

Solving Eigenvalue Problems with LU Iterations:

The basic idea is to perform an LU decomposition, , multiply the factors in the reverse order $U*L$, and iterate. The resulting matrix will be converging to diagonal matrix, and its diagonal elements are converging to the eigenvalues of the matrix.

MATLAB CODE:

```
function [eigvals, nit] = luiters(A, etol, maxit)
% This function calculates the eigenvalues of a given matrix
% INPUT:
%   A = the matrix
%   etol = the error tolerance
%   maxit = max. no. of iterations
% OUTPUT:
%   eigvals = the eigenvalues of the matrix
%   nit = actual no. of iterations

nit = 0;
while nit < maxit
    nit = nit + 1;
    tempdiag = diag(A);
    [L, U] = lu(A); %LU factorization
    A = U*L;
    if (norm((diag(A)-tempdiag), 2)) <= etol, break, end
end
eigvals = diag(A);
end
```

```
>> A
A =

     6     4     3
     4    10     6
     3     6     8

>> [val, nit] = luiters(B, 10^(-10), 100)

val =

    17.2904
     3.8704
     2.8392

nit = 70
```