This note repeated Lin's result with our code, and got the right result.

Table of Contents

The input parameters.

clear; clc;

```
import model.phy.PhysicalObject.Lens
import model.phy.PhysicalObject.LaserBeam.ParaxialBeam.ParaxialLaguerreGaussianBea
import model.phy.PhysicalObject.LaserBeam.OpticalField
%%Lens
f=1.0;%focal distance in mm
NA=0.95; working_medium='vacuum';
lens=Lens(f, NA, working_medium);
%%incBeam
power=0.1;
% This power is used to calc the incbeam parameters.
%Also used as the focal plane power.
wavelength=1.064; waist=950.0; center=[0, 0, 0];  %in micron
%filling_factor = n_work_medium*waist/(f*1000)/NA
px=1.0; py=0.0; p=0; l=1;
incBeam1=ParaxialLaguerreGaussianBeam(wavelength, power, waist, center, p, l, px,
lg1=model.phy.PhysicalObject.LaserBeam.AplanaticBeam.LinearCircularPol(lens, incBe
lg1.calcAmpFactor(power);
%%scatter
k=lg1.focBeam.k; n_relative=1.46; radius =0.05;
%%Now begin calcualte.
Nmax=60;%It's fast in Fortran.Also in example_lg. But not ours, strange.
Nmax=ott13.ka2nmax(k*radius);Nmax=Nmax*5;
```

get ab and T matrix.

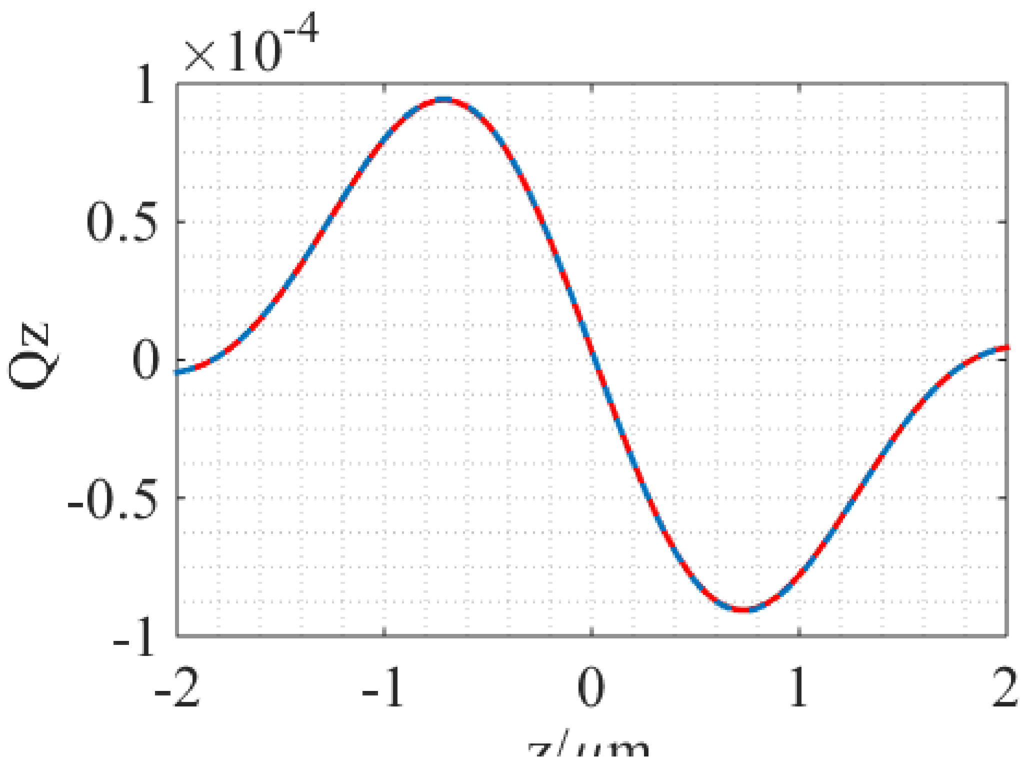
```
lg1.getVSWFcoeff(Nmax);
lg1.focBeam.aNNZ;
a0=lg1.focBeam.aNNZ(:,3);
b0=lg1.focBeam.bNNZ(:,3);
n=lg1.focBeam.aNNZ(:,1);
m=lg1.focBeam.aNNZ(:,2);
```

