```
% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 % radar_optimizaton Function Definition
 % This function generates a population of pareto optimal solutions, each
 % representing one sparse radar array.
 % It receives parameters from the main_live_script and uses cost_function.m
 % and loop_gain.m to calculate the cost and gain of its members.
 % ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 % Parameters
 % styles: % A style is a group of congruent radar arrays.
           % They share the same diameter, power, and type (see type on line 18).
 %
 % min_T_styles: Minimum number of styles of transmitters in the array
 % min_R_styles: Minimum number of styles of receivers in the array
 % min_quantity: Minimum number of antennas of one style
 % max_quantity: Maximum number of antennas of one style
 % min_diameter: Minimum diameter (meters) of the antenna of one style
 % max_diameter: Maximum diameter (meters) of the antenna of one style
 % min_power: Minimum power (watts) to each transmitter of one style
 % max_power: Maximum power (watts) to each transmitter of one style
 % year: Year array will be built
 % include_monostatic: % If allow_monostatic is set to "T", each style has either
                             % transmitter, reciever, or monostatic types of antennas
                             % If allow_monostatic is set to "F", each style has
                             % transmitter or reciever types of antennas.
 % k: Efficiency of antennas
 % lambda: Wavelength (meters)
 % maximum_elements: Maximum number of antennas in the entire array
 % ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 % Return
 % x: matrix of pareto optimal solutions containing type, quantity,
 % diameter and power for each antenna style
 % fval: matrix of values of objective functions (cost and gain)
 % ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


 function [x,fval] = radar_optimization(num_styles, min_t_styles,min_r_styles, ↙
min_quantity,max_quantity,min_diameter,max_diameter,min_power,max_power,year_built, ↙
include_monostatic,k,nu,max_antennas)

    num_parameters = 4; % currently four parameters: type, quantity, diameter, and ↙
power
    num_vars = num_parameters*num_styles; % number of variables used for the matrix x
    % options for gamultiobj
    options = optimoptions('gamultiobj', 'FunctionTolerance',1e-4,'PopulationSize', ↙
150, 'MaxGenerations', 150);

    %speed of light = wavelength*frequency
    %speed of light in a vacuum, which decreases based on the density
    %of the medium
    c = physconst('LightSpeed');
    %the livescript has an input of Gigahertz, so iGHz = 10^9 Hz
    %calculating the wavelength in meters to be used in the loop gain
    nuHZ = nu*10^9;
    %function
```