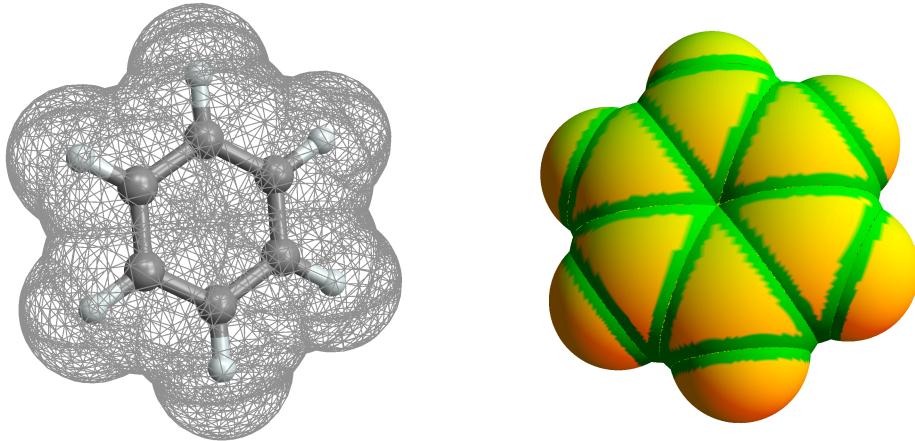


Molecular Surface Modeling

Anton Antonov

Wolfram Research, Inc.

This article demonstrates that *Mathematica* can be used for fast prototyping of various molecular surface meshing algorithms.



Introduction: molecular surface mesh generation

When simulating chemical reactions in liquids, the molecular volume of the solute is represented by Ω , the geometrical domain of the molecular volume in the tridimensional space \mathcal{R}^3 . Boundary $\partial\Omega$ is referred to as a molecular surface. Problems relating to physical chemistry can be expressed in terms of partial differential equations and solved using for instance the finite element method or the boundary element method. These methods are based on a spatial discretization, or mesh, of the surface $\partial\Omega$ (and also the volume Ω in some cases). It is thus necessary to construct a partition of the molecular surface $\partial\Omega$ in geometrically simple elements, such as triangles or quadrilaterals.

Below we develop a prototype for mesh generation over the van der Waals surface.

Geometric definitions of molecular surfaces

The basic idea in modeling an individual molecule is to represent each atom as a ball B_i with a size determined by the Van der Waals radius. The cluster of atoms is represented by $\bigcup B_i$, which is the union of all balls (many of which are overlapping). It is then possible to define different kinds of surfaces, most commonly the van der Waals surface (VWS), the solvent-accessible surface (SAS) and the solvent-excluded surface (SES). The VWS is simply the boundary of the union $\bigcup B_i$.

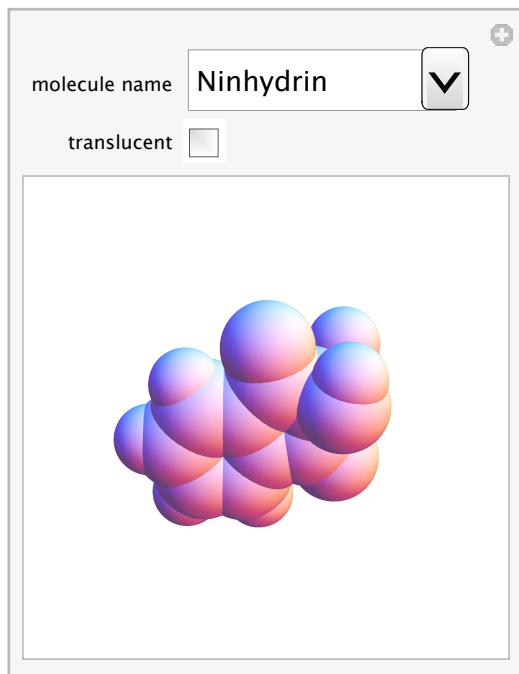
Implementation of VWS for several molecules is illustrated in the graphics below. By checking "translucent" -- one can see that the van der Waals surfaces overlap. The next section describes the algorithm used to remove such overlaps.

```

Manipulate[
DynamicModule[{apos, etypes, vtypes, vdWrs, radii},
  apos = ChemicalData[mname, "AtomPositions"];
  etypes = ChemicalData[mname, "ElementTypes"];
  vtypes = ChemicalData[mname, "VertexTypes"];
  vdWrs = ElementData[#, "VanDerWaalsRadius"] & /@ etypes;
  radii = N[vtypes /. Thread[etypes -> vdWrs]];

Graphics3D[{{
  Opacity[If[translucent, 0.4, 1]],
  MapThread[Sphere[{#1, #2} &, {apos, radii}]],
  Boxed -> False, ImageSize -> 200, SphericalRegion -> True]},
{{mname, "Benzene", "molecule name"}, {"Water", "Benzene", "Ethanol", "Ninhydrin", "OxalaceticAcid", "Isopropanol", "T-Butanol", "Caffeine", "NitricAcid" -> "Nitric Acid"}}, {translucent, {False, True}}]

