VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

MATHEMATICAL MODELING (CO2011)

Assignment

Dynamical systems in forecasting Greenhouse Micro-climate

Advisor: Nguyen Tien Thinh
Nguyen An Khuong

TA: Tran Trung Hieu (tthieu.sdh20@hcmut.edu.vn)

Students: Doan Anh Tien - 1852789 (Class CC01, **Team Leader**)
Ho Hoang Thien Long - 1852161 (Class CC01)
Nguyen Duy Tinh - 1852797 (Class CC05)
Bui Hoang Phuc - 1952925 (Class CC02)
Le Minh Dang - 1952041 (Class CC02)

HO CHI MINH CITY, 28TH AUGUST 2021

Contents

1 Member list & Workload

No.	Fullname	Student ID	Problems	Percentage of work
1	Doan Anh Tien	1852789	- Theory: Question 1a - Application: Question 2, 3, 4, 5 - Task Tracking & Progress Log	100%
2	Ho Hoang Thien Long	1852161	- Theory: Question 1a - Application: Question 2, 3, 4, 5 - Final Check	100%
2	Nguyen Duy Tinh	1852797	- Theory: Question 1b,c,d - Application: Question 3, 4, 5 - Writing Report	100%
2	Bui Hoang Phuc	1952925	- Theory: Question 1b,c,d,e - Application: Question 4, 5 - Writing Report	100%
2	Le Minh Dang	1952041	- Theory: Question 1b,c,d,e - Application: Question 4,5 - Writing Report	100%

2 Background Section

2.1 Problem A

Present the definition and classification according to different criteria, the general form of dynamical systems, and especially first-order differential equations systems with initial condition at time t0, which are continuous dynamical systems used in this assignment.

2.1.1 Definition

There are many definitions of dynamical system arcording to different studies. In this report, definitions and theories of dynamical system is based on [GFH13], *"A dynamical system is a relationship among terms in a sequence"*, whose *"a sequence is a function whose domain is the set of all non-negative integers and whose range is a subset of the real numbers"*.

In simple term, relationship here represents the change of quantities usually respecting to time domain depends on its current states, values or multiple states, values in the sequence (table of values) or external terms from one period to the next, particularly represented as a set of algebraic equations. This relationship represents dynamical system.

The dynamical system that describes the change of quantities respect to n domain can be represented as the general equation

$$change = \Delta a_n = some\,function\; f(terms\,in\,the\,sequence,\; external\,terms)$$

Whose Δa_n is the change value between some quantity values in the interval n, which is a value in a discrete period in discrete systems or an instantaneous rate of change in a extremely small interval that can be consider as an instantaneous point in continuous systems.

Solutions of the above equation is the series of values of the specific quantity.

2.1.2 Classification

There are 2 main types of dynamical system: Discrete dynamical system (Difference equations) and Continuous dynamical system (Differential equations)

2.1.2.a Discrete dynamical system

Discrete dynamical system describes the change in behaviors of some quantities in discrete periods, usually time periods, represented as difference equations.

Difference equations represent the relationship between the change of a current value and successive values of the function respect to a discrete domain. The difference equations is categorized into n-order, linear/non-linear and homogenuous/non-homogenuous. The report will introduce the first-order homogenuous difference equations.

For a sequence of numbers $A = \{a_0, a_1, a_2, a_3, ..., a_n\}, n \,\epsilon\, \mathbb{N}$. The n^{th} first order homogenuous differences, with a_0 is the initial value, are:

$$\Delta a_n = a_{n+1} - a_n = some\,function\; f(n, a_n)$$

Figure 1 shows the vertical difference or change between 2 discrete consecutive numbers in sequence A in 1 discrete period of time between n and $n+1$. This change allows us to know whether the trend of the system quantity is increasing, decreasing or unchange respect to a time period.

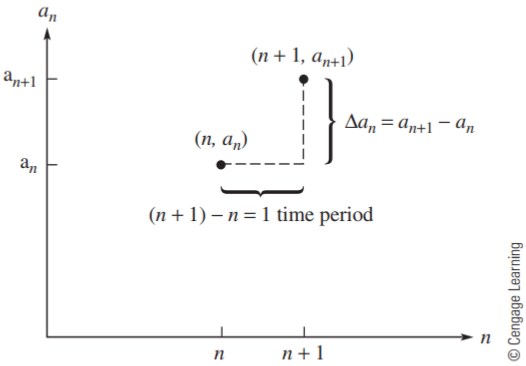


Figure 1: The n^{th} first differences

Solutions of difference equations is a sequence of numerical values of a quantity that can be determined by starting with an initial value of the difference equation and iterating the sequence of subsequent values, then we can get the pattern of the system. Let's consider an example.

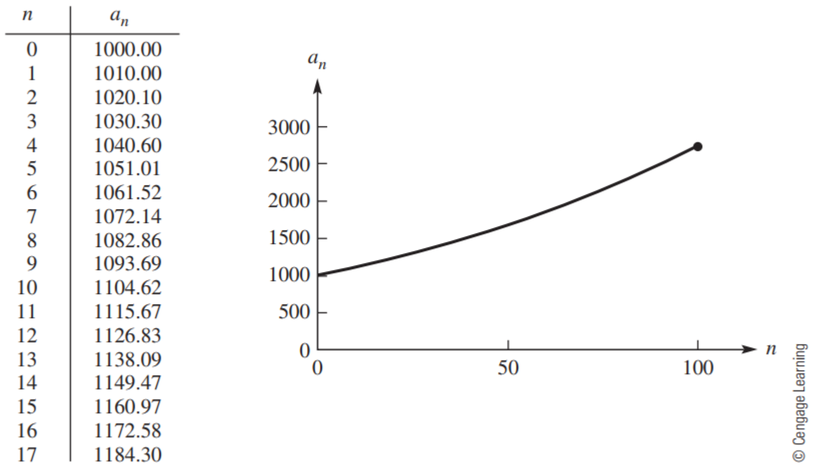
Consider the value of a savings certificate initially worth $1000 that accumulates interest paid each month at 1% per month. Let n represent the month n, a_n represent the value of saving certificate in that month. Determining the dynamical system

$$\Delta a_n = a_{n+1} - a_n = 0.01a_n, n = 0, 1, 2, 3, ...$$

$$a_0 = 1000$$

By iterating the sequence of the above difference equation to compute the numerical solutions a_n, we can plot the graph represented in Figure 2 and observe the rising trend in the value of savings certicate by every month that seems to have no bound.

In overall, the discrete dynamical system can represent the changes and trends of a particular system in discrete time domain. This model is suitable to solve problem that only consider discrete intervals, such as the savings certificate above, the value only change when the next month comes, which means it is meaningless to consider the change of value in the middle of that month. Moreover, if there are more demand in study the change of many dependent variables, the system of difference equations should be considered.



n	a_n
0	1000.00
1	1010.00
2	1020.10
3	1030.30
4	1040.60
5	1051.01
6	1061.52
7	1072.14
8	1082.86
9	1093.69
10	1104.62
11	1115.67
12	1126.83
13	1138.09
14	1149.47
15	1160.97
16	1172.58
17	1184.30

Figure 2: Growth of the values of a savings certificate

2.1.2.b Continuous dynamical system

While discrete dynamical system describes change of behaviours in the discrete time intervals (change of money in savings certificate every month), continuous dynamical system describes this change throughtout the time domain continuously (change of the amount of CO_2 in the greenhouse respect to time). This kind of dynamical system is represented as differential equations.

Differential equations represents the relationship between the rates of change of functions in terms of derivatives and their functions in terms of quantities. There are 3 main kinds of differential equations categorized by Newton. This report will introduce one form of differential equations, the ordinary first-order diffential equation (ODE) which represents the relationship between the quantity functions and its first derivative linearly.

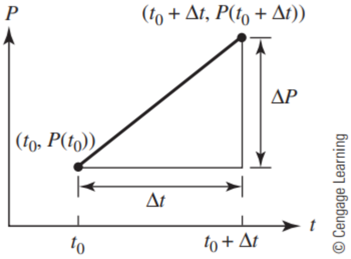
The general form of the first-order differential equation in which P is a function in t domain, allow t to approach 0 we get the rate of change of function P respect to t, in addition with initial value of P at initial t_0 are known

$$\lim_{x \to 0} \frac{\Delta P}{\Delta t} = \frac{dP}{dt} = f(t, P)$$

$$P(t_0) = P_0$$

In this case, the general change Δa_n is replaced by the instantaneous rate of change $\dfrac{dP}{dt}$, where P and t are dependent variables (appear explicily on the right side of the equations).

Figure 3 shows ΔP and Δt is a extremely small difference between 2 values P and between 2 values t, respectively; $\frac{\Delta P}{\Delta t}$ can be intepret as the slope of the line segment connecting 2 points $(t_0, P(t_0))$ and $(t_0 + \Delta t, P(t_0 + \Delta t))$. Intuiatively speaking, if the difference ΔP is larger, then the fraction $\frac{\Delta P}{\Delta t}$ is bigger, which makes the slope goes more "upward", and so on.



© Cengage Learning

Figure 3: The first-order differential

The solution of differential equation is the form

$$P = P(t)$$

Whenever we specify an initial condition $P(t_0) = P_0$ for the solutions of differential equation $P' = f(t, P)$, these solutions can be represented as the solution curve of the function $P(t)$ in (P,t)-plane (graph of the solution) required to pass through the point (t_0, P_0) and has a slope $f(t_0, P_0)$. The exact solutions can be specfied by analytical method such as integral applied for some simple equations. However, more difficult equations that are hard to apply analytical method can apply approximate method such as Euler to approximate the solutions.

