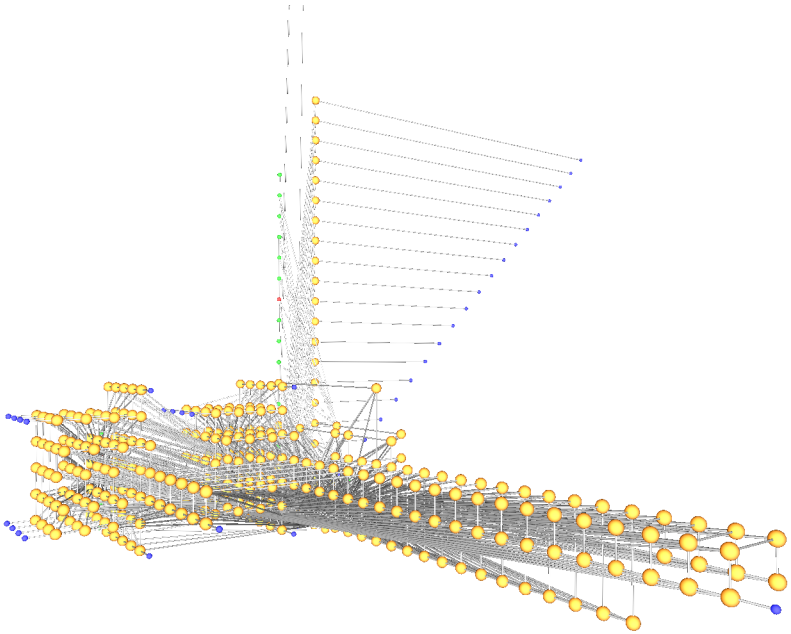
Perspex Machine VI:
A Graphical User Interface to the Perspex Machine

Christopher J.A. Kershaw & James A.D.W. Anderson
Computer Science, The University of Reading, England

All Turing programs can be compiled automatically to networks of geometrical transformations expressed as artificial neurons. These neural networks can be manipulated in a Graphical User Interface.

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

```
double starts, 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]=1;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;


starts = 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 = starts;
  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 + 3;
    }
    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];
}
```



The Walnut Cake Theorem shows that, in general, the digital computation of a series with sub-quadratic convergence, converges non-monotonically.

This is a surprise for those who believe that the calculus of continuous functions is a sufficient model of digital arithmetic.

It has some very surprising consequences which will be examined in future work.

1. The scientific literature, and all literary enterprises, go through paradigm shifts because of the Walnut Cake effect.

2. Digital encodings, such as DNA, cause punctuated equilibria in systems which evolve sub-quadratically.

3. Systems which evolve sub-quadratically, such as the Terrestrial biosphere, cause digital encodings, such as DNA.