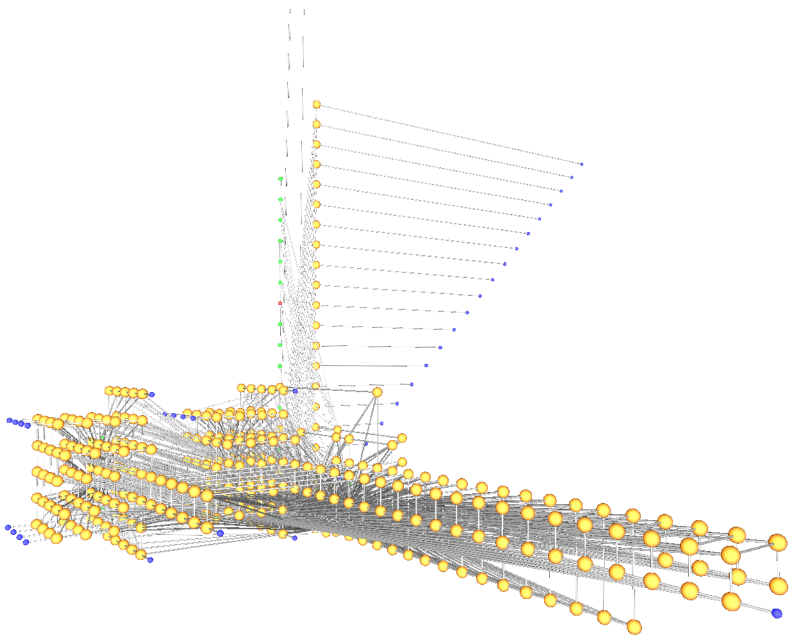
Perspex Machine VII:
The Universal Perspex Machine

James A.D.W. Anderson
Computer Science, The University of Reading, England

The Universal Perspex Machine is a theoretical, continuous, super-Turing machine that operates geometrically. It can be simulated approximately on a digital computer using transrational arithmetic. It is a Single Instruction, Zero Exception (SIZE) machine. It is the ultimate RISC!

The figures below show C code for the Travelling Salesman problem compiled into perspective simplexes, i.e. into perspexes.

```
double startx, starty, finx, finy, marked[4][4], distance[4][4];

marked[0][0]=1;marked[1][0]=1;marked[2][0]=1;marked[3][0]=1;
marked[0][1]=1;marked[1][1]=0;marked[2][1]=0;marked[3][1]=1;
marked[0][2]=1;marked[1][2]=0;marked[2][2]=0;marked[3][2]=1;
marked[0][3]=1;marked[1][3]=1;marked[2][3]=1;marked[3][3]=1;


startx = 2;
starty = 1;
finx = 1;
finy = 2;


void main()
{
double  finished, x, y, currentx, currenty, least, leastx, leasty, tx, ty, tdist, logic;

  finished = 1;
  currentx = startx;
  currenty = starty;

  while (finished == 1)
  {
    marked[currentx][currenty] = 1;
    logic = marked[finx][finy];
    if (logic == 1)
    {
      finished = 0;
    }
    else
    {
       for (x = -1; x < 2; x++)
       {
 for (y = -1; y < 2; y++)
         {
           tx = currentx + x;
   ty = currenty + y;
   logic = marked[tx][ty];
           if (logic == 0)
   {
             tdist = distance[currentx][currenty];

     if (x + y == -1)
     {
       tdist = tdist + 1;
     }
      else if (x + y == 1)
     {
        tdist = tdist + 1;
     }
     else
     {
        tdist = tdist + 7_/5;
     }

     logic = distance[tx][ty];
             if (tdist < logic)
     {
       distance[tx][ty] = tdist;
     }
     else if (logic == 0)
     {
       distance[tx][ty] = tdist;
     }
   }
 }
       }

       least = 100;
       leastx = -1;
       leasty = -1;

       for (x = 1; x < 3; x++)
       {
 for (y = 1; y < 3; y++)
         {
           logic = marked[x][y];
   if (logic == 0)
   {
      tdist = distance[x][y];
     if (tdist > 0)
     {
       if (tdist < least)
               {
                 least = tdist;
   leastx = x;
 leasty = y;
       }
     }
   }
 }
       }

       if (least == 100)
       {
         finished = 0;
       }
       else
       {
         currentx = leastx;
         currenty = leasty;
       }
    }
  }

  tdist = distance[finx][finy];
}
```



Transrational arithmetic is a total arithmetic that allows division by zero.

This does not give rise to contradictions because transrational arithmetic has an algebraic structure that is more general than a field. In particular, division is defined in a way that is more general than the multiplicative inverse.

The whole of existing mathematics remains intact, but is extended by transrational arithmetic.

Transrational arithmetic has a number at each extreme of the number line: $-\infty = -1/0$ and $\infty = 1/0$. It also has a number, nullity, $\Phi = 0/0$, that lies off the number line.

This exposes some faults in IEEE floating-point arithmetic. For example, it takes two bits, not one bit, to encode the sign of a number: negative, positive, zero, nullity.