One differential equation is good for representing the rate of change of one quantiy respect to a certain domain. Howerver, some real world systems have more than one quantity that affect each other. For instance, the study of dynamic population growths of varity of animal species has to point out how one animal population affect others and so on. In order to model those systems, the system of differential equations is considered. This report will represent one form of the systems of differential equations involving 2 ODEs.

The general form of the system

$$\begin{cases} \dfrac{dx}{dt} = f(x, y) \\ \dfrac{dy}{dt} = g(x, y) \end{cases} \tag{1}$$

This system is called an autonomous system of differential equations due to the absent of independent variable t on the right handside, which means the system do not depend on time and only depends on 2 dependent variables x and y.

The solutions of this system of diferential equations is a pair of 2 parametric equations

$$\begin{cases} x = x(t) \\ y = y(t) \end{cases} \tag{2}$$

representing as a curve in xy-plane describing the dependence of x and y. The system of diferential equations can be solved using analytical methods to find the exact solutions, but some difficult system that is hard to apply analytical methods can be estimated the approximate solutions by using aproximation methods such as Euler or Runge-Kutta algorithms.

Examples about diffential equation and system of diffential equations is presented in **Problem C**.

2.2 Problem B

Introduce a necessary and sufficient condition for the above systems of differential equations to exist and have unique solutions.

In the system of differential of equations describing CO_2 concentration and its rate of change in the greenhouse, we can factor out CO_{2Air} and CO_{2Top} multiply by the results of operations on parameters (from the CO_2 fluxes formulas); moreover, the photothesis rate P only takes 1 solution as well as independent of time. Therefore, the system is linear independently respect to 2 variable CO_{2Air} and CO_{2Top}, we can represent the 2 equations in the form of linear system

$$\begin{cases} x' = a(t)x + b(t)y \\ y' = c(t)x + d(t)y \end{cases} \tag{3}$$

converted to matrix form

$$\begin{cases} \begin{bmatrix} x \\ y \end{bmatrix}' = \begin{bmatrix} a(t) & b(t) \\ c(t) & d(t) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \end{cases} \tag{4}$$

The solutions are in the form

$$s = c_1 s_1 + c_2 s_2$$

where $x = CO2_{Air}$, $y = CO2_{Top}$, $a(t), b(t), c(t), d(t)$ are functions of t representing the results of operations on greenhouse parameters multiply by x, y, in this case t is independent and can be thought as a horizontal line in $at/bt/ct/dt$-plane, no s_i is constant muliple of each other.

If we call the matrix $A(t)$ and s represent the matrix of functions of t and solutions of the equation (4) x, y, respectively, and declare an initial condition to solve the above system, we get

$$\begin{cases} s' = A(t)s \\ s(t_0) = s_0 \end{cases} \tag{5}$$

According to the theorem of linear systems stated in [Ope11], the condition for the above system to exist and have a unique solutions is *"if the entries of the square matrix A(t) are continuous on an open interval I containing* t_0*, then the initial value problem of the system (5) has one and only one solution s(t) on the interval I."*.

It can be intepreted in the systems of CO_2 concentration is that if the results of operations on the parameters of the greenhouse affecting the system is defined (continuous) and have values at every value of time, then the system exists and has a unique pair of CO_2 concentration Air, Top.

2.3 Problem C

Give some examples of solvable first-order differential equations and their exact solutions.

Example 1:
a) Solve the differential equation $\frac{dx}{dy} = \frac{y^2}{x^2}$
b) Find the solution of this equation that satisfies the initial condition

$$y\left(0\right) = 2$$

Solution for Example 1:
a) We write the equation in terms of differentials and integrate both sides:

$$y^2 dy = x^2 dx$$
$$\int \mathrm{y}^2 \, \mathrm{d}y = \int \mathrm{x}^2 \, \mathrm{d}x$$
$$\tfrac{1}{3}y^3 = \tfrac{1}{3}x^3 + C$$

where C is an arbitrary constant. (We could have used a constant C_1 on the left size and other constatnt C_2 on the right size. But then we could combine these constants by writing
$C = C_2 - C_1$
Solving for y, we will get

$$y = \sqrt[3]{x^3 + 3C}$$

We could leave the solution like this or we could write it in the form

$$y = \sqrt[3]{x^3 + K}$$

where $K = 3C$.(Since C is an arbitrary constant, so is K.)
b) If we put $x = 0$ in the general solution in part (a), we get $y\left(0\right) = \sqrt[3]{K}$.To satisfy the initial condition $y\left(0\right) = 2$, we must have $\sqrt[3]{K} = 2$ and so $K = 8$. Thus the solution of the initial-value problem is

$$y = \sqrt[3]{x^3 + 8}$$

Example 2: Solve system of differential equations:

$$\begin{cases} \dfrac{dy_1}{dx} = 4y_1 + y_2 - e^{2x} \\ \dfrac{dy_2}{dx} = -2y_1 + y_2 \end{cases} \tag{6}$$

We consider the homogenous system:

$$\begin{cases} \dfrac{dy_1}{dx} = 4y_1 + y_2 \\ \dfrac{dy_2}{dx} = -2y_1 + y_2 \end{cases} \tag{7}$$

Coefficient matrix:

$$A = \begin{pmatrix} 3 & 1 \\ -2 & 1 \end{pmatrix}$$

Characteristic equation:

$$A = \begin{vmatrix} 4-\lambda & 1 \\ -2 & 1-\lambda \end{vmatrix} = 0$$
$$\Leftrightarrow (4-\lambda)(1-\lambda)+2=0$$
$$\Leftrightarrow \lambda^2-5\lambda+6=0 \Leftrightarrow \left[\begin{array}{l} \lambda=2 \\ \lambda=3 \end{array}\right.$$

Find general solution:
For $\lambda = 2$, coordinates of the particular vector of the euquations above is:

$$\begin{cases} 2x_1+x_2=0 \\ -2x_1+x_2=0 \end{cases} \Leftrightarrow x_2=-2x_1 \tag{8}$$

Particular vector $v_1(1;-2)$, basic solution

$$Y_1 = \begin{pmatrix} e^{2x} \\ -2e^{2x} \end{pmatrix} \tag{9}$$

For $\lambda = 3$ Particular vector $v_1(1;-1)$, basic solution

$$Y_2 = \begin{pmatrix} e^{3x} \\ -e^{3x} \end{pmatrix} \tag{10}$$

Find particular solution
Let $Y^* = C_1(x)Y_1 + C_2(x)Y_2$ be a particular solution.

$$Y^* = \begin{pmatrix} C_1(x)e^{2x}+C_2(x)e^{3x} \\ -2C_1(x)e^{2x}-C_2(x)e^{3x} \end{pmatrix} \tag{11}$$

Placing Y^* in the original equations, we have:

$$\begin{cases} C_1'(x)e^{2x}+C_2'(x)e^{3x}=-e^{2x} \\ -2C_1'(x)e^{2x}-C_2'(x)e^{3x}=0 \end{cases} \Leftrightarrow \begin{cases} C_1'(x)=1 \\ C_2'(x)=-2e^{-x} \end{cases} \Leftrightarrow \begin{cases} C_1'(x)=x \\ C_2'(x)=2e^{-x} \end{cases} \tag{12}$$

General solution
Therefore,we have:

$$Y^* = \begin{pmatrix} xe^{2x}+2e^{2x} \\ -2xe^{2x}-2e^{2x} \end{pmatrix} \tag{13}$$

General solution:

$$Y = AY_1+BY_2+Y^* = \begin{pmatrix} Ae^{2x}+Be^{3x}+(x+2)e^{2x} \\ -2Ae^{2x}-Be^{3x}-(2x+2)e^{2x} \end{pmatrix} \tag{14}$$

(A, B are constants)

2.4 Problem D

Introduce and present the approximation steps of the Explicit Euler and Explicit Runge–Kutta of order 4 algorithms to solve general first-order differential equations.

- **What is Explicit Euler's Method?**

  This fundamental idea is based on a very simple principle. Suppose that a particle is moving in such a way that, at an initial time x_0, its position is equal to y_0 and that, at this time, the velocity is known to be v_0. The simple principle is that, in a short period of time, so short that there has not been time for the velocity to change significantly from v_0, the change in position will be approximately equal to the change in time multiplied by v_0. If the motion of the particle is governed by a differential equation, the value of v_0 will be known as a function of x_0 and y_0. Hence, given x_0 and y_0, the solution at x_1, assumed to be close to x_0, can be calculated as:

  $$y_1 = y_0 + (x_1 - x_0)\, v_0$$

  which can be found from known values only of x_0, x_1 and y_0. Assuming that v_1, found using the differential equation from the values x_1 and y_1, is sufficiently accurate, a second step can be taken to find y2, an approximate solution at x_2, using the formula

  $$y_2 = y_1 + (x_2 - x_1)\, v_1$$

  A sequence of approximations y_1, y_2, y_3, . . . to the solution of the differential equation at x_1, x_2, x_3, . . . is intended to lead eventually to acceptable approximations, at increasingly distant times from where the initial data was given.

