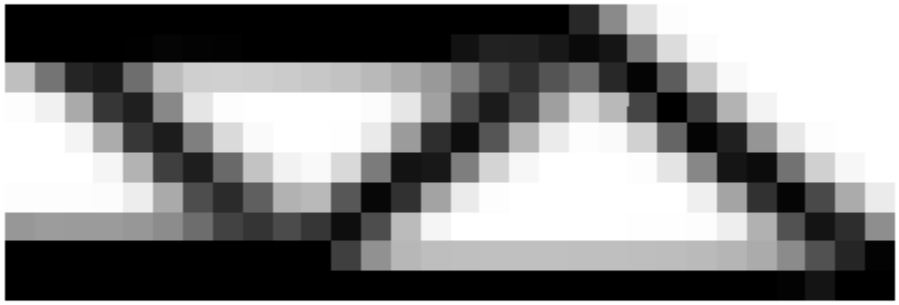
```
%%%% A 99 LINE TOPOLOGY OPTIMIZATION CODE BY OLE SIGMUND, JANUARY 2000 %%%
%%%% CODE MODIFIED FOR INCREASED SPEED, September 2002, BY OLE SIGMUND %%%
nelx = 30; % Example values, adjust as needed
nely = 10;
volfrac = 0.5;
penal = 3.0;
rmin = 1.5;

% INITIALIZE
x(1:nely, 1:nelx) = volfrac;
loop = 0;
change = 1.;

% START ITERATION
while change > 0.01
    loop = loop + 1;
    xold = x;

    % FE-ANALYSIS
    [U, K] = FE(nelx, nely, x, penal); % Capture K matrix from FE

    % OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
    [KE] = lk;
    c = 0.;
    for ely = 1:nely
        for elx = 1:nelx
            n1 = (nely+1)*(elx-1) + ely;
            n2 = (nely+1)*elx + ely;
            Ue = U([2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1;
2*n1+2], 1);
            c = c + x(ely, elx)^penal * Ue' * KE * Ue;
            dc(ely, elx) = -penal * x(ely, elx)^(penal-1) * Ue' * KE * Ue;
        end
    end

    % FILTERING OF SENSITIVITIES
    [dc] = check(nelx, nely, rmin, x, dc);

    % DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
    [x] = OC(nelx, nely, x, volfrac, dc);

    % PRINT RESULTS
    change = max(max(abs(x - xold)));
    disp([' It.: ' sprintf('%4i', loop) ' Obj.: ' sprintf('%10.4f', c) ...
          ' Vol.: ' sprintf('%6.3f', sum(sum(x)) / (nelx * nely)) ...
          ' ch.: ' sprintf('%6.3f', change)])

    % PLOT DENSITIES (if needed)
    colormap(gray); imagesc(-x); axis equal; axis tight; axis off;
pause(1e-6);
end
```

```
%%%%%%%%%% OPTIMALITY CRITERIA UPDATE %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [xnew]=OC(nelx,nely,x,volfrac,dc)
l1 = 0; l2 = 100000; move = 0.2;
while (l2-l1 > 1e-4)
  lmid = 0.5*(l2+l1);
  xnew = max(0.001,max(x-move,min(1.,min(x+move,x.*sqrt(-dc./lmid)))));
  if sum(sum(xnew)) - volfrac*nelx*nely > 0
    l1 = lmid;
  else
    l2 = lmid;
  end
end
end
%%%%%%%%%% MESH-INDEPENDENCY FILTER %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [dcn]=check(nelx,nely,rmin,x,dc)
dcn=zeros(nely,nelx);
for i = 1:nelx
  for j = 1:nely
    sum=0.0;
    for k = max(i-floor(rmin),1):min(i+floor(rmin),nelx)
      for l = max(j-floor(rmin),1):min(j+floor(rmin),nely)
        fac = rmin-sqrt((i-k)^2+(j-l)^2);
        sum = sum+max(0,fac);
        dcn(j,i) = dcn(j,i) + max(0,fac)*x(l,k)*dc(l,k);
      end
    end
    dcn(j,i) = dcn(j,i)/(x(j,i)*sum);
  end
end
end
%%%%%%%%%% FE-ANALYSIS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [U, K] = FE(nelx, nely, x, penal)
    [KE] = lk;
    K = sparse(2*(nelx+1)*(nely+1), 2*(nelx+1)*(nely+1));
    F = sparse(2*(nely+1)*(nelx+1), 1);
    U = zeros(2*(nely+1)*(nelx+1), 1);

    for elx = 1:nelx
        for ely = 1:nely
            n1 = (nely+1)*(elx-1) + ely;
            n2 = (nely+1)*elx + ely;
            edof = [2*n1-1; 2*n1; 2*n2-1; 2*n2; 2*n2+1; 2*n2+2; 2*n1+1;
2*n1+2];
            K(edof, edof) = K(edof, edof) + x(ely, elx)^penal * KE;
        end
    end
% DEFINE LOADS AND SUPPORTS (HALF MBB-BEAM)
F(2, 1) = -1;
    fixeddofs = union([1:2:2*(nely+1)], [2*(nelx+1)*(nely+1)]);
    alldofs = [1:2*(nely+1)*(nelx+1)];
    freedofs = setdiff(alldofs, fixeddofs);

    % SOLVING
    U(freedofs, :) = K(freedofs, freedofs) \ F(freedofs, :);
```

```
    U(fixeddofs, :) = 0;
end
%%%%%%%%%% ELEMENT STIFFNESS MATRIX %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [KE]=lk
E = 1.;
nu = 0.3;
k=[ 1/2-nu/6   1/8+nu/8 -1/4-nu/12 -1/8+3*nu/8 ...
