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 Sep 2021
Some parts by Bill Nelson.

Set Run Time Parameters, Initialize Arrays and Tables

```
In[1]:= m = 20 000 000;
trialDeg = 721;
s1 = ConstantArray[0, m];
s2 = ConstantArray[0, m];
λ1 = ConstantArray[0, m];
λ2 = ConstantArray[0, m];
outA12 = Table[{0, 0}, m];
outA22 = Table[{0, 0}, m];
outB12 = Table[{0, 0}, m];
outB22 = Table[{0, 0}, m];
aa = ConstantArray[0, m];
bb = ConstantArray[0, m];
a1 = ConstantArray[0, m];
b1 = ConstantArray[0, m];
A1 = ConstantArray[0, m];
B1 = ConstantArray[0, m];
a2 = ConstantArray[0, m];
b2 = ConstantArray[0, m];
A2 = ConstantArray[0, m];
B2 = 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];
nPP2 = ConstantArray[0, trialDeg];
nNN2 = ConstantArray[0, trialDeg];
nPN2 = ConstantArray[0, trialDeg];
nNP2 = ConstantArray[0, trialDeg];
nAP2 = ConstantArray[0, trialDeg];
nBP2 = ConstantArray[0, trialDeg];
nAN2 = ConstantArray[0, trialDeg];
nBN2 = ConstantArray[0, trialDeg];
ϕ = 3; β = 0.2; ξ = 0; (*Adustable parameters for fine tuning*)
```

Generating Particle Data with Three Independent Do-Loops 0.93865

In[38]:=
```
Do[θ = RandomReal[{0, 360}]; (*Singlet vector angle*) (*Hidden Variable*)
 λ1[[i]] = β (Cos[θ/ϕ]^2);
 λ2[[i]] = 0.99 (Cos[θ/ϕ]^2);
 s1[[i]] = θ;
 s2[[i]] = θ + 180, {i, m}]     (*Conservation of angular momentum*)
```

In[39]:=
```
Do[a = RandomInteger[{0, 360}]; (*Detector vector angle 1 degree increments*)
 aa[[i]] = a;
 If[Abs[Cos[(a - s1[[i]]) Degree]] > λ1[[i]],
  Aa1 = Sign[Cos[(a - s1[[i]]) Degree]], Aa1 = { }];
 outA12[[i]] = {a, Aa1}, {i, m}]
Do[If[Abs[Cos[(aa[[i]] - s1[[i]]) Degree]] < λ2[[i]],
  Aa2 = Sign[Sin[(aa[[i]] - s1[[i]] + ξ) Degree]], Aa2 = { }];
 outA22[[i]] = {aa[[i]], Aa2}, {i, m}]
outA1 = Take[outA12, 0.9 * m];
outA2 = Take[outA22, 0.9 * m];
a1 = outA1[[All, 1]];
A1 = outA1[[All, 2]];
a2 = outA2[[All, 1]];
A2 = outA2[[All, 2]];
```

In[47]:=
```
Do[b = RandomInteger[{0, 360}]; (*Detector vector angle 1 degree increments*)
 bb[[i]] = b;
 If[Abs[Cos[(b - s2[[i]]) Degree]] > λ1[[i]],
  Bb1 = Sign[Cos[(b - s2[[i]]) Degree]], Bb1 = { }];
 outB12[[i]] = {b, Bb1}, {i, m}]
Do[If[Abs[Cos[(bb[[i]] - s2[[i]]) Degree]] < λ2[[i]],
  Bb2 = Sign[Sin[(bb[[i]] - s2[[i]] + ξ) Degree]], Bb2 = { }];
 outB22[[i]] = {bb[[i]], Bb2}, {i, m}]
outB1 = Take[outB12, 0.9 * m];
outB2 = Take[outB22, 0.9 * m];
b1 = outB1[[All, 1]];
B1 = outB1[[All, 2]];
b2 = outB2[[All, 1]];
B2 = outB2[[All, 2]];
```

Statistical Analysis of the Particle Data Received from Alice and Bob

In[55]:=
```
Do[θ1 = a1[[i]] - b1[[i]] + 361;
 (*All angles are shifted by 361 degrees since θ1 is an index*)
 aliceD = A1[[i]]; bobD = B1[[i]];
 If[aliceD == 1, nAP[[θ1]] ++];
 If[bobD == 1, nBP[[θ1]] ++];
 If[aliceD == -1, nAN[[θ1]] ++];
 If[bobD == -1, nBN[[θ1]] ++];
 If[aliceD == 1 && bobD == 1, nPP[[θ1]] ++];
 If[aliceD == 1 && bobD == -1, nPN[[θ1]] ++];
 If[aliceD == -1 && bobD == 1, nNP[[θ1]] ++];
 If[aliceD == -1 && bobD == -1, nNN[[θ1]] ++], {i, 0.9 * m}]
```

