```
% This program is use for Debri Colision Prediction
%
% Based on OrbProp developed by Damon P. DeLuca, USN CIV
% Email: damondeluca@gmail.com
% Adapted and Modified by Phd. Eliecer Hernandez, ABAE
% Email: ehernandez@abae.gob.ve
%
% Two files should be provided
%   First the configuration file named config.dcp
%       which will include the Code and TLE of the satellite that we want
%       to supervise
%
%   Second the TLE catalog:  named catalog.tle
%       Updated catalog of objects provided by NORAD and downloaded through
%       space-track
%
% NOTE: Output generated by this program has not been validated, other than
% ensuring close agreement with AGI STK results.
%
% This script runs the SGP4 propagator using the manual method, afspc, with
% WGS72 ellipsoid. SPG4 output in ManualOut.out is transformed from TEME
% to ECEF using functions from David Vallado. The transformation teme2ecef
% is executed for each line of output in ManualOutput.out.
%
% Input: See GUI. TLE input is a single two line file.
%
% Output:
%
%
%SGP4 functions were % downloaded from www.celestrak.com during the week
%of 18 JAN 2016. Credit below:
% Author:
%   Jeff Beck
%   beckja@alumni.lehigh.edu
%%

global mu  opsmode infile outfile  satrec startmfe stopmfe deltamin typerun...
    startedit stopedit s1
clear adum grob azob HorEl ElSR ElGR

tic

%% Configuration Section

typerun = 'm';
typeinput = 'e';
whichconst = 72;
rad = 180.0 / pi;

% Epoch
startedit='26-Jan-2017 00:00:00'; % UTC start time  01-Feb-2016 02:25:03
```

```
stopedit='29-Jan-2017 00:00:00';
s1='1'; %step resolution (min)

min_dist=10; % Min Distance threshold in Km
rp_tolerance=10; % perigee radius tolerance in Km
% Epoch

% File names IO
% Inputs
subject_file='\ExampleInput\vrss1.txt';
subject_output='subject_output.txt';
subject_output_refined='subject_output_refined.txt';
catalog_file='\ExampleInput\Full_Catalog_1.txt';
close_approach_file='sat_names_output.txt';
close_approach_refined='sat_close_approach_refined_';

%% Loading Files both input & output

inpath=subject_file;
infilename = inpath;
infile = fopen(infilename, 'r');
if (infile == -1)
    fprintf(1,'Failed to open file: %s\n', infilename);
    return;
end

% output file for RV Distance Date etc
outpath = subject_output;
outfilename = outpath;
outfile = fopen(outfilename, 'wt+');
if (outfile == -1)
    fprintf(1,'Failed to open file: %s\n', outfilename);
    return;
end

%% Load TLE from file

opsmode= 'a';

%typeinput = 'e'; %     typeinput = input('input mfe, epoch (YMDHMS), or dayofyr↙
approach, m,e,d:','s');

longstr1 = fgets(infile, 130);
while ( (longstr1(1) == '#') && (feof(infile) == 0) )
    longstr1 = fgets(infile, 130);
end
longstr2 = fgets(infile, 130);
catno = strtrim(longstr1(3:7)); % Satelite Number
%                 // convert the char string to sgp4 elements
%                 // includes initialization of sgp4
[satrec, startmfe, stopmfe, deltamin] = twoline2rv( whichconst, ...
longstr1, longstr2);



%% main routine

% --- Executes on button press in run.