```



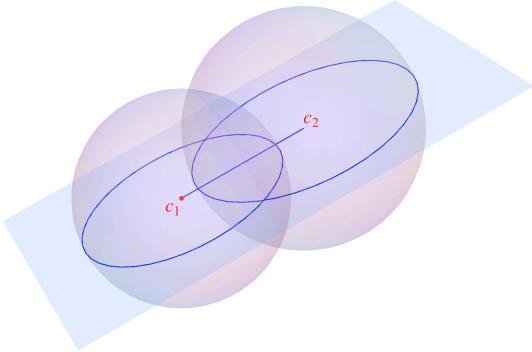
Algorithm

The algorithm finds the intersection circle for each couple of spheres. The equations of the planes on which these circles lie are used to form region functions for ParametricPlot3D. In this way only the boundary of the union of all balls is plotted.

■ Find the intersection circle of two spheres

Consider two spheres with centers c_1 and c_2 and radii r_1 and r_2 .

To find the intersection between two spheres consider the circles that lie in the same plane as the segment that connects their centers.



These leads us to the solution of the following system of equations:

$$\begin{aligned} \text{Solve}\left[\begin{array}{l} \left(\frac{d}{2} + x\right)^2 + y^2 = r_1^2, \quad \left(-\frac{d}{2} + x\right)^2 + y^2 = r_2^2 \\ \{x, y\} \end{array} \right] \\ \left\{ \begin{array}{l} y \rightarrow -\frac{1}{2} \sqrt{-d^2 + 2r_1^2 + 2r_2^2 - \frac{(r_1^2 - r_2^2)^2}{d^2}}, \quad x \rightarrow \frac{r_1^2 - r_2^2}{2d} \\ y \rightarrow \frac{1}{2} \sqrt{-d^2 + 2r_1^2 + 2r_2^2 - \frac{(r_1^2 - r_2^2)^2}{d^2}}, \quad x \rightarrow \frac{r_1^2 - r_2^2}{2d} \end{array} \right\} \end{aligned}$$

, where $d = |c_2 - c_1|$.

■ Find the equation of the dividing plane

We form the equation of the plane that passes through these two points and perpendicular to the segment connecting the centers.

■ Multiple spheres intersection

We need to find the intersections for all pairs of spheres. Clearly some pairs do not intersect. We consider only the pairs for which $r_1 + r_2 > d$.

