CIVE60005: Fluid Mechanics 3

Controlling outdoor and indoor spread of COVID-19

Name: Mikaella Liveri

CID: 02032739

Table of Contents

1. Outdoor transmission of COVID-19 1
1.1 Question 1a 1
1.2 Question 1b 2
1.3 Question 1c 3
1.4 Question 1d 4
2. Indoor transmission of COVID-19 6
2.1 Question 2a 6
2.2 Question 2b 7
References 9
Appendix A: MATLAB Code for Question 1 10
Appendix B: MATLAB Code for Question 2 16

1. Outdoor transmission of COVID-19

The purpose of this report is to refine government guidelines to reduce the spread of COVID-19 in outdoor spaces, by airborne guidelines. Transport of SARS-CoV-2, the virus responsible for COVID-19, is measured in quanta. One quantum corresponds to the virus amount required for infection.

The government wants to allow citizens to attend concerts again and is examining the design that would reduce contamination. Figure 1.1 shows a diagram of the considered format. There are 25 viewing platforms, with each 2m diameter platform holding a maximum of 4 people, and a stage. The platforms are at ground level and have a centre-to-centre spacing of 4m. The stage has a width of 8m, while it is 3m above ground level. Moreover, the concert duration considered is 3 hours.

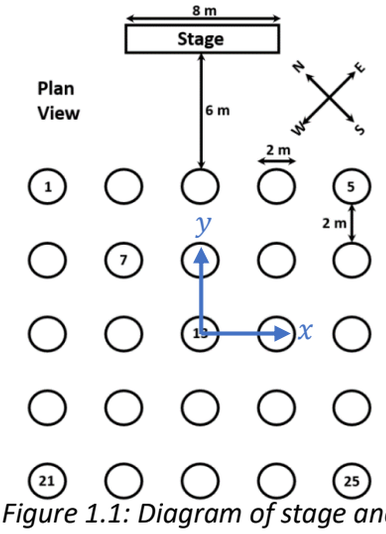


Figure 1.1: Diagram of stage and viewing areas with defined origin.

Assumptions:

- All 4 people on the central viewing platform 13 are infected with SARS-CoV-2 and are contagious.
- The average head height is calculated assuming the gender distribution of festival attendees in the UK (54% female and 46% male) (Götting, 2021). The average height is then calculated using the average female and male heights, which are 1.62 and 1.76m respectively (NHS, 2022).
  Hence, $h = 0.54(1.62) + 0.46(1.76) = 1.70m$
- Datum, z=0 is considered at head height.
- The turbulent diffusion coefficient is constant with a value of $D_T = 1m^2/s$, and hence turbulent diffusion is assumed to be homogeneous and isotropic.
- Molecular diffusion coefficient, D is negligible since $D \ll D_T$.
- No wind implies only diffusion, and hence advection is not considered as a transport process.
- The release of the virus is considered continuous, passive, neutrally buoyant, and non-reactive.
- The floor is modelled as a perfect reflector, since it is not permeable to SARS-CoV-2.
- The worst case scenario is considered and it is therefore assumed that all 4 people on platform 13 are speaking loudly/singing, which has the highest quanta emission rate equal to $98hr^{-1}$.

1.1 Question 1a

The temporal evolution of the virus concentration at head height at platforms 1, 3, 7 and 13 are calculated below. The radial distances from the origin of each of the platforms are calculated using Equation 1.1 and displayed in Table 1.1. r_{13} was calculated assuming that each person on the platform is at a distance of half the radius, hence 0.5m from the centre, since using (0,0) would cause singularity.

$$r = \sqrt{x^2 + y^2 + z^2} \tag{1.1}$$

Table 1.1: Radial distances from the origin at head height $z = 0$.

$r_1\ [m]$	$r_3\ [m]$	$r_7\ [m]$	$r_{13}\ [m]$
$\sqrt{8^2 + 8^2} = 8\sqrt{2}$	$\sqrt{8^2} = 8$	$\sqrt{4^2 + 4^2} = 4\sqrt{2}$	$\sqrt{0.5^2 + 0.5^2} = \sqrt{2}/2$

The solution ignores effects of advection as there is no wind and it is governed by the continuous 3D diffusion equation, shown in Equation 1.2.

$$\frac{\partial C}{\partial t} = D_T \nabla^2 C + \dot{M}\delta(x, y, z) \tag{1.2}$$

Since the floor is a perfect reflector, there is a boundary at $z = -h$ where the same number of virus particles arrive and leave. Applying the principle of superposition, yields Equation 1.3. r_I is the radial distance to the image source given as $r_I = \sqrt{x^2 + y^2 + (z + 2h)^2}$ and $\dot{M} = 4 \times 98/3600 = 392hr^{-1}$.

$$C = \frac{\dot{M}}{4\pi D_T r} erfc\left(\frac{r}{\sqrt{4D_T t}}\right) + \frac{\dot{M}}{4\pi D_T r_I} erfc\left(\frac{r_I}{\sqrt{4D_T t}}\right) \tag{1.3}$$

An image source is placed below the boundary at a distance equal to the one of the ground and the source, $(z = -2h)$. The concentration gradient of the image source should be equal and opposite to the concentration gradient of the source to give a net mass flux of zero. The Neumann boundary condition is therefore defined as $q_z(z = -h) = -D\frac{\partial C}{\partial z}|_{z=-h} = 0$.

Figure 1.2 plots the temporal evolution of the virus concentration at head heights on the platforms 1, 3, 7 and 13 against duration of the concert, which increases at a decreasing rate.

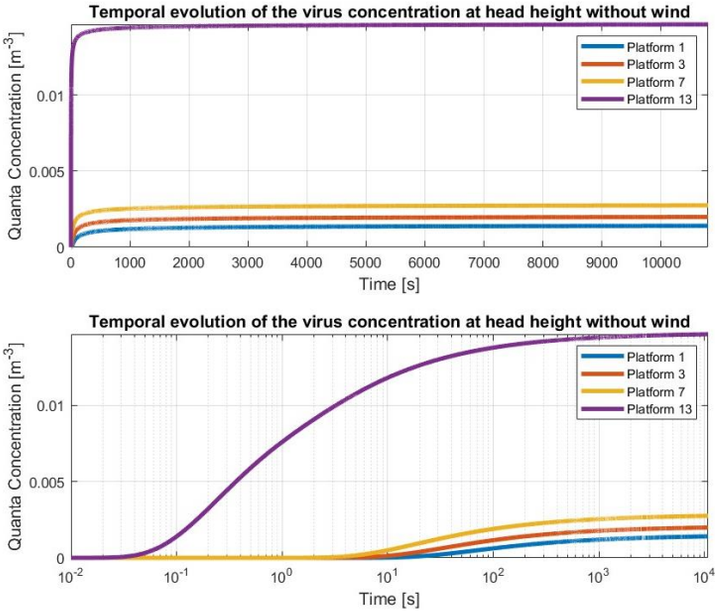


Figure 1.2: Temporal evolution of virus concentration.

