Joy Christian's S^7 model for GHZ 4 particle 2D correlations using Geometric Algebra by Fred Diether Apr. 2021

In[183]:=
```
<< "clifford.m"
s = 0; t = 0; u = 0; w = 0; x1 = 0;
m = 30000;
plotArray = Table[{0, 0}, m];
plotArray1 = Table[{0, 0}, m];
plotArray2 = Table[{0, 0}, m];
plotArray3 = Table[{0, 0}, m];
plotArray4 = Table[{0, 0}, m];
plotArray5 = Table[{0, 0}, m];
plotArray6 = Table[{0, 0}, m];
I3 = Pseudoscalar[3];
```

In[194]:=
```
Do[ar = ToBasis[RandomPoint[Circle[]]] / √2;  (*convert to GA*)
 ad = ToBasis[Normalize[ToVector[InnerProduct[ar, (e[1] * e[2])], 2]]] / √2;
 Da = GeometricProduct[I3, ar] + GeometricProduct[ad, e[4]];
 br = ToBasis[RandomPoint[Circle[]]] / √2;  (*convert to GA*)
 bd = ToBasis[Normalize[ToVector[InnerProduct[br, (e[1] * e[2])], 2]]] / √2;
 Db = GeometricProduct[I3, br] + GeometricProduct[bd, e[4]];
 cr = ToBasis[RandomPoint[Circle[]]] / √2;  (*convert to GA*)
 cd = ToBasis[Normalize[ToVector[InnerProduct[cr, (e[1] * e[2])], 2]]] / √2;
 Dc = GeometricProduct[I3, cr] + GeometricProduct[cd, e[4]];
 dr = ToBasis[RandomPoint[Circle[]]] / √2;  (*convert to GA*)
 dd = ToBasis[Normalize[ToVector[InnerProduct[dr, (e[1] * e[2])], 2]]] / √2;
 Dd = GeometricProduct[I3, dr] + GeometricProduct[dd, e[4]];
 s11 = RandomPoint[Circle[]];  (*Singlet spin vector 2D*)
 s1 = ToBasis[s11];  (*Singlet spin vector 2D*)
 s33 = RandomPoint[Circle[]]; (*Singlet spin vector 2D*)
 s3 = ToBasis[s33]; (*Singlet spin vector 2D*)
 Ls1 = InnerProduct[I3, s1];  (*singlet bivector*)
 Ls2 = -Ls1;  (*conservation of angular momentum*)
 Ls3 = InnerProduct[I3, s3];  (*singlet bivector*)
 Ls4 = -Ls3;  (*conservation of angular momentum*)
 A = Sign[InnerProduct[Da, Ls1]]; (*A detector full polarization*)
 B = Sign[InnerProduct[Db, Ls2]]; (*B detector full polarization*)
 C1 = Sign[InnerProduct[Dc, Ls3]]; (*C detector full polarization*)
 D1 = Sign[InnerProduct[Dd, Ls4]]; (*D detector full polarization*)
 (*Sign functions are not used in product calc because they
    are a simplification of the limit replacement process
    and the limits drop out because Ls1**Ls2 = 1*)
 LA = GeometricProduct[-Da, Ls1];  (*Detector - spin interaction*)
```

```
LB = GeometricProduct[ Ls2, Db];
LC = GeometricProduct[-Dc, Ls3];
LD = GeometricProduct[Ls4, Dd];
LA1 = GeometricProduct[-Da, Ls1];
LB1 = GeometricProduct[ Ls3, Db];
LC1 = GeometricProduct[-Dc, Ls4];
LD1 = GeometricProduct[Ls2, Dd];