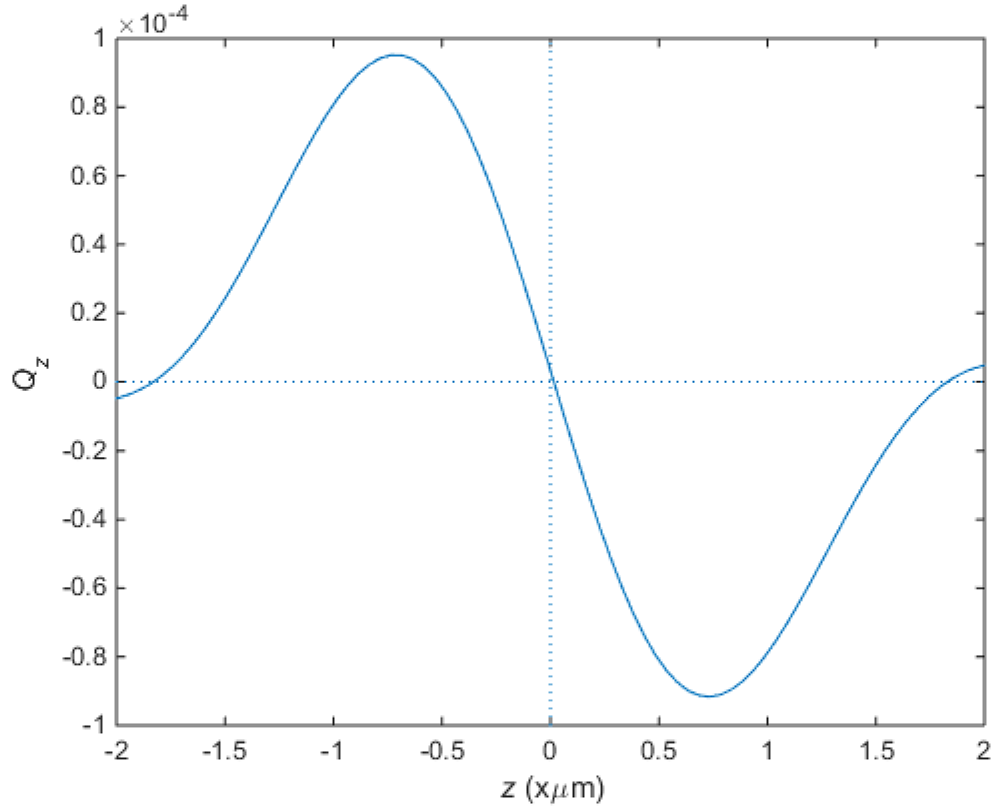
```
[a0,b0,n,m]=abLin2Nie(a0,b0,n,m);
[a,b,n,m] = ott13.make_beam_vector(a0,b0,n,m);

T = ott13.tmatrix_mie(Nmax,k,k*n_relative,radius);
```