We find the pairs as combinations of two elements:

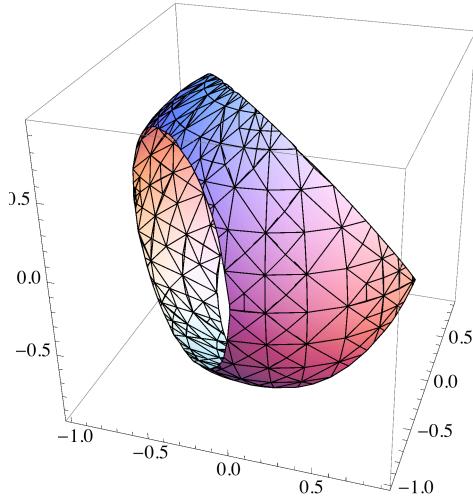
```
Combination[{}]:= {};
Combination[elems_List]:= 
  Join[{First@elems, #}&/@Rest[elems], Combination[Rest[elems]]];
```

```
Combination[{a, b, c, d}]
{ {a, b}, {a, c}, {a, Sqrt[3]}, {b, c}, {b, Sqrt[3]}, {c, Sqrt[3]} }
```

■ Plot of a single intersecting sphere

To plot a single sphere if all its parts that are inside other spheres removed, we can use ParametricPlot3D and its option RegionFunction:

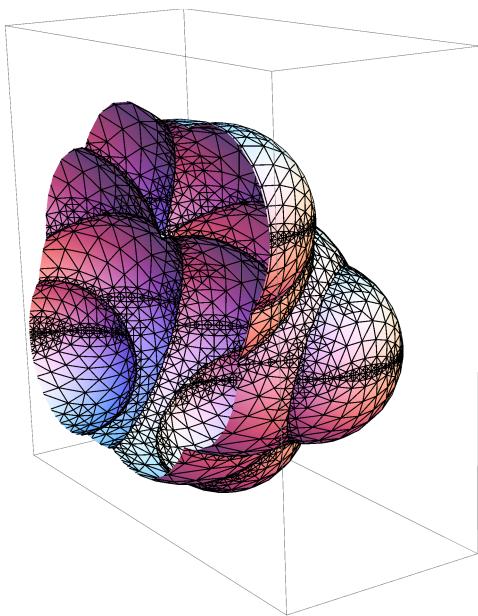
```
ParametricPlot3D[ {Cos[\phi] Cos[\theta], Sin[\phi] Cos[\theta], Sin[\theta]}, {\phi, 0, 2 \pi}, {\theta, -\pi, \pi}, Mesh \rightarrow All,
RegionFunction \rightarrow (\#1 + \#2 + \#3 < 1 / 2 \&\& \#1 + \#2 - \#3 < 1 / 2 \&\& (1 / 2 + \#1) + (1 / 2 + \#2) > 1 / 30 \&)]
```



■ Combine all single sphere plots

■ Definitions

- **Visualization**

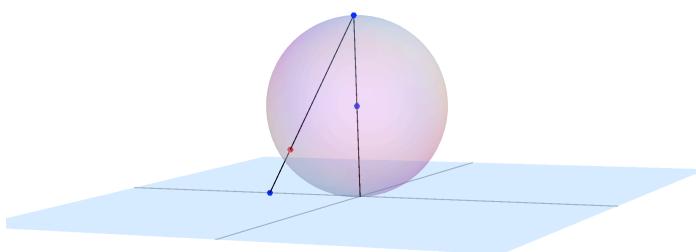


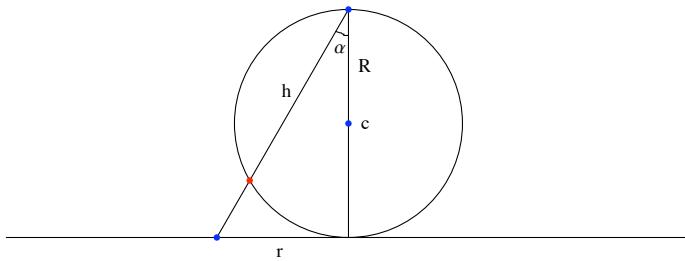
- **Better triangulations with stereographic projection**

ParametricPlot3D makes a lot of small, thin wedge shapes (i.e. triangles for which one of the sides is much smaller than the other sides) near the poles. A better triangulation can be achieved using the stereographic projection.

- **Derivation**

The diagrams below is an aid for the derivation of the formulas of the stereographic projection.