LA2 = GeometricProduct[-Da, Ls1];
LB2 = GeometricProduct[-Db, Ls3];
LC2 = GeometricProduct[Ls2, Dc];
LD2 = GeometricProduct[Ls4, Dd];
AB = GeometricProduct[LA, LB];
CD = GeometricProduct[LC, LD];
q = 0; q1 = 0; q2 = 0; q3 = 0; q4 = 0; q5 = 0; q6 = 0;
If[s11⟦2⟧ > 0, q = GeometricProduct[AB, CD], q = GeometricProduct[CD, AB]];
(*product calc*)
If[s11⟦2⟧ > 0, q1 = GeometricProduct[LA, LB], q1 = GeometricProduct[LB, LA]];
If[s33⟦2⟧ > 0, q2 = GeometricProduct[LC, LD], q2 = GeometricProduct[LD, LC]];
If[s11⟦2⟧ > 0, q3 = GeometricProduct[LA1, LD1], q3 = GeometricProduct[LD1, LA1]];
If[s33⟦2⟧ > 0, q4 = GeometricProduct[LB1, LC1], q4 = GeometricProduct[LC1, LB1]];
If[s11⟦2⟧ > 0, q5 = GeometricProduct[LA2, LC2], q5 = GeometricProduct[LC2, LA2]];
If[s33⟦2⟧ > 0, q6 = GeometricProduct[LB2, LD2], q6 = GeometricProduct[LD2, LB2]];
s = s + q;
t = t + A;
u = u + B;
w = w + C1;
x1 = x1 + D1;
ϕA = ArcTan[Expand[-Da / OuterProduct[e[3], e[1]]]⟦1⟧,
   Expand[Da / OuterProduct[e[2], e[3]]]⟦1⟧] / Degree;
ϕB = ArcTan[Expand[Db / OuterProduct[e[3], e[1]]]⟦1⟧,
   Expand[Db / OuterProduct[e[2], e[3]]]⟦1⟧] / Degree;
ϕC = ArcTan[Expand[Dc / OuterProduct[e[3], e[1]]]⟦1⟧,
   Expand[Dc / OuterProduct[e[2], e[3]]]⟦1⟧] / Degree;
ϕD = ArcTan[Expand[Dd / OuterProduct[e[3], e[1]]]⟦1⟧,
   Expand[-Dd / OuterProduct[e[2], e[3]]]⟦1⟧] / Degree;
angle  =  ϕA + ϕB - ϕC - ϕD;
If[ϕA * ϕB > 0, angle1 = ArcCos[GeometricProduct[Da, Db]⟦1⟧] / Degree,
 angle1 = (-ArcCos[GeometricProduct[Da, Db]⟦1⟧]) / Degree];
If[ϕC * ϕD > 0, angle2 = ArcCos[GeometricProduct[Dc, Dd]⟦1⟧] / Degree,
 angle2 = (-ArcCos[GeometricProduct[Dc, Dd]⟦1⟧]) / Degree];
If[ϕA * ϕD > 0, angle3 = ArcCos[GeometricProduct[Da, Dd]⟦1⟧] / Degree,
 angle3 = (-ArcCos[GeometricProduct[Da, Dd]⟦1⟧]) / Degree];
If[ϕB * ϕC > 0, angle4 = ArcCos[GeometricProduct[Db, Dc]⟦1⟧] / Degree,
 angle4 = (-ArcCos[GeometricProduct[Db, Dc]⟦1⟧]) / Degree];
If[ϕA * ϕC > 0, angle5 = ArcCos[GeometricProduct[Da, Dc]⟦1⟧] / Degree,
 angle5 = (-ArcCos[GeometricProduct[Da, Dc]⟦1⟧]) / Degree];
If[ϕB * ϕD > 0, angle6 = ArcCos[GeometricProduct[Db, Dd]⟦1⟧] / Degree,
```

```
 angle6 = (-ArcCos[GeometricProduct[Db, Dd]⟦1⟧]) / Degree];
plotArray1⟦i⟧ = {angle1, q1⟦1⟧};
plotArray2⟦i⟧ = {angle2, q2⟦1⟧};
plotArray3⟦i⟧ = {angle3, q3⟦1⟧};
plotArray4⟦i⟧ = {angle4, q4⟦1⟧};
plotArray5⟦i⟧ = {angle5, q5⟦1⟧};
plotArray6⟦i⟧ = {angle6, q6⟦1⟧};
plotArray⟦i⟧ = {angle, q⟦1⟧}, {i, m}]
```