The graph clearly shows that the quanta concentration of platform 13 is much higher compared to the other platforms who have a quanta concentration of an order of magnitude less. It can also be concluded that the virus concentration decreases as the distance from the source increases, as they reach a lower steady state concentration. The reason for this, is that as concentration increases in the arena due to the spreading of the virus, the concentration gradient decreases. Diffusion is driven by this concentration gradient, and is resisted by collisions between air particles with SARS-CoV-2 particles. The further away from the source, the more the collisions and hence the resistance force, but the smallest the propellant force caused by diffusion. Therefore, a higher concentration gradient is required to oppose this resistance force, which is not the case, and hence a smaller final concentration on further away platforms is reached. The gradients of the lines show that the rate of concentration increase is greatest for platform 13. The most significant variations in virus concentration happen in very short periods of time, and they stagnate rapidly. For $t \to \infty, erfc(t^{-1/2}) \to 1$, therefore a steady state $C = \frac{\dot{M}}{4\pi D_T r} + \frac{\dot{M}}{4\pi D_T r_I}$ is reached.

1.2 Question 1b

In the following section, the spatial distribution of the virus concentration within the public area at head height is considered at the end of the concert, and hence after 3 hours which is equal to 10800s. The ground is still considered as a perfect reflector. The above is analysed under two conditions: i) no wind and ii) a spatially uniform wind with a speed of 3m/s blowing from the south-west. As shown in Figure 1.1, the coordinate system is set such that the y-direction is towards the south-west wind direction. The wind velocity components are therefore $(0, U, 0)$. The x-direction is perpendicular to this.

For no wind conditions, Equation 1.3 is adopted since there is no advection. For the wind conditions however, there is now a need to consider both diffusion, and advection in the y direction. This is modelled as a steady problem (as proved by the steady-states reached above), and as a continuous 3-D point source release using the 3-D Advection-Diffusion Equation. It is also assumed that advection dominates over diffusion in the y wind direction. x and y are varied between -9 and 9, which is the area of the public space. Equation 1.4 below is the Governing equation and Equation 1.5 shows the solution.

$$U\frac{\partial C}{\partial x} = D_T\left(\frac{\partial^2 C}{\partial y^2} + \frac{\partial^2 C}{\partial z^2}\right) + \dot{M}\delta(x, y, z) \tag{1.4}$$

$$C(x,y,z) = \frac{\dot{M}}{4\pi D_T r} e^{-\frac{U}{2D_T}(r-y)} + \frac{\dot{M}}{4\pi D_T r_I} e^{-\frac{U}{2D_T}(r_I-y)} \quad (1.5)$$

Figures 1.3 and 1.4 show plots of the spatial distribution without and with wind conditions respectively, plotted using MATLAB.

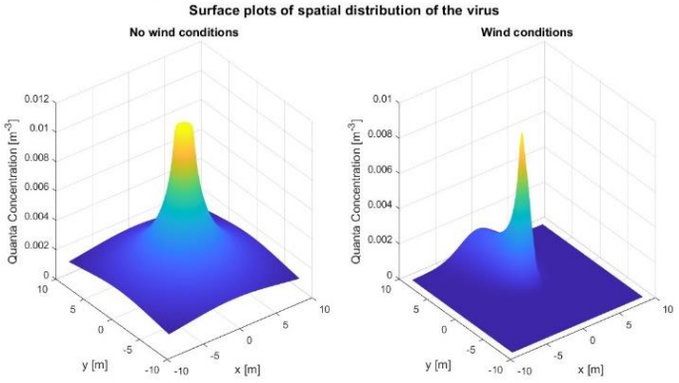


Figure 1.3: Surface plots of the spatial distribution of the virus with i) no wind ii) wind.

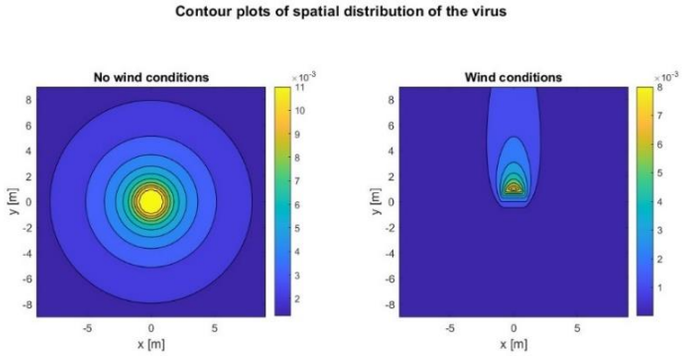


Figure 1.4: Contour plots of the spatial distribution of the virus with i) no wind ii) wind.

Comparing the two contour plots, it is evident that there are similarities and differences. For both conditions, the highest concentration occurs at the vicinity of the origin, platform 13, and it decreases as the distance from the source increases. To avoid singularities, concentrations going to infinity, the concentration at the source was set equal to the one at $r = 1\ (0.0111m^{-3})$. However, the peak concentration for the wind model, is slightly offset from the defined origin towards the positive y direction, at the edge of platform 13. The shapes of the distributions are also very different. The no wind model distribution is uniform in space with a decreasing concentration radially, and hence any point at the same distance from the source will have an equal concentration since only diffusion takes place. In contrast, the wind model distribution spans across the direction of the wind, due to advection. There is almost no concentration in the negative y direction since the particles are being transported in the positive y direction by the wind. Even though concentration gradients are greater behind the source, quanta concentration spreads significantly in the positive y direction which shows that the dominant transportation process is advection. The spreading of SARS-CoV-2 in the wind direction due to advection, causes quanta concentration to be greater in the positive y direction. Additionally, when comparing the maximum concentration of the 2 conditions, the no wind condition has a higher peak concentration compared to the wind condition, with values of $0.0111m^{-3}$ and $0.0086m^{-3}$ respectively. For the no wind model, peak concentration spans over the circular area of 1m around the centre of platform 13, since the radial distance from the source was constraint to a minimum of 1.

1.3 Question 1c

In the analysis that follows, it is now assumed that a sufficient amount of disinfectant is applied to all ground surfaces within the public area. This would to kill the SARS-CoV-2 virus upon contact, meaning that virus concentration is zero at the boundary, $C|_{z=-h} = 0$. Since the boundary now absorbs the SARS-CoV-2 particles, it is now modelled as a perfect absorber (Dirichlet Boundary Condition). Therefore, using the principle of superposition, an image source of equal strength is subtracted from the plane source, as shown in Equation 1.6 and 1.7 for no wind and wind conditions respectively.

$$C = \frac{\dot{M}}{4\pi D_T r} erfc\left(\frac{r}{\sqrt{4D_T t}}\right) - \frac{\dot{M}}{4\pi D_T r_I} erfc\left(\frac{r_I}{\sqrt{4D_T t}}\right) \quad (1.6)$$

$$C = \frac{\dot{M}}{4\pi D_T r} e^{-\frac{U}{2D_T}(r-y)} - \frac{\dot{M}}{4\pi D_T r_I} e^{-\frac{U}{2D_T}(r_I-y)} \quad (1.7)$$

Figure 1.5 and Figure 1.6, show the spatial distribution of the virus for each condition.