   -1/4+nu/12 -1/8-nu/8  nu/6       1/8-3*nu/8];
KE = E/(1-nu^2)*[ k(1) k(2) k(3) k(4) k(5) k(6) k(7) k(8)
                  k(2) k(1) k(8) k(7) k(6) k(5) k(4) k(3)
                  k(3) k(8) k(1) k(6) k(7) k(4) k(5) k(2)
                  k(4) k(7) k(6) k(1) k(8) k(3) k(2) k(5)
                  k(5) k(6) k(7) k(8) k(1) k(2) k(3) k(4)
                  k(6) k(5) k(4) k(3) k(2) k(1) k(8) k(7)
                  k(7) k(4) k(5) k(2) k(3) k(8) k(1) k(6)
                  k(8) k(3) k(2) k(5) k(4) k(7) k(6) k(1)];
end
% Total it takes 72 iteration.
```

```
 It.:    1 Obj.:   984.5548 Vol.:  0.500 ch.:  0.200
 It.:    2 Obj.:   580.4116 Vol.:  0.500 ch.:  0.200
 It.:    3 Obj.:   419.0640 Vol.:  0.500 ch.:  0.200
 It.:    4 Obj.:   357.2235 Vol.:  0.500 ch.:  0.194
 It.:    5 Obj.:   337.8750 Vol.:  0.500 ch.:  0.125
 It.:    6 Obj.:   327.5757 Vol.:  0.500 ch.:  0.141
 It.:    7 Obj.:   319.9654 Vol.:  0.500 ch.:  0.105
 It.:    8 Obj.:   312.9614 Vol.:  0.500 ch.:  0.113
 It.:    9 Obj.:   307.3377 Vol.:  0.500 ch.:  0.090
 It.:   10 Obj.:   302.7199 Vol.:  0.500 ch.:  0.095
 It.:   11 Obj.:   298.9022 Vol.:  0.500 ch.:  0.075
 It.:   12 Obj.:   295.5064 Vol.:  0.500 ch.:  0.080
 It.:   13 Obj.:   292.2006 Vol.:  0.500 ch.:  0.068
 It.:   14 Obj.:   288.9708 Vol.:  0.500 ch.:  0.066
 It.:   15 Obj.:   285.5598 Vol.:  0.500 ch.:  0.062
 It.:   16 Obj.:   281.9851 Vol.:  0.500 ch.:  0.063
 It.:   17 Obj.:   277.8991 Vol.:  0.500 ch.:  0.059
 It.:   18 Obj.:   273.2379 Vol.:  0.500 ch.:  0.066
 It.:   19 Obj.:   267.7815 Vol.:  0.500 ch.:  0.062
 It.:   20 Obj.:   261.5262 Vol.:  0.500 ch.:  0.073
 It.:   21 Obj.:   254.5783 Vol.:  0.500 ch.:  0.064
 It.:   22 Obj.:   247.6060 Vol.:  0.500 ch.:  0.061
 It.:   23 Obj.:   241.9718 Vol.:  0.500 ch.:  0.048
 It.:   24 Obj.:   238.3026 Vol.:  0.500 ch.:  0.031
 It.:   25 Obj.:   236.0399 Vol.:  0.500 ch.:  0.025
 It.:   26 Obj.:   234.4791 Vol.:  0.500 ch.:  0.023
 It.:   27 Obj.:   233.3757 Vol.:  0.500 ch.:  0.020
 It.:   28 Obj.:   232.5235 Vol.:  0.500 ch.:  0.022
 It.:   29 Obj.:   231.8796 Vol.:  0.500 ch.:  0.019
 It.:   30 Obj.:   231.3397 Vol.:  0.500 ch.:  0.021
 It.:   31 Obj.:   230.9233 Vol.:  0.500 ch.:  0.019