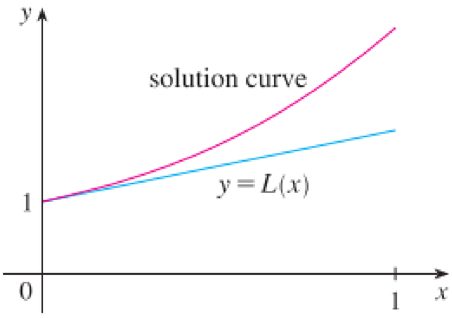
  [BG08]

- **An example for Euler's Method**

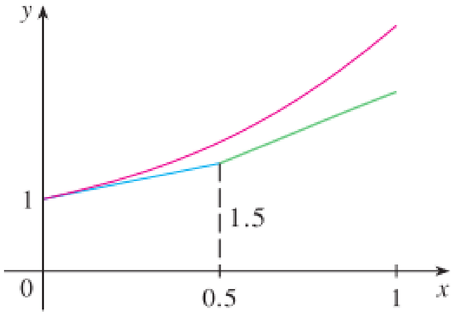
  Suppose we are asked to sketch the graph of the solution of the initial-value problem

  $$y^{'} = x + y \qquad y\,(0) = 1$$

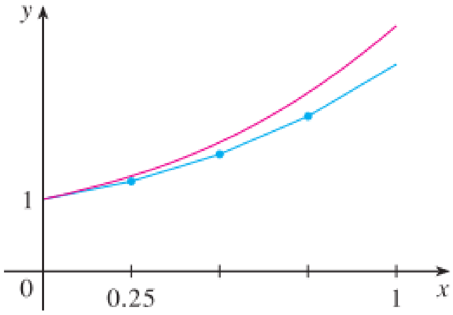
  The differential equation tells us that $y'\,(0) = 0 + 1 = 1$, so the solution curve has slope 1 at the point $(0, 1)$. As a first approximation to the solution we could use the linear approximation $L\,(x) = x + 1$. In other words, we could use the tangent line at $(0, 1)$ as a rough approximation to the solution curve



Euler's idea was to improve on this approximation by proceeding only a short distance along this tangent line and then making a midcourse correction by changing direction as indicated by the direction field. Figure 12 shows what happens if we start out along the tangent line but stop when $x = 0,5$ (This horizontal distance traveled is called the step size.)



In general, Euler's method says to start at the point given by the initial value and proceed in the direction indicated by the direction field. Stop after a short time, look at the slope at the new location, and proceed in that direction. Keep stopping and changing direction according to the direction field. Euler's method does not produce the exact solution to an initial-value problem—it gives approximations. But by decreasing the step size (and therefore increasing the number of midcourse corrections), we obtain successively better approximations to the exact solution



Approximate values for the solution of the initial-value problem $y' = F(x, y)$, $y(x_0) = y_0$ with the step size h, at $x_n = x_{n+1} + h$
[SCW20]

$$y_n = y_{n+1} + h, \quad \text{with: } n = 1, 2, 3, ...$$

- **Doing Explicit Euler's Method step by step**

  For an example we have the formular:

$$y' = y, \quad y(0) = 1$$

with the step size $\Delta h = 1$ and the formular $f(x_n, y_n) = x + y + 1$

and we would like to use this method to approximate y_4.

The Explicit Euler's Method formular is: $y_{n+1} = y_n + f(x_n, y_n)\Delta h$

Step 1: we solve the equation $f(x_n, y_n)$

$$f(x_n, y_n) = f(0, 1) = 0 + 1 + 1 = 2$$

$f(x_n, y_n) = 2$ is the slope of this equation, reminding that the slope of the equation is the change in y divded by the change in x or we have $\frac{\Delta y}{\Delta x}$

Step 2: Multiply to step size Δh

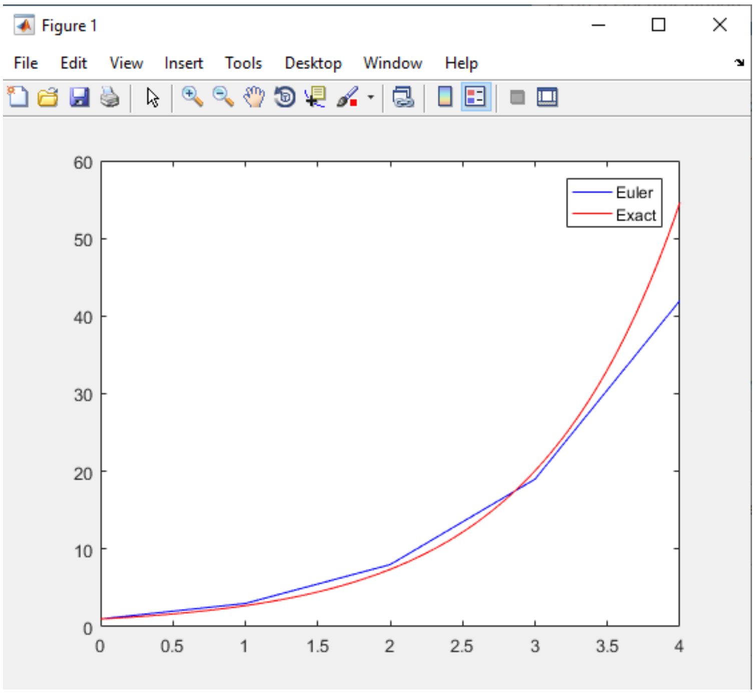
$$f(x_n, y_n)\Delta h = 2.1 = 2$$

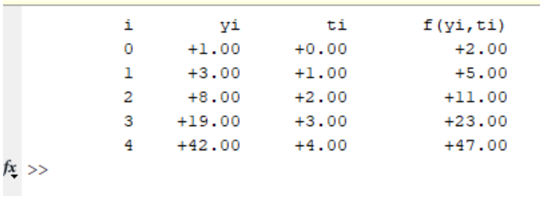
Step 3: add the y_0 after multiplying the step size with the function f then we get the new $y1$ value

$$y_1 = y_0 + f(0, 1)\Delta h = 1 + 2.1 = 3$$

Step 4: Do the same thing with y_2, y_3, y_4

$$y_2 = y_1 + f(1, 3)\Delta h = 3 + 5 = 8$$
$$y_3 = y_2 + f(2, 8)\Delta h = 11 + 8 = 19$$
$$y_4 = y_3 + f(3, 19)\Delta h = 19 + 23 = 42$$





```
i       yi        ti      f(yi,ti)
0    +1.00     +0.00        +2.00
1    +3.00     +1.00        +5.00
2    +8.00     +2.00       +11.00
3   +19.00     +3.00       +23.00
4   +42.00     +4.00       +47.00
>>
```

- **What is Explicit Runge - Kutta Method**

  The Fourth Order-Runge Kutta Method is used to approximate the solution to a first order differential equation, for example:

$$\frac{y(t)}{dt} = y'(t) = f(y(t), y), \ \text{with } y_{t0} = y_0$$

$$\begin{aligned} k_1 &= f(y(t_0), t_0) \\ k_2 &= f(y(t_0) + k_1\tfrac{h}{2}, t_0 + \tfrac{h}{2}) \\ k_3 &= f(y(t_0) + k_2\tfrac{h}{2}, t_0 + \tfrac{h}{2}) \\ k_4 &= f(y(t_0) + k_3 h, t_0 + h) \end{aligned}$$

  Each of these slope estimates can be described verbally.

  - k_1 is the slope at the beginning of the time step (this is the same as k_1 in the first and second order methods).
  - If we use the slope k_1 to step halfway through the time step, then k_2 is an estimate of the slope at the midpoint. This is the same as the slope, k_2, from the second order midpoint method. This slope proved to be more accurate than k1 for making new approximations for y(t).
  - If we use the slope k_2 to step halfway through the time step, then k_3 is another estimate of the slope at the midpoint.
  - Finally, we use the slope, k_3, to step all the way across the time step $(t_0 t_0 + h)$, and k_4 is an estimate of the slope at the endpoint.

$$\begin{aligned} y_{t_0+h} = y_{t_0} + \tfrac{k_1+2k_2+2k_3+k_4}{6}h = y(t_0) + (\tfrac{1}{6}k_1 + \tfrac{1}{3}k_2 + \tfrac{1}{3}k_3 + \tfrac{1}{6}k_4)h \\ = y(t_0) + mh \ \text{(m is a weighted average slope approximation)} \end{aligned}$$

2.5 Problem E

Using Explicit Euler and Explicit Runge–Kutta, give approximate values of the exact solutions of the above examples at time t0, t0 +h, t0 +2h, ..., t0 +5h with optional h.