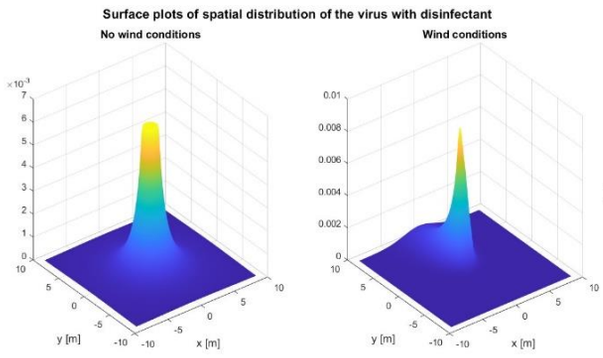


Figure 1.5: Surface plots of spatial distribution of the virus with disinfectant, with i) no wind ii) wind.

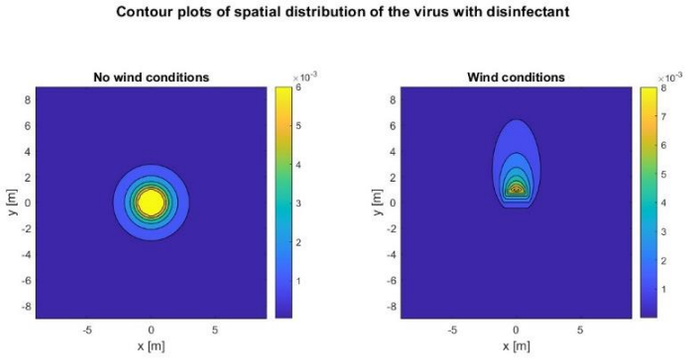
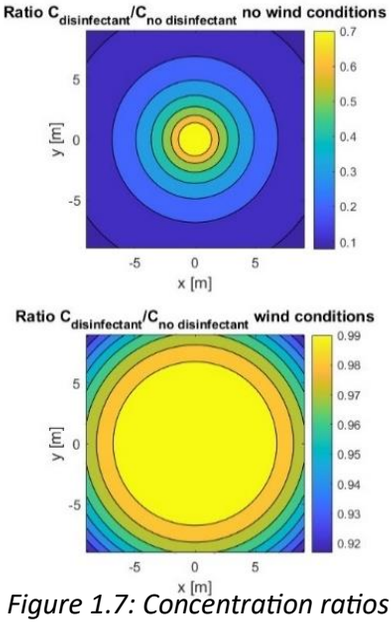


Figure 1.6: Contour plots of spatial distribution of the virus with disinfectant, with i) no wind ii) wind.

Additionally, Table 1.2 compares the magnitudes of maximum and average quanta concentrations when no disinfectant is applied and when it is.

Table 1.2: Peak and average concentrations for each scenario.

	No wind conditions	**Wind conditions**
	Maximum quanta concentration [m^{-3}]	
No disinfectant	0.0111	0.00863
Disinfectant	0.00621	0.00852
	Average quanta concentration [m^{-3}]	
No disinfectant	0.00272	0.000288
Disinfectant	0.000386	0.000220



As expected and evident from Figures 1.5 and 1.6, and Table 1.2, the disinfectant absorbs the virus, reducing the maximum and average quanta concentrations as there is no more reflection taking place, for both wind and no wind conditions. However it can be seen that the difference is much greater in the no wind conditions compared to the windy one. In the wind model, the order of magnitude of maximum and average quanta concentration does not change. This is because advection dominates over diffusion, and the wind transports the SARS-CoV-2 particles over a greater distance before they settle at the boundary. Hence, less SARS-CoV-2 particles are absorbed and killed by the disinfectant on the boundary. Therefore, the disinfectant is less effective under windy conditions, especially in high wind velocities. Figure 1.7 plots the ratio of concentrations with and without disinfectant, providing further evidence for this argument.

Figure 1.7: Concentration ratios with and without disinfectant.

1.4 Question 1d

In order to respond to Chris Witty about whether raising each viewing platform within the public area to a height of 2m above the ground would lower the exposure to the SARS-CoV-2 virus, the distributions of all scenarios are considered again, with the new head height of $h_{new} = h + 2 = 1.7 + 2 = 3.7m$. The datum is moved to 3.7m so that z=0, is still at head height. The same equations are used as before, with the only difference being the distance between the source and the boundary, which affects the distance to the image source, updated in Equation 1.8.

$$r_I = \sqrt{x^2 + y^2 + [z - (-2h_{new})^2]} \tag{1.8}$$

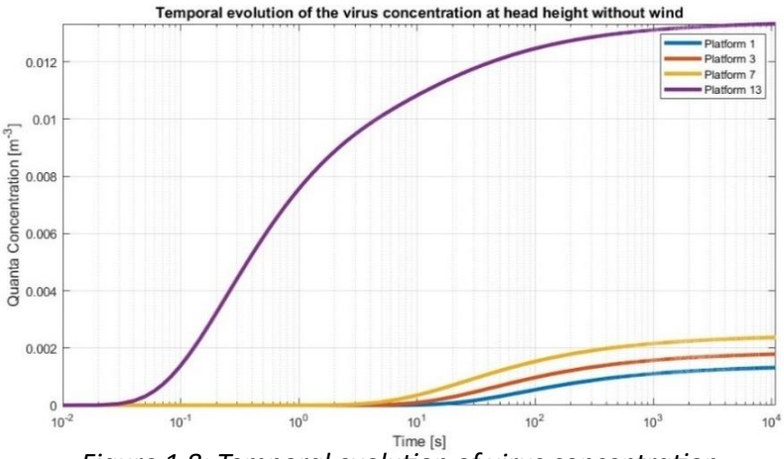


Figure 1.8: Temporal evolution of virus concentration.

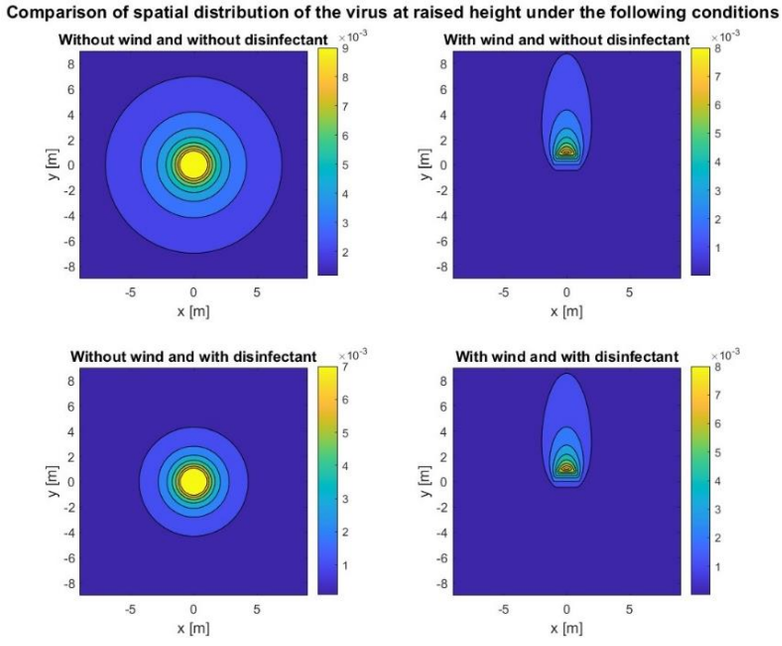


Figure 1.9: Contour plots of virus concentration.