%   add operation smode for afspc (a) or improved (i)
% opsmode= input('input opsmode afspc a, improved i ','s');
% opsmode= 'a';

%EH comented
%         //typerun = 'c' compare 1 year of full satcat data
%         //typerun = 'v' verification run, requires modified elm file with
%         //typerun = 'm' maunual operation- either mfe, epoch, or dayof yr
%         //              start stop and delta times

%     typerun = input('input type of run c, v, m: ','s');
%outfile = fopen('ManualOutput.out', 'wt');


%% ----------------- test simple propagation -------------------

longstr1 = fgets(infile, 130);
while ( (longstr1(1) == '#') && (feof(infile) == 0) )
    longstr1 = fgets(infile, 130);
end


[~, ro ,vo] = sgp4 (satrec,  0.0);  %[satrec, ro ,vo] = sgp4 (satrec,  0.0);

tsince = startmfe;


%                // check so the first value isn't written twice
if ( abs(tsince) > 1.0e-8 )
    tsince = tsince - deltamin;
end

%%     loop to perform the main propagation
k=1;
%
[p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,mu);

ra_0=a*(1+ecc)+rp_tolerance;

while ((tsince < stopmfe) && (satrec.error == 0))

    tsince = tsince + deltamin;
    if(tsince > stopmfe)
        tsince = stopmfe;
    end

    [~, ro, vo] = sgp4 (satrec,  tsince);  % [satrec, ro, vo] = sgp4 (satrec,  tsince);
    if (satrec.error > 0)
        fprintf(1,'# *** error: t:= %f *** code = %3i\n', tsince, satrec.error);
    end

    if (satrec.error == 0)
            jd = satrec.jdsatepoch + tsince/1440.0;
            [year,mon,day,hr,minute,sec] = invjday ( jd );

            %Print the below for TEME state
            fprintf(outfile,...
                ' %16.8f %16.8f %16.8f %12.9f %12.9f %12.9f %5i%3i%3i %2i %2i %9.6f↙
\n',...
                ro(1),ro(2),ro(3),vo(1),vo(2),vo(3),year,mon,day,hr,minute,sec );
            r_o(k,1)=ro(1);
            r_o(k,2)=ro(2);
            r_o(k,3)=ro(3);
            %[p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,mu);
    end %// if satrec.error == 0
    k=k+1;
end %// while propagating the orbit

%% Catalog propagation propagation and coarse close approach estimation




%% Input and out files for the catalog

inpath=catalog_file;
infilename = inpath;
infile = fopen(infilename, 'r');
if (infile == -1)
    fprintf(1,'Failed to open file: %s\n', infilename);
    return;
end


% output file for RV Distance Date etc
outpath = close_approach_file;
outfilename = outpath;
outfile = fopen(outfilename, 'wt+');
if (outfile == -1)
    fprintf(1,'Failed to open file: %s\n', outfilename);
    return;
end
```

```
disp('Loading Satellite Catalog, initiating propagation');

%% Loading TLE drom file and propagating from catalog


r_1=zeros(size(r_o,1),3);
delta_r=zeros(size(r_o,1),3);


i=1; %counter for satellite number list


while (feof(infile) == 0)

    longstr1 = fgets(infile, 130);
    while ( (longstr1(1) == '#') && (feof(infile) == 0) )
        longstr1 = fgets(infile, 130);
    end
    longstr2 = fgets(infile, 130);
    catno = strtrim(longstr1(3:7)); % Satelite Number
    %disp(strcat('Propagating satellite: ',num2str(catno)));
    % Convert the TLE to RV

    [satrec, startmfe, stopmfe, deltamin] = twoline2rv( whichconst, ...
    longstr1, longstr2);

    %Test Propagation
    [~, ro ,vo] = sgp4 (satrec,  0.0);  %[satrec, ro ,vo] = sgp4 (satrec,  0.0);
    tsince = startmfe;
    %                // check so the first value isn't written twice
    if ( abs(tsince) > 1.0e-8 )
        tsince = tsince - deltamin;
    end

    %%     loop to perform the main propagation
    k=1;
    tic
    %
    while ((tsince < stopmfe) && (satrec.error == 0))
        if k==1
         [p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,mu);
         rp_1=a*(1-ecc)-rp_tolerance;
         if rp_1>ra_0
          break
         end
        end
        tsince = tsince + deltamin;

        if(tsince > stopmfe)
```

```
            tsince = stopmfe;
        end

        [~, ro, vo] = sgp4 (satrec,  tsince);  % [satrec, ro, vo] = sgp4 (satrec,↙
tsince);
        if (satrec.error > 0)
            fprintf(1,'# *** error: t:= %f *** code = %3i\n', tsince, satrec.error);
        end

        if (satrec.error == 0)
                jd = satrec.jdsatepoch + tsince/1440.0;
                [year,mon,day,hr,minute,sec] = invjday ( jd );
                %Print the below for TEME state
                %fprintf(outfile,...