Example 1:

Solve the differential equation

$$\frac{dy}{dt} = \frac{t^2}{y^2}, \ with \ y(t_0) = 2, \ h = 0.01, \ t0 = 0 \tag{15}$$

Step 1: Solve the equation $f(t_n, y_n)$

$$\begin{aligned} \tfrac{dy}{dy} = \tfrac{t^2}{y^2} => f(t_n, y_n) = \tfrac{t^2}{y^2} \\ f(t_n, y_n) = \tfrac{t^2}{y^2} = \tfrac{0^2}{2} = 0 \end{aligned}$$

Step 2: Multiply $f\left(t_n, y_n\right)$ with the step size

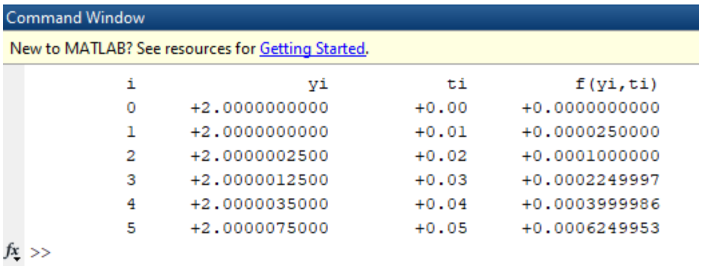
$$f(t_n, y_n) \times \Delta h = 0 \times 0.01 = 0 \tag{16}$$

Step 3: Adding the y_0 after multiplying the step size with the function f then we get the new y value

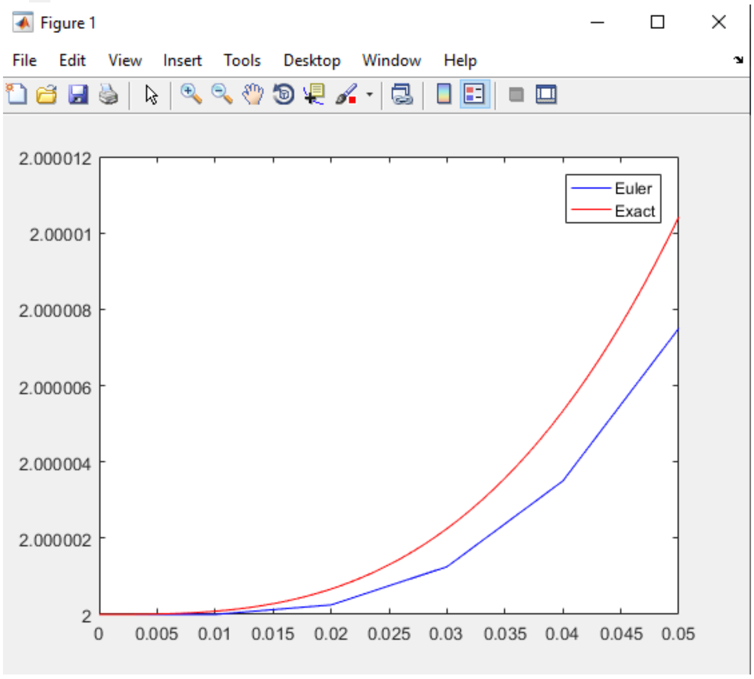
$$y_1 = f\left(t_n, y_n\right) \times \Delta h + y_0 = (0 \times 0.01) + 2 = 2$$

(17)

Step 4: Repeat Step 1 to step 3 until we reach $t_0 = t_0 + 5$



```
        i                  yi          ti            f(yi,ti)
        0        +2.0000000000       +0.00       +0.0000000000
        1        +2.0000000000       +0.01       +0.0000250000
        2        +2.0000002500       +0.02       +0.0001000000
        3        +2.0000012500       +0.03       +0.0002249997
        4        +2.0000035000       +0.04       +0.0003999986
        5        +2.0000075000       +0.05       +0.0006249953
fx >>
```



As we can see form the graph, red line is label for Exact Solution of the ODE and blue line is label for the Approximate solution that using Euler method to calculate, we can easily see that The trend of the blue line, the farther back to the values from the initial value, the higher the error will be compared to the exact value. But the error between Exact and Euler at each specific step is quite large when we approach to the higher value of "h"

The MATLAB code and explaination for this implementation: euler- ex1.m

Runge-Kutta order 4 algorithms for Example 1

We have 6 step:

Step 1: Calculate $K1$

$$K1 = f(t_0, y_0) = \frac{{t_0}^2}{{y_0}^2} = 0$$

Step 2: Calculate $K2$

$$K2 = f(t_0 + \frac{h}{2}, y_0 + \frac{K1}{2} * h) = \frac{0+\frac{0.01}{2}}{2+\frac{K2*h}{2}} = 6.25 \times 10^{-8}$$

Step 3: Calculate $K3$

$$K3 = f(t_0 + \frac{h}{2}, y_0 + \frac{K2}{2} * h) = \frac{0+\frac{0.01}{2}}{2+\frac{K1*h}{2}} = 6.249999805 \times 10^{-6}$$

Step 4: Calculate $K4$

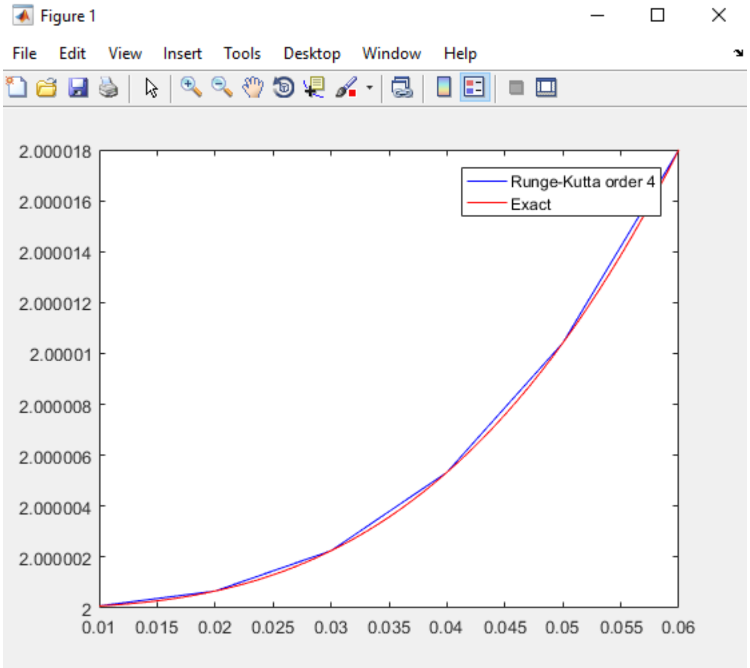
$$K4 = f(t_0 + h, y_0 + K3 * h) = 2.4999999844 \times 10^{-5}$$

Step 5: Calculate the next value of y

We use the equation $y_{i+1} = y_i + (K1 + 2 \times K2 + 2 \times K3 + K4) \times \frac{h}{6}$

$$y_1 = y_0 + (K1 + 2 \times K2 + 2 \times K3 + K4) \times \frac{h}{6} = 2,000000083$$

Step 6: Calutlate y_i and repeat until we reach $t = t + 5h$



```
i              k1i              k2i              k3i              k4i          y(i+1)
0   +0.000000000000  +0.000006250000  +0.000006250000   0.000024999998  2.000000083333
1   +0.000024999998  +0.000056249988  +0.000056249979   0.000099999935  2.000000666666
2   +0.000099999933  +0.000156249818  +0.000156249774   0.000224999498  2.000002249997
3   +0.000224999494  +0.000306248966  +0.000306248842   0.000399997875  2.000005333319
4   +0.000399997867  +0.000506246288  +0.000506246019   0.000624993503  2.000010416612
5   +0.000624993490  +0.000756239759  +0.000756239263   0.000899983819  2.000017999838
```

The MATLAB file already attached in to latex

As we can see from the graph, the explicit Runge-Kutta Order 4 has the less diffrent between value of each line at a specific step The blue line which is labelled for RK4 is apporoximate the red line which is labelled for Exact Solution. We can conclude that the effiency of RK4 is more than explicit Euler

Example 2:

Solve system of differential equations:

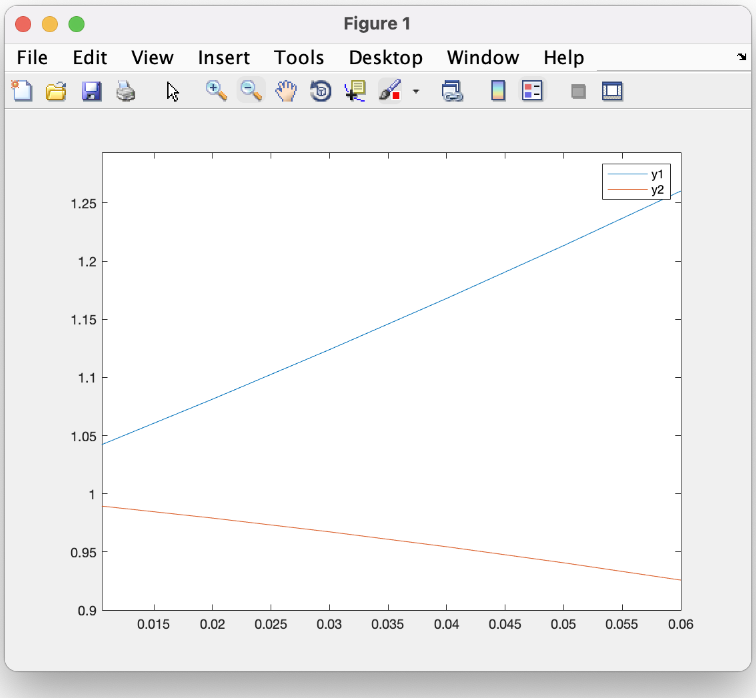
$$\begin{cases} \dfrac{dy_1}{dx} = 4y_1 + y_2 - e^{2x} \\ \dfrac{dy_2}{dx} = -2y_1 + y_2 \end{cases} \tag{18}$$

with $y_1(0) = 1$ and $y_2(0) = 1$

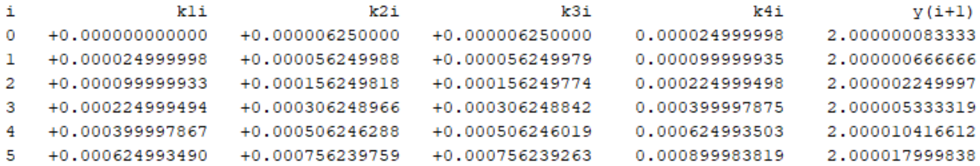
We will do step by step as we did in Example 1

Using Explicit Euler's Method to solve (Let $h = 0.01$)

