```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% BILEVEL OPTIMIZER MAIN SCRIPT
% FOR FALL 2017 ME 6101 FINAL PROJECT ON GREEN MODULAR DESIGN OPTIMIZATION
% ENGINEERS: JAMES S COLLINS
%            BEN DUSSALT
%            NAMKHA NORSANG
%            NISHANTH KATHIRVEL
%
% PROJECT: ME 6101 GREEN MODULAR DESIGN GROUP PROJECT
% DATE: NOVEMBER 2017
% LOCATION: GEORGIA INSTITUTE OF TECHNOLOGY. ATL, GA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
clc

load Vg.mat;
load Vh.mat;
load Vs.mat;


Fstar=0.01;
fstar=0.006;

components=37;
ULiterations=2000;
LLiterations=2000;
population=700;

m=round(sqrt(components)); %inital max number of modules

% Store the alpha, beta (>=0) and the corresponding chromosome
A=cell(1,5);

ULmembers = randi(m,[population components]);
LLmembers = zeros(population, components);
ULscores = randi(10,[population 1]);

for i=1:ULiterations
    % Perform CGA on ULmembers
    ULmembers=GreenModGeneticAlgorithmUL(ULmembers,ULscores);
    % now we have a new population of ULCs
    for j=1:population
        [X,binX,~,m] = chromoSort(ULmembers(j,:)); %decompose each ULC
        F=fitnessFunctionF(vh,vs,ULmembers(j,:));
        ULscores(j,1)=F;
        alpha=(F-Fstar)/(1-Fstar);
        %%%%Evaluate Alpha(X)%%%
        if alpha<0 || isnan(alpha)
            break
        else
            binY=binX;
            Y=X;
            f=fitnessFunctionff(vg,Y);
            beta=(f-fstar)/(1-fstar);
```

```
            %%%%Evaluate Beta(Y)%%%
            if beta>=0 && ~isnan(beta)
                [currentAPop,~]=size(A);
                A(currentAPop+1,:)={[alpha],[beta],[F],[f],ULmembers(j,:)};
            else

                for k=1:LLiterations
                    %%%%%perform CGA on Y%%%%%
                    Y=GreenModGeneticAlgorithmLL(Y);
                    for l=1:size(Y,1)
                        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                        f = fitnessFunctionff(vg,Y);
                        %%%%evaluate beta(Y) %%%%
                        beta=(f-fstar)/(1-fstar);
                        if beta<0 || isnan(beta)
                            break
                        else
                            Y=sum(Y,1);
                            F=fitnessFunctionF(vh,vs,Y);
                            %%%%evaluate alpha(X)%%%%%%
                            alpha=(F-Fstar)/(1-Fstar);
                            if alpha<0 || isnan(alpha)
                                break
                            else
                                [currentAPop,~]=size(A);
                                A(currentAPop+1,:)={[alpha],[beta],[F],[f],Y(l,:)};
                            end
                        end
                    end
                end
            end
        end
    end
end
```

Published with MATLAB® R2017b