```
In[21]:= (* Two spheres, or other shapes, of opposite exactly
          overlap.  Plotting is impossible unless we use the Pfaffian.  *)

         (*Pfaffian code from Harvard postdoc Everett You who
          addapted C/FORTRAN code created by R. W. Cherng for Mathematica*)
         BeginPackage["Pfaffian`"];
         Pf::usage = "Pf[A] gives the pfaffian of the skew symmetric A.";
         Begin["`Private`"];
         Pf[A_] := Switch[Length[A], 0, 1, _?OddQ, 0, _?EvenQ, xPf[A, 1]];
         xPf[A_, p0_] := Module[{A0, n, pivot, sign = 1, A1, p1}, n = Length[A] / 2;
            If[n ≠ 1, A0 = A;
             pivot = First[Ordering[Normal[Abs[A0[[2 n - 1, All]]]], -1]];
             If[pivot ≠ 2 n, A0[[{pivot, 2 n}, All]] = A0[[{2 n, pivot}, All]];
              A0[[All, {pivot, 2 n}]] = A0[[All, {2 n, pivot}]];
              sign = -1;];
             p1 = A0[[2 n - 1, 2 n]];
             A1 = p1 A0[[1 ;; 2 n - 2, 1 ;; 2 n - 2]];
             A1 += (# - Transpose[#]) &@
              Outer[Times, A0[[1 ;; 2 n - 2, 2 n]], A0[[1 ;; 2 n - 2, 2 n - 1]]];
             A1 /= p0;
             sign xPf[A1, p1], A[[1, 2]]]];
         End[];
         EndPackage[];

In[28]:= sigma1 = {{0, 1}, {1, 0}};

In[29]:= sigma2 = {{0, -I}, {I, 0}};

In[30]:= sigma3 = {{1, 0}, {0, -1}};

In[31]:= sig = {{0, -1}, {1, 0}};

In[32]:= sig1 = {{0, 1}, {-1, 0}};

In[33]:= Z = (KroneckerProduct[sig, IdentityMatrix[2]] );

In[34]:= MatrixForm[Z];

In[35]:= Q = (KroneckerProduct[IdentityMatrix[4], IdentityMatrix[2]] +
            KroneckerProduct[-sigma2, Z] );

In[36]:= MatrixForm[Q];

In[37]:= X = {{0, 1 - 2 * s, 0, s}, {1 - 2 * s, 0, -s, 0}, {0, -s, 0, 1 - 2 * s}, {s, 0, 1 - 2 * s, 0}};
```

In[38]:=
```
MatrixForm[X]
```

Out[38]//MatrixForm=

$$\begin{pmatrix} 0 & 1-2s & 0 & s \\ 1-2s & 0 & -s & 0 \\ 0 & -s & 0 & 1-2s \\ s & 0 & 1-2s & 0 \end{pmatrix}$$

In[39]:=
```
Y = {{0, -i, 0, 0}, {i, 0, 0, 0}, {0, 0, 0, i}, {0, 0, -i, 0}};
```

In[40]:=
```
MatrixForm[Y]
```

Out[40]//MatrixForm=

$$\begin{pmatrix} 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \\ 0 & 0 & 0 & i \\ 0 & 0 & -i & 0 \end{pmatrix}$$

In[41]:=
```
Z = {{1 - s, 0, 0, 0}, {0, -1 + s, 0, 0}, {0, 0, 1 - s, 0}, {0, 0, 0, -1 + s}};
```

In[42]:=
```
MatrixForm[Z]
```

Out[42]//MatrixForm=

$$\begin{pmatrix} 1-s & 0 & 0 & 0 \\ 0 & -1+s & 0 & 0 \\ 0 & 0 & 1-s & 0 \\ 0 & 0 & 0 & -1+s \end{pmatrix}$$

In[43]:=
```
loclzr = KroneckerProduct[sigma1, X - x * IdentityMatrix[4]] +
    KroneckerProduct[sigma2, Y - y * IdentityMatrix[4]] +
    KroneckerProduct[sigma3, Z - z * IdentityMatrix[4]] ;
```

In[44]:=
```
MatrixForm[loclzr]
```

Out[44]//MatrixForm=

$$\begin{pmatrix} 1-s-z & 0 & 0 & 0 & -x+i\,y & -2s & 0 & s \\ 0 & -1+s-z & 0 & 0 & 2-2s & -x+i\,y & -s & 0 \\ 0 & 0 & 1-s-z & 0 & 0 & -s & -x+i\,y & 2-2s \\ 0 & 0 & 0 & -1+s-z & s & 0 & -2s & -x+i\,y \\ -x-i\,y & 2-2s & 0 & s & -1+s+z & 0 & 0 & 0 \\ -2s & -x-i\,y & -s & 0 & 0 & 1-s+z & 0 & 0 \\ 0 & -s & -x-i\,y & -2s & 0 & 0 & -1+s+z & 0 \\ s & 0 & 2-2s & -x-i\,y & 0 & 0 & 0 & 1-s+z \end{pmatrix}$$