 It.:   32 Obj.:   230.5497 Vol.:  0.500 ch.:  0.019
 It.:   33 Obj.:   230.2440 Vol.:  0.500 ch.:  0.018
 It.:   34 Obj.:   229.9589 Vol.:  0.500 ch.:  0.018
 It.:   35 Obj.:   229.7230 Vol.:  0.500 ch.:  0.018
```

```
It.:   36 Obj.:   229.4823 Vol.:  0.500 ch.:  0.017
It.:   37 Obj.:   229.2861 Vol.:  0.500 ch.:  0.017
It.:   38 Obj.:   229.0731 Vol.:  0.500 ch.:  0.016
It.:   39 Obj.:   228.8956 Vol.:  0.500 ch.:  0.016
It.:   40 Obj.:   228.7072 Vol.:  0.500 ch.:  0.015
It.:   41 Obj.:   228.5415 Vol.:  0.500 ch.:  0.015
It.:   42 Obj.:   228.3703 Vol.:  0.500 ch.:  0.014
It.:   43 Obj.:   228.2082 Vol.:  0.500 ch.:  0.015
It.:   44 Obj.:   228.0390 Vol.:  0.500 ch.:  0.014
It.:   45 Obj.:   227.8872 Vol.:  0.500 ch.:  0.014
It.:   46 Obj.:   227.7236 Vol.:  0.500 ch.:  0.014
It.:   47 Obj.:   227.5839 Vol.:  0.500 ch.:  0.013
It.:   48 Obj.:   227.4199 Vol.:  0.500 ch.:  0.014
It.:   49 Obj.:   227.2842 Vol.:  0.500 ch.:  0.013
It.:   50 Obj.:   227.1235 Vol.:  0.500 ch.:  0.014
It.:   51 Obj.:   226.9944 Vol.:  0.500 ch.:  0.013
It.:   52 Obj.:   226.8463 Vol.:  0.500 ch.:  0.014
It.:   53 Obj.:   226.7106 Vol.:  0.500 ch.:  0.013
It.:   54 Obj.:   226.5837 Vol.:  0.500 ch.:  0.014
It.:   55 Obj.:   226.4665 Vol.:  0.500 ch.:  0.012
It.:   56 Obj.:   226.3483 Vol.:  0.500 ch.:  0.013
It.:   57 Obj.:   226.2407 Vol.:  0.500 ch.:  0.012
It.:   58 Obj.:   226.1522 Vol.:  0.500 ch.:  0.013
It.:   59 Obj.:   226.0659 Vol.:  0.500 ch.:  0.012
It.:   60 Obj.:   226.0011 Vol.:  0.500 ch.:  0.012
It.:   61 Obj.:   225.9469 Vol.:  0.500 ch.:  0.011
It.:   62 Obj.:   225.9007 Vol.:  0.500 ch.:  0.012
It.:   63 Obj.:   225.8573 Vol.:  0.500 ch.:  0.011
It.:   64 Obj.:   225.8175 Vol.:  0.500 ch.:  0.011
It.:   65 Obj.:   225.7725 Vol.:  0.500 ch.:  0.011
It.:   66 Obj.:   225.7431 Vol.:  0.500 ch.:  0.011
It.:   67 Obj.:   225.7040 Vol.:  0.500 ch.:  0.011
It.:   68 Obj.:   225.6692 Vol.:  0.500 ch.:  0.010
It.:   69 Obj.:   225.6473 Vol.:  0.500 ch.:  0.010
It.:   70 Obj.:   225.6202 Vol.:  0.500 ch.:  0.010
It.:   71 Obj.:   225.6070 Vol.:  0.500 ch.:  0.010
It.:   72 Obj.:   225.5880 Vol.:  0.500 ch.:  0.010
```



Published with MATLAB® R2024a