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 Oct. 2021
With 3D Vectors!

Set Run Time Parameters, Initialize Arrays and Tables

```
In[1]:= m = 5 000 000;
trialDeg = 360;
s1 = ConstantArray[0, m];
s2 = ConstantArray[0, m];
λ1 = ConstantArray[0, m];
λ2 = ConstantArray[0, m];
aa1 = ConstantArray[0, m];
bb1 = ConstantArray[0, m];
outA1 = Table[{0, 0, 0}, m];
outA2 = Table[{0, 0, 0}, m];
outB1 = Table[{0, 0, 0}, m];
outB2 = Table[{0, 0, 0}, m];
a1 = ConstantArray[0, m];
b1 = ConstantArray[0, m];
A = ConstantArray[0, m];
B = 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];
ϕ = 3; β = 0.285; ξ = 0.888; (*Adustable parameters for fine tuning*)
```

Generating Particle Data with Three Independent Do-Loops

```
In[26]:= Do[θ = RandomPoint[Sphere[]]; (*Singlet 3D vector*) (*Hidden Variable*)
 θ1 = ToSphericalCoordinates[θ][[3]] * 180 / π;
 θ2 = ToSphericalCoordinates[θ][[2]];
 λ1[[i]] = β (Cos[θ1/ϕ]^2);
 λ2[[i]] = (Cos[(θ2 * ξ)/2]^2);
 s1[[i]] = θ;
 s2[[i]] = -θ, {i, m}]
```

In[27]:=
```
Do[a = RandomPoint[Sphere[]];  (*Detector 3D vector angle*)
 aa1[[i]] = a;
 If[Abs[a.s1[[i]]] > λ1[[i]],
  A1 = Sign[a.s1[[i]]], A1 = Sign[Sin[ArcCos[a.s1[[i]]]]] + 0.001];
 outA1[[i]] = {a, A1, i}, {i, m}]
Do[If[Abs[aa1[[i]].s1[[i]]] > λ2[[i]],
  A2 = Sign[aa1[[i]].s1[[i]]], A2 = Sign[Sin[ArcCos[aa1[[i]].s1[[i]]]]] + 0.001];
 outA2[[i]] = {aa1[[i]], A2, i + m}, {i, m}]
outA = Catenate[{outA1, outA2}];
```

In[30]:=
```
Do[b = RandomPoint[Sphere[]];  (*Detector 3D vector angle*)
 bb1[[i]] = b;
 If[Abs[b.s2[[i]]] > λ1[[i]],
  B1 = Sign[b.s2[[i]]], B1 = Sign[Sin[ArcCos[b.s2[[i]]]]] + 0.001];
 outB1[[i]] = {b, B1, i}, {i, m}]
Do[If[Abs[bb1[[i]].s2[[i]]] > λ2[[i]],
  B2 = Sign[bb1[[i]].s2[[i]]], B2 = Sign[Sin[ArcCos[bb1[[i]].s2[[i]]]]] + 0.001];
 outB2[[i]] = {bb1[[i]], B2, i + m}, {i, m}]
outB = Catenate[{outB1, outB2}];
```

Statistical Analysis of the Particle Data Received from Alice and Bob

In[33]:=
```
m2 = 2 m;
theta = ConstantArray[0, m2];
th1 = ConstantArray[0, m2];
a1 = outA[[All, 1]];
A = outA[[All, 2]];
b1 = outB[[All, 1]];
B = outB[[All, 2]];
Do[ϕA1 = ArcTan[a1[[i]][[1]], a1[[i]][[2]]] / 50;
 ϕB1 = ArcTan[b1[[i]][[2]], b1[[i]][[1]]] / 50;
 If[ϕA1 * ϕB1 > 0, th1[[i]] = ArcCos[a1[[i]].b1[[i]]],
  th1[[i]] = 2 π - ArcCos[a1[[i]].b1[[i]]]];
 theta[[i]] = Round[th1[[i]] * 180 / π];
 th = theta[[i]];
 aliceD = A[[i]]; bobD = B[[i]];
 If[aliceD == 1, nAP[[th]] ++];
 If[bobD == 1, nBP[[th]] ++];
 If[aliceD == -1, nAN[[th]] ++];
 If[bobD == -1, nBN[[th]] ++];
 If[aliceD == 1 && bobD == 1, nPP[[th]] ++];
 If[aliceD == 1 && bobD == -1, nPN[[th]] ++];
 If[aliceD == -1 && bobD == 1, nNP[[th]] ++];
 If[aliceD == -1 && bobD == -1, nNN[[th]] ++], {i, m2}]
```