                %    ' %16.8f %16.8f %16.8f %12.9f %12.9f %12.9f %5i%3i%3i %2i %2i %9.6f↙
\n',...
                %    ro(1),ro(2),ro(3),vo(1),vo(2),vo(3),year,mon,day,hr,minute,sec );
                r_1(k,1)=ro(1);
                r_1(k,2)=ro(2);
                r_1(k,3)=ro(3);
                delta_r=((r_1(k,1)-r_o(k,1))^2+(r_1(k,2)-r_o(k,2))^2+(r_1(k,3)-r_o(k,3))↙
^2)^(1/3);
                %[p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,↙
mu);
        end %// if satrec.error == 0

        % Check for closest aproach distance and save it in file
        if delta_r<=min_dist
         fprintf(outfile,'%5s %16.8f %5i%3i%3i %2i:%2i:%9.6f \n',catno,delta_r,year,mon,↙
day,hr,minute,sec);
         disp('********************   Close approach   *****************');
         disp(str2double(catno));
         disp(delta_r);
         sat_no_list(i,1)=str2double(catno);
         i=i+1;
         break;
        end
        k=k+1;
    end %// while propagating the orbit


end %// While Catalog

fclose('all');

disp('***************  Refining Propagations ********************');

%% Refined Propagation

% Configuration Section
```

```
s1='1'; % Refined Step Resolution

%% propagate the subject satellite orbit

%% Loading Files both input & output

inpath=subject_file;
infilename = inpath;
infile = fopen(infilename, 'r');
if (infile == -1)
    fprintf(1,'Failed to open file: %s\n', infilename);
    return;
end


% output file for RV Distance Date etc
outpath = subject_output_refined;
outfilename = outpath;
outfile = fopen(outfilename, 'wt+');
if (outfile == -1)
    fprintf(1,'Failed to open file: %s\n', outfilename);
    return;
end

%% Load TLE from file

opsmode= 'a';

%typeinput = 'e'; %     typeinput = input('input mfe, epoch (YMDHMS), or dayofyr↙
approach, m,e,d:','s');

longstr1 = fgets(infile, 130);
while ( (longstr1(1) == '#') && (feof(infile) == 0) )
    longstr1 = fgets(infile, 130);
end
longstr2 = fgets(infile, 130);
catno = strtrim(longstr1(3:7)); % Satelite Number
%                // convert the char string to sgp4 elements
%                // includes initialization of sgp4
[satrec, startmfe, stopmfe, deltamin] = twoline2rv( whichconst, ...
longstr1, longstr2);


%% Main Routine
%%----------------- test simple propagation -------------------

longstr1 = fgets(infile, 130);
while ( (longstr1(1) == '#') && (feof(infile) == 0) )
    longstr1 = fgets(infile, 130);
end
```

```matlab
[~, ro ,vo] = sgp4 (satrec,  0.0);  %[satrec, ro ,vo] = sgp4 (satrec,  0.0);

tsince = startmfe;

%                 // check so the first value isn't written twice
if ( abs(tsince) > 1.0e-8 )
    tsince = tsince - deltamin;
end

%%     loop to perform the main propagation
k=1;
%
[p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,mu);

ra_0=a*(1+ecc)+rp_tolerance;

while ((tsince < stopmfe) && (satrec.error == 0))

    tsince = tsince + deltamin;

    if(tsince > stopmfe)
        tsince = stopmfe;
    end

    [~, ro, vo] = sgp4 (satrec,  tsince);  % [satrec, ro, vo] = sgp4 (satrec,  tsince);
    if (satrec.error > 0)
        fprintf(1,'# *** error: t:= %f *** code = %3i\n', tsince, satrec.error);
    end


    if (satrec.error == 0)
        jd = satrec.jdsatepoch + tsince/1440.0;
        [year,mon,day,hr,minute,sec] = invjday ( jd );

        %Print the below for TEME state
        fprintf(outfile,...