Here is an algorithm to find the coordinates of the red point on the diagram:

```
 $\sigma[R_, u_, v_] :=$ 
 $\text{Block}[\{\alpha, r, h, k\},$ 
 $r = \sqrt{u^2 + v^2};$ 
 $\alpha = \text{ArcTan}[r / (2 R)];$ 
 $h = \text{Cos}[\alpha] 2 R;$ 
 $k = h \text{Sin}[\alpha] / r;$ 
 $\{u k, v k, 2 R - h \text{Cos}[\alpha]\}$ 
];
```

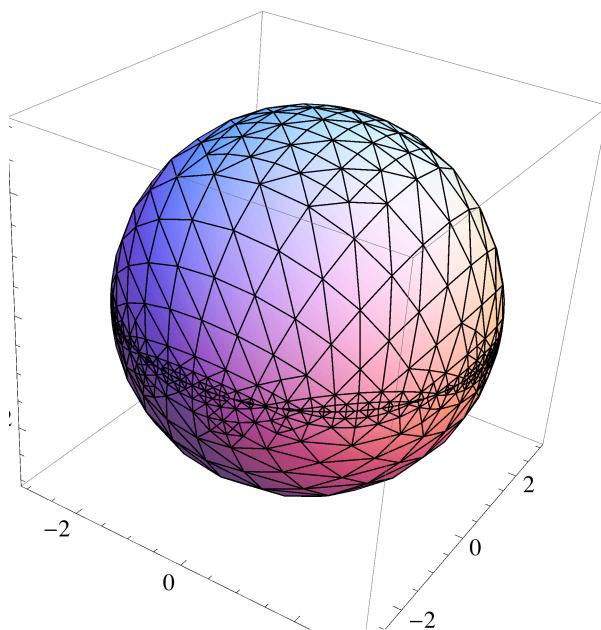
Applying the algorithm on symbols and using symbolic simplification we obtain the stereographic projection formulas:

```
 $\sigma[R, u, v] // \text{Simplify}$ 
```

$$\left\{ \frac{4 u}{4 + u^2 + v^2}, \frac{4 v}{4 + u^2 + v^2}, \frac{2 (u^2 + v^2)}{4 + u^2 + v^2} \right\}$$

Hence we can simply define:

$$\sigma[R_, u_, v_] := \frac{2 R}{4 R^2 + u^2 + v^2} \{2 R u, 2 R v, (u^2 + v^2)\}$$