Figure 1.8 shows the quanta concentration without wind for platforms 1, 3, 7 and 13. When compared to before, it is noted that the concentration has decreased by 6.58%, 10.16%, 13.91% and 9.07% for each of these platforms respectively.

Figure 1.9, shows contour plots of the virus concentrations with the raised platforms. The key point here, is that raising the platforms lowers the concentration gradient between the platform and the floor. This reduces the downward flux of the virus particles. When no disinfectant is applied, there is a small decrease in the virus concentrations, when comparing Table 1.3 to Table 1.2. This is because as the head height increases, so does r_I. Consequently, $\frac{\dot{M}}{4\pi D r_I} erfc\left(\frac{r_I}{\sqrt{4D_T t}}\right)$ and $\frac{\dot{M}}{4\pi D r_I} e^{-\frac{U}{2D}(r_I - y)}$, which correspond to the image source terms in Equations 1.2 and 1.4, become smaller. Since these terms are added to the overall concentration, it is less than before. The opposite occurs when disinfectant is applied. Since these terms are now smaller and subtracted due to the boundary being modelled as an absorber, this leads to an increase in concentration when compared to before. Physically, this is because the particles are further away from the absorbing boundary.

Table 1.3: Peak and average concentrations for each scenario with raised viewing platforms.

	No wind conditions	**Wind conditions**
	Maximum quanta concentration [m^{-3}]	
No disinfectant	0.00974	0.00858
Disinfectant	0.00750	0.00858
	Average quanta concentration [m^{-3}]	
No disinfectant	0.00236	0.000255
Disinfectant	0.000738	0.000253

In conclusion, to respond to Chris Whitty, raising the platforms reduces the exposure to concert goers to the SARS-CoV-2 virus only when there is no disinfectant. Raising the platforms reduces the effectiveness of the disinfectant. However, this can be altered by placing disinfectant boards at a height closer to the audience, since the lowest concentration is when disinfectant is applied without the platforms being raised. However, the disinfectant would need to be inspected as over time spilling of drinks or walking over it, might reduce its effectiveness. Lastly, in all cases virus concentration is higher in windy conditions, hence adding windbreakers would decrease the effect of wind and ensure the particles reach the boundary in a shorter time.

2. Indoor transmission of COVID-19

This report further investigates the concentration of SARS-CoV-2 in a naturally ventilated classroom with a single high-level window, as seen in Figure 2.1. Guidelines for the operation of the classroom are to be set such that for each 1 hour class, the probability of infection does not exceed 0.01, and the temperature of the room does not fall below 17°C for occupancy comfort.

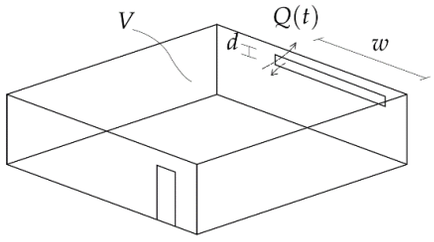


Figure 2.1: Diagram of naturally ventilated classroom.

Assumptions:

- Only one person in the classroom is infected with the virus.
- The window is the only source of ventilation and the door is assumed to remain closed.
- The classroom is well insulated, with zero buoyancy transfer across solid boundaries.
- Air in the room is well mixed, with a uniform density, buoyancy and hence temperature.
- Hydrostatic variation of pressure is assumed in the room since the window area, where pressure gradients exist, is significantly smaller compared to the area of the interior space.
- Quasi steady state is assumed, meaning that the interface between the layers moves relatively slowly.
- Quanta emission rate, G is $1hr^{-1}$ assuming room occupants are breathing without speaking.
- The volume flux of respiration per person is $r = 0.0002m^3s^{-1}$.
- Each occupant of the space contributes a heat load of $100W$, in addition to a constant $1000W$ associated with central heating.
- The outdoor temperature, T_0 is $7°C$ and the initial temperature of the classroom is $17°C$.
- The volume of the classroom is $200m^3$ and the window width is $6m$.

2.1 Question 2a

It is asked to determine the maximum number of occupants in the classroom, by changing the window opening depth, d, based on the specified constraints. Equations 2.1, 2.2 and 2.3 show a system of ordinary differential equations, modelling the quanta concentration (c), probability of infection (p) and buoyancy (b), as functions of time respectively.

$$\frac{dc}{dt} = \frac{G}{V} - \frac{Qc}{V} \tag{2.1}$$

$$\frac{dp}{dt} = r(N-1)(1-p)c \tag{2.2}$$

$$\frac{db}{dt} = \frac{F - Qb}{V} \tag{2.3}$$

G is the quanta emission rate per infected person, V is the volume of the classroom, Q is the volume flux of outdoor air entering the classroom, r is the volume flux of respiration per person, N is the number of occupants in the space and F is the buoyancy flux. Since the classroom is modelled as a box with a single opening in a vertical plane, Q is defined in Equation 2.4 where $k \approx 0.5$, w is the width of the window opening, and d is its depth.

$$Q = \frac{k}{3} w\sqrt{bd^3} \tag{2.4}$$

Additionally, in air $100W$ is equivalent to a buoyancy flux of $F = 0.0028m^4s^{-3}$, hence in this scenario $F = 0.000028(1000 + 100N)$ because of the constant heat load associated with central heating and the heat load contributed per person. Lastly, Equation 2.5 below calculates the buoyancy, where T is the temperature in Kelvin.

$$b = \frac{g(T - T_0)}{T_0} \tag{2.5}$$

The ODE system is solved numerically in MATLAB. The initial conditions used are $c = 0$ since there is no quanta concentration at $t = 0$ and therefore the probability of infection is also zero, $p = 0$. The initial value for the buoyancy is $b_0 = \frac{9.81(17-7)}{7+273} = 0.3504\ m/s^2$.

Through trial and error, N and d were varied until the maximum number of occupants was obtained such that the probability of infection did not exceed the limit of 0.1, and the temperature remained above $17°C$. The optimal values obtained were $N = 14$ and $d = 0.47m$. It is worth noting that a smaller number of d such as 0.455 also leads to $N = 14$, however it was assumed that a greatest depth is preferred. Figure 2.2 plots the temporal evolutions of the virus concentration, probability of infection and temperature as a function of time, showing the probability and temperature constraints.

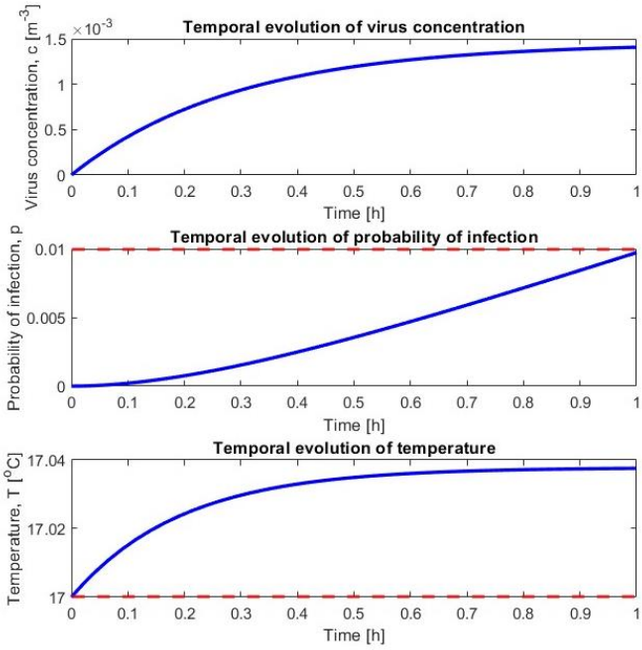


Figure 2.2: Simulation of room for 1 hour $(N = 14, d = 0.47m)$.

