Adapted from Michel Fodje's epr-simple simulation translated from Python to Mathematica by John Reed 13 Nov 2013
Using Joy Christian's complete state parameters
Made even more simple by Fred Diether 11 Oct 2015
(It no longer depends on particle angle "e".)

Set run time parameters, initialize arrays

In[121]:=
```
trials = 5 000 000;
trialDeg = 360;
```

In[123]:=
```
aliceDeg = ConstantArray[0, trials];
bobDeg = ConstantArray[0, trials];
aliceDet = ConstantArray[0, trials];
bobDet = ConstantArray[0, trials];
```

In[127]:=
```
nPP = ConstantArray[0, trialDeg];
nNN = ConstantArray[0, trialDeg];
nPN = ConstantArray[0, trialDeg];
nNP = ConstantArray[0, trialDeg];
```

Complete State Selection

In[131]:=
```
test1[angle_, λ_] := Module[{c, out},
  c = -Cos[angle];
  If[λ ≥ Abs[c], out = 0, out = Sign[c]];
  out]
test2[angle_, λ_] := Module[{c, out},
  c = -Cos[angle];
  If[λ ≥ Abs[c], out = 0, out = -Sign[c]];
  out]
```

Generate Particle Data

In[133]:=
```
Do[
 t = RandomReal[{0, π}];
 λ = 2 / Sqrt[1 + 3 t / π] - 1;
 aliceAngle = RandomReal[{0, 2 π}];
 aliceDeg[[i]] = aliceAngle / Degree;
 bobAngle = RandomReal[{0, 2 π}];
 bobDeg[[i]] = bobAngle / Degree;
 aliceDet[[i]] = test1[aliceAngle, λ];
 bobDet[[i]] = test2[bobAngle, λ],
 {i, trials}]
```

Statistical Analysis of Particle Data

In[134]:=
```
Do[
 θ = Round[aliceDeg[[i]] - bobDeg[[i]]];
 aliceD = aliceDet[[i]]; bobD = bobDet[[i]];
 If[aliceD == 1 && bobD == 1, nPP[[θ]]++];
 If[aliceD == 1 && bobD == -1, nPN[[θ]]++];
 If[aliceD == -1 && bobD == 1, nNP[[θ]]++];
 If[aliceD == -1 && bobD == -1, nNN[[θ]]++],
 {i, trials}]
```

Calculate mean values and plot

In[135]:=
```
pPP = 0; pPN = 0; pNP = 0; pNN = 0;
```

In[136]:=
```
mean = ConstantArray[0, trialDeg];
```

In[137]:=
```
Do[
 sum = nPP[[i]] + nPN[[i]] + nNP[[i]] + nNN[[i]];
If[sum == 0, Goto[jump],
  {pPP = nPP[[i]] / sum;
   pNP = nNP[[i]] / sum;
   pPN = nPN[[i]] / sum;
   pNN = nNN[[i]] / sum;
mean[[i]] = pPP + pNN - pPN - pNP}];
 Label[jump],
 {i, trialDeg}]
```

In[138]:=
```
simulation = ListPlot[mean, PlotMarkers → {Automatic, Tiny}];
```

In[139]:=
```
cos = Plot[-Cos[x Degree], {x, 0, 360}, PlotStyle → {Red}];
```

Compare mean values with -Cosine Curve

In[140]:=
```
Show[simulation, cos]
```

Out[140]=