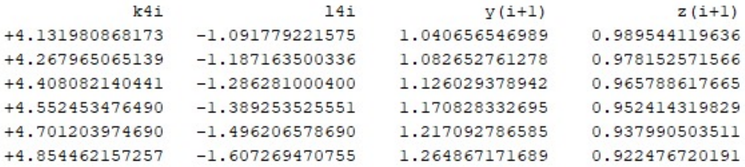
```
i        x_i      y1_i      y2_i
0       0.00      1.00      1.00
1       0.01      1.04      0.99
2       0.02      1.08      0.98
3       0.03      1.12      0.97
4       0.04      1.17      0.95
5       0.05      1.21      0.94
```



Using Runge-Kutta order 4 to solve



```
i               k1i              k2i              k3i              k4i            y(i+1)
0   +0.000000000000  +0.000006250000  +0.000006250000   0.000024999998   2.000000083333
1   +0.000024999998  +0.000056249988  +0.000056249979   0.000099999935   2.000000666666
2   +0.000099999933  +0.000156249818  +0.000156249774   0.000224999498   2.000002249997
3   +0.000224999494  +0.000306248966  +0.000306248842   0.000399997875   2.000005333319
4   +0.000399997867  +0.000506246288  +0.000506246019   0.000624993503   2.000010416612
5   +0.000624993490  +0.000756239759  +0.000756239263   0.000899983819   2.000017999838
```



```
            k4i              l4i           y(i+1)           z(i+1)
+4.131980868173  -1.091779221575   1.040656546989   0.989544119636
+4.267965065139  -1.187163500336   1.082652761278   0.978152571566
+4.408082140441  -1.286281000400   1.126029378942   0.965788617665
+4.552453476490  -1.389253525551   1.170828332695   0.952414319829
+4.701203974690  -1.496206578690   1.217092786585   0.937990503511
+4.854462157257  -1.607269470755   1.264867171689   0.922476720191
```

3 Application Section

3.1 Exercise 2

3.1.1 Problem A

Restate the model for the CO_2 concentration in greenhouses in detail.

3.1.1.a Greenhouse and CO_2 fluxes system

In this section, we will analyze deeper into the application of dynamic system involving the greenhouses model - a method that mostly used in agriculture. Regarding the beneficial reasons of greenhouses, one can not only protects the crop against pests, insects and extreme climate conditions, but also enables an ariel envinronment that independent from that outside atmosphere (Vanthdoor et al., 2011).

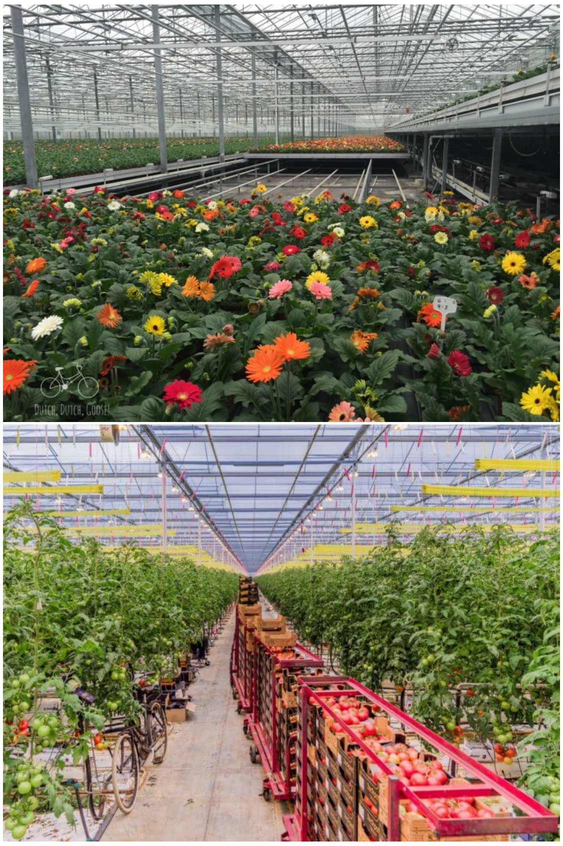


Figure 4: Greenhouses in Pijnacker (above) and Kwintsheul (below), Canda

"In high-technology greenhouses, one can adapt the light level, temperature, CO2 concentration and relative humidity to the needs of the crop." [Van11]. The main climatic factors contributing the greenhouse system's mechanism include temperature, vapor pressure of water and especially CO_2 - an element that directly affect the crop yield.

In this section, we will take CO_2 concentration into account as a vital factor to describe and analyze its rate of change in the greenhouse. The model was based on the following assumptions:

1. There are no spatial difference in temperature, vapor pressure and CO_2 concentration; therefore, all the model fluxes will describe per square meter m^2.
2. The model will be equipped with a thermal screen[1] which divides the greenhouse into two compartments: above and below the screen. [Van11].
3. The greenhouse parameters such as dimension, thermal screen parameters and related equipment paramaters imitate a certain greenhouse design in Netherlands and are constants.
4. The temperature and other related environment parameters are constants and based on the environmnet data collection of team AiCU on Autonomous Greenhouse Challenge 2018.
5. Respiration rate is assumed to be 10% of the photothesis rate.
6. The plant under investigation is cucumber and its properties' parameters are constants.
7. The Photosynthetically Active Radiation do not take the light reflection factor and light absorption of greenhouse objects.

The upper part is often narrower than the lower one which results in different concentrations of CO_2. The main entrance of the system for the flow CO_2 to go through is in the lower compartment. In addition to this, not only the amount of CO_2 brought from natural airflow but also from different sources, which are listed below:

- pad system
- direct air heaters
- third party

From the lower compartment, a portion of CO_2 will be lost due to the fluxes and CO_2 exchange mechanism:

- flow transferred to upper part
- exit the greenhouse through the fan system
- CO_2 exchange of plants for photosynthesis and respiration.

The upper compartment's flux is less complicated, in which amount of CO_2 is mainly received from the exchange with the lower compartment and will release to outside through the roof ventilation (if any). A schematic of the CO_2 concentration cycle that is remade by our group is shown in the Figure 5.

[1] Thermal screens are made of many different materials such as metal or elastic-plastic. They are used to protect crops from damage caused by direct sunlight as well as from freezing in winter.

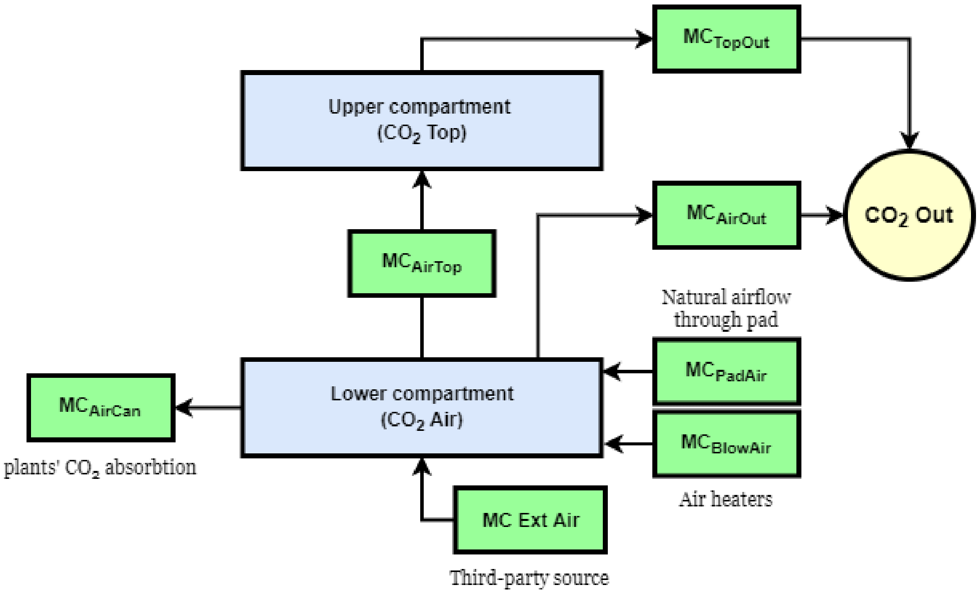


Figure 5: Main CO_2 fluxes in the system

From the figure 5, there will be two differential equations represent the fluctuation of CO_2 concentration in the lower and upper compartment of the greenhouse:

$$\begin{cases} cap_{CO_{2Air}}\dot{CO}_{Air} = MC_{BlowAir} + MC_{ExtAir} + MC_{PadAir} - MC_{AirCan} - MC_{AirTop} - MC_{AirOut} \\ cap_{CO_{2Top}}\dot{CO}_{Top} = MC_{AirTop} - MC_{TopOut} \end{cases}$$

There are some notations on these equations to understand the system more easily:

- A,B: source of CO_2 flow
- cap_A: the capacity to store CO_2 in A (m)
- CO_{2A}: the CO_2 concentration in A (mg m^{-3})
- $\dot{CO}_{2A}$: the rate of change of CO_2 concentration in A (mg m^{-3} s^{-1})
- MC_{AB}: the net mass CO_2-flux density from A to B (mg m^{-2} s^{-1})

From the system of ODEs, we will specifically break them down into different **components** including the system parameters. Our step to implement the system in terms of theory and coding listed below:

1. Investigate the meaning of each equation
2. Investigate the needed variables in each equation
3. Combine all functions into the main system of ODEs
4. Verify the written functions and ODEs.