It is observed that the trend of the temporal evolution of virus concentration is very similar to the one of temperature. They both increase with a decreasing rate until they reach a constant asymptotic value. Steady state is achieved at those points, where virus concentration and temperature are maximum. Since buoyancy is a function of temperature, it also becomes maximum at the same point $\left(\frac{dc}{dt} = 0, \frac{dT}{dt} = 0, \frac{db}{dt} = 0\right)$. Hence, $F = Qb$, which essentially means that the buoyancy flux as a result of the room occupants is equal to the buoyancy flux as a result of the ventilation.

2.2 Question 2b

The classroom is used for consecutive classes throughout the day. Based on the same constraints, it is again asked to determine the maximum number of occupants in the classroom, by changing the window opening depth, d, when a time interval of 0.5 hours is left between classes to ventilate the space with no occupants in the room. The classroom is simulated for 1.5 hours and it should be ensured that the temperature and virus concentration at the beginning of each class is equal. Essentially, $T(0) = T(1.5\ hours)$ and $c(0) = c(1.5\ hours)$. It is assumed that the virus concentration is zero at the beginning of the first class and the initial temperature is again 17°C. The door opening is not considered as it is assumed that the time taken for the occupants to exit the room through the door is insignificant.

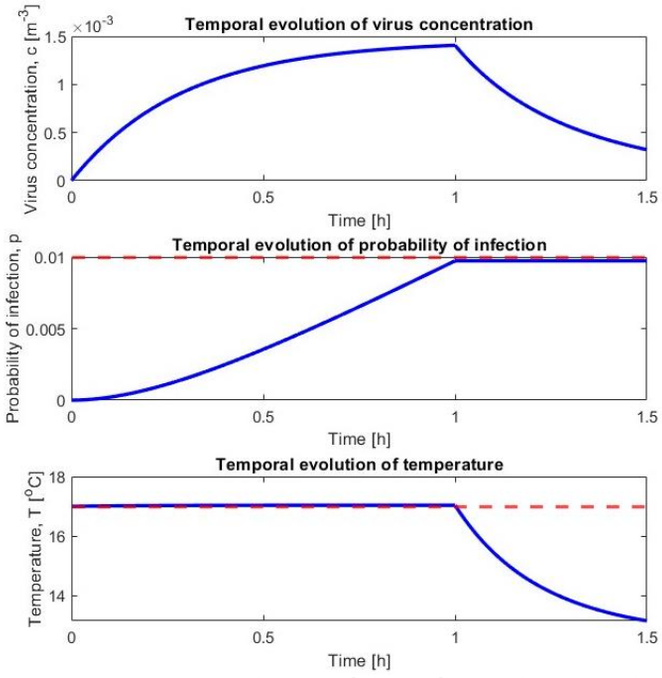


Figure 2.3: Simulation of room for 1.5 hours with unacceptable parameters $(N = 14, d = 0.47m)$.

Figure 2.3 plots the simulation described above using the parameters found in Question 2a $(N = 14, d = 0.47m)$. However, after the ventilation period, the final virus concentration is $3.22 \times 10^{-4} m^{-3}$, hence it does not return to the initial value of zero, while the temperature drops to $13.16°C$. The temperature at t=1.5h should also be $17°C$, such that the next class has an initial temperature of $17°C$. The probability of infection remains less than 0.01, however this could be exceeded in the next class if there is someone infected in the current

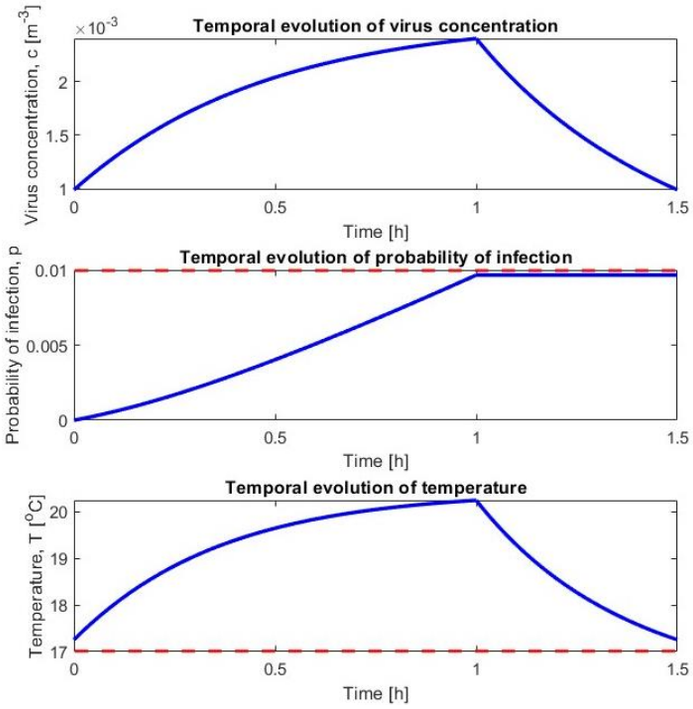


Figure 2.4: Simulation of room after purging the space for 0.5hr with optimized parameters $(N = 8, d = 0.29m)$.

class. These values of N and d are not optimal, and hence trial and error is performed in MATLAB to find appropriate values which satisfy all conditions.

When trying to find a value of N, it is know that it will be less than 14 obtained in Question 2a since it was assumed that initial concentration is zero which is unrealistic and optimistic. Also, the window depth should be less than 0.47m to prevent the excessive heat loss which results in the drop of temperature in Figure 2.3. However, this in turn would decrease ventilation, and again the maximum occupants allowed in the room should be decreased. Since the system is repetitive throughout the day the initial conditions used have to be replaced with periodic boundary conditions, such that the virus concentration and buoyancy at the start of the simulation correspond to their values at the end of the simulation. The probability at $t = 0$ is always zero. Adopting this approach using trial and error, was found that the optimal number of occupants is $N = 8$ and the depth of the window $d = 0.29m$. Figure 2.4 plots the simulation described, which works for all constraints. The probability does not exceed the set limit of 0.01. Table 2.1 summarises the concentration and temperature values which make $c(0) = c(1.5)$ and $T(0) = T(1.5)$. An occupancy of 8 is a small number compared to the usual number of pupils in a class, and therefore more heating with a greater window depth could be a solution in increasing the allowed room capacity, without compromising comfort.

Table 2.1: Final values of concentration and temperature.