Van der Waals surfaces examples

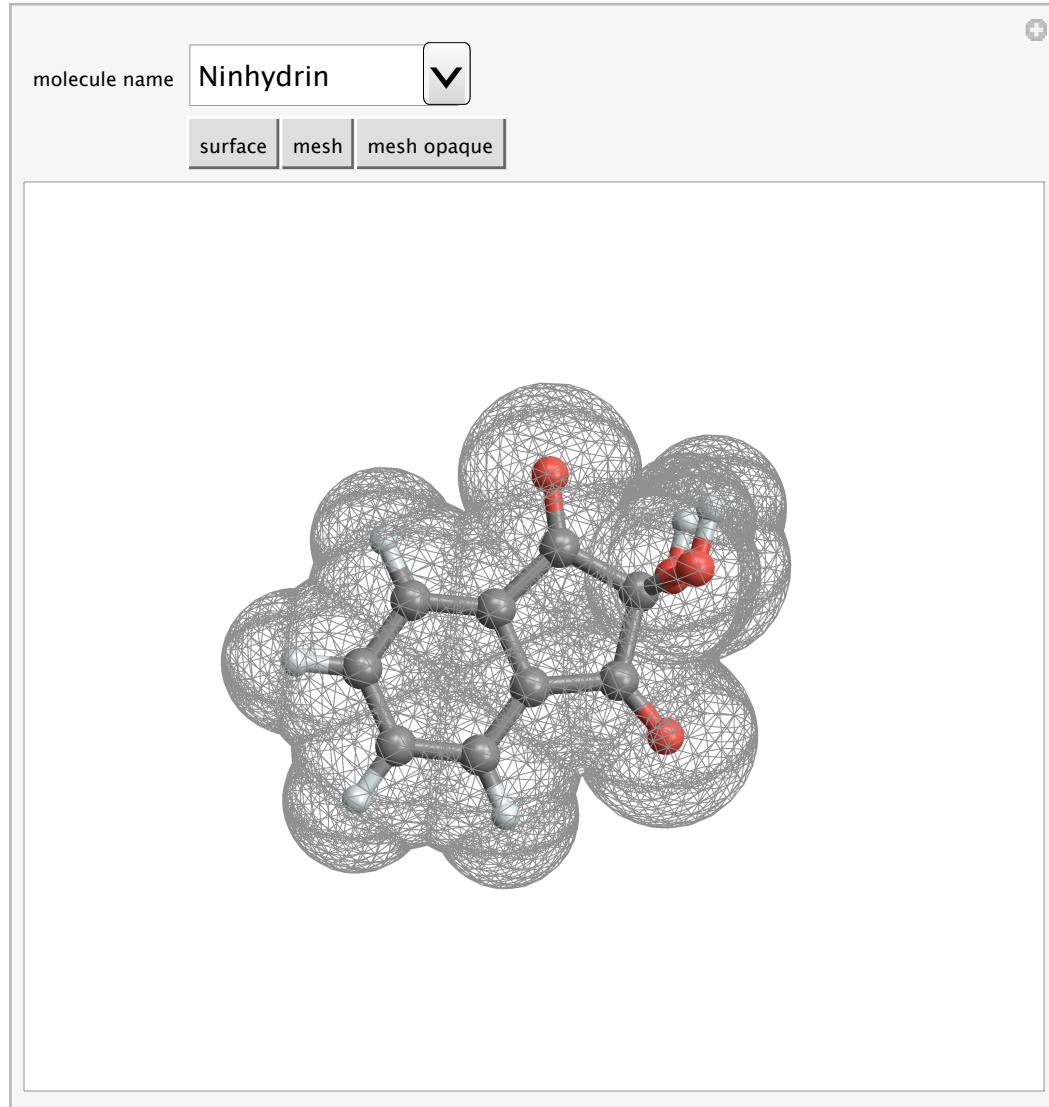
Below are given plots of the van der Waals surfaces of several molecules.

▪ Definitions

▪ Manipulate

```
In[91]:= Manipulate[
 DynamicModule[{regions, gr, mp, apos, radiuses, vtypes, etypes, vdWrs},
  mp = ChemicalData[mname, "MoleculePlot"];
  apos = ChemicalData[mname, "AtomPositions"];
  vtypes = ChemicalData[mname, "VertexTypes"];
  etypes = ChemicalData[mname, "ElementTypes"];
  vdWrs = ElementData[#, "VanDerWaalsRadius"] & /@ etypes;
  radiuses = vtypes /. Thread[etypes → vdWrs] // N;
  regions = Flatten[CheckPairs[apos, radiuses][[2]], 1];

 Show[mp,
  MapThread[PlotOnRegions[#1, #2, #3, MaxRecursion → 2,
   Sequence @@ Which[
    surfaces === "surface",
    {PlotStyle → {Green, Opacity[.4]}, Mesh → None},
    surfaces === "mesh",
    {PlotStyle → {FaceForm[None]}, MeshStyle → Gray, Mesh → All},
    True,
    {PlotStyle → {FaceForm[White]}, MeshStyle → Gray, Mesh → All}]] &,
  {apos, radiuses, regions}],
  Boxed → False, ImageSize → {450, 400}, SphericalRegion → True]],
 {{mname, "Benzene", "molecule name"}, {"Water", "Benzene", "Ethanol", "Ninhydrin",
  "OxalaceticAcid", "Isopropanol", "T-Butanol", "Caffeine", "NitricAcid", "FullereneC60"}},
 {{surfaces, "mesh", ""}, {"surface", "mesh", "mesh opaque"}}]
```