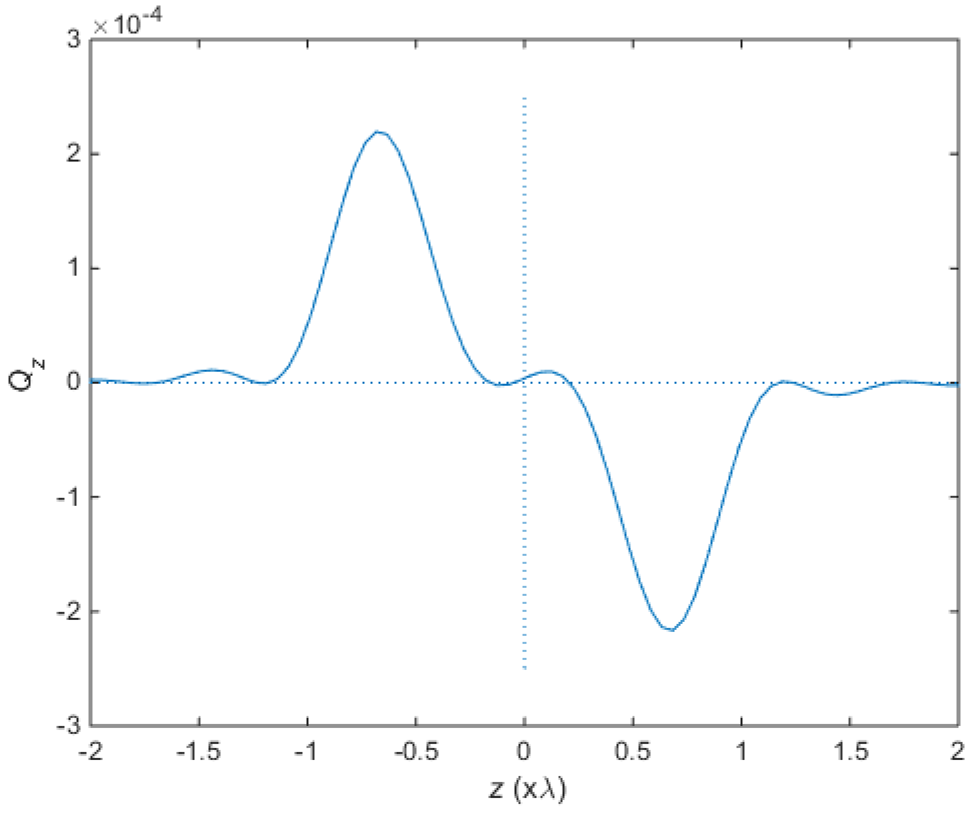
plot Qz

```
z = linspace(-2,2,80);
r = linspace(-2,2,80);
z = z/wavelength;
r = z/wavelength;

fz = zeros(size(z));
fr = zeros(size(r));

%root power for nomalization to a and b individually.
pwr = sqrt(sum( abs(a).^2 + abs(b).^2 ));

%normalize total momentum of wave sum to 1. Not good for SI EM field.
a=a/pwr;
b=b/pwr;

%calculate the force along z
for nz = 1:length(z)
%     nz
    [A,B] = ott13.translate_z(Nmax,z(nz));
    a2 = ( A*a + B*b );
    b2 = ( A*b + B*a );

    pq = T * [ a2; b2 ];
    p = pq(1:length(pq)/2);
    q = pq(length(pq)/2+1:end);

    fz(nz) = ott13.force_z(n,m,a2,b2,p,q);

end

figure;
plot(z*wavelength,fz);
xlabel('{\it z} (x\mum)');
ylabel('{\it Q_z}');
aa = axis;
hold on
line(aa(1:2),[ 0 0 ],'linestyle',':');
line([0 0],aa(3:4),'linestyle',':');
%compare with Lin
open('D:\mywork\zhoulm\OpticalTrap\FScat\SphereScat\SphereScat\calibration1\Qz50nm
%Nieminen's result.
open('D:\mywork\zhoulm\OpticalTrap\FScat\SphereScat\SphereScat\calibration1\Qz50nm
```

This note repeated Lin's result with
our code, and got the right result.