$\boldsymbol{c(0) = c(1.5)}$	$9.87 \times 10^{-4}\ m^{-3}$
$\boldsymbol{T(0) = T(1.5)}$	$17.25\ °C$

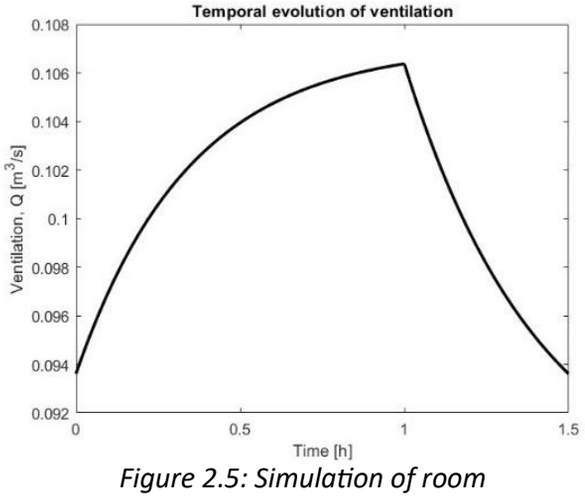


Figure 2.5: Simulation of room ventilation for 1.5 hours.

Figure 2.5 plots the room ventilation within the 1.5 hour period. For a box model this was defined previously in Equation 2.4, as a function of buoyancy and window depth. Since buoyancy itself is a function of temperature, the shape of the ventilation plot is equal to the one of the temperature variation. The peak is after 1 hour, at the end of the class when the temperature is greatest, after the 8 occupants have produced a heat equal to 800W, followed by a rapid decrease during the half an hour where the room is unoccupied, and hence the 800W leave.

References

Götting, M.C. (2021) *Gender profile of festival-goers in the United Kingdom (UK) from 2012 to 2016*. https://www.statista.com/statistics/282836/gender-distribution-of-visitors-to-uk-music-festivals/ [Accessed 20th November 2023].

NHS. (2022) *Health Survey for England, 2021 part 1*. https://digital.nhs.uk/data-and-information/publications/statistical/health-survey-for-england/2021/part-4-trends [Accessed 20th November 2023].