SES and SAS type of plots

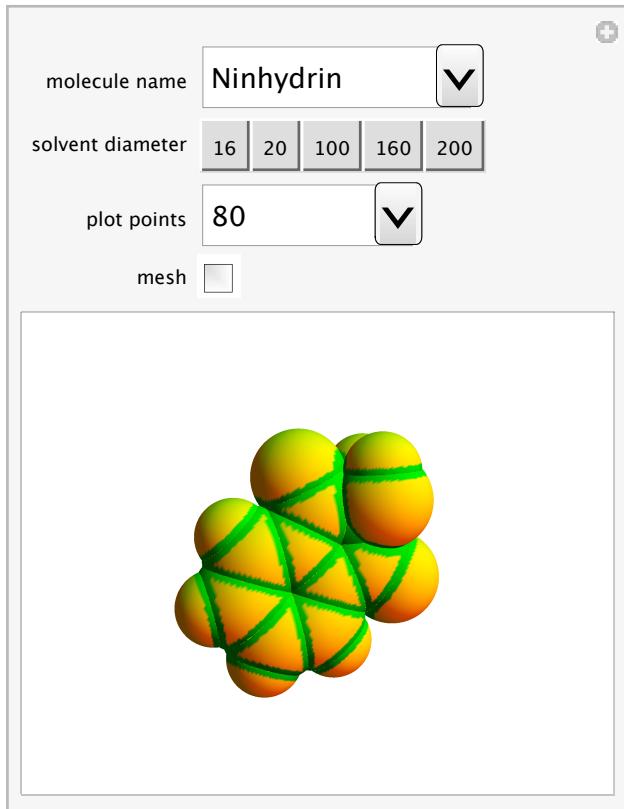
The meshes generated for the van der Waals surfaces can be used to make computations and plots for SES and SAS.

■ Definitions

■ Visualisation

```
In[92]:= Manipulate[
DynamicModule[{regions, gr, mp, apos, radiuses, vtypes, etypes, vdWrs},
mp = ChemicalData[mname, "MoleculePlot"];
apos = ChemicalData[mname, "AtomPositions"];
vtypes = ChemicalData[mname, "VertexTypes"];
etypes = ChemicalData[mname, "ElementTypes"];
vdWrs = ElementData[#, "VanDerWaalsRadius"] & /@ etypes;
radiuses = N[vtypes /. Thread[etypes → (d / 2 + vdWrs)]];
regions = Flatten[CheckPairs[apos, radiuses][[2]], 1];
gr = MapThread[PlotOnRegionsSAS[#1, #2, #3, d, PlotPoints → ppoints,
Mesh → If[mesh, All, None]] &, {apos, radiuses, regions}];

Graphics3D[First /@ gr, Boxed → False, ImageSize → {250, 200}, SphericalRegion → True]],
{{mname, "Benzene", "molecule name"}, {"Water", "Benzene", "Ethanol", "Ninhydrin",
"OxalaceticAcid", "Isopropanol", "T-Butanol", "Caffeine", "NitricAcid"}},
{{d, 16, "solvent diameter"}, {16, 20, 100, 160, 200}},
{{ppoints, Automatic, "plot points"}, {Automatic, 20, 40, 60, 80, 120, 200}},
{mesh, {False, True}}]
```



Summary

This article demonstrates the advantages of using *Mathematica* for prototyping and investigating molecular surface meshing algorithms and molecular visualisation techniques. The presented implementation of the van der Waals mesh algorithm utilizes *Mathematica*'s extensive chemical molecules and element database, solvers and symbolic mathematical manipulations, and the powerful plotting options *RegionFunction* and *ColorFunction*.

References

- [1] P. Laug, H. Borouchaki, Molecular Surface Modeling and Meshing, Journal Engineering with Computers (2002) 18:199-210.
- [2] C. Le Bris, editor "Computational Chemistry, Special Volume of Handbook of Numerical Analysis, vol X", Elsevier Science B.V., 2003, series editor: Ph. G. Ciarlet.