution).

9 Question Nine, Ten

See LIZFCM Software manual and shared library in `dist` after compiling.