Infinite 3D medium, Isotropic Point Source, Isotropic Scattering

BesselK0 Random Flight

This is code to accompany the book:

A Hitchhiker's Guide to Multiple Scattering

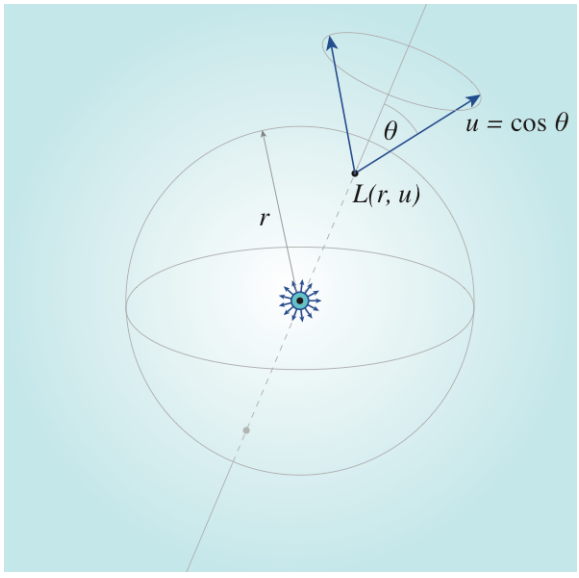
www.eugenedeon.com/hitchhikers

Path Setup

Put a file at ~/.hitchhikerpath with the path to your hitchhiker repo so that these worksheets can find the MC data from the C++ simulations for verification

In[2681]:=
```
SetDirectory[Import["~/.hitchhikerpath"]]
```

Notation



c - single-scattering albedo

r - radial position coordinate in medium (distance from point source at origin)

u = cos θ - direction cosine

Namespace

In[2831]:=
```
Begin["inf3DisopointIsoscatterK0`"]
```
Out[2831]=
```
inf3DisopointIsoscatterK0`
```

Analytical results

Collision rate density

collision rate density Cc due to correlated emission:

derivation

result

In[2882]:=
```
Ccexact[r_, c_] := NIntegrate[ u^2 ArcSinh[u] Sinc[r u] / (2 π^2 (u - c ArcSinh[u])),
  {u, 0, Infinity}, Method → "ExtrapolatingOscillatory"]
```

load MC data

In[2893]:=
```
ppoints[xs_, dr_, maxx_] :=
 Table[{dr (i) - 0.5 dr, xs[[i]]}, {i, 1, Length[xs]}][[1 ;; -2]]
```

In[2894]:=
```
ppointsu[xs_, du_, Σt_] :=
 Table[{-1.0 + du (i) - 0.5 du, xs[[i]] / (2 Σt)}, {i, 1, Length[xs]}][[1 ;; -1]]
```

In[2895]:=
```
fs = FileNames["code/3D_medium/infinite3Dmedium/Isotropicpointsource/MCdata/
     inf3D_isotropicpoint_isotropicscatter_bessel*"];
```

In[2896]:=
```
index[x_] := Module[{data, c},
   data = Import[x, "Table"];
   c = data[[2, 3]];
   {c, data}];
simulations = index /@ fs;
cs = Union[#[[1]] & /@ simulations]
```
Out[2898]=
```
{0.01, 0.1, 0.3, 0.5, 0.7, 0.8, 0.9, 0.95, 0.99, 0.999}
```

In[2899]:=
```
numcollorders = simulations[[1]][[-1]][[2, 13]];
```

Compare analytic and MC

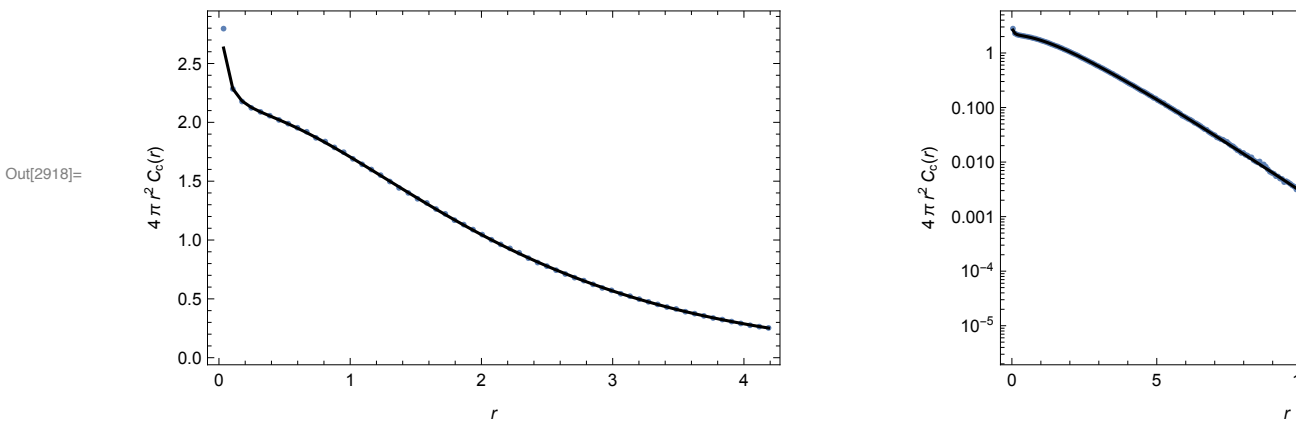
Collision-rate density - Exact solution - comparison to MC

In[2900]:=
```
{ActionMenu["Set c", "c = " <> ToString[#] :> (c = #;) & /@ cs], Dynamic[c]}
```

Out[2900]= { Set c , 0.8 }

In[2910]:=
```
data = SelectFirst[simulations, #[[1]] == c &][[2]];
maxr = data[[2, 5]];
dr = data[[2, 7]];
MCCollisionRate = ppoints[data[[4]], dr, maxr];
exact1CRShallow =
  Quiet[{#[[1]], 4 Pi #[[1]]^2 Ccexact[#[[1]], c]}] & /@ MCCollisionRate[[1 ;; 60]];
exact1CR = Quiet[{#[[1]], 4 Pi #[[1]]^2 Ccexact[#[[1]], c]}] & /@
   MCCollisionRate[[61 ;; -1 ;; 10]];
plotϕshallow = Quiet[Show[
    ListPlot[MCCollisionRate[[1 ;; 60]],
     PlotRange → All, PlotStyle → PointSize[.01]],
    ListPlot[exact1CRShallow, PlotRange → All, Joined → True, PlotStyle → Black],
    Frame → True,
    FrameLabel -> {{4 π r^2 C"c"[r],}, {r,}}
   ]];
logplotϕ = Quiet[Show[
    ListLogPlot[MCCollisionRate, PlotRange → All, PlotStyle → PointSize[.01]],
    ListLogPlot[exact1CR, PlotRange → All, Joined → True, PlotStyle → Black],
    ListLogPlot[exact1CRShallow,
     PlotRange → All, Joined → True, PlotStyle → Black],
    Frame → True,
    FrameLabel -> {{4 π r^2 C"c"[r],}, {r,}}
   ]];
Show[GraphicsGrid[{{plotϕshallow, logplotϕ}}, ImageSize → 800],
 PlotLabel -> "Infinite 3D, isotropic point source,
     Isotropic scattering, BesselK0 random flight - correlated
     emission\nCollision-rate density Cc[r], c = " <> ToString[c]]
```

Out[2918]=



Namespace

In[2919]:=
```
End[]
```

Out[2919]=
```
inf3DisopointIsoscatterK0`
```