In[56]:=
```
Do[θ2 = a2[[i]] - b2[[i]] + 361;
 (*All angles are shifted by 361 degrees since θ1 is an index*)
 aliceD2 = A2[[i]]; bobD2 = B2[[i]];
 If[aliceD2 == 1, nAP2[[θ2]] ++];
 If[bobD2 == 1, nBP2[[θ2]] ++];
 If[aliceD2 == -1, nAN2[[θ2]] ++];
 If[bobD2 == -1, nBN2[[θ2]] ++];
 If[aliceD2 == 1 && bobD2 == 1, nPP2[[θ2]] ++];
 If[aliceD2 == 1 && bobD2 == -1, nPN2[[θ2]] ++];
 If[aliceD2 == -1 && bobD2 == 1, nNP2[[θ2]] ++];
 If[aliceD2 == -1 && bobD2 == -1, nNN2[[θ2]] ++], {i, 0.9 * m}]
```

Calculating Mean Values of A, B, and AB, and Plotting the Results

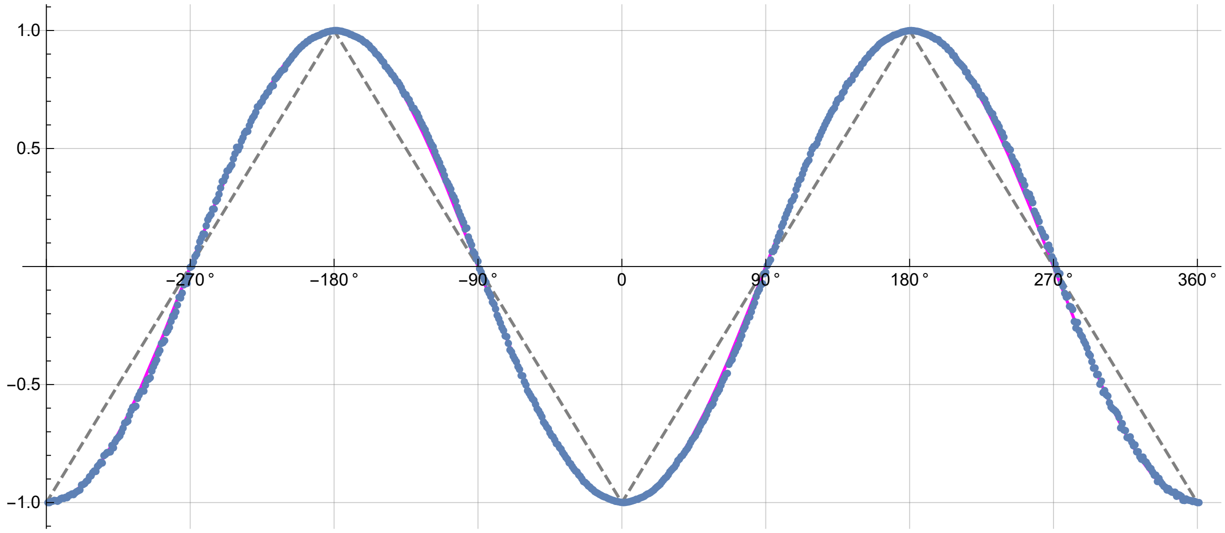
In[57]:=
```
sum1 = ConstantArray[0, trialDeg];
sum2 = ConstantArray[0, trialDeg];
mean = ConstantArray[0, trialDeg];
Do[sum1[[i]] = (nPP[[i]] + nNN[[i]] - nPN[[i]] - nNP[[i]]);
 sum2[[i]] = nPP[[i]] + nPN[[i]] + nNP[[i]] + nNN[[i]] + 0.0000001;
 mean[[i]] = sum1[[i]] / sum2[[i]], {i, trialDeg}]
sum12 = ConstantArray[0, trialDeg];
sum22 = ConstantArray[0, trialDeg];
mean2 = ConstantArray[0, trialDeg];
Do[sum12[[i]] = (nPP2[[i]] + nNN2[[i]] - nPN2[[i]] - nNP2[[i]]);
 sum22[[i]] = nPP2[[i]] + nPN2[[i]] + nNP2[[i]] + nNN2[[i]] + 0.0000001;
 mean2[[i]] = sum12[[i]] / sum22[[i]], {i, trialDeg}]
```

In[65]:=
```
simulation = ListPlot[(mean + mean2) / 2, PlotMarkers → {Automatic, Tiny}];
negcos = Plot[-Cos[x Degree], {x, 0, 720}, PlotStyle → {Magenta}, AspectRatio → 7 / 16,
   Ticks → {{{0, -360 °}, {90, -270 °}, {180, -180 °}, {270, -90 °}, {360, 0 °}, {450, 90 °},
      {540, 180 °}, {630, 270 °}, {720, 360 °}}, Automatic}, GridLines → Automatic];
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}];
p3 = Plot[-5 + 2 x Degree / π, {x, 360, 540}, PlotStyle → {Gray, Dashed}];
p4 = Plot[7 - 2 x Degree / π, {x, 540, 720}, PlotStyle → {Gray, Dashed}];
```