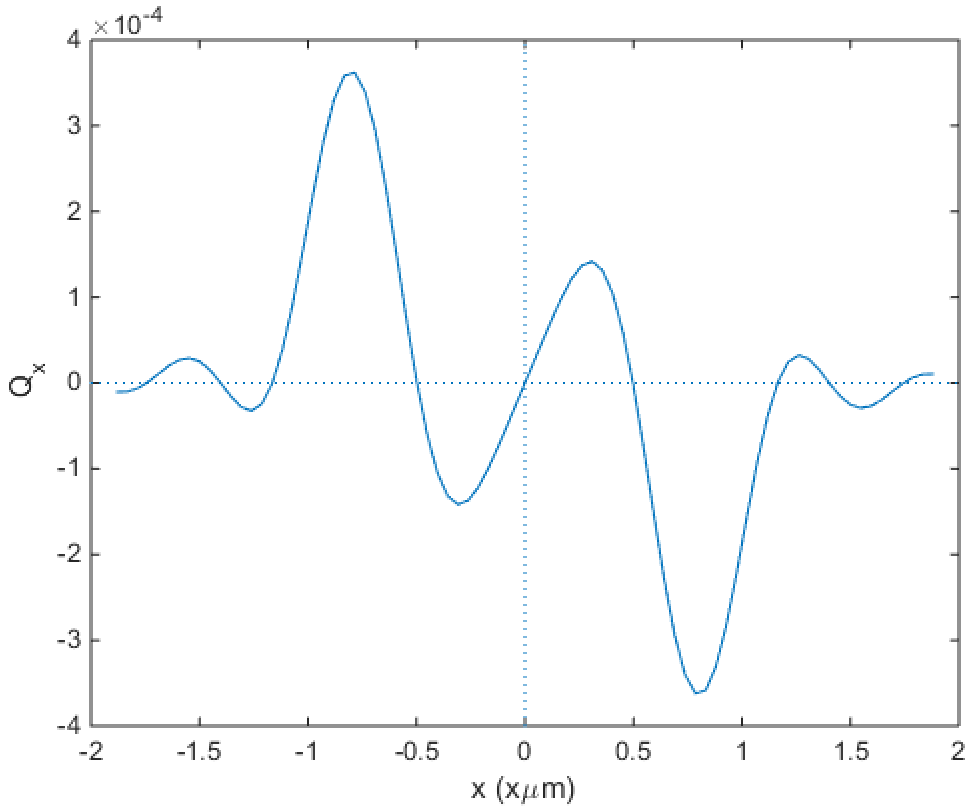




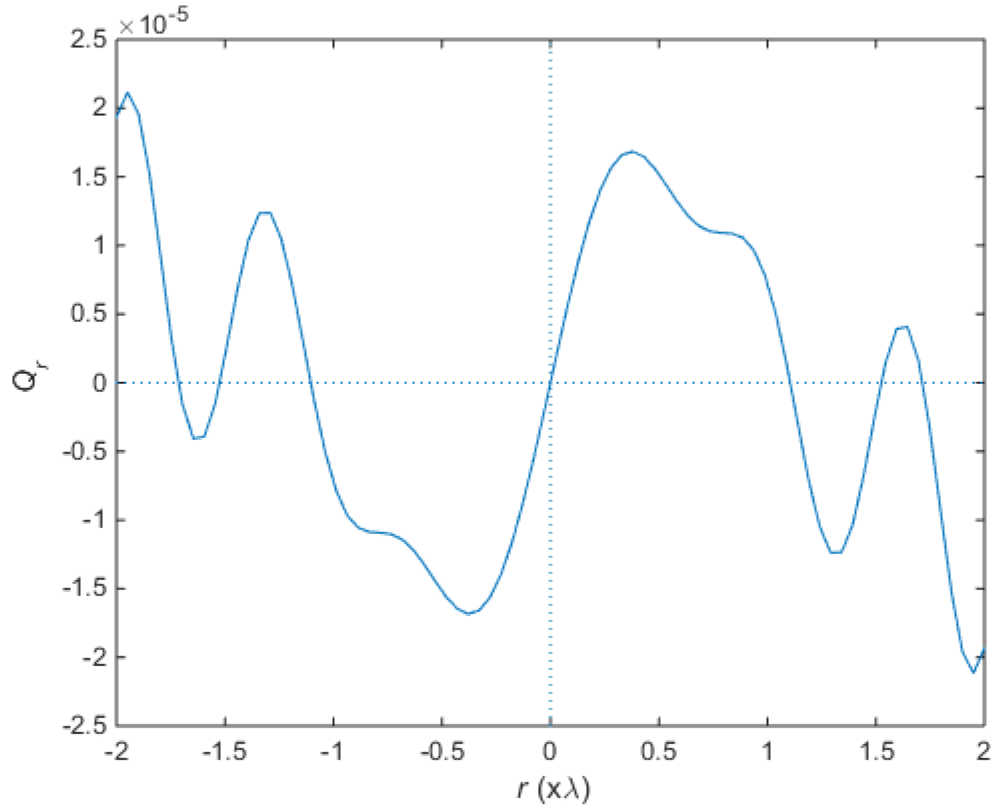
plot Qx

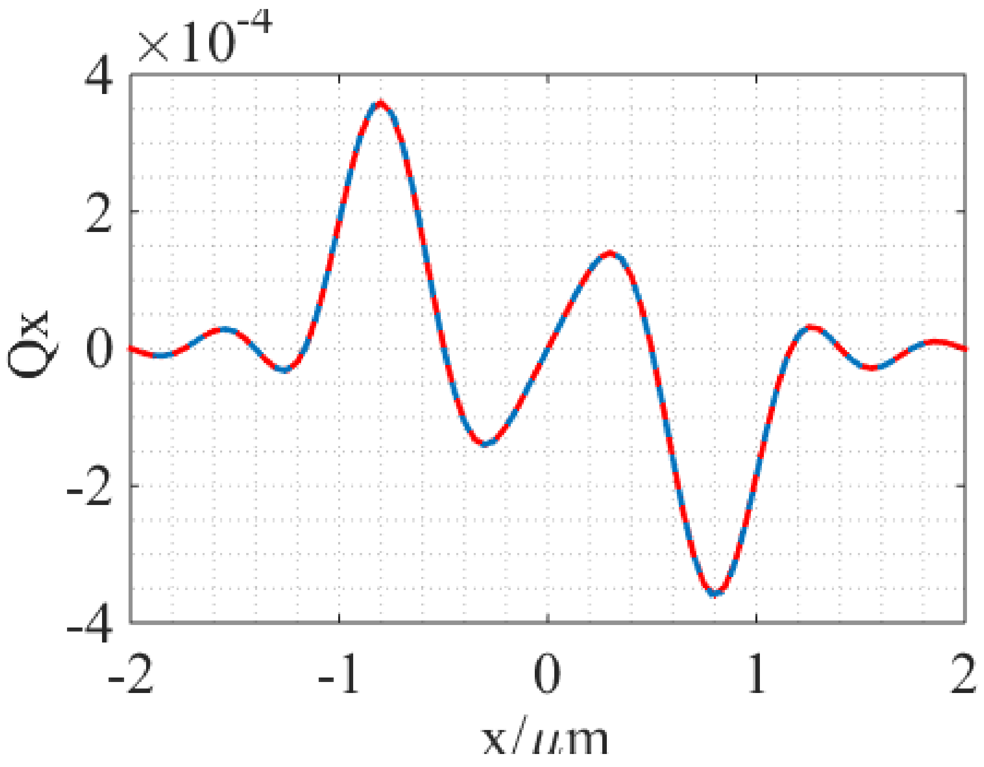
```
%calculate the x-axis coefficients for force calculation.
%now work out spherical coordinates along that axis:
zeq=0;
[rt,theta,phi]=ott13.xyz2rtp(r,0,zeq);
Rx = ott13.z_rotation_matrix(pi/2,0);
% Dx = ott13.wigner_rotation_matrix(Nmax,Rx);
Dx = wigner_rotation_matrix1(Nmax,Rx');

for nr = 1:length(r)
%     nr
    R = ott13.z_rotation_matrix(theta(nr),phi(nr)); %calculates an appropriate axi
%     D = ott13.wigner_rotation_matrix(Nmax,R);
    D = wigner_rotation_matrix1(Nmax,R');

    [A,B] = ott13.translate_z(Nmax,rt(nr));
    a2 = D'*(  A * D*a +  B * D*b ); % Wigner matricies here are hermitian. Theref
    b2 = D'*(  A * D*b +  B * D*a ); % In MATLAB operations on vectors are done fi

    pq = T * [ a2; b2 ];
    p = pq(1:length(pq)/2);
    q = pq(length(pq)/2+1:end);

    fr(nr) = ott13.force_z(n,m,Dx*a2,Dx*b2,Dx*p,Dx*q); %Dx makes the z-force calcu
```

```
end
%     timetakes(ii)=toc;
% end
%
% plot(log([4:length(timetakes)])/log(10),log(timetakes(4:end)-timetakes(3:end-1))
% plot([1:length(timetakes)-1],timetakes(2:end)-timetakes(1:end-1))


figure; plot(r*wavelength,fr);
xlabel('{x} (x\mum)');
ylabel('{Q_x}');
aa = axis;
hold on
line(aa(1:2),[ 0 0 ],'linestyle',':');
line([0 0],aa(3:4),'linestyle',':');
%compare with Lin
open('D:\mywork\zhoulm\OpticalTrap\FScat\SphereScat\SphereScat\calibration1\Qx50nm
%Nieminen's result.
open('D:\mywork\zhoulm\OpticalTrap\FScat\SphereScat\SphereScat\calibration1\Qr50nm
```



This note repeated Lin's result with
our code, and got the right result.





Published with MATLAB® R2014b