            ' %16.8f %16.8f %16.8f %12.9f %12.9f %12.9f %5i%3i%3i %2i %2i %9.6f \n',...
            ro(1),ro(2),ro(3),vo(1),vo(2),vo(3),year,mon,day,hr,minute,sec );
        r_o(k,1)=ro(1);
        r_o(k,2)=ro(2);
        r_o(k,3)=ro(3);
        %[p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,mu);
    end %// if satrec.error == 0
    k=k+1;
end %// while propagating the orbit
```

```
%% Input and out files for the catalog


inpath=catalog_file;
infilename = inpath;
infile = fopen(infilename, 'r');
if (infile == -1)
    fprintf(1,'Failed to open file: %s\n', infilename);
    return;
end

disp('Fetching Close Satellites from the Catalog, initiating propagation');

%% Loading TLE drom file and propagating from catalog

r_1=zeros(size(r_o,1),3);
delta_r=zeros(size(r_o,1),3);

i=1; %counter for satellite number list

while (feof(infile) == 0)

    min_distance=1000;

    longstr1 = fgets(infile, 130);
    while ( (longstr1(1) == '#') && (feof(infile) == 0) )
        longstr1 = fgets(infile, 130);
    end
    longstr2 = fgets(infile, 130);
    catno = strtrim(longstr1(3:7)); % Target Satelite Number
    if str2double(catno)~=sat_no_list(i,1)
        continue;
    end

    % output file for RV Distance Date etc
    outpath = strcat(close_approach_refined,num2str(catno),'.txt');
    outfilename = outpath;
    outfile = fopen(outfilename, 'wt+');
    if (outfile == -1)
        fprintf(1,'Failed to open file: %s\n', outfilename);
        return;
    end

    disp(strcat('Propagating satellite: ',num2str(catno)));
    % Convert the TLE to RV

    [satrec, startmfe, stopmfe, deltamin] = twoline2rv( whichconst, ...
    longstr1, longstr2);

    %Test Propagation
    [~, ro ,vo] = sgp4 (satrec,  0.0);  %[satrec, ro ,vo] = sgp4 (satrec,  0.0);
    tsince = startmfe;
    %                // check so the first value isn't written twice
    if ( abs(tsince) > 1.0e-8 )
        tsince = tsince - deltamin;
    end

    %%     loop to perform the main propagation
    k=1;
    %
    while ((tsince < stopmfe) && (satrec.error == 0))

        tsince = tsince + deltamin;

        if(tsince > stopmfe)
            tsince = stopmfe;
        end

        [~, ro, vo] = sgp4 (satrec,  tsince);  % [satrec, ro, vo] = sgp4 (satrec,↙
tsince);
        if (satrec.error > 0)
            fprintf(1,'# *** error: t:= %f *** code = %3i\n', tsince, satrec.error);
        end

        if (satrec.error == 0)
            jd = satrec.jdsatepoch + tsince/1440.0;
            [year,mon,day,hr,minute,sec] = invjday ( jd );
            %Print the below for TEME state
            %fprintf(outfile,...
            %    ' %16.8f %16.8f %16.8f %12.9f %12.9f %12.9f %5i%3i%3i %2i %2i %9.6f↙
\n',...
            %    ro(1),ro(2),ro(3),vo(1),vo(2),vo(3),year,mon,day,hr,minute,sec );
            r_1(k,1)=ro(1);
            r_1(k,2)=ro(2);
            r_1(k,3)=ro(3);
            delta_r=((r_1(k,1)-r_o(k,1))^2+(r_1(k,2)-r_o(k,2))^2+(r_1(k,3)-r_o(k,3))^2)^↙
(1/3);
            if delta_r<min_distance
                min_distance=delta_r;

            end
            %[p,a,ecc,incl,node,argp,nu,m,arglat,truelon,lonper ] = rv2coe (ro,vo,mu);
            fprintf(outfile,'%5s %16.8f %16.8f %16.8f %16.8f %5i %3i %3i %2i %2i %9.6f↙
\n',catno,ro(1),ro(2),ro(3),delta_r,year,mon,day,hr,minute,sec);
        end %// if satrec.error == 0

        % Check for closest aproach distance and save it in file

    end %// while propagating the orbit
    disp(min_distance);
    i=i+1;
    if i>size(sat_no_list,1)
```

```
        break
    end
end %// While Catalog

disp('End of Propagation');
fclose('all');

toc

%%% Change Log

% 30012017
%   Saving sat_no_list with the closest satellites for later use
%   All files are define in the configuration section
%   Added the refining routine
```