Review of "Quantitative Models of In Vitro Bacteriophage–Host Dynamics and Their Application to Phage Therapy"
Cairns et al, 2009

This mathematical model describes a population of *Campylobacter jejuni* bacteria that interact with a population of phages, using a system of coupled differential equations. The bacteria are grouped into four states: susceptible, resistant and infected. The susceptible bacteria can be infected by phages or mutate into resistant bacteria (the resistant bacteria cannot be infected by phages). Infected bacteria undergo a latency period, and are then lysed (broken apart) by the phages. Phages replicate when inside the infected bacteria, and are released when the bacteria is lysed.

The change in the number of susceptible bacteria over time is given by:

$$\frac{dS}{dt} = aS - fS - bSV$$

where

S is the number of susceptible bacteria
t is time
a is the growth rate of bacteria
f is the mutation rate of bacteria
b is the binding rate of phages
V is the number of phages

The number of susceptible bacteria increases when the bacteria multiply, and decreases when they are mutated or succumb to infection.

The change in the number of resistant bacteria is over time is given by:

$$\frac{dR}{dt} = aR + fS$$

where

R is the number of resistant bacteria
t is time
a is the growth rate of bacteria
f is the mutation rate of bacteria
S is the number of susceptible bacteria

The existing resistant bacteria multiply, and new resistant bacteria are created when susceptible bacteria are mutated.

The change in the number of infected bacteria over time is given by:

$$\frac{dI}{dt} = bSV - bS(t-K)V(t-K)$$

where

I is the number of infected bacteria
t is time
b is the binding rate of phages
S is the number of susceptible bacteria
V is the number of phages
K is the latent period between infection and lysis
S(t-K) is the number of susceptible bacteria at time t-K
V(t-K) is the number of phages at time t-K

The number of infected bacteria is increased when susceptible bacteria are infected, and decreased in a time-delayed fashion when infected cells are lysed.

The change in the number of phages over time is given by:

$$\frac{dV}{dt} = hbS(t-K)V(t-K) - bSV - mV$$

where

V is the number of free phages
t is time
h is the average number of phages released per cell lysed
b is the binding rate of phages
K is the latent period between infection and lysis
S(t-K) is the number of susceptible bacteria at time t-K
V(t-K) is the number of phages at time t-K
S is the number of susceptible bacteria
m is the phage decay rate

The number of free phages is increased in a time-delayed fashion when infected cells are lysed, and decreased when phages infect cells (these phages are no longer "free"). The number of free phages also decays over time, as phages are not stable outside of a host cell.

This model can be used to find useful data for industrial and medical applications. *C. jejuni* is a food-borne pathogen, which is increasingly becoming resistant to traditional antibiotics. The use of phages as anti-bacterial agents... Two useful parameters that can be easily determined experimentally are the "inundation threshold" for initial phage concentration, V_I, above which the number of susceptible bacteria decreases - and the proliferation threshold for the susceptible bacteria concentration, S_P, above which the concentration of

phages increases.

The inundation threshold can be calculated by:

$$V_I = \frac{a-f}{b}$$

where

VI is the inundation threshold
a is the growth rate of bacteria
f is the mutation rate of bacteria
b is the binding rate of phages

The proliferation threshold can be calculated by:

$$S_P = \frac{m}{b(h-1)}$$

where

SP is the proliferation threshold
m is the phage decay rate
b is the binding rate of phages
h is the number of virus particles released per lysed bacteria

References

Cairns BJ, Timms AR, Jansen VAA, Connerton IF, Payne RJH (2009) Quantitative Models of In Vitro Bacteriophage-Host Dynamics and Their Application to Phage Therapy. PLoS Pathog 5(1): e1000253. doi:10.1371/journal.ppat.1000253

Modifying the FLAME model

1. Community of agents

In order to convert the model into a single community of bacterial agents, I created a modified 0.xml file containing the starting positions of all the bacteria from 0forVibrio.xml, 0forSAR11.xml and 0forBacteriaX.xml. I changed the BacteriaIDs so that they numbered from 1-30, and copied the phage agents from 0forBacteriaX.xml.

