Common Quantities and Functions

This is code to accompany the book:

A Hitchhiker's Guide to Multiple Scattering

www.eugenedeon.com

Surface Area of Unit Sphere in d Dimensions

In[1324]:=
```
Ω[d_, r_] := d π^(d/2) r^(d-1) / Gamma[d/2 + 1]
```

Spherical Diffusion Mode in d Dimensions

In[1328]:=
```
diffusionMode[v_, d_, r_] := (2 π)^(-d/2) r^(1-d/2) v^(-1-d/2) BesselK[1/2 (-2 + d), r/v]
```

In[1329]:=
```
Table[{d, FullSimplify[diffusionMode[v, d, r], Assumptions → v > 0 && r > 0]},
  {d, 1, 3}] // TableForm
```

Out[1329]//TableForm=

1	$\frac{e^{-\frac{r}{v}}}{2v}$
2	$\frac{\text{BesselK}\left[0,\frac{r}{v}\right]}{2\pi v^2}$
3	$\frac{e^{-\frac{r}{v}}}{4\pi r v^2}$

In[1330]:=
```
Integrate[Ω[d, r] diffusionMode[v, d, r],
 {r, 0, Infinity}, Assumptions → v > 0 && d ≥ 1]
```

Out[1330]= 1

Caseology Quantities

Definitions

```
CaseN0[c_, v0_] := 1/2 c v0^3 (c/(v0^2 - 1) - 1/v0^2)
```

In[1361]:=
```
Casev0[c_?NumericQ] :=
 FindRoot[c v ArcTanh[1/v] - 1, {v, 1 + 10^-10, 10^10}, Method → "Brent"][[1]][[2]]
```

In[1337]:=
```
Casev0[c_, prec_] := ReplaceAll[Abs[v],
   First[FindRoot[c v ArcTanh[1/v] - 1, {v, 2}, WorkingPrecision → prec]]];
```

```
CaseN[c_, v_] := v (Caseλ[v, c]^2 + (π c v/2)^2)
```

```
Caseλ[v_, c_] := 1 - c v ArcTanh[v]
```

Approximations

Approximation from [Case and Zweifel 1967]

In[1342]:=
```
k0low[c_] :=
  1 - 2 E^(-2/c) (1 + (4 - c)/c E^(-2/c) + (24 - 12 c + c^2)/c^2 E^(-4/c) + (512 - 384 c + 72 c^2 - 3 c^3)/(3 c^3) E^(-6/c));
k0high[c_] := Sqrt[3 (1 - c)] (1 - 2/5 (1 - c) - 12/175 (1 - c)^2 - 2/125 (1 - c)^3 + 166/67 375 (1 - c)^4);
Casev0approx[c_] := If[c > 0.56, 1/k0high[c], 1/k0low[c]]
```

Benchmark Values for Discrete Eigenvalue v_0

In[1369]:=
```
v0BenchTable = TableForm[
  Join[{{"α", "v0"}}, Map[{N[#], Casev0[#, 40]} &, {1/100, 5/100, 1/10, 2/10, 3/10, 5/10, 7/10,
    8/10, 85/100, 9/10, 95/100, 98/100, 99/100, 995/1000, 999/1000, 9999/10 000, 99 999/100 000, 999 999/1 000 000}]]]
```

FindRoot::cvmit : Failed to converge to the requested accuracy or precision within 100 iterations. >>

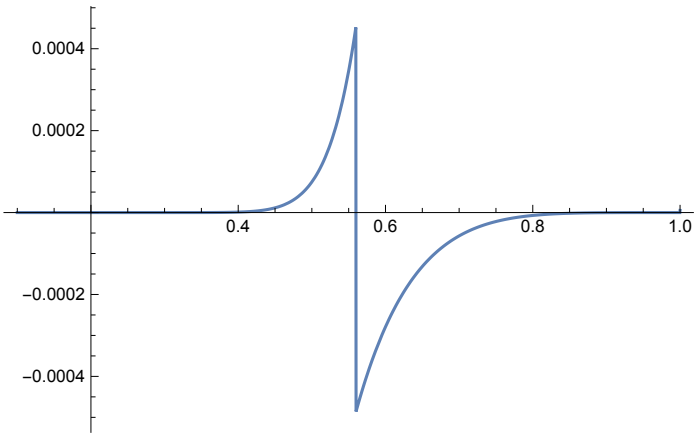
Out[1369]//TableForm=

α	v_0
0.01	1.00000000000000000000000000000093600630
0.05	1.00000000000000000849670851058318091 4518
0.1	1.00000004122307593242207339133885345957
0.2	1.00009088654438071082110919216032696 3735
0.3	1.00259288879322319914298250164296409 2168
0.5	1.04438203376083348498401390634474776 0869
0.7	1.20680425398528603357214453710544839 7639
0.8	1.40763430906277201589007182580816383 6056
0.85	1.58855862536317969642842131770450166 3412
0.9	1.90320485604484771898056123745778081 6825
0.95	2.63514883426873917731167996758654952 2622
0.98	4.11552047631644542127143179268299575 3409
0.99	5.79672945130200230977583636559759879 3316
0.995	8.18134253585742032173001303338091747 5302
0.999	18.26472572652667373356350462926948043 553
0.9999	57.73733645201289717419088459805147261 345
0.99999	182.57491613597186024303362834132987373 41
0.999999	577.35050012986540621312920596107734327 21

Evaluate Case approximation

In[1367]:=
```
p = Plot[Casev0[c] - Casev0approx[c], {c, 0.1, 1}, PlotRange → All]
```

Out[1367]=



In[1368]:=
```
p = Plot[Casev0[c] / Casev0approx[c], {c, 0.1, 1}, PlotRange → All]
```

Out[1368]=