Appendix A: MATLAB Code for Question 1

```
%% CIVE60005 COURSEWORK
% By: Mikaella Liveri
% Question 1 -  Outdoor transmission of COVID-19

clear;
clc;
close all;

%% Loading data
% Quanta emission rates depending on action and level of activity [s^-1]
B_rest=1/3600;
B_heavy_act=8/3600;
S_light_act=15/3600;
S_loudly=98/3600;

% Turbulent diffusivity [m^2/s]
D_T=1;

% Duration of concert [s]
T=3*3600;

% For linspace
n=1000;
%% Question 1a
M_dot=S_loudly*4; % 4 people per platform
h=1.7;

% Setting the coordinates of the platform from the defined origin [m]
x_1=-8; y_1=8;
x_3=0; y_3=8;
x_7=-4; y_7=4;
x_13=0.5; y_13=0.5;

% Creating x and y vectors
x_a=[x_1, x_3, x_7, x_13];
y_a=[y_1, y_3, y_7, y_13];

% Finding radial distances
r_a=sqrt(x_a.^2+y_a.^2); % plane source

% Since datum is defined at z=h, perfect reflector is at z=-2h
z=0;

% Finding radial distances of image source
r_image=sqrt(x_a.^2+y_a.^2+(z+2*h)^2); % image source

% Time vector
t_log= [0:0.01:5 5:10800];

% Finding concentrations
C=zeros(length(r_a),length(t_log));
for j=1:4
C(j,:)= (M_dot/(4*pi*D_T*r_a(j)))*erfc(r_a(j)./(sqrt(4*D_T.*t_log))) +
(M_dot/(4*pi*D_T*r_image(j)))*erfc(r_image(j)./(sqrt(4*D_T.*t_log)));
end

% Plotting
figure;
```

```
subplot(2,1,1)
plot(t_log,C,'LineWidth',3);
xlabel('Time [s]', 'FontSize',12);
ylabel('Quanta Concentration [m^{-3}]', 'FontSize',12);
legend('Platform 1','Platform 3','Platform 7', 'Platform 13');
title('Temporal evolution of the virus concentration at head height without wind',
'FontSize',12);
axis tight; grid on; box on;

% Log scale plot
subplot(2,1,2)
semilogx(t_log,C,'LineWidth',3);
xlabel('Time [s]', 'FontSize',12);
ylabel('Quanta Concentration [m^{-3}]', 'FontSize',12);
legend('Platform 1','Platform 3','Platform 7', 'Platform 13');
title('Temporal evolution of the virus concentration at head height without wind',
'FontSize',12);
axis tight; grid on; box on;

%% Question 1b
% i) no wind conditions
% Creating space meshgrid at head height
x=linspace(-9,9);
y=linspace(-9,9);
[X,Y]=meshgrid(x,y);

% Distance from the source
r=sqrt(X.^2+Y.^2);
% Interested in concentration around the infected platform
r(r<1)=1; % To avoid singularity
r_im=sqrt(X.^2+Y.^2+(z+2*h)^2);
C_nowind=(M_dot./(4*pi*D_T*r)).*erfc(r./(sqrt(4*D_T*T))) +
(M_dot./(4*pi*D_T*r_im)).*erfc(r_im./(sqrt(4*D_T*T)));

% ii) Spatially uniform wind with a speed of 3m/s from S-W
U=3; %[m/s]
C_wind=(M_dot./(4*pi*D_T*r)).*exp(-U*(r-Y)/(2*D_T))+(M_dot./(4*pi*D_T*r_im)).*exp(-
U*(r_im-Y)/(2*D_T));

% Surface plot of concentration with no wind condition and wind conditions
% respectively
figure;
sgtitle('Surface plots of spatial distribution of the
virus','FontSize',14,'FontWeight','bold')
subplot(1,2,1)
surf(X,Y,C_nowind);
shading interp
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
zlabel('Quanta Concentration [m^{-3}]','FontSize',12)
title('No wind conditions','FontSize',12)

subplot(1,2,2)
surf(X,Y,C_wind);
shading interp
xlabel('x [m]', 'FontSize',12);
ylabel('y [m]','FontSize',12);
zlabel('Quanta Concentration [m^{-3}]','FontSize',12)
title('Wind conditions','FontSize',12)
```

```
% Contour plot of concentration with no wind condition and wind conditions
% respectively
figure;
subplot(1,2,1)
sgtitle('Contour plots of spatial distribution of the
virus','FontSize',14,'FontWeight','bold')
contourf(X,Y,C_nowind);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('No wind conditions','FontSize',12)

subplot(1,2,2)
contourf(X,Y,C_wind);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Wind conditions','FontSize',12)

% Finding maximum concentrations of the 2 conditions
bmaxC_nowind=max(max(C_nowind));
bmaxC_wind=max(max(C_wind));

%% Question 1c
% Boundary modelled as perfect absorber
C_nowind2=(M_dot./(4*pi*D_T*r)).*erfc(r./(sqrt(4*D_T*T)))-
(M_dot./(4*pi*D_T*r_im)).*erfc(r_im./(sqrt(4*D_T*T)));
C_wind2=(M_dot./(4*pi*D_T*r)).*exp(-U*(r-Y)/(2*D_T))-
(M_dot./(4*pi*D_T*r_im)).*exp(-U*(r_im-Y)/(2*D_T));

% Surfance plot of concentration with no wind condition and wind conditions
figure;
sgtitle('Surface plots of spatial distribution of the virus with
disinfectant','FontSize',14,'FontWeight','bold')
subplot(1,2,1)
surf(X,Y,C_nowind2);
shading interp
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('No wind conditions','FontSize',12)

subplot(1,2,2)
surf(X,Y,C_wind2);
shading interp
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Wind conditions','FontSize',12)

% Contour plot of concentration with no wind condition and wind conditions
figure;
sgtitle('Contour plots of spatial distribution of the virus with
disinfectant','FontSize',14,'FontWeight','bold')
subplot(1,2,1)
contourf(X,Y,C_nowind2);
axis equal;
colorbar;
```

```
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('No wind conditions','FontSize',12)

subplot(1,2,2)
contourf(X,Y,C_wind2);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Wind conditions','FontSize',12)

% Mean concentrations
mean_C_nowind=mean(C_nowind(:));
mean_C_wind=mean(C_wind(:));
mean_C2_nowind=mean(C_nowind2(:));
mean_C2_wind=mean(C_wind2(:));

% Maximum concentrations
max_C_nowind=max(C_nowind(:));
max_C_wind=max(C_wind(:));
max_C2_nowind=max(C_nowind2(:));
max_C2_wind=max(C_wind2(:));

fprintf('QUESTION 1c \n')
fprintf('Mean Concentrations: \n')
fprintf('No Wind: No disinfectant: %g Disinfectant %g \n', mean_C_nowind,
mean_C2_nowind)
fprintf('Wind: No disinfectant: %g Disinfectant %g \n', mean_C_wind, mean_C2_wind)
fprintf('Maximum Concentrations: \n')
fprintf('No Wind: No disinfectant: %g Disinfectant %g \n', max_C_nowind,
max_C2_nowind)
fprintf('Wind: No disinfectant: %g Disinfectant %g \n\n', max_C_wind, max_C2_wind)

% Plots showing the ratio of concentrations with and without disinfectant
figure;
subplot(2,1,1)
contourf(X,Y,C_nowind2./C_nowind);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Ratio C_{disinfectant}/C_{no disinfectant} no wind
conditions','FontSize',12)

subplot(2,1,2)
contourf(X,Y,C_wind2./C_wind);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Ratio C_{disinfectant}/C_{no disinfectant} wind conditions','FontSize',12)

%% Question 1d
% head height is raised by 2m
h_new=2+h; %[m]

% Assuming no wind conditions
% Finding concentrations
```

```
r_ima=sqrt(x_a.^2+y_a.^2+(2*h_new)^2);
C=zeros(length(r_a),length(t_log));
for j=1:4
C(j,:)= (M_dot/(4*pi*D_T*r_a(j)))*erfc(r_a(j)./(sqrt(4*D_T.*t_log))) +
(M_dot/(4*pi*D_T*r_ima(j)))*erfc(r_ima(j)./(sqrt(4*D_T.*t_log)));
end

figure;
semilogx(t_log,C,'LineWidth',3);
xlabel('Time [s]', 'FontSize',12);
ylabel('Quanta Concentration [m^{-3}]', 'FontSize',12);
legend('Platform 1','Platform 3','Platform 7', 'Platform 13');
title('Temporal evolution of the virus concentration at head height without wind',
'FontSize',12);
axis tight; grid on; box on;

r_im=sqrt(X.^2+Y.^2+(z+2*h_new)^2);
C_nowind=(M_dot./(4*pi*D_T*r)).*erfc(r./(sqrt(4*D_T*T))) +
(M_dot./(4*pi*D_T*r_im)).*erfc(r_im./(sqrt(4*D_T*T)));
C_wind=(M_dot./(4*pi*D_T*r)).*exp(-U*(r-Y)/(2*D_T))+(M_dot./(4*pi*D_T*r_im)).*exp(-
U*(r_im-Y)/(2*D_T));
C_nowind2=(M_dot./(4*pi*D_T*r)).*erfc(r./(sqrt(4*D_T*T)))-
(M_dot./(4*pi*D_T*r_im)).*erfc(r_im./(sqrt(4*D_T*T)));
C_wind2=(M_dot./(4*pi*D_T*r)).*exp(-U*(r-Y)/(2*D_T))-
(M_dot./(4*pi*D_T*r_im)).*exp(-U*(r_im-Y)/(2*D_T));

figure
subplot(2,2,1)
sgtitle('Comparison of spatial distribution of the virus at raised height under the
following conditions','FontSize',14,'FontWeight','bold')
contourf(X,Y,C_nowind);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Without wind and without disinfectant','FontSize',12)

subplot(2,2,2)
contourf(X,Y,C_wind);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('With wind and without disinfectant','FontSize',12)

subplot(2,2,3)
contourf(X,Y,C_nowind2);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
title('Without wind and with disinfectant','FontSize',12)

subplot(2,2,4)
contourf(X,Y,C_wind2);
axis equal;
colorbar;
xlabel('x [m]','FontSize',12);
ylabel('y [m]','FontSize',12);
```

```
title('With wind and with disinfectant','FontSize',12)

% Mean concentrations
mean_C_nowind=mean(C_nowind(:));
mean_C_wind=mean(C_wind(:));
mean_C2_nowind=mean(C_nowind2(:));
mean_C2_wind=mean(C_wind2(:));

% Maximum concentrations
max_C_nowind=max(C_nowind(:));
max_C_wind=max(C_wind(:));
max_C2_nowind=max(C_nowind2(:));
max_C2_wind=max(C_wind2(:));

fprintf('QUESTION 1d: Raised Platforms \n')
fprintf('Mean Concentrations: \n')
fprintf('No Wind: No disinfectant: %g Disinfectant %g \n', mean_C_nowind,
mean_C2_nowind)
fprintf('Wind: No disinfectant: %g Disinfectant %g \n', mean_C_wind, mean_C2_wind)
fprintf('Maximum Concentrations: \n')
fprintf('No Wind: No disinfectant: %g Disinfectant %g \n', max_C_nowind,
max_C2_nowind)
fprintf('Wind: No disinfectant: %g Disinfectant %g \n', max_C_wind, max_C2_wind)
```