Since each different species of bacteria has a different radius, I used conditional formatting in the FLAME visualiser to differentiate between them in the visualisation.

bacteriaRadius == 910
> green sphere with radius 50
bacteriaRadius == 500
> blue sphere with radius 20
bacteriaRadius == 100
> pink sphere with radius 15

2. Converting to 3D

Editing header files
Added to Phage_agent_header.h

```
#define PHAGEZ (current_xmachine_Phage->phageZ)
```

Added to Bacteria_agent_header.h

```
#define BACTERIAZ (current_xmachine_Bacteria->bacteriaZ)
```

Editing phageAndBacteriaV6.c
I modified the getDistance function to find the distance in 3d space:

```
double getDistance(double x, double y, double z, double tx, double ty, double tz) {
	return sqrt(pow(tx - x, 2) + pow(ty - y, 2) + pow(tz - z, 2)); }
```

I duplicated the checkBacteriaPositionY function and converted it to the Z axis, which I have set as 900 units long. I set the boundaries for the different types of bacteria based on their radius from the X and Y boundaries – 24 units for SAR11, 40 units for Bacteria X and 57 units for Vibrio.

```
double checkBacteriaPositionZ(double bacteriaPosition) {

	double zUpperBorder, zLowerBorder  = 0;

	if (BACTERIARADIUS == 100){
		//SAR11
		zUpperBorder = 24;
		zLowerBorder = 876;
	} else if (BACTERIARADIUS == 500){
		//Bacteria X
		zUpperBorder = 40;
		zLowerBorder = 860;
	} else {
		//Vibrio
		zUpperBorder = 57;
		zLowerBorder = 843;
	}

	double newPosition = bacteriaPosition;

	if (bacteriaPosition < zUpperBorder) {
		newPosition = zUpperBorder;
	}
```

```
        if (bacteriaPosition > zLowerBorder) {
                newPosition = zLowerBorder;
        }
        return newPosition;
}
```

Similarly, I duplicated the checkPhagePositionY function and converted it for the Z plane.

```
double checkPhagePositionZ(double
phagePosition) {

        double zUpperBorder = 21;
        double zLowerBorder = 879;

        double newPosition = phagePosition;

        if (phagePosition < zUpperBorder) {
                newPosition = zUpperBorder;
        }

        if (phagePosition > zLowerBorder) {
                newPosition = zLowerBorder;
        }
        return newPosition;
}
```

I modified the function add_bacteriaInformation_message() to contain BACTERIAZ:

```
add_bacteriaInformation_message(BACTERIAID,
BACTERIAX, BACTERIAY, BACTERIAZ,
BACTERIARADIUS, DEFENSEFACTOR);
```

To the functions phageRandomMove(), walkRandomly(), detectBacteria() and bacteriaRandomMove(), I added Z co-ordinates wherever there were X and Y co-ordinates. For example:

```
int detectBacteria(double* bacteriaX, double*
bacteriaY, double* bacteriaZ, int* bacteriaID) {

        double currentX = PHAGEX;
        double currentY = PHAGEY;
        double currentZ = PHAGEZ;
...}
```

Editing phageAndBacteriaV6.xml

Using the FLAME editor, I added the variable "bacteriaZ" as a double to the Bacteria agent and the bacteriaInformation message. I also added the variable "phageZ" as a double to the Phage agent and the phageInformation message.

Editing 0.xml

Using "sed" and $RANDOM at the command line, I added random Z co-ordinates (between 60 and 840) to each bacteria – for example:

```
<xagent>
  <name>Bacteria</name>
  <bacteriaID>1</bacteriaID>
  <bacteriaX>270.131299536107</bacteriaX>
  <bacteriaY>141.46062863118757</bacteriaY>
  <bacteriaZ>790</bacteriaZ>
  <bacteriaRadius>500</bacteriaRadius>
  <defenseFactor>98</defenseFactor>
 </xagent>
```

and similarly to each phage:

```
<xagent>
  <name>Phage</name>
  <phageID>1</phageID>
  <phageLife>10000</phageLife>
  <phageX>927.2916122831458</phageX>
  <phageY>632.66254371618</phageY>
  <phageZ>762</phageZ>
  <phageRadius>25</phageRadius>
  <setSeed>0</setSeed>
 </xagent>
```

3. Adding phage replication/burst

I edited phageAndBacteriaV6.c to create the function makeNewPhage(), which uses add_Phage_agent() repeatedly to make new phages. The number of new phages created depends on the size of the bacteria killed.