In[45]:=
```
loclzrSkew = Simplify[I * (1 / 2) * ConjugateTranspose[Q].loclzr.Q];
```

In[46]:=
```
MatrixForm[loclzrSkew]
```

Out[46]//MatrixForm=

$$\begin{pmatrix} 0 & s & x & 2s & -y & 0 & 1-s-z & 0 \\ -s & 0 & 2(-1+s) & x & 0 & -y & 0 & -1+s-z \\ -x & 2-2s & 0 & s & -1+s+z & 0 & -y & 0 \\ -2s & -x & -s & 0 & 0 & 1-s+z & 0 & -y \\ y & 0 & 1-s-z & 0 & 0 & -s & -x & 2-2s \\ 0 & y & 0 & -1+s-z & s & 0 & -2s & -x \\ -1+s+z & 0 & y & 0 & x & 2s & 0 & -s \\ 0 & 1-s+z & 0 & y & 2(-1+s) & x & s & 0 \end{pmatrix}$$

In[47]:=
```
archpoly = FullSimplify[Pf[loclzrSkew]]
```

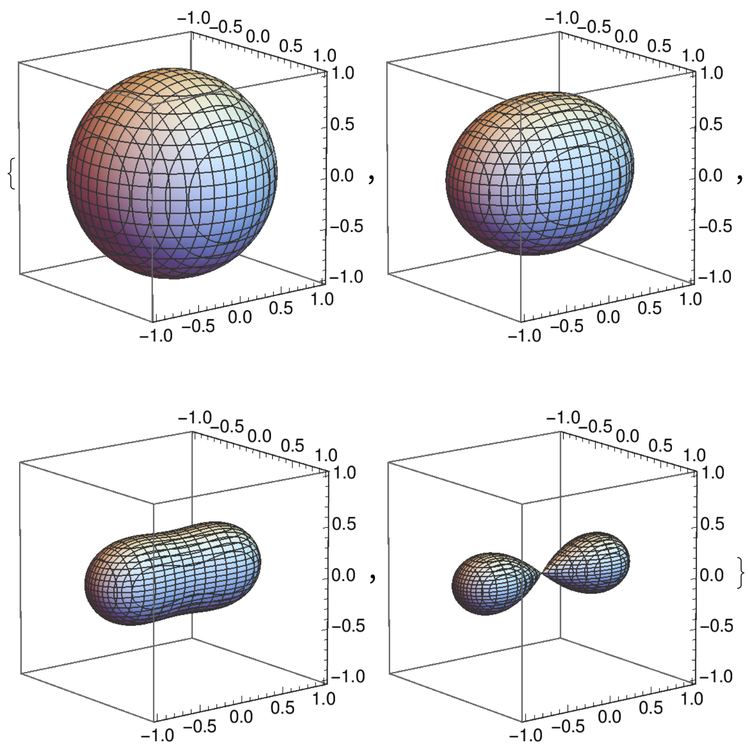
Out[47]=

$$-16\,s^3 + 16\,s^4 + 4\,s\,\left(3 + x^2 - 3\,y^2 - z^2\right) + \left(-1 + x^2 + y^2 + z^2\right)\,\left(3 + x^2 + y^2 + z^2\right) + s^2\,\left(-8 - 8\,x^2 + 12\,y^2 + 8\,z^2\right)$$

In[48]:=
```
step = 1/6;
```

In[49]:=
```
plots = ParallelTable[ContourPlot3D[archpoly == 0, {x, -1.0, 1.0},
    {y, -1.0, 1.0}, {z, -1.0, 1.0}, Contours → {{1, LightBlue}},
    PlotPoints → 100, ViewPoint → {18, -14, 6}], {s, 0, 3/6, step}]
```

Out[49]=



In[50]:=
```
Export["ClassAIIspherePfaff0_6.eps", plots[[1]], ImageSize → 2.5 * 72];
```

In[51]:=
```
Export["ClassAIIspherePfaff1_6.eps", plots[[2]], ImageSize → 2.5 * 72];
```

In[52]:=
```
Export["ClassAIIspherePfaff2_6.eps", plots[[3]], ImageSize → 2.5 * 72];
```

In[53]:=
```
Export["ClassAIIspherePfaff3_6.eps", plots[[4]], ImageSize → 2.5 * 72];
```