In this Exercise 2, we will analyze the functions only and such specific values will be **discussed later** on the Exercise 3 (where we assume the reasonable values assigned for each input parameters)

3.1.1.b BlowAir Flux - Air Heater to Greenhouse Lower Compartment

The amount of CO_2 going from the heater into the greenhouse air as follows:

$$MC_{BlowAir} = \frac{\eta_{HeatCO_2} U_{Blow} P_{Blow}}{A_{Flr}} \tag{19}$$

where
η_{HeatCO_2}: amount of CO_2 generated when 1 Joule of sensible heat is generated by the heater (mg CO_2 J^{-1})
U_{Blow}: control the amount of CO_2 generated by the heater (dimensionless, [0,1])
A_{Flr}: area of the greenhouse (m^2)

3.1.1.c ExtAir Flux - Third Party Source to Greenhouse Lower Compartment

The amount of CO_2 pumped by external CO_2-source into the greenhouse air as follows:

$$MC_{ExtAir} = \frac{U_{ExtCO_2} \phi_{ExtCO_2}}{A_{Flr}} \tag{20}$$

where
U_{ExtCO_2}: adjust the rate at which the gas is injected into the greenhouse (dimensionless, [0,1])
ϕ_{ExtCO_2}: ability of third party source to pump CO_2 (mg s^{-1})

3.1.1.d PadAir Flux - Pad System to Greenhouse Lower Compartment

Through the pad system, the amount of CO_2 can enters the greenhouse due to the difference in the concentration of CO_2 inside and outside. In addition to this, the ability of the pad system can be adjusted to let in more air. The formula is described as follows:

$$MC_{PadAir} = f_{Pad}(CO_{2Out} - CO_{2Air}) = \frac{U_{Pad}\phi_{Pad}}{A_{Flr}}(CO_{2Out} - CO_{2Air}) \tag{21}$$

where
f_{Pad}: a product of $\frac{U_{Pad}\phi_{Pad}}{A_{Flr}}$ (m s^{-1})
U_{Pad}: permability of the pad (dimensionless, [0,1])
ϕ_{Pad}: ability for the airflow to pass through $(m^3\ s^{-1})$

3.1.1.e AirTop Flux - Greenhouse Lower Compartment to Upper Compartment

The net flux of CO2 from the lower compartment to the upper compartment of the greenhouse is more complicated and it depends on the difference in temperature and air density between the two compartments.

$$MC_{AirTop} = f_{ThScr}(CO_{2Air} - CO_{2Top}) \tag{22}$$

$$f_{ThScr} = U_{ThScr} K_{ThScr} |T_{Air} - T_{Top}|^{\frac{2}{3}} + (1 - U_{ThScr})[\frac{g(1 - U_{ThScr})}{2\rho_{Air}^{Mean}} |\rho_{Air} - \rho_{Top}|]^{\frac{1}{2}} \tag{23}$$

where
f_{ThScr}: the airflow rate through the thermal screen (m s^{-1})

The details of how this function is too complicated and the meaning of such parameters can be read more from the provided information in page 5 of assignment [NN20].

3.1.1.f AirOut Flux - Greenhouse Upper Compartment to Outside Space

$$MC_{AirOut} = (f_{VentSide} + f_{VentForced})(CO_{2Air} - CO_{2Out}) \tag{24}$$

where
$f_{VentSide}$: the flux due to the fan system on the sidewalls of the greenhouse (m s^{-1})
$f_{VentForced}$: the flux due to the fan system inside the greenhouse (m s^{-1})

The details of $f_{VentSide}$ and $f_{VentForced}$ functions and their conditions can be read from equations (10), (11), (12), (13), (14) in page 7 of the assignment [NN20].

3.1.1.g TopOut Flux - Greenhouse Upper Compartment to Upper Compartment

$$MC_{TopOut} = f_{VentRoof}(CO_{2Top} - CO_{2Out}) \tag{25}$$

where
$f_{VentRoof}$: the flux due to the fan system inside the greenhouse (m s^{-1})

The details of $f_{VentRoof}$ function and its conditions can be read from equations (16), (17) in page 7 of the assignment [NN20].

3.1.1.h AirCan Flux - Greenhouse Lower Compartment to Leaves

$$MC_{AirCan} = M_{CH_2O}h_{C_{Buf}}(P - R) \tag{26}$$

where
M_{CH_2O}: the molar mass of CH_2O (mg μmol^{-1})
P: photosynthetic rate (μmol CO_2 m^{-2} s^{-1})
R: respiration rate (μmol CO_2 m^{-2} s^{-1})
$h_{C_{Buf}}$: shows the cessation of photosynthesis (dimensionless, [0,1])

The details of photosynthesis aspects containing functions and parameters can be read from the Section 4, from page 8 to page 14 in the assignment [NN20].

3.1.1.i Comment on model

From the main components of the model mentioned above, we will focus more on the rate of change CO_{2Air} and CO_{2Top} to analyze how they affect each other, and their contribution along with other environment factors to the amount of CO_2 concentration throughout the two main compartments of the system.

Because all of the components affecting the systems above will be represented as mathematic polynomials and equations, we can model this system by mathematic computation.

3.1.2 Problem B

Write programs that calculate the net CO_2 flux from one place to an other place using the formulas provided in the assignment. Then write a program that returns the right side of the system's formulas divided by $cap_{CO_{2Air}}$ and $cap_{CO_{2Top}}$ respectively and named this function dx.

To easily implement and calculate the model, we use Python and Jupyter Notebook.

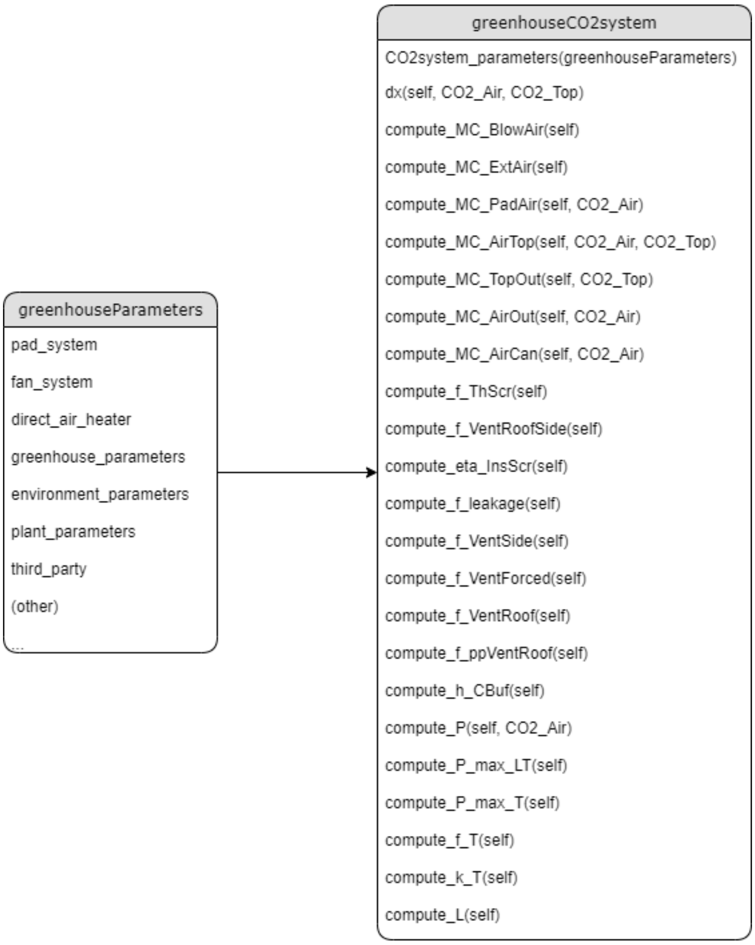


Figure 6: CO_2 concentration model program layout

Figure 6 shows how the program is organized. There are 2 classes is used to represent the CO_2 system

- greenhouseParameters - holds all the parameters constant values of the greenhouse which is used for CO_2 system as well as VP system.
- greenhouseCO2system - holds functions that process system formulas, compute CO_2 fluxes and the rates of change of the CO_2 concentration.

Technically, because there are some formulas such as photosynthesis rate that needs input many parameters, we will not list all the parameters of functions, instead we create the greenhouseParameters class to hold all parameters and pass to the greenhouseCO2system to call neccessary parameters by using $self$ attribute.

In the greenhouseCO2system, CO_2 fluxes are computed using 7 functions *compute_MC_X()*; there are some helpers functions to compute specific parameters used to compute fluxes.

The dx function computes the derivatives of the pair CO_{2Air} and CO_{2Top} by 3 steps: compute CO_2 fluxes; then compute the pair of derivatives based on the system of differential equations by dividing net fluxes by $cap_{CO_{2Air}}$, $cap_{CO_{2Top}}$ and assign the results to variable d_CO2_Air and d_CO2_Top, respectively; finally return the results as the dictionary of 2 elements, $'d\_Air'$ holds d_CO2_Air and $'d\_Top'$ holds holds d_CO2_Top. Figure 7 show the computation algorithm of dx function.

The more complete description and assumption of functions is described in Figure 8.

