

Simulation Based on Michel Fodje's epr-simple simulation translated from Python to Mathematica by John Reed 13 Nov 2013 and Quaternions Modified by Fred Diether for Completely Local-Realistic Sep 2021 Some parts by Bill Nelson. Includes Joy's S^3 Quaternion Model.

Load Quaternion Package, Set Run Time Parameters, Initialize Arrays and Tables

```
In[423]:= << Quaternions` ;
β0 = Quaternion[1, 0, 0, 0];
β1 = Quaternion[0, 1, 0, 0];
β2 = Quaternion[0, 0, 1, 0];
β3 = Quaternion[0, 0, 0, 1];
Qcoordinates = {β1, β2, β3};
m = 50000;
trialDeg = 721;
Ls1 = ConstantArray[0, m];
Ls2 = ConstantArray[0, m];
λ1 = ConstantArray[0, m];
λ2 = ConstantArray[0, m];
outAa = Table[{0, 0, 0, 0, 0, 0}, m];
outBb = Table[{0, 0, 0, 0, 0, 0}, m];
outA12 = Table[{0, 0, 0, 0, 0, 0}, m];
outA22 = Table[{0, 0, 0, 0, 0, 0}, m];
outB12 = Table[{0, 0, 0, 0, 0, 0}, m];
outB22 = Table[{0, 0, 0, 0, 0, 0}, m];
plotq = Table[{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 = 3.1; β = 0.32; ξ = - $\frac{4\pi}{45}$ ; (*Adustable parameters for fine tuning*)
```

Generating Particle Data with Three Independent Do-Loops

```
In[455]:= Do[θ = RandomReal[{-179, 180}]; (*Singlet vector angle*) (*Hidden Variable*)
λ1[[i]] = β (Cos[ $\frac{\theta}{\phi}$ ] ^ 2);
λ2[[i]] = Sign[θ];
eθ = N[Flatten[{FromPolarCoordinates[{1, θ * π/180}], 0}]];
Ls1[[i]] = λ2[[i]] * eθ.Qcoordinates;
Ls2[[i]] = -Ls1[[i]], {i, m} (*Conservation of angular momentum*)
```

```
In[456]:= Do[a = RandomInteger[{-179, 180}]; (*Detector vector angle 1 degree increments*)
  aa = N[Flatten[{FromPolarCoordinates[{1, a *  $\pi$ /180}], 0}]];
  Da = aa.Qcoordinates; (*Convert to quaternion coordinates*)
  qa = Da ** Ls1[[i]];
  aq = -(Ls1[[i]] ** Da);
  If[Abs[Re[qa]] >  $\lambda$ 1[[i]],
    Aa = Re[Da ** Limit[Ls1[[i]], Ls1[[i]] -> Sign[Re[Da ** Ls1[[i]]]] Da]],
    Aa =  $\lambda$ 2[[i]] * Sign[qa[[4]] +  $\xi$ ]];
  A0 =  $\lambda$ 2[[i]] * Sign[qa[[4]] +  $\xi$ ]];
  outAa[[i]] = {a, Aa, i, A0, qa, aq};
  If[Abs[Re[qa]] >  $\lambda$ 1[[i]], outA12[[i]] = outAa[[i]], outA22[[i]] = outAa[[i]], {i, m}]
outA1 = DeleteCases[outA12, {0, 0, 0, 0, 0, 0}]; (*Split outAa into outA1 and outA2*)
outA2 = DeleteCases[outA22, {0, 0, 0, 0, 0, 0}];
```

```
In[458]:= Do[b = RandomInteger[{-179, 180}]; (*Detector vector angle 1 degree increments*)
  bb = N[Flatten[{FromPolarCoordinates[{1, b *  $\pi$ /180}], 0}]];
  Db = bb.Qcoordinates; (*Convert to quaternion coordinates*)
  qb = Ls2[[i]] ** Db;
  bq = -(Db ** Ls2[[i]]);
  If[Abs[Re[qb]] >  $\lambda$ 1[[i]],
    B = Re[Db ** Limit[Ls2[[i]], Ls2[[i]] -> Sign[Re[Db ** Ls2[[i]]]] Db]],
    B = - $\lambda$ 2[[i]] * Sign[qb[[4]] +  $\xi$ ]];
  B0 = - $\lambda$ 2[[i]] * Sign[qb[[4]] +  $\xi$ ]];
  outBb[[i]] = {b, B, i, B0, qb, bq};
  If[Abs[Re[qb]] >  $\lambda$ 1[[i]], outB12[[i]] = outBb[[i]], outB22[[i]] = outBb[[i]], {i, m}]
outB1 = DeleteCases[outB12, {0, 0, 0, 0, 0, 0}]; (*Split outBb into outB1 and outB2*)
outB2 = DeleteCases[outB22, {0, 0, 0, 0, 0, 0}];
```

