

Weierstrass-Enneper procedure: construct minimal surfaces from complex data

The following code will work for local construction of minimal surfaces from analytic data, provided the data is simple enough for Maple to evaluate. For reconstruction over non-simply connected domains, or over Riemann surfaces, or for complicated data, the code would need to be customized. This code is adapted from the textbook by Oprea, "Differential Geometry and its Applications", page 249

```
[ > restart:with(plots):  
Warning, the name changecoords has been redefined  
  
[ > assume(u,real);  
  assume(v,real);  
  assume(t,real);  
  
[ > #Procedure which uses general gauss map g, adapted from Oprea page  
  249  
  WE1:=proc(f,g)  
    local Z1,Z2,Z3, X1,X2,X3,X;  
    Z1:=int(f*(1-g^2),z);  
    Z2:=int(I*f*(1+g^2),z);  
    Z3:=int(2*f*g,z);  
    X1:=evalc(Re(subs(z=u+I*v,Z1)));  
    X2:=evalc(Re(subs(z=u+I*v,Z2)));  
    X3:=evalc(Re(subs(z=u+I*v,Z3)));  
    X:=simplify([X1,X2,X3]);  
  end;
```

Test it on Enneper.

```
[ > Enne1:=WE1(1,z);  
  plot3d(Enne1,u=-1..1,v=-1..1,color=black,style=wireframe,  
  scaling=constrained,axes=boxed);
```

Here's a variation:

```
[ > Enne2:=WE1(1,z^2);  
  plot3d(Enne2,u=-1..1,v=-1..1,color=black,style=wireframe,  
  scaling=constrained,axes=boxed);
```

Interesting families of surfaces: This family interpolates the catenoid to the helicoid, through a family of isometric surfaces.

```
[ > Helcat:=WE1(-exp(I*t)*exp(-z)/2,-exp(z));  
  
[ > animate3d(Helcat,u=-1..1,v=0..2*Pi,t=0..2*Pi,frames=28,  
  scaling=constrained);
```

If the following surface is the catenoid when $t=1$, what will the family look like?

```
[ > duh:=WE1(-t*exp(-z)/2,-exp(z));  
  animate3d(duh,u=-1..1,v=0..2*Pi,t=0..2*Pi,frames=20,  
  scaling=constrained);
```

[

Something picked at random:

```
[ > play:=WE1(exp(t*z),z);  
  animate3d(play,u=-1..1,v=-1..1,t=1..3,frames=20,  
  scaling=constrained);
```

[