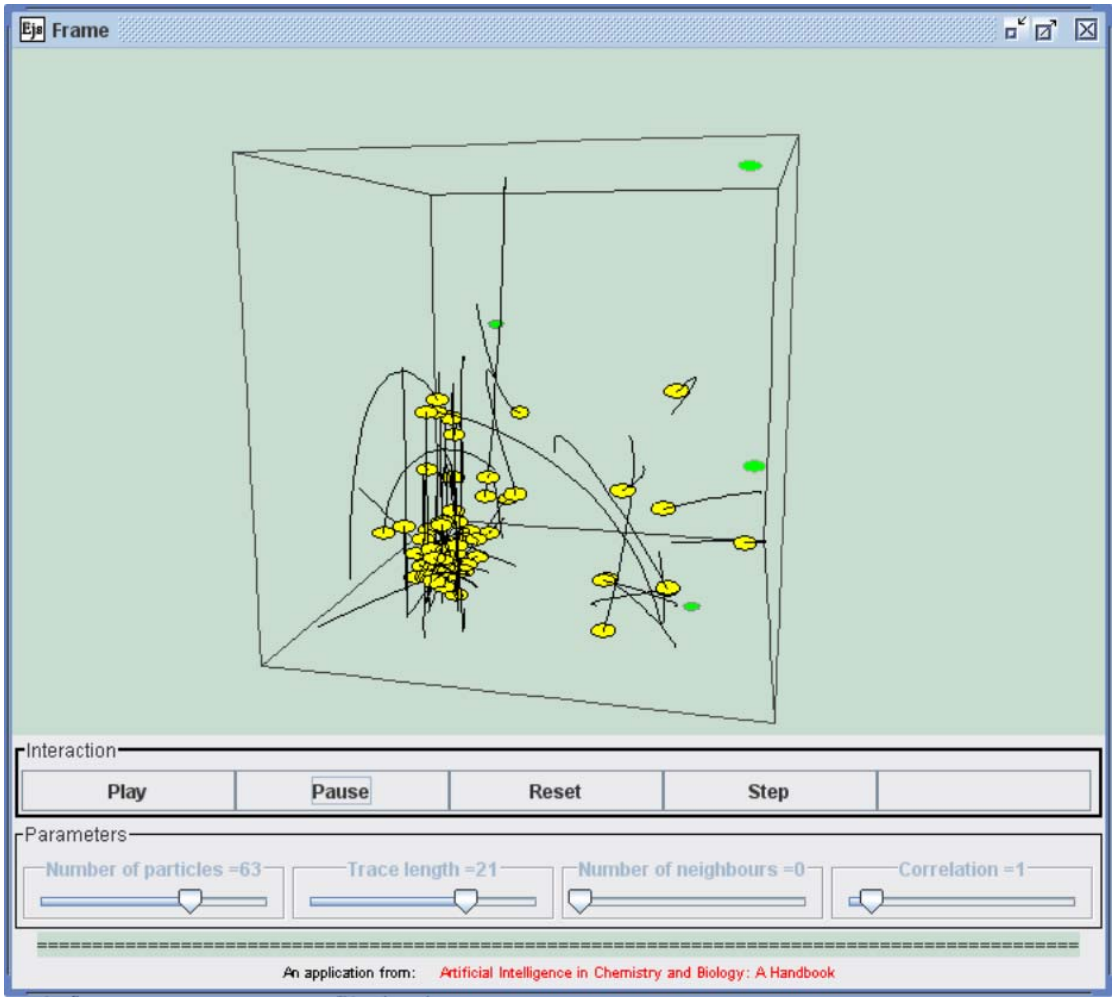
Simple Swarm Program

In the swarm program particles still move around much as they did when flocking, but now their movement is influenced by the characteristics of the search space in which they move, not by their neighbors.



The five green blobs represent the location, in 3-d space, of high-quality solutions. The particles are not aware of these solutions themselves, but do experience a field that is the summation of the effect of all solutions:

```
for (j=0; j<5; j++)
    {
        distance=Math.sqrt(Math.pow((x[i]-
blackholex[j]),2)+Math.pow((y[i]-
blackholey[j]),2)+Math.pow((z[i]-blackholez[j]),2));
        field=field-1.0/(1.0+distance);
    }
```

Because the particles do not have any data about the gradient of the field, they experience, they do not know in what direction to move to get closer to a solution, but they do keep a record of how strong the field is, and at what location they sampled it:

```
lbestfield[i]=field;
lbestx[i]=x[i];
lbesty[i]=y[i];
lbestz[i]=z[i];
```

If this field is better (more negative) than that ever experienced by any particle, this fact also is noted:

```
if (field<gbest)
{
    gbest=field;
    gbestx=x[i];
    gbesty=y[i];
    gbestz=z[i];
}
```

The particles each adjust their velocity so that they move slightly towards the best solution they have every encountered and also towards the best solution any particle has ever seen:

```
for (i=0; i<np; i++)
{
    vx[i]=vx[i]+0.01*(lbestx[i]-x[i])+0.001*(gbestx-
x[i])+0.02*(Math.random()-0.5);
    vy[i]=vy[i]+0.01*(lbesty[i]-y[i])+0.001*(gbesty-
y[i])+0.02*(Math.random()-0.5);
    vz[i]=vz[i]+0.01*(lbestz[i]-z[i])+0.001*(gbestz-
z[i])+0.02*(Math.random()-0.5);

totalv=totalv+Math.abs(vx[i])+Math.abs(vy[i])+Math.abs(vz
[i]);
}
```

This leads to clustering of particles in the region of the solutions, so the search space around those solutions can be thoroughly investigated.

BUTTONS

Button	Function
Play	Restart the program if it has previously been paused.
Pause	Temporarily halt execution.
Reset	Restart the program from scratch; all parameters are reset to their default values.
Step	Execute a single cycle of the program.

SLIDERS

Slider	Default	Comment
No. of particles	2	Number of particles to be displayed.
Trace length	12	The length of the "tail" that each particle leaves as it moves. Note (a) a trace length of zero is not permitted by the program, but this may easily be altered by changing the lower limit for this slider in the View section of the program. A trace length of zero gives a trace which grows without limit, which slows the program; (b) there appears to be a bug in the method Ejs uses to draw the traces, since a change in trace length is sometimes ignored until the number of particles is changed.
Number of neighbors	0	(Not used in this version.)
Correlation strength	4	(Not used in this version.)

Investigations and Exercises

1. Default parameters

Run the program and note the way in which particles tend to gather around one (or more than one) of the solutions. The display can be rotated in three dimensions by clicking and dragging using the left mouse button, and this may make it simpler to see the clustering.

You should be able to click and drag a solution away from the clustered particles, using the left mouse button to pick up the solution when the cursor is on a green blob. When the solution is dropped in its new position, the particles may scatter as they look for better solutions elsewhere.