Matching Events Observed by Alice and Bob using Trial Numbers

```
In[460]:= list13 = outA1[[All, 3]]; (*Two lists of only trial numbers used for matching*)
list23 = outB1[[All, 3]];
```

Local Detection Analysis of the Events Observed by Alice

```
In[462]:= listA3 = Select[outA1, Intersection[#[[3]], list23] != {#[[3]]} &];
M1 = Length[listA3];
tna = listA3[[All, 3]];
listA4 = Select[outA1, Intersection[#[[3]], tna] == {#[[3]]} &];
outA1a = Select[outA1, Intersection[#[[3]], tna] != {#[[3]]} &];
outA5 = Table[{0, 0, 0, 0, 0, 0}, M1];
qaaq = ConstantArray[0, M1];
listA35 = listA3[[All, 5]];
listA36 = listA3[[All, 6]];
Do[If[listA4[[i]][[2]] == listA4[[i]][[4]],
  qaaq[[i]] = 1, qaaq[[i]] = Re[listA35[[i]] ** listA36[[i]]]];
  outA5[[i]] = {listA4[[i]][[1]], qaaq[[i]] * listA4[[i]][[2]], listA4[[i]][[3]], listA4[[i]][[4]], listA4[[i]][[5]], listA4[[i]][[6]]}, {i, M1}] (*spinorial sign change*)
outA2a = Sort[Catenate[{outA5, outA2}], #1[[3]] < #2[[3]] &]; (*Combine lists and sort*)
outA = Sort[Catenate[{outA1a, outA2a}], #1[[3]] < #2[[3]] &];
a1 = outA[[All, 1]]; (*These results are what Alice observes as defined in Eq.(??)*)
A = outA[[All, 2]];

Local Detection Analysis of the Events Observed by Bob
```

```
In[474]:= listB3 = Select[outB1, Intersection[{#[[3]]}, list13] != {#[[3]]} &];
M2 = Length[listB3];
tnb = listB3[[All, 3]];
listB4 = Select[outB1, Intersection[{#[[3]]}, tnb] == {#[[3]]} &];
outB1b = Select[outB1, Intersection[{#[[3]]}, tnb] != {#[[3]]} &];
outB5 = Table[{0, 0, 0, 0, 0, 0}, M2];
qbbq = ConstantArray[0, M2];
listB35 = listB3[[All, 5]];
listB36 = listB3[[All, 6]];
Do[If[listB4[[i]][[2]] == listB4[[i]][[4]],
  qbbq[[i]] = 1, qbbq[[i]] = Re[listB35[[i]] ** listB36[[i]]];
  outB5[[i]] = {listB4[[i]][[1]], qbbq[[i]] * listB4[[i]][[2]], listB4[[i]][[3]], listB4[[i]][[4]], listB4[[i]][[5]], listB4[[i]][[6]]}, {i, M2}] (*spinorial sign change*)
outB2b = Sort[Catenate[{outB5, outB2}], #1[[3]] < #2[[3]] &]; (*Combine lists and sort*)
outB = Sort[Catenate[{outB1b, outB2b}], #1[[3]] < #2[[3]] &];
b1 = outB[[All, 1]]; (*These results are what Bob observes as defined in Eq.(??)*)
B = outB[[All, 2]];
```