Appendix B: MATLAB Code for Question 2

```
%% CIVE60005 COURSEWORK
% By: Mikaella Liveri
% Question 2 - Indoor transmission of COVID-19

clear;
clc;
close all;

%% Data
G=1/3600; % [s^-1]
r=0.0002; % [m^3/s]
To=7+273; % [K]
T=17+273; % [K]
w=6; % [m]
k=0.5; % opening in vertical plane [dimensionless]
g=9.81; % [m/s^2]
V=200; % [m^3]

%% Question 2a
% Maximize N by changing d
d=0.47; % Depth of window
N=14; % Number of people
t=[0, 3600]; %  Interested in 1 hour classes

F=0.000028*(1000+N*100);

% ODE initial conditions
b_0=g*(T-To)/To; % buoyancy
c_0=0; % concentration at t=0 is 0
p_0=0; % probability of infection at t=0 ia 0
I_0=[b_0; c_0; p_0]; % initial conditions to solve the ode

% Solving ODE
[t_s, A]=ode45(@ode,t,I_0,[],k,w,G,V,r,N,F,d);

% Solutions
b=A(:,1);
c=A(:,2);
p=A(:,3);

Temp =b*To/g+To-273; % [Celsius]

% Plots of concentration, probability, and temperature
figure;

% Plotting concentration
subplot(3,1,1);
plot(t_s/3600,c,'b','LineWidth',2);
xlabel('Time [h]');
ylabel('Virus concentration, c [m^{-3}]');
title('Temporal evolution of virus concentration');

% Plotting probability
subplot(3,1,2);
plot(t_s/3600,p,'b','LineWidth',2);
hold on; yline(0.01,'r--','LineWidth',2);
xlabel('Time [h]');
ylabel('Probability of infection, p');
```

```
title('Temporal evolution of probability of infection');

% Plotting temperature
subplot(3,1,3);
plot(t_s/3600,Temp,'b','LineWidth',2);
yline(17,'r--','LineWidth',2);
xlabel('Time [h]');
ylabel('Temperature, T [^oC]');
title('Temporal evolution of temperature');

%% Question 2b
% Maximise N through changing d (window depth), modelling 1.5 hours. During
% the first hour there are N students and during the last 0.5 hour all
% occupants leave the room to ventilate the space.
c0 = 0;
C0 = [b_0;c0;0];

[t_b, B]=ode45(@ode,[0 3600],C0,[],k,w,G,V,r,N,F,d);

% Solutions
b2=B(:,1);
c2=B(:,2);
p2=B(:,3);

% Using the solutions as initial conditions for the 30 minutes after class,
% when the room is empty
b_1=B(end,1);
c_1=B(end,2);
p_1=B(end,3);
I_0=[B(end,1); B(end,2); B(end,3)];

% Defining the new parameters of the problem
N=0; % empty room
r=0; % no respiration in room
G=0; % no quanta emission
F=0.000028*(1000+N*100); % new buoyancy flux

[t_b2, C]=ode45(@ode,[3600 5400],I_0,[],k,w,G,V,r,N,F,d);

% Solutions
b3=C(:,1);
c3=C(:,2);
p3=C(:,3);

% Temperatures
T_2=b2*To/g+To-273; % [Celsius]
T_3=b3*To/g+To-273; % [Celsius]

% Plots of concentration, probability, and temperature
figure;

% Plotting concentration
subplot(3,1,1);
plot(t_b/3600,c2,'b', t_b2/3600, c3,'b','LineWidth',2);
xlabel('Time [h]');
ylabel('Virus concentration, c [m^{-3}]');
title('Temporal evolution of virus concentration');

% Plotting probability
```

```
subplot(3,1,2);
plot(t_b/3600,p2,'b',t_b2/3600, p3,'b','LineWidth',2);
hold on; yline(0.01,'r--','LineWidth',2);
xlabel('Time [h]');
ylabel('Probability of infection, p');
title('Temporal evolution of probability of infection');

% Plotting temperature
subplot(3,1,3);
plot(t_b/3600,T_2,'b',t_b2/3600, T_3,'b','LineWidth',2);
yline(17,'r--','LineWidth',2);
xlabel('Time [h]');
ylabel('Temperature, T [^oC]');
title('Temporal evolution of temperature');

% Final concentration and temperature values
c3_final=c3(end);
T_3_final=T_3(end);

% Finding optimal number of people such that T(0)=T(1.5hours) and
% c(0)=c(1.5hours)
diff_c=abs(B(1,2)-C(end,2))/B(1,2);
diff_b=abs(B(1,1)-C(end,1))/B(1,1);

if diff_c>diff_b
    diff=diff_c;
else
    diff=diff_b;
end

% A while loop with a tolerance of the difference of concentration and buoyancy  of
0.0000001 is set up
while diff>0.0000001
    N=8;
    d=0.29;
    r=0.0002; % respiration in room
    G=1/3600; % quanta emission
    F=0.000028*(1000+N*100); % buoyancy flux
    I_0=[C(end,1); C(end,2); 0];
    [t_1, B]=ode45(@ode,[0 3600],I_0,[],k,w,G,V,r,N,F,d);
    b3=B(end,1);
    c3=B(end,2);
    p3=B(end,3);
    I_0=[b3; c3; p3];
    N=0; r=0; G=0; F=0.000028*(1000+N*100);
    [t_2, C]=ode45(@ode,[3600 5400],I_0,[],k,w,G,V,r,N,F,d);

    diff_c=abs(B(1,2)-C(end,2))/B(1,2);
    diff_b=abs(B(1,1)-C(end,1))/B(1,1);

    if diff_c>diff_b
        diff=diff_c;
    else
        diff=diff_b;
    end
end

% Solutions
b4=B(:,1);
```

```
c4=B(:,2);
p4=B(:,3);
T_4=b4*To/g+To-273; % [Degrees Celsius]

b5=C(:,1);
c5=C(:,2);
p5=C(:,3);
T_5=b5*To/g+To-273; % [Degrees Celsius]

% Plots of concentration, probability and temperature
figure;

% Plotting concentration
subplot(3,1,1);
plot(t_1/3600,c4,'b', t_2/3600, c5,'b','LineWidth',2);
xlabel('Time [h]');
ylabel('Virus concentration, c [m^{-3}]');
title('Temporal evolution of virus concentration');

% Plotting probability
subplot(3,1,2);
plot(t_1/3600,p4,'b',t_2/3600, p5,'b','LineWidth',2);
hold on; yline(0.01,'r--','LineWidth',2);
xlabel('Time [h]');
ylabel('Probability of infection, p');
title('Temporal evolution of probability of infection');

% Plotting temperature
subplot(3,1,3);
plot(t_1/3600,T_4,'b',t_2/3600, T_5,'b','LineWidth',2);
yline(17,'r--','LineWidth',2);
xlabel('Time [h]');
ylabel('Temperature, T [^oC]');
title('Temporal evolution of temperature');

% Calculation of ventilation Q over time for case found
Q1 = k/3*w*sqrt(B(:,1)*d^3);
Q2 = k/3*w*sqrt(C(:,1)*d^3);

figure;
grid on; box on;
plot(t_1/3600,Q1,'k',t_2/3600,Q2,'k','LineWidth',2);
xlabel('Time [h]');
ylabel('Ventilation, Q [m^{3}/s]');
title('Temporal evolution of ventilation');

% Finding the final values of c(0)=c(1.5) and T(0)=T(1.5)
c_final=c4(1);
T_final=T_4(1);

%% FUNCTION FOR SYSTEM OF ODES
function dydt=ode(t,I_0,k,w,G,V,r,N,F,d)

% initial conditions
b=I_0(1);
c=I_0(2);
p=I_0(3);

% Volume flux
```

```
Q=(k/3)*w*sqrt(b*d^3);

dbdt=(F-Q*b)/V;
dcdt=G/V-Q*c/V;
dpdt=r*(N-1)*(1-p)*c;

dydt=[dbdt; dcdt; dpdt];
end
```