Heine's non-local HV model translated from R and modified by Fred Diether Feb 2021

Some parts by John Reed

Set Run Time Parameters, Initialize Arrays and Tables

In[385]:=
```
trials = 5 000 000;
trialDeg = 360;
CA = Table[{0, 0}, trials];
CB = Table[{0, 0}, trials];
a1 = ConstantArray[0, trials];
b1 = ConstantArray[0, trials];
A1 = ConstantArray[0, trials];
B1 = ConstantArray[0, trials];
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[401]:=
```
Do[λ = RandomReal[{-1, 1}];
 a = RandomPoint[Sphere[]];
 b = RandomPoint[Sphere[]];
 L = (1 + a.b) / 4;
 If[Abs[λ] < L, A = Sign[λ],
  If[Abs[λ] < 2 * L, A = -Sign[λ], If[Abs[λ] < L + 0.5, A = -1, A = 1]]];
 If[Abs[-λ] < L, B = Sign[-λ], If[Abs[-λ] < 2 * L,
   B = -Sign[-λ], If[Abs[-λ] < L + 0.5, B = -1, B = 1]]];
 CA[[j]] = {a, A};
 CB[[j]] = {b, B}, {j, trials}]
```

Statistical Analysis of Particle Data

In[402]:=
```
a1 = CA[[All, 1]]; b1 = CB[[All, 1]]; A1 = CA[[All, 2]]; B1 = CB[[All, 2]];
Do[ϕA = ArcTan[a1[[j]][[2]], a1[[j]][[1]]];
 ϕB = ArcTan[b1[[j]][[1]], b1[[j]][[2]]];
 If[ϕA * ϕB > 0, angle = ArcCos[a1[[j]].b1[[j]]] / Degree,
  angle = (-ArcCos[a1[[j]].b1[[j]]]) / Degree + 360];
 θ = Round[angle];
 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, trials}]
```

Calculate Mean Values

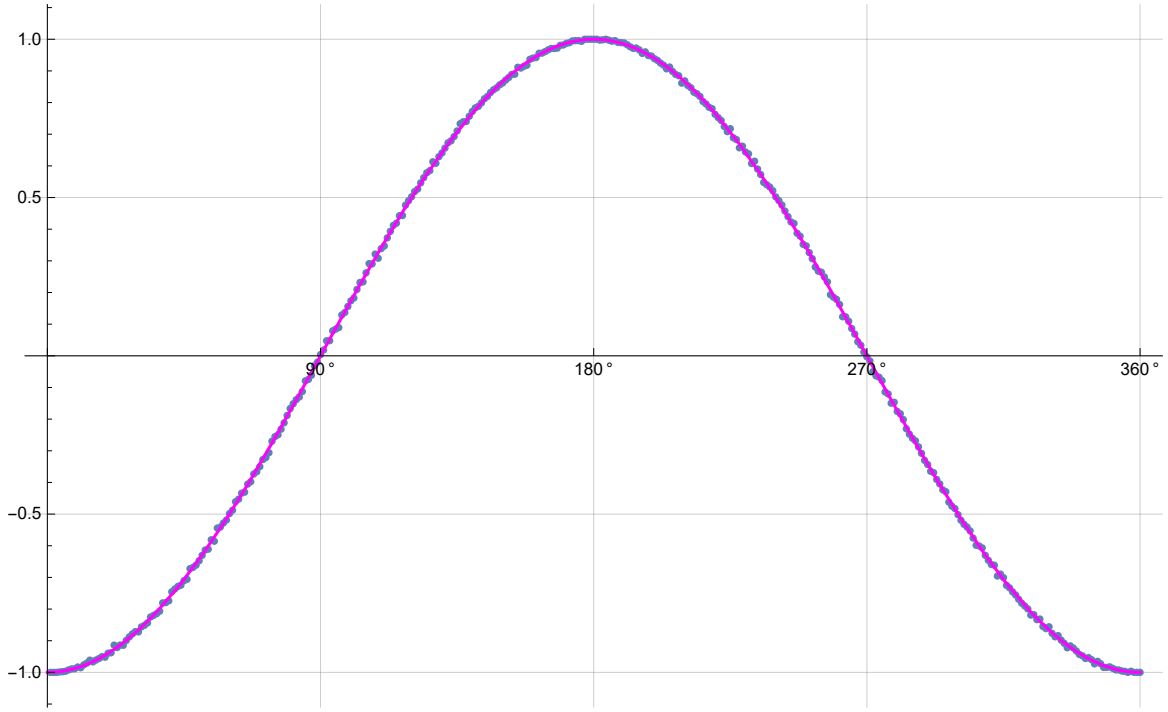
In[404]:=
```
pPP = 0; pPN = 0; pNP = 0; pNN = 0;
mean = ConstantArray[0, trialDeg];
Do[sum = nPP[[i]] + nPN[[i]] + nNP[[i]] + nNN[[i]];
If[sum == 0, Goto[jump],
  {pPP = nPP[[i]] / sum;
   pNP = nNP[[i]] / sum;
   pPN = nPN[[i]] / sum;
   pNN = nNN[[i]] / sum;
mean[[i]] = pPP + pNN - pPN - pNP}];
 Label[jump], {i, trialDeg}]
```

In[407]:=
```
simulation = ListPlot[mean, PlotMarkers → {Automatic, Tiny},
   Ticks → {{{0, 0 °}, {90, 90 °}, {180, 180 °}, {270, 270 °}, {360, 360 °}}, Automatic},
   GridLines → Automatic];
negcos = Plot[-Cos[x Degree], {x, 0, 360}, PlotStyle → {Magenta}] ;
```

Compare mean values with -Cosine Curve and compute averages

In[409]:=
```
Show[simulation, negcos]
```

Out[409]=



In[410]:=
```
AveA = N[Sum[A1[[i]], {i, trials}] / trials];
AveB = N[Sum[B1[[i]], {i, trials}] / trials];
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[[22]]] + N[mean[[68]]] + N[mean[[45]]] - N[mean[[135]]]]
```

```
AveA = -0.0000748
AveB = 0.0000616
P(A+)= 0.499963
P(B+)= 0.500031
```

Out[422]= 4 999 948

Out[423]= 0.249831

Out[424]= 0.249837

Out[425]= 0.250132

Out[426]= 0.2502

Out[427]= 2.7079