Statistical Analysis of the Particle Data Received from Alice and Bob

```
In[486]:= theta = ConstantArray[0, m];
Do[ $\theta_1 = a_1[[j]] - b_1[[j]] + 361$ ;
  (*All angles are shifted by 361 degrees since  $\theta_1$  is an index*)
  theta[[j]] =  $\theta_1$ ;
  aliceD = A[[j]]; bobD = B[[j]];
  If[aliceD == 1, nAP[[ $\theta_1$ ]] ++];
  If[bobD == 1, nBP[[ $\theta_1$ ]] ++];
  If[aliceD == -1, nAN[[ $\theta_1$ ]] ++];
  If[bobD == -1, nBN[[ $\theta_1$ ]] ++];
  If[aliceD == 1 && bobD == 1, nPP[[ $\theta_1$ ]] ++];
  If[aliceD == 1 && bobD == -1, nPN[[ $\theta_1$ ]] ++];
  If[aliceD == -1 && bobD == 1, nNP[[ $\theta_1$ ]] ++];
  If[aliceD == -1 && bobD == -1, nNN[[ $\theta_1$ ]] ++], {j, m}]
```

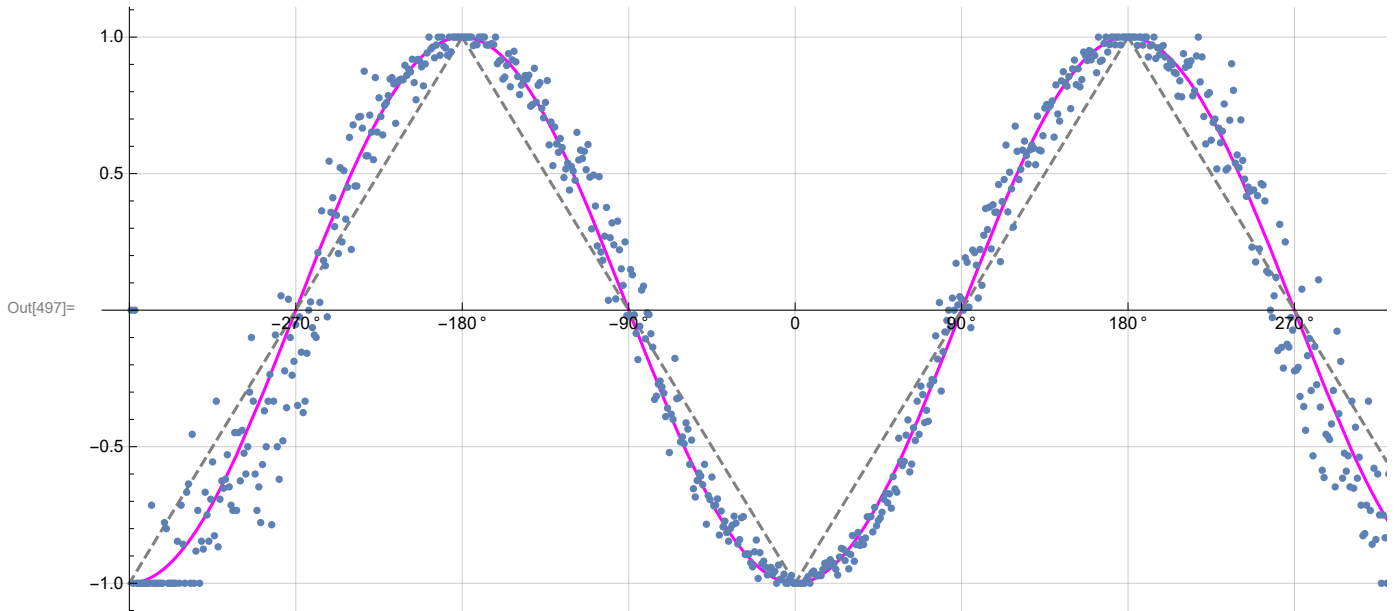
Calculating Mean Values of A, B, and AB, and Plotting the Results

```
In[488]:= 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[491]:= simulation = ListPlot[mean, 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 /  $\pi$ , {x, 0, 180}, PlotStyle -> {Gray, Dashed}];
p2 = Plot[3 - 2 x Degree /  $\pi$ , {x, 180, 360}, PlotStyle -> {Gray, Dashed}];
p3 = Plot[-5 + 2 x Degree /  $\pi$ , {x, 360, 540}, PlotStyle -> {Gray, Dashed}];
p4 = Plot[7 - 2 x Degree /  $\pi$ , {x, 540, 720}, PlotStyle -> {Gray, Dashed}];
```