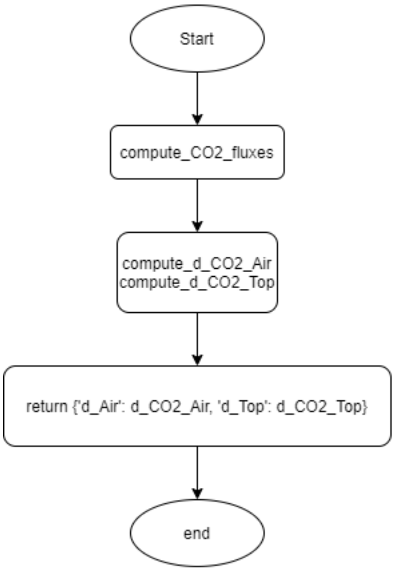
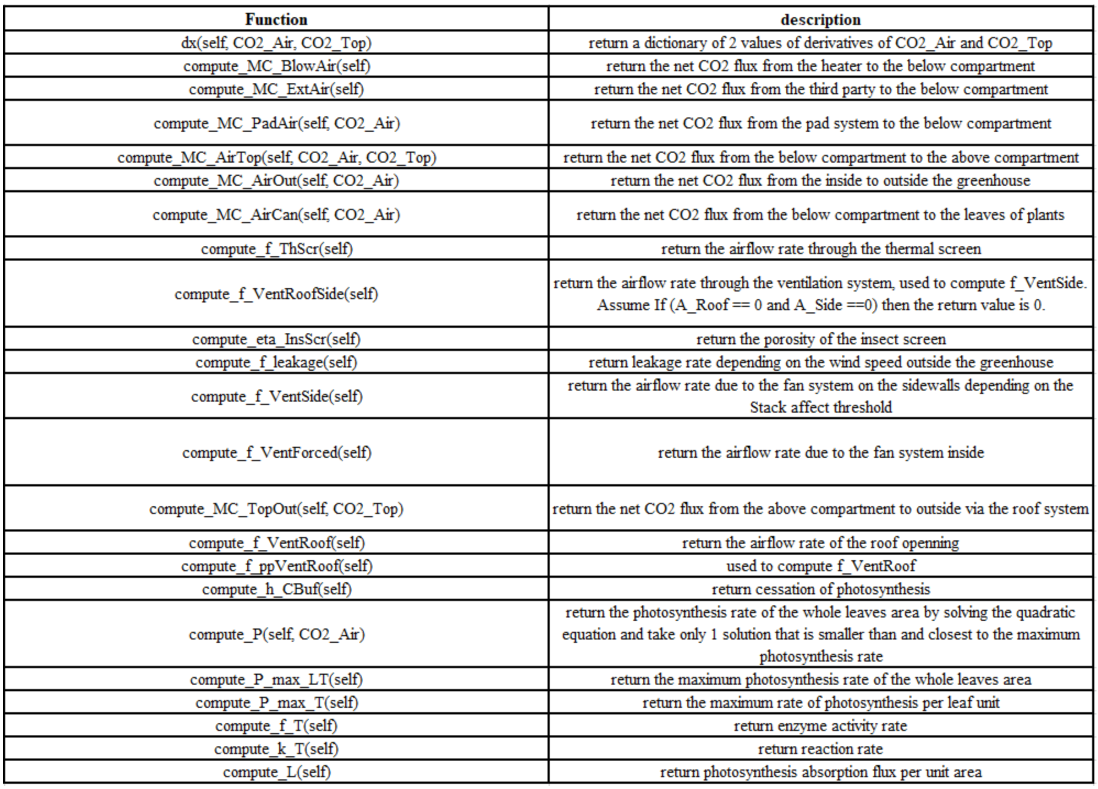


Figure 7: computation of derivatives of the pair CO_{2Air} and CO_{2Top} algorithm



Function	description
dx(self, CO2_Air, CO2_Top)	return a dictionary of 2 values of derivatives of CO2_Air and CO2_Top
compute_MC_BlowAir(self)	return the net CO2 flux from the heater to the below compartment
compute_MC_ExtAir(self)	return the net CO2 flux from the third party to the below compartment
compute_MC_PadAir(self, CO2_Air)	return the net CO2 flux from the pad system to the below compartment
compute_MC_AirTop(self, CO2_Air, CO2_Top)	return the net CO2 flux from the below compartment to the above compartment
compute_MC_AirOut(self, CO2_Air)	return the net CO2 flux from the inside to outside the greenhouse
compute_MC_AirCan(self, CO2_Air)	return the net CO2 flux from the below compartment to the leaves of plants
compute_f_ThScr(self)	return the airflow rate through the thermal screen
compute_f_VentRoofSide(self)	return the airflow rate through the ventilation system, used to compute f_VentSide. Assume If (A_Roof == 0 and A_Side ==0) then the return value is 0.
compute_eta_InsScr(self)	return the porosity of the insect screen
compute_f_leakage(self)	return leakage rate depending on the wind speed outside the greenhouse
compute_f_VentSide(self)	return the airflow rate due to the fan system on the sidewalls depending on the Stack affect threshold
compute_f_VentForced(self)	return the airflow rate due to the fan system inside
compute_MC_TopOut(self, CO2_Top)	return the net CO2 flux from the above compartment to outside via the roof system
compute_f_VentRoof(self)	return the airflow rate of the roof openning
compute_f_ppVentRoof(self)	used to compute f_VentRoof
compute_h_CBuf(self)	return cessation of photosynthesis
compute_P(self, CO2_Air)	return the photosynthesis rate of the whole leaves area by solving the quadratic equation and take only 1 solution that is smaller than and closest to the maximum photosynthesis rate
compute_P_max_LT(self)	return the maximum photosynthesis rate of the whole leaves area
compute_P_max_T(self)	return the maximum rate of photosynthesis per leaf unit
compute_f_T(self)	return enzyme activity rate
compute_k_T(self)	return reaction rate
compute_L(self)	return photosynthesis absorption flux per unit area

Figure 8: The description of functions of the system

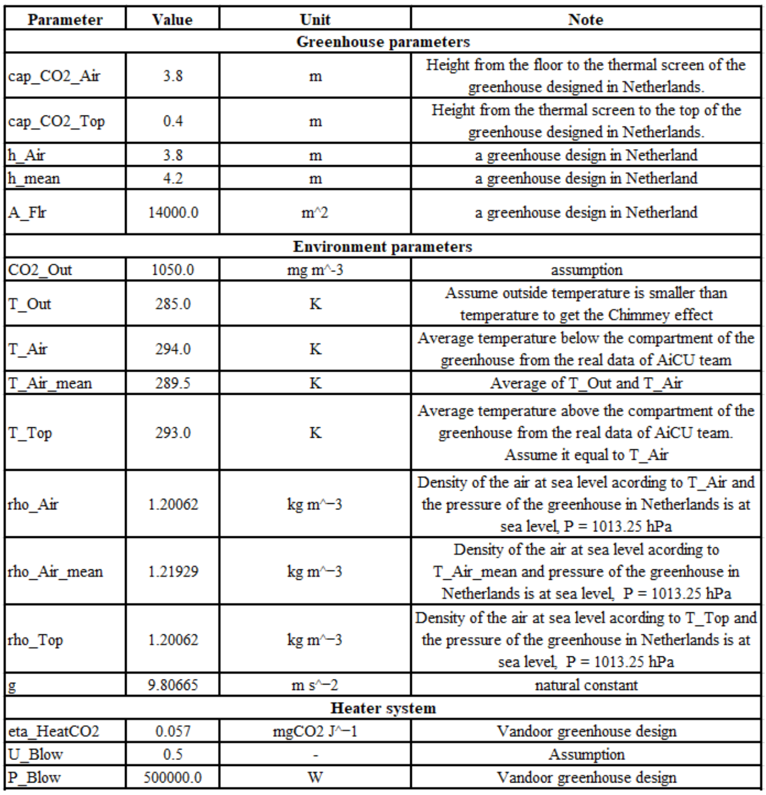
The more detail Python code for this program: greenhouse-system.ipynb

3.2 Exercise 3

Assuming constant temperature difference and constant air density difference, study from [Van11] and related citations to find specific and reasonable values for each input parameter of the function dx including the difference in temperature and in air density except for the variables CO_{2Air} and CO_{2Top}

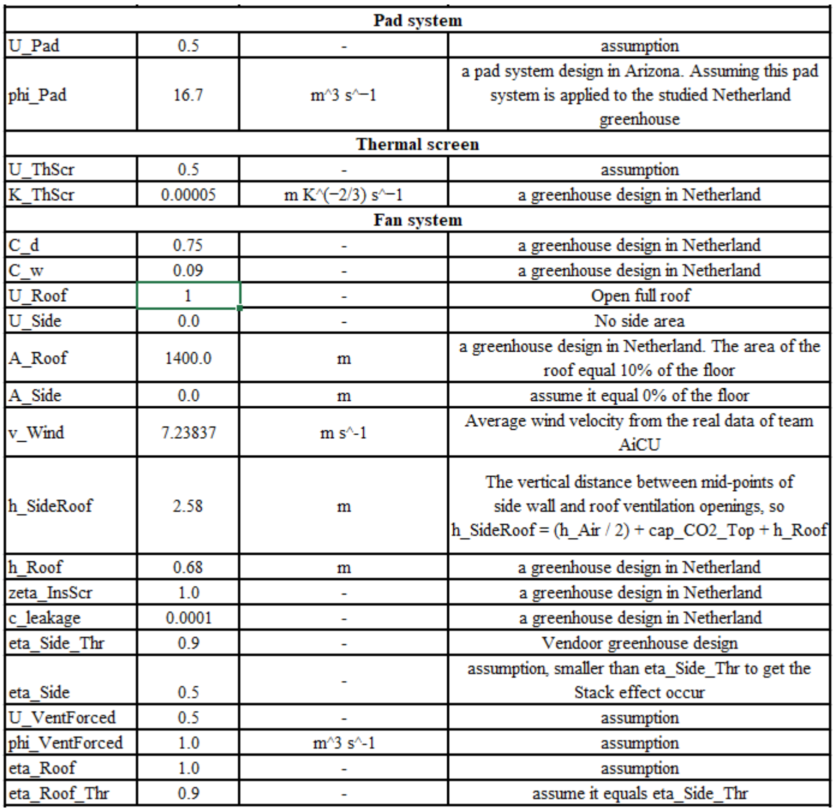
Because of the lack on some greenhouse parameters information on the data of team AiCU, so we assume to use the parameters value of a certain greenhouse in Netherlands and try to modified reasonably and estimate to reduce the error with the real $CO2$ concentration data in order to be useful to predict in the future.