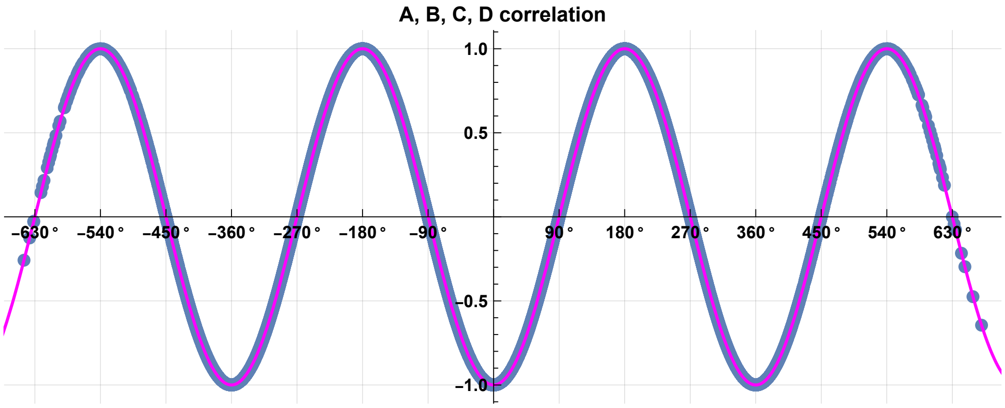
In[195]:=
```
meanq = Expand[N[s / m]];  (*shows the vanishing of cross products*)
aveA = N[t / m];
aveB = N[u / m];
aveC = N[w / m];
aveD = N[x1 / m];
Print["  <A> = ", aveA, "  <B> = ", aveB, "  <C> = ", aveC, "  <D> = ", aveD];
Print["Cross products vanish, meanq = ", meanq];
simulation = ListPlot[plotArray, PlotLabel → "A, B, C, D correlation",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   AspectRatio → 3 / 8, Ticks → {{{-720, -720 °}, {-630, -630 °}, {-540, -540 °},
      {-450, -450 °}, {-360, -360 °}, {-270, -270 °}, {-180, -180 °}, {-90, -90 °},
      {0, 0 °}, {90, 90 °}, {180, 180 °}, {270, 270 °}, {360, 360 °}, {450, 450 °},
      {540, 540 °}, {630, 630 °}, {720, 720 °}}, Automatic}, GridLines → Automatic];
simulation1 = ListPlot[plotArray1, PlotLabel → "A correlated to B",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   Ticks → {{{-180, -180 °}, {-90, -90 °}, {0, 0 °}, {90, 90 °}, {180, 180 °}}, Automatic},
   GridLines → Automatic];
simulation2 = ListPlot[plotArray2, PlotLabel → "C correlated to D",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   Ticks → {{{-180, -180 °}, {-90, -90 °}, {0, 0 °}, {90, 90 °}, {180, 180 °}}, Automatic},
   GridLines → Automatic];
simulation3 = ListPlot[plotArray3, PlotLabel → "A correlated to D",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   Ticks → {{{-180, -180 °}, {-90, -90 °}, {0, 0 °}, {90, 90 °}, {180, 180 °}}, Automatic},
   GridLines → Automatic];
simulation4 = ListPlot[plotArray4, PlotLabel → "B correlated to C",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   Ticks → {{{-180, -180 °}, {-90, -90 °}, {0, 0 °}, {90, 90 °}, {180, 180 °}}, Automatic},
   GridLines → Automatic];
simulation5 = ListPlot[plotArray5, PlotLabel → "A correlated to C",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   Ticks → {{{-180, -180 °}, {-90, -90 °}, {0, 0 °}, {90, 90 °}, {180, 180 °}}, Automatic},
   GridLines → Automatic];
simulation6 = ListPlot[plotArray6, PlotLabel → "B correlated to D",
   LabelStyle → Directive[Bold, Black], PlotMarkers → {Automatic, Small},
   Ticks → {{{-180, -180 °}, {-90, -90 °}, {0, 0 °}, {90, 90 °}, {180, 180 °}}, Automatic},
   GridLines → Automatic];
negcos = Plot[-Cos[x Degree], {x, -720, 720}, PlotStyle → {Magenta}];
Show[simulation, negcos ]
Show[simulation1, negcos ]
Show[simulation2, negcos ]
Show[simulation3, negcos ]
Show[simulation4, negcos ]
Show[simulation5, negcos ]
Show[simulation6, negcos ]
```

