%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% TEST FILE - Merit fuction map of a doublet with 2 variables

%

% Based on doublet design used in "Fractal Basins of Attraction" paper

% (Turnhout et al, 2008)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% constants - all come from Turnhout et. al. (2008)

EFL = 100; % effective focal length

r1 = 1/0.020; % radius of curvature of first glass surface

r4 = inf; % radius of curvature of last glass surface - solved for later

d0 = 150; % distance from object to first element (not from paper - made up)

d1 = 10.346; % thickness of first element

d2 = 1; % air gap between first and second element

d3 = 2.351; % thickeness of second element;

d4 = 0; % distance from last surface to image plane - solved for later

n1 = 1.618; % index of refraction of first element

n2 = 1.717; % index of refraction of second element

na = 1; % index of refraction of air

sd = 33.33/2; % semidiamter of first element

seed = 1899345; % seed for calculating random rmse

% field points

sourcex = [10]; sourcey = [10];

% variables

%r2 = 1./linspace(-0.025, 0.040, 20);

%r3 = 1./linspace(-0.045, 0.075, 20);

r2 = 1./linspace(-0.045, 0.075, 100);

r3 = 1./linspace(-0.045, 0.075, 100);

% create camera

clear camera

camera(1) = struct('R', inf,'d', d0, 'n', na, 'sd', inf); % Object plane

camera(2) = struct('R', r1, 'd', d1, 'n', n1, 'sd', sd);

%camera(3) = struct('R', r2, 'd', d2, 'n', na, 'sd', sd);

%camera(4) = struct('R', r3, 'd', d3, 'n', n2, 'sd', sd);

camera(5) = struct('R', r4, 'd', d4, 'n', na, 'sd', sd);

%[camera, r4] = calc\_lastr(camera, EFL); % set last radius of curvature, r4

%[camera, d4] = calc\_lastd(camera); % set distance to image plane, d4

%%

N = 1000;

rmse = zeros(numel(r2), numel(r3));

for i = 1:numel(r2)

disp(i)

camera(3) = struct('R', r2(i), 'd', d2, 'n', na, 'sd', sd);

for j = 1:numel(r3)

camera(4) = struct('R', r3(j), 'd', d3, 'n', n2, 'sd', sd);

[camera, r4] = calc\_lastr(camera, EFL); % set last radius of curvature, r4

[camera, d4] = calc\_lastd(camera); % set distance to image plane, d4

[ xout, xtout, yout, ytout ] = traceRayForward( 0, 0, atan(.9\*sd/d0), 0, camera );

if isnan(xout)

rmse(i,j) = nan;

else

rmse\_points = zeros(size(sourcex));

for n = 1:numel(sourcex)

rmse\_points(n) = calc\_rmseCam(camera, sourcex(n), sourcey(n), N, seed);

end

rmse(i,j) = rms(rmse\_points);

end

end

end

%%

[R2, R3] = meshgrid(r2, r3);

figure; surf(1./r3, 1./r2, rmse','EdgeColor','none');

ylabel('c2 (mm^{-1})')

xlabel('c3 (mm^{-1})')

colorbar

%caxis([0 5])

%%

% Visual single point on merit function graph

c3 = 1/r3(49); c2 = 1/r2(89);

%camera(3) = struct('R', r2(39), 'd', d2, 'n', na, 'sd', sd);

%camera(4) = struct('R', r3(27), 'd', d3, 'n', n2, 'sd', sd);

camera(3) = struct('R', 1/c2, 'd', d2, 'n', na, 'sd', sd);

camera(4) = struct('R', 1/c3, 'd', d3, 'n', n2, 'sd', sd);

[camera, r4] = calc\_lastr(camera, EFL); % set last radius of curvature, r4

[camera, d4] = calc\_lastd(camera); % set distance to image plane, d4

rmse\_points = zeros(size(sourcex));

figure; h1 = subplot(2,1,1); h2 = subplot(2,1,2);

viz\_cameraWithRay(camera, 0, 0, atan(sd\*.6/d0), 0, 'fwd', h1);

title(sprintf('c2 = %1.4f, c3 = %1.4f', c2, c3));

viz\_spotdiag(camera, sourcex(1), sourcey(1), 1000, seed, h2);

rmse\_thisPoint = calc\_rmseCam(camera, sourcex(1), sourcey(1), N, seed)