Comparing Mean Values with -Cosine Function and Computing Averages

In[71]:=
```
Show[negcos, p1, p2, p3, p4, simulation]
```

Out[71]=



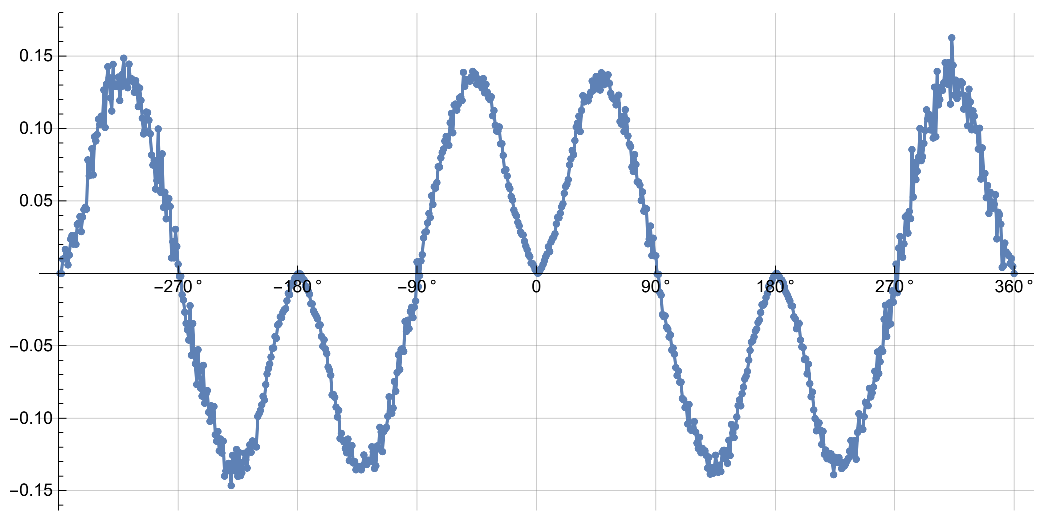
In[72]:=
```
A3 = DeleteCases[A1, {}];
m2 = Length[A3];
B3 = DeleteCases[B1, {}];
m3 = Length[B3];
AveA = Expand[N[Sum[A3[[i]], {i, m2}] / m2]];
AveB = Expand[N[Sum[B3[[i]], {i, m3}] / m3]];
Print["AveA = ", AveA]
Print["AveB = ", AveB]
PAP = N[Sum[nAP[[i]], {i, trialDeg}]];
PBP = N[Sum[nBP[[i]], {i, 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}];
totAB2 = Sum[nPP2[[i]] + nNN2[[i]] + nPN2[[i]] + nNP2[[i]], {i, trialDeg}];
totAB3 = totAB + totAB2;
Print["AB1 Events1 = ", totAB]
Print["AB2 Events2 = ", totAB2]
Print["Total Events = ", totAB3]
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];
Print["Ave ++ = ", PP]
Print["Ave -- = ", NN]
Print["Ave +- = ", PN]
Print["Ave -+ = ", NP]
mean3 = (mean + mean2) / 2;
CHSH = Abs[N[mean3[[315]]] - N[mean3[[225]]] + N[mean3[[405]]] + N[mean3[[45]]]];
Print["Approx. CHSH = ", CHSH]
```

```
AveA = -0.000124542

AveB = 0.000146885

P(A+) = 0.499938

P(B+) = 0.500073

AB1 Events1 = 15 800 158

AB2 Events2 = 4 197 480

Total Events = 19 997 638

Ave ++ = 0.24998

Ave -- = 0.249992

Ave +- = 0.249925

Ave -+ = 0.250103

Approx. CHSH = 2.88153
```

Deviation from negative cosine curve

In[105]:=
```
dev1 = ConstantArray[2, 720];
dev2 = ConstantArray[2, 720];
dev3 = ConstantArray[2, 720];
Do[dev1 = mean[[i]]; dev2[[i]] = {dev1, i}, {i, 720}]
devang = dev2[[All, 2]] - 361;
Do[dev3[[i]] = mean[[i]] + Cos[devang[[i]] Degree], {i, 720}]
ListPlot[N[dev3], PlotMarkers → {Automatic, Tiny}, Joined → True, AspectRatio → 1/2,
 Ticks → {{{0, -360 °}, {90, -270 °}, {180, -180 °}, {270, -90 °}, {360, 0 °}, {450, 90 °},
    {540, 180 °}, {630, 270 °}, {720, 360 °}}, Automatic}, GridLines → Automatic]
```

Out[111]=



In[112]:=
```
Mean[N[dev3]]
Mean[N[Abs[dev3]]]
```

Out[112]= 0.00032054

Out[113]= 0.0773954