

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 non-local Aug 2021

Set Run Time Parameters, Initialize Arrays and Tables

```
In[761]:= m = 20000000;
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[776]:= Do[e = RandomReal[{0, 360}];
λ = 0.32 (Cos[ $\frac{e}{3.1}$ ] ^ 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 - 16) 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 - 16) Degree]], B = Sign[Cos[(b - e) Degree]]];
AB[[i]] = {a, b, A, B}, {i, m}]
```

Statistical Analysis of Particle Data

```
In[777]:= 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[780]:= 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[783]:= simulation = ListPlot[mean, PlotMarkers -> {Automatic, Tiny}];
negcos = Plot[-Cos[x Degree], {x, 0, 720}, PlotStyle -> {Magenta}, 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];

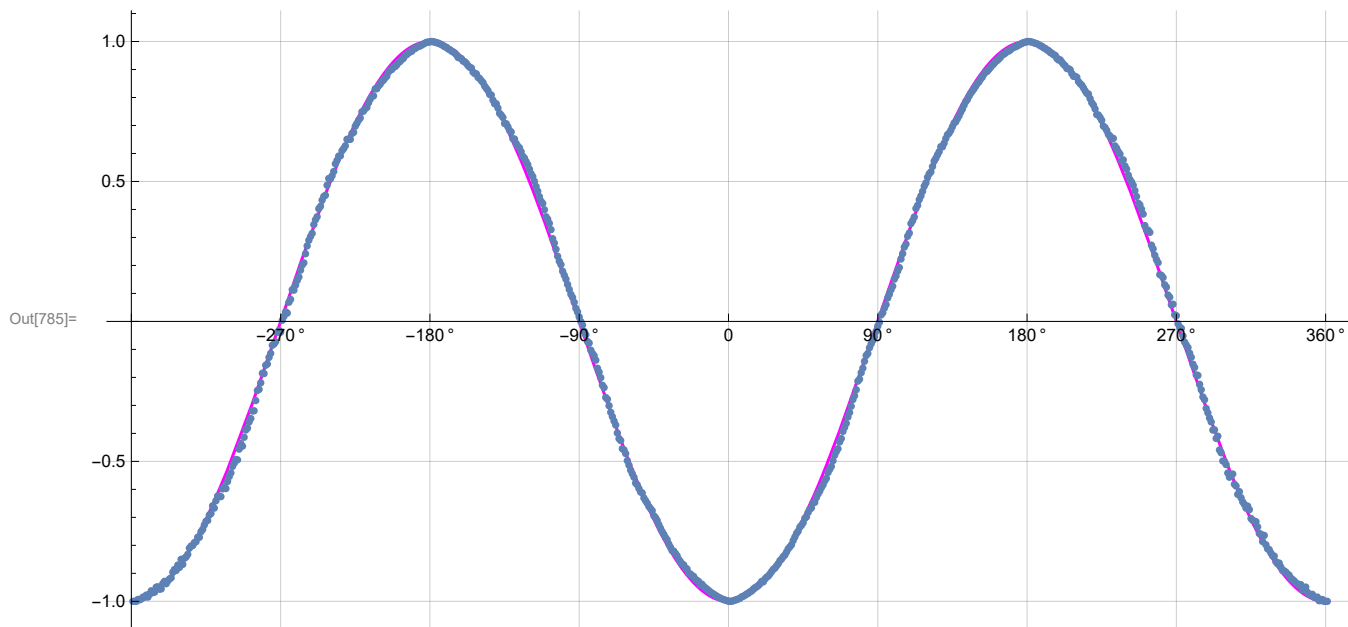
```

Compare mean values with -Cosine Curve and compute averages

```

In[785]:= Show[negcos, simulation]

```



```

In[786]:= 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 = -0.0001282

AveB = 0.000058

P(A+) = 0.499936

P(B+) = 0.500029

Out[798]= 20 000 000

Out[799]= 0.249964

Out[800]= 0.249999

Out[801]= 0.249972

Out[802]= 0.250065

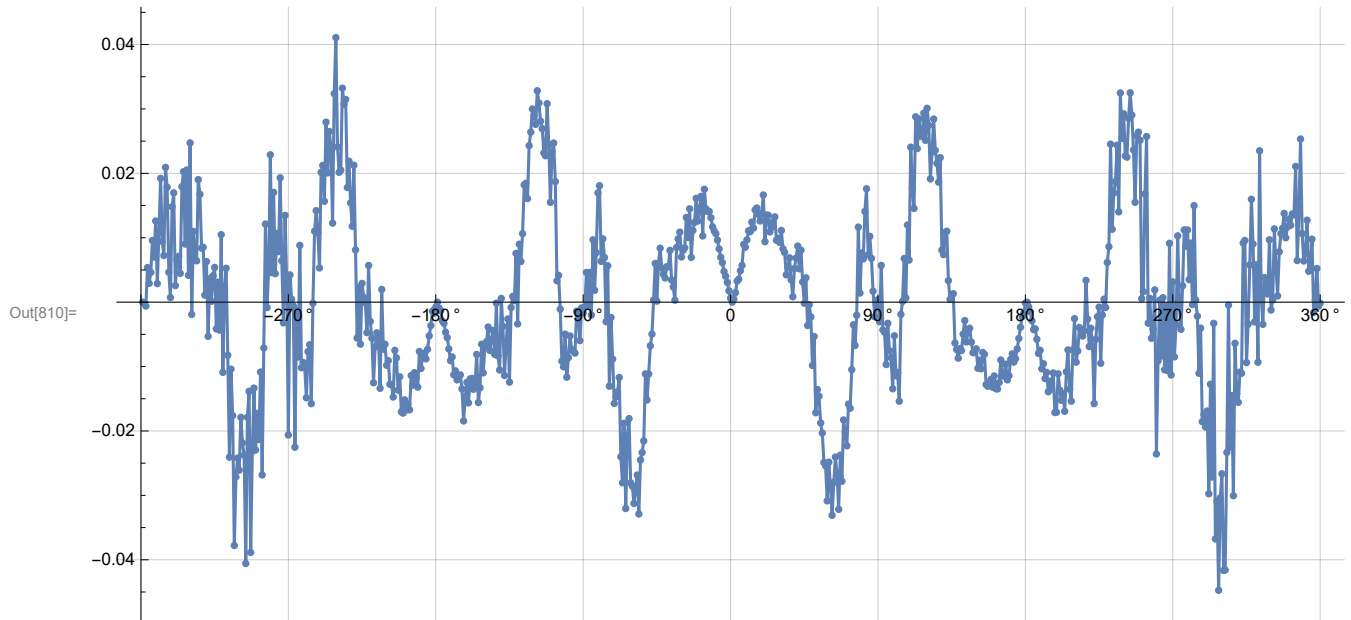
Out[803]= 2.83488

```

```

In[804]:= 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]

```



```

In[811]:= N[Mean[Abs[dev3]]]

```

Out[811]= 0.0114478