<A> = −0.000333333 <B> = −0.00126667 <C> = 0.0064 <D> = 0.0088

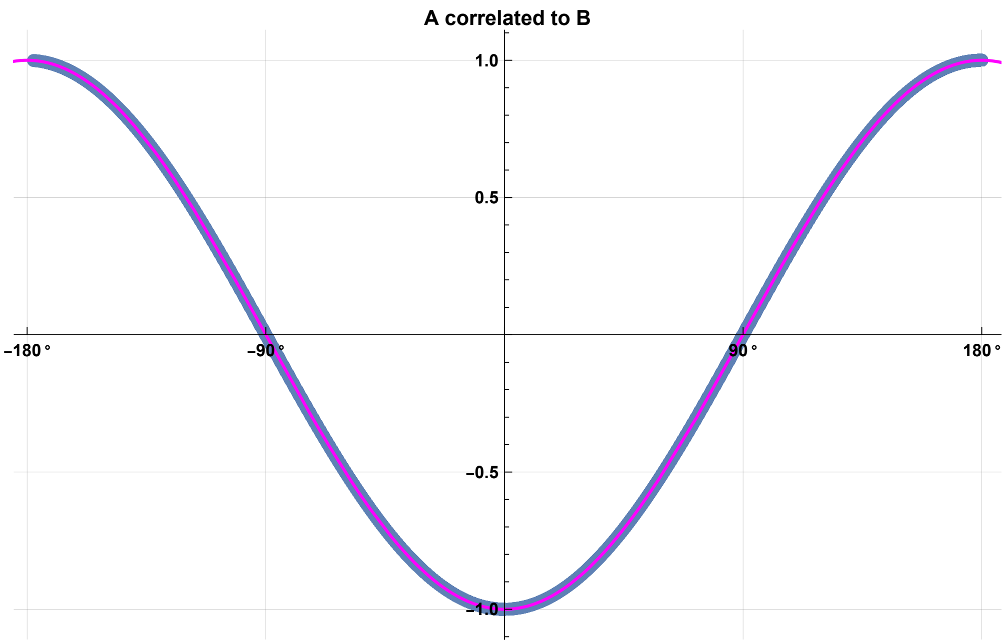
Cross products vanish, meanq =

$-0.00561156 + 0.000272403\, e_{1.}\, e_{2.} - 3.34369\times10^{-19}\, e_{3.}\, e_{4.} + 6.59348\times10^{-20}\, e_{1.}\, e_{2.}\, e_{3.}\, e_{4.}$

