Simulation Based on Michel Fodje's epr-simple simulation translated from Python to Mathematica by John Reed 13 Nov 2013
Modified by Fred Diether for Completely Local-Realistic Jan 2021

Set Run Time Parameters, Initialize Arrays and Tables

In[2906]:=
```
m = 20 000 000;
trialDeg = 721;
AB = Table[{0, 0, 0, 0}, m];
a1 = ConstantArray[0, m];
b1 = ConstantArray[0, m];
A1 = ConstantArray[0, m];
B1 = ConstantArray[0, m];
nPP = ConstantArray[0, trialDeg];
nNN = ConstantArray[0, trialDeg];
nPN = ConstantArray[0, trialDeg];
nNP = ConstantArray[0, trialDeg];
nAP = ConstantArray[0, trialDeg];
nBP = ConstantArray[0, trialDeg];
nAN = ConstantArray[0, trialDeg];
nBN = ConstantArray[0, trialDeg];
```

Generate Particle Data

In[2921]:=
```
Do[e = RandomReal[{0, 360}];
 λ = 0.3 (Cos[e/3]^2);
 a = RandomInteger[{0, 360}];
 b = RandomInteger[{0, 360}];
 If[Abs[Cos[(a - e) Degree]] < λ || Abs[Cos[(b - e) Degree]] < λ,
  A = -Sign[Sin[(a - e - 15) Degree]], A = -Sign[Cos[(a - e) Degree]]];
 If[Abs[Cos[(a - e) Degree]] < λ || Abs[Cos[(b - e) Degree]] < λ,
  B = Sign[Sin[(b - e - 15) Degree]], B = Sign[Cos[(b - e) Degree]]];
 AB[[i]] = {a, b, A, B}, {i, m}]
```

Statistical Analysis of Particle Data

In[2922]:=
```
theta = ConstantArray[0, m];
a1 = AB[[All, 1]]; b1 = AB[[All, 2]]; A1 = AB[[All, 3]]; B1 = AB[[All, 4]];
Do[θ = a1[[j]] - b1[[j]] + 361;  (*All angles are shifted by 2π since θ is an index*)
 theta[[j]] = θ;
 aliceD = A1[[j]]; bobD = B1[[j]];
 If[aliceD == 1, nAP[[θ]]++];
 If[bobD == 1, nBP[[θ]]++];
 If[aliceD == -1, nAN[[θ]]++];
 If[bobD == -1, nBN[[θ]]++];
 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]]++], {j, m}]
```

Calculate Mean Values and Plot

In[2925]:=
```
pPP = 0; pPN = 0; pNP = 0; pNN = 0;
mean = ConstantArray[0, trialDeg];
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[2928]:=
```
simulation = ListPlot[mean, PlotMarkers → {Automatic, Tiny}];
negcos = Plot[-Cos[x Degree], {x, 0, 720}, PlotStyle → {Magenta},
   Ticks → {{{0, -360 °}, {90, -270 °}, {180, -180 °}, {270, -90 °}, {360, 0 °}, {450, 90 °},
      {540, 180 °}, {630, 270 °}, {720, 360 °}}, Automatic}, GridLines → Automatic];
```

Compare mean values with -Cosine Curve and compute averages

In[2930]:=
```
Show[negcos, simulation]
```

Out[2930]=

In[2931]:=
```
AveA = N[Sum[A1[[i]], {i, m}] / m];
AveB = N[Sum[B1[[i]], {i, m}] / m];
Print["AveA = ", AveA]
Print["AveB = ", AveB]
PAP = N[Sum[nAP[[i]], {i, trialDeg}]];
PBP = N[Sum[nBP[[i]], {i, trialDeg}]];
PAN = N[Sum[nAN[[i]], {i, trialDeg}]];
PBN = N[Sum[nBN[[i]], {i, trialDeg}]];
PA1 = PAP / (PAP + PAN);
PB1 = PBP / (PBP + PBN);
Print["P(A+)= ", PA1]
Print["P(B+)= ", PB1]
totAB = Sum[nPP[[i]] + nNN[[i]] + nPN[[i]] + nNP[[i]], {i, trialDeg}]
PP = N[Sum[nPP[[i]], {i, trialDeg}] / totAB]
NN = N[Sum[nNN[[i]], {i, trialDeg}] / totAB]
PN = N[Sum[nPN[[i]], {i, trialDeg}] / totAB]
NP = N[Sum[nNP[[i]], {i, trialDeg}] / totAB]
CHSH = Abs[N[mean[[315]]] - N[mean[[225]]] + N[mean[[405]]] + N[mean[[45]]]]
```

```
AveA = 0.0003019

AveB = 0.000249

P(A+)= 0.500151

P(B+)= 0.500125
```

Out[2943]= 20 000 000

Out[2944]= 0.250215

Out[2945]= 0.24994

Out[2946]= 0.249936

Out[2947]= 0.249909

Out[2948]= 2.7948

In[2949]:=
```
Histogram[theta, 70]
```

Out[2949]=