I called this function inside bacteriaRandomMove() when the bacteria agent is destroyed by a phage (just before it returns 1).

```
int makeNewPhage(int lphageID, double
lphageLife, double lphageX, double lphageY,
double lphageZ, double lphageRadius, int
lsetSeed, double bactRadius) {
        int burstSize = 0;
        if (bactRadius == 100){
                //SAR11
                burstSize = 1;
        } else if (bactRadius == 500){
                //Bacteria X
```

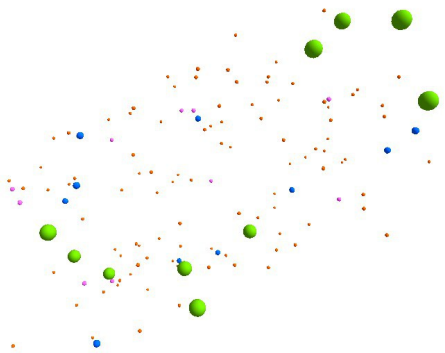
```
                burstSize = 5;
        } else {
                //Vibrio
                burstSize = 9;
        }
        printf("%d\n",burstSize);
        int i;
        for(i=0;i<burstSize;i++) {
add_Phage_agent(lphageID,lphageLife,lphageX,lp
hageY,lphageZ,lphageRadius,lsetSeed);
        }
        return 0;
}
```

4. Extra: added bacterial replication

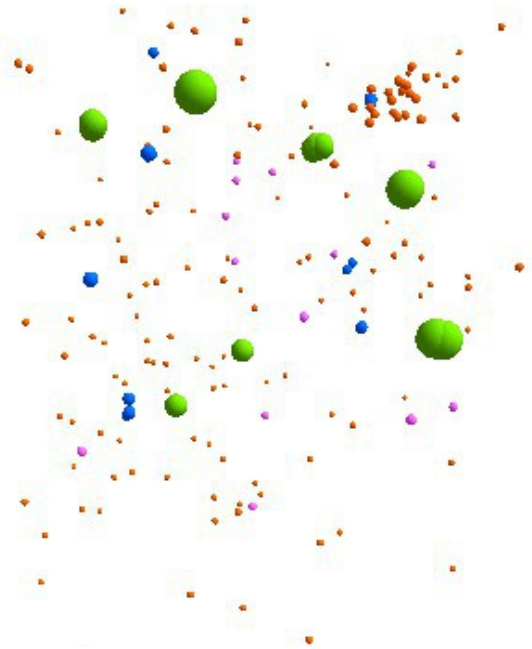
I thought it would be interesting to include bacterial replication, so I used add_Bacteria_agent() to create new bacteria based on a random number during bacteriaRandomMove().

```
if (BACTERIARADIUS == 100){
        //SAR11
        dividingProbability = 0.01;
} else if (BACTERIARADIUS == 500){
        //Bacteria X
        dividingProbability = 0.01;
} else if (BACTERIARADIUS == 910){
        //Vibrio
        dividingProbability = 0.01;
}
double diceRoll = 0;
diceRoll = ((double)rand()/
((double)RAND_MAX)); //0 and 1
        if (diceRoll <= dividingProbability){
                printf("Bacteria %d at bx %f by %f
bz %f has divided \n", BACTERIAID,
BACTERIAX, BACTERIAY, BACTERIAZ);
                add_Bacteria_agent(newb_ID,
BACTERIAX, BACTERIAY, BACTERIAZ,
BACTERIARADIUS, DEFENSEFACTOR);
                }
```

Snapshot at time = 0



Snapshot at time = 65 from a different angle, showing phage burst and bacteria division



References

Suttle, CA (2007) Marine viruses — major players in the global ecosystem . Nature Rev Microbiology 5, 801-812