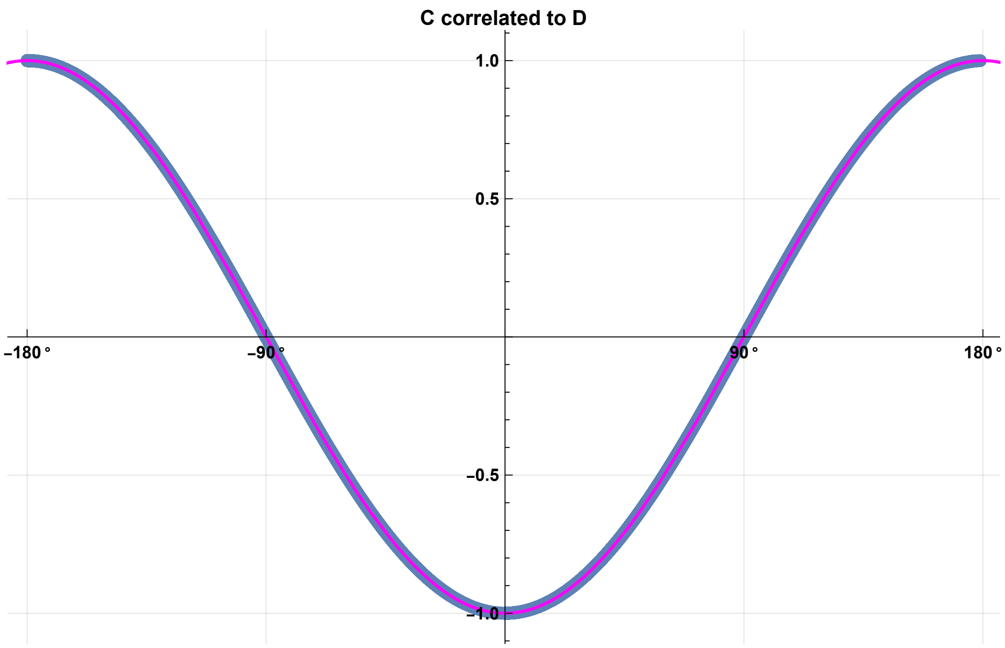
Out[209]=



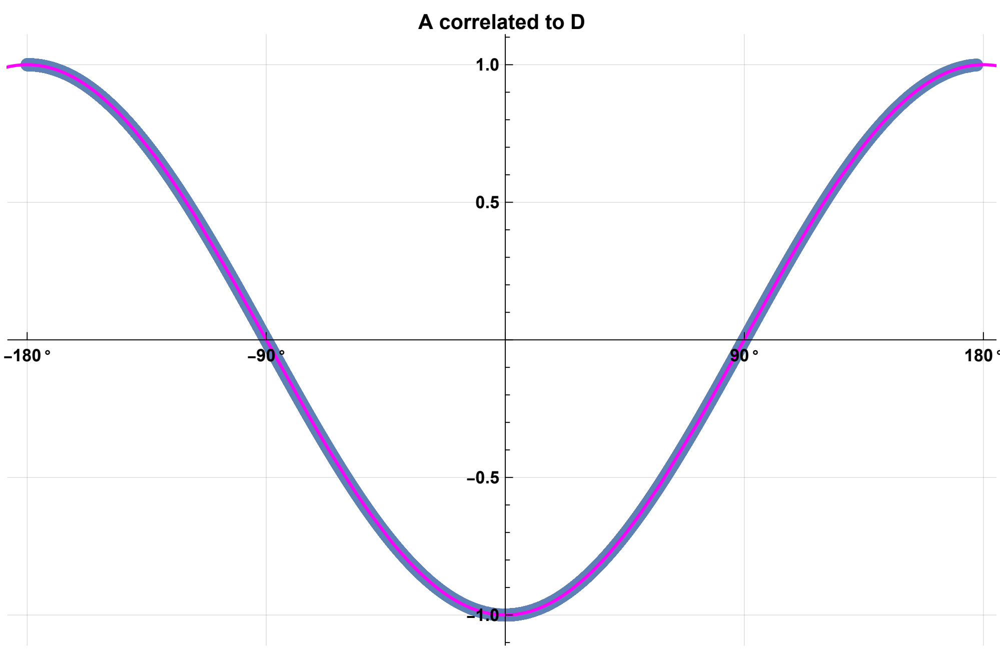
Out[210]=