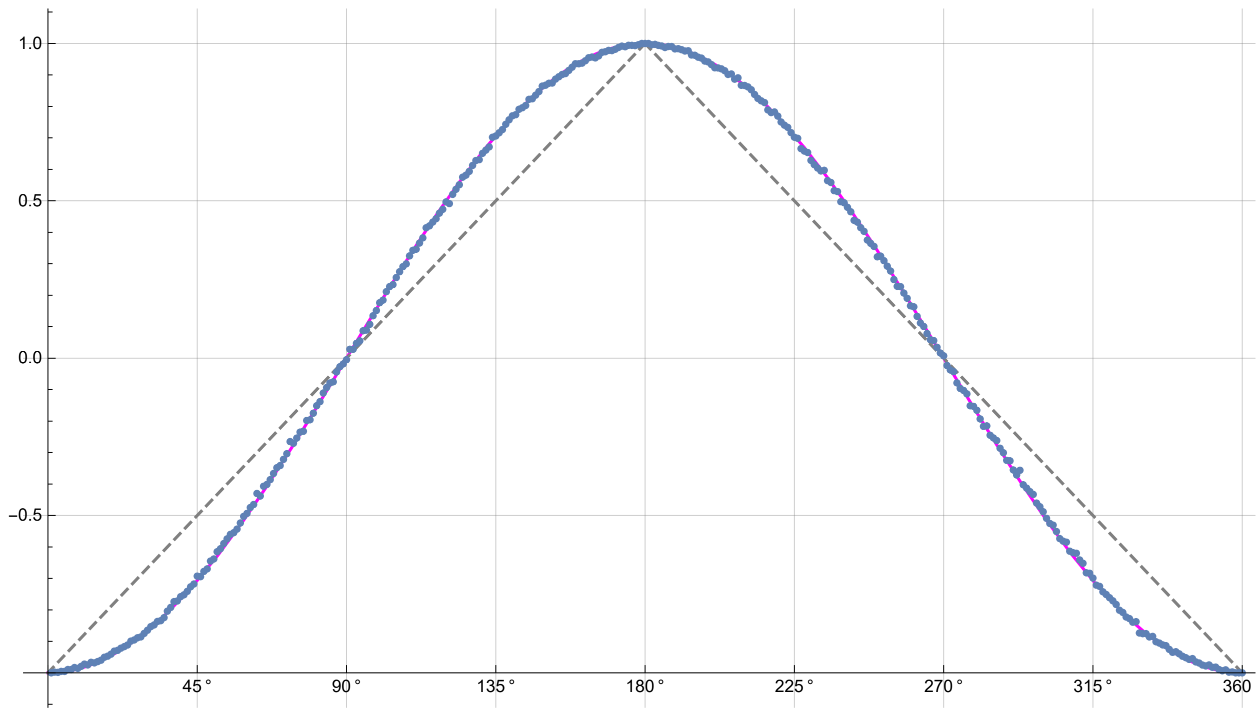
Calculating Mean Values of AB, and Plotting the Results

In[38]:=
```
mean = ConstantArray[0, trialDeg];
sum1 = ConstantArray[0, trialDeg];
sum2 = ConstantArray[0, trialDeg];
Do[sum1[[i]] = (nPP[[i]] + nNN[[i]] - nPN[[i]] - nNP[[i]]);
 sum2[[i]] = nPP[[i]] + nPN[[i]] + nNP[[i]] + nNN[[i]] + 0.0000001;
mean[[i]] = sum1[[i]] / sum2[[i]], {i, trialDeg}]
```

Comparing Mean Values with -Cosine Function and Computing Averages

In[42]:=
```
simulation = ListPlot[mean, PlotMarkers → {Automatic, Tiny}];
negcos = Plot[-Cos[x Degree], {x, 0, 360}, PlotStyle → {Magenta},
   AspectRatio → 9 / 16, Ticks → {{{0, 0 °}, {45, 45 °}, {90, 90 °}, {135, 135 °},
      {180, 180 °}, {225, 225 °}, {270, 270 °}, {315, 315 °}, {360, 360 °}}, Automatic},
   GridLines → Automatic, AxesOrigin → {0, -1.0}];
p1 = Plot[-1 + 2 x Degree / π, {x, 0, 180}, PlotStyle → {Gray, Dashed}];
p2 = Plot[3 - 2 x Degree / π, {x, 180, 360}, PlotStyle → {Gray, Dashed}];
Show[negcos, p1, p2, simulation]
```

Out[46]=



In[47]:=
```
A1 = ConstantArray[0, m2];
B1 = ConstantArray[0, m2];
Do[If[A[[i]] == 1 || A[[i]] == -1, A1[[i]] = A[[i]]];
 If[B[[i]] == 1 || B[[i]] == -1, B1[[i]] = B[[i]]], {i, m2}]
AveA = N[Sum[A1[[i]], {i, m2}] / m2];
AveB = N[Sum[B1[[i]], {i, m2}] / m2];
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}];
Print["Total Events = ", totAB]
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];
totP = PP + NN + PN + NP;
Print["Ave ++ = ", PP]
Print["Ave -- = ", NN]
Print["Ave +- = ", PN]
Print["Ave -+ = ", NP]
CHSH = Abs[N[mean[[23]]] - N[mean[[135]]] + N[mean[[68]]] + N[mean[[45]]]];
Print["Approx. CHSH = ", CHSH]
```

```
AveA = 0.0002832

AveB = 0.0002644

P(A+)= 0.500221

P(B+)= 0.500205

Total Events = 4980732

Ave ++ = 0.250328

Ave -- = 0.249775

Ave +- = 0.249944

Ave -+ = 0.249953

Approx. CHSH = 2.68206
```