the dipole moment of the spherical NP is given by:

where is the external field, is the field due to other oscillating dipoles and is the polarizability of the average NP. the NP are takent bo have no off diagonal components meaning that the fields are decoupled in to two parallel and one perpendicular component and using giving the parallel and perpendicular components as

function [kGenper] = kGenper(eps1,epsxpll,epsxper,angl,w)

%out puts the perpendicular wave vector beyond the first layer

%epsxpll is the dielectric constant for that layer

%epsxper is the dielectric constant for that layer

%eps1 is the dielectric constant for the first layer

%angl is the angl of incident light on the first layer

kGenper=(w/197.4)\*sqrt(epsxpll/epsxper)\*sqrt(epsxper-eps1\*((sin(angl))^2));

end

function [kGenpll] = kGenpll(eps1,epsxpll,epsxper,angl,w)

%out puts the parallel wave vector beyond the first layer

%epsxpll is the dielectric constant for that layer

%eps1 is the dielectric constant for the first layer

%angl is the angl of incident light on the first layer

x=epsxper;

kGenpll=(w/197.4)\*sqrt(epsxpll-eps1\*((sin(angl))^2));

end

function [kGen] = kGen(eps1,epsx,angl,w)

%out puts the wave vector for nom metamaterials beyond first layer

%epsx is the dielectric constant for that layer

%eps1 is the dielectric constant for the first layer

%angl is the angl of incident light on the first layer

kGen=(w/197.4)\*sqrt(epsx-eps1\*((sin(angl))^2));

end

function [rsij] = rsij(kipll,kjpll)

%reflection coeficient for s polarised light between layers i and j

% kipll is the wave vector for layer i

% kjpll is the wave vector for layer j

rsij=(kipll-kjpll)/(kipll+kjpll);

end

function [rpij] = rpij(kiper,kjper,epsipll,epsjpll)

%reflection coeficient for p polarised light between layers i and j

% kiper is the wave vector for layer i

% kjper is the wave vector for layer j

% epsipll is the parallel component for the dielectric constant for layer i

% epsjpll is the parallel component for the dielectric constant for layer j

rpij=(epsipll\*kjper-epsjpll\*kiper)/(epsipll\*kjper+epsjpll\*kiper);

end

function [tpij] = tpij(kiper,kjper,epsipll,epsjpll)

%reflection coeficient for p polarised light between layers i and j

% kiper is the wave vector for layer i

% kjper is the wave vector for layer j

% epsipll is the parallel compondent of the dielectric component for layer i

% epsjpll is the parallel component for the dielectric constant for layer j

tpij=(2\*sqrt(epsipll)\*sqrt(epsjpll)\*kiper)/(epsipll\*kjper+epsjpll\*kiper);

end

function [tsij] = tsij(kipll,kjpll)

%reflection coeficient for s polarised light between layers i and j

% kipll is the wave vector for layer i

% kjpll is the wave vector for layer j

tsij=(2\*kipll)/(kjpll+kipll);

end

function [Mn] = Mn(rij,tij,dj)

%transfer matix for two layers where n is i and n+1 is j

% tij is the transmition corficient for layers i,j

% rij is the relfectance coeficcient for layers i,j

% dj is the phase shift

Mn=(1/tij)\*[exp(-1i\*dj) rij\*exp(1i\*dj);rij\*exp(-1i\*dj) exp(1i\*dj)];

end

function [M] = M(M1,M2,M3,M4,M5,M6,M7,rij,tij)

%total transfer matix

% M1-M8 tranfer matixices for individual transitions

% tij is the transmition corficient for final transiton

% rij is the relfectance coeficcient for final transiton

M9=(1/tij)\*[1 rij;rij 1];

M=M1\*M2\*M3\*M4\*M5\*M6\*M7\*M9;

end