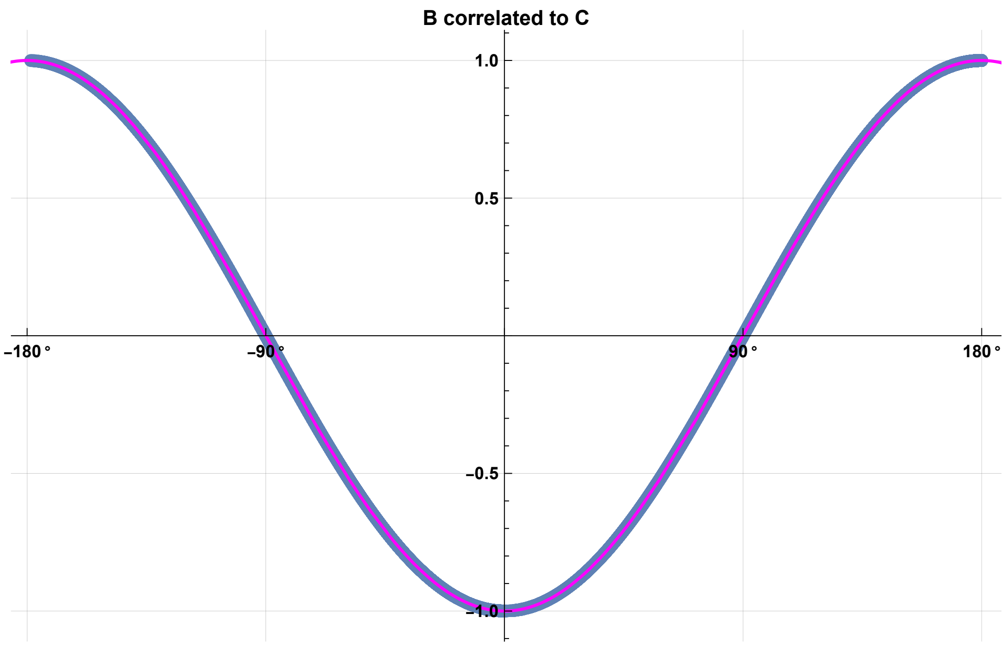
Out[211]=



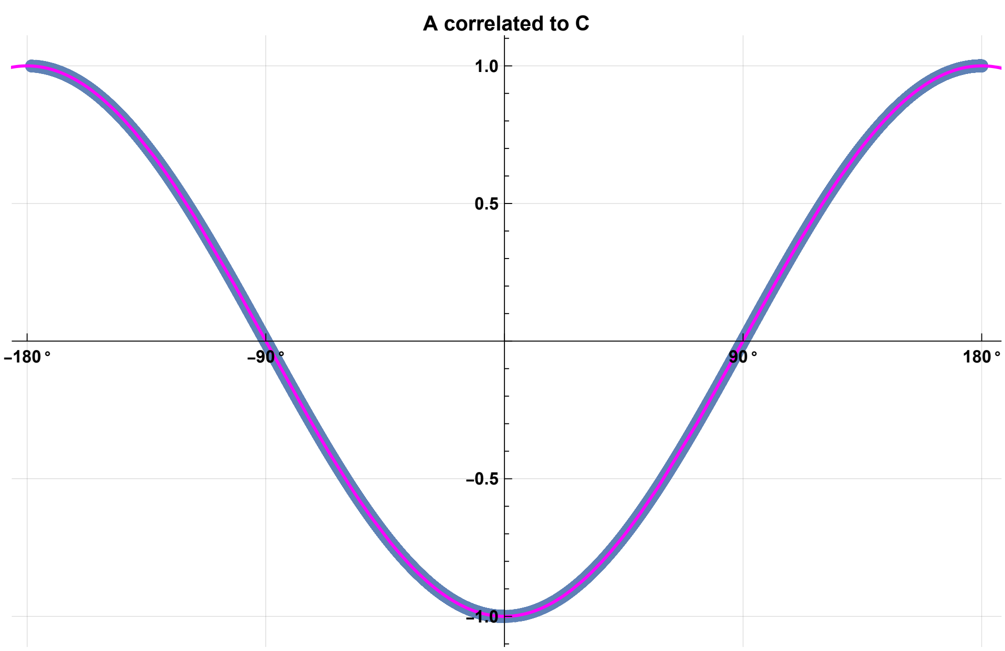
Out[212]=



Out[213]=



Out[214]=



Out[215]=