Comparing Mean Values with -Cosine Function and Computing Averages

In[497]:= Show[negcos, p1, p2, p3, p4, simulation]



```
In[498]:= AveA = N[Sum[A[[i]], {i, m}]/m];
AveB = N[Sum[B[[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 = -0.00476

AveB = -0.00172

P(A+) = 0.49762

P(B+) = 0.49914

Out[510]= 50 000

Out[511]= 0.24954

Out[512]= 0.25278

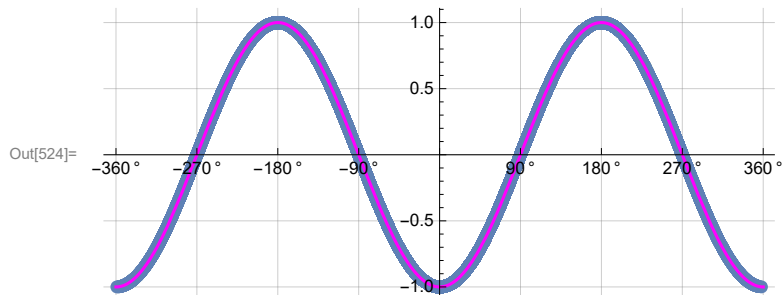
Out[513]= 0.24808

Out[514]= 0.2496

Out[515]= 2.69781

Product Calculation

```
In[516]:= q = 0;
q2a = outA[[All, 6]];
q2b = outB[[All, 6]];
plotq = Table[{0, 0}, m];
angle = ConstantArray[0, m];
Do[If[λ2[[i]] == 1, q = q2a[[i]] ** q2b[[i]], q = q2b[[i]] ** q2a[[i]]];
  angle = theta[[i]] - 361;
  plotq[[i]] = {angle, Re[q]}, {i, m}]
sim = ListPlot[plotq, PlotMarkers → {Automatic, Small}, AspectRatio → 7/16,
  Ticks → {{{-360, -360°}, {-270, -270°}, {-180, -180°}, {-90, -90°}, {0, 0°}, {90, 90°},
    {180, 180°}, {270, 270°}, {360, 360°}}, Automatic}, GridLines → Automatic];
negcos1 = Plot[-Cos[x Degree], {x, -360, 360}, PlotStyle → {Magenta}];
Show[sim, negcos1]
```

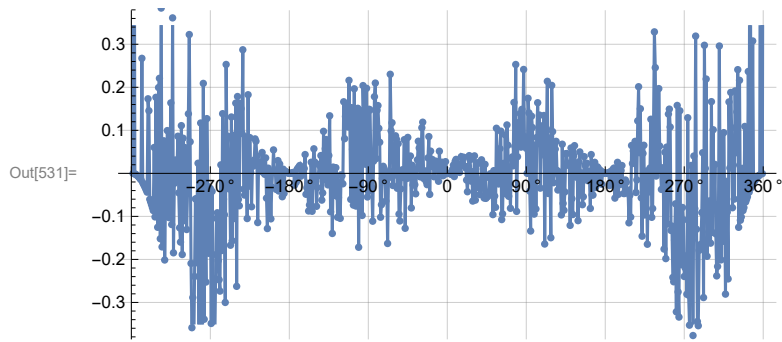


Blue is the correlation data and magenta is the -cosine curve for an exact match.

```

In[525]:= 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[532]:= Mean[N[dev3]]
Mean[N[Abs[dev3]]]

```

Out[532]= 0.00024208

Out[533]= 0.0876574