Acording to the assumptions of the CO_2 system we have stated in Exercise 2, team AiCU data, [Van11], assignment description etc. the parameters' values holds as the following table



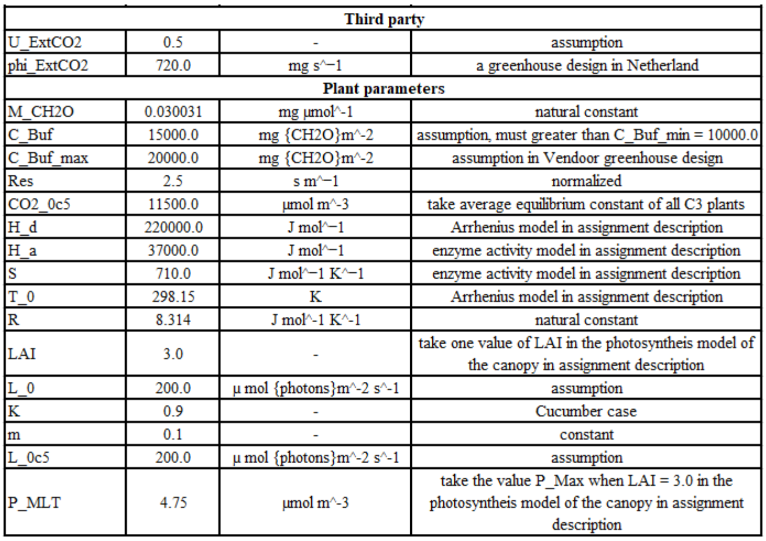
Parameter	Value	Unit	Note
Greenhouse parameters			
cap_CO2_Air	3.8	m	Height from the floor to the thermal screen of the greenhouse designed in Netherlands.
cap_CO2_Top	0.4	m	Height from the thermal screen to the top of the greenhouse designed in Netherlands.
h_Air	3.8	m	a greenhouse design in Netherland
h_mean	4.2	m	a greenhouse design in Netherland
A_Flr	14000.0	m^2	a greenhouse design in Netherland
Environment parameters			
CO2_Out	1050.0	mg m^-3	assumption
T_Out	285.0	K	Assume outside temperature is smaller than temperature to get the Chimmey effect
T_Air	294.0	K	Average temperature below the compartment of the greenhouse from the real data of AiCU team
T_Air_mean	289.5	K	Average of T_Out and T_Air
T_Top	293.0	K	Average temperature above the compartment of the greenhouse from the real data of AiCU team. Assume it equal to T_Air
rho_Air	1.20062	kg m^−3	Density of the air at sea level acording to T_Air and the pressure of the greenhouse in Netherlands is at sea level, P = 1013.25 hPa
rho_Air_mean	1.21929	kg m^−3	Density of the air at sea level acording to T_Air_mean and pressure of the greenhouse in Netherlands is at sea level, P = 1013.25 hPa
rho_Top	1.20062	kg m^−3	Density of the air at sea level acording to T_Top and the pressure of the greenhouse in Netherlands is at sea level, P = 1013.25 hPa
g	9.80665	m s^−2	natural constant
Heater system			
eta_HeatCO2	0.057	mgCO2 J^−1	Vandoor greenhouse design
U_Blow	0.5	-	Assumption
P_Blow	500000.0	W	Vandoor greenhouse design

Figure 9: CO_2 system parameters



Pad system			
U_Pad	0.5	-	assumption
phi_Pad	16.7	m^3 s^−1	a pad system design in Arizona. Assuming this pad system is applied to the studied Netherland greenhouse
Thermal screen			
U_ThScr	0.5	-	assumption
K_ThScr	0.00005	m K^(−2/3) s^−1	a greenhouse design in Netherland
Fan system			
C_d	0.75	-	a greenhouse design in Netherland
C_w	0.09	-	a greenhouse design in Netherland
U_Roof	1	-	Open full roof
U_Side	0.0	-	No side area
A_Roof	1400.0	m	a greenhouse design in Netherland. The area of the roof equal 10% of the floor
A_Side	0.0	m	assume it equal 0% of the floor
v_Wind	7.23837	m s^-1	Average wind velocity from the real data of team AiCU
h_SideRoof	2.58	m	The vertical distance between mid-points of side wall and roof ventilation openings, so h_SideRoof = (h_Air / 2) + cap_CO2_Top + h_Roof
h_Roof	0.68	m	a greenhouse design in Netherland
zeta_InsScr	1.0	-	a greenhouse design in Netherland
c_leakage	0.0001	-	a greenhouse design in Netherland
eta_Side_Thr	0.9	-	Vendoor greenhouse design
eta_Side	0.5	-	assumption, smaller than eta_Side_Thr to get the Stack effect occur
U_VentForced	0.5	-	assumption
phi_VentForced	1.0	m^3 s^-1	assumption
eta_Roof	1.0	-	assumption
eta_Roof_Thr	0.9	-	assume it equals eta_Side_Thr

Figure 10: CO_2 system parameters



Third party			
U_ExtCO2	0.5	-	assumption
phi_ExtCO2	720.0	mg s^−1	a greenhouse design in Netherland
Plant parameters			
M_CH2O	0.030031	mg µmol^-1	natural constant
C_Buf	15000.0	mg {CH2O}m^-2	assumption, must greater than C_Buf_min = 10000.0
C_Buf_max	20000.0	mg {CH2O}m^-2	assumption in Vendoor greenhouse design
Res	2.5	s m^−1	normalized
CO2_0c5	11500.0	µmol m^-3	take average equilibrium constant of all C3 plants
H_d	220000.0	J mol^−1	Arrhenius model in assignment description
H_a	37000.0	J mol^−1	enzyme activity model in assignment description
S	710.0	J mol^−1 K^−1	enzyme activity model in assignment description
T_0	298.15	K	Arrhenius model in assignment description
R	8.314	J mol^-1 K^-1	natural constant
LAI	3.0	-	take one value of LAI in the photosyntheis model of the canopy in assignment description
L_0	200.0	µ mol {photons}m^-2 s^-1	assumption
K	0.9	-	Cucumber case
m	0.1	-	constant
L_0c5	200.0	µ mol {photons}m^-2 s^-1	assumption
P_MLT	4.75	µmol m^-3	take the value P_Max when LAI = 3.0 in the photosyntheis model of the canopy in assignment description

Figure 11: CO_2 system parameters

3.3 Exercise 4

3.3.1 Problem A

Study explicit Euler and Runge–Kutta of order 4 algorithms for solving first-order differ- ential equations. Write two programs performing the two algorithms and named them euler and rk4 respectively with inputs: a callable function as dx, the values at time t of the two variables CO_2, the time step h. These solvers return the approximate values of CO_{2Air} and CO_{2Top} at time t+h.

To facilitate the calculation, we have created two functions $euler_{CO2}()$ and $rk4_{CO2}()$ to calculate each conversion step of CO2 concentration and repeats it many times in main function to create a list n data. We'll explain the parameters below, but let's look at the functions first:

euler_CO2(dx, Air, Top, h): returns a dictionary of the 2 next value of CO2 above and below the screen calculated by using the explicit Euler method.

rk4_CO2(dx, Air, Top, h): returns the next value of CO2 above and below the screen calculated by using the explicit Runge-Kutta Four method.

Main function: With n values, the main function repeat the functions n time **euler_CO2(dx, Air, Top, h)** and **rk4_CO2(dx, Air, Top, h)** then append results into two list $CO2\_euler$ and $CO2\_rk4$ to plot data.

dx(): The function calculating the rate of change with respect to time of two input: $CO2_{Air}$ and $CO2_{Top}$.

Air: Initial value of the CO2 concentration below the screen used to estimate the next value.

Top: Initial value of the CO2 concentration above the screen used to estimate the next value.

h: Step of each calculation.

We have attached the source code to this report, please check for more details.

3.3.2 Problem B

Select specific values of CO_{2Air} and CO_{2Top} at time t from the data set as initial values to run the solvers and find the approximate values of CO_{2Air} and CO_{2Top} in the next 5 minutes, 10 minutes, 20 minutes, ... and calculate the difference of the result from the actual data. Comment on the accuracy of the model and present details in the report..

Having completed calculating, we try to plot the values collected with respect to time (s). We compare the real values collected by measurement to the values estimated with the same initial value to observe the difference between these two. Therefore, we can estimate the accuracy of two methods. The plotting part shows as below:

Dataset description: Measured values were obtained from the greenhouse cucumber dataset of team AiCU in the Autonomous Greenhouse Challenge 2018. The dataset is published at https://github.com/CEAOD/Data. The parameters including AssimLight, BlackScr, CO2air, EnScr, GHtime, HumDef, PipeGrow, PipeLow, RHair, Tair, VentLee, Ventwind were measured every 5 minutes.

Dataset processing: Only the first 288 values, recorded every 5 minutes (measured in 1 day overall) of CO2air (CO2 concentration below the screen) is taken to maintain the uniformity of ambient conditions of each measurement while NaN values were ignored. Besides, we form an average line replace each sets of 5 values with a mean value of them to reduce the confusion of the actual data and observe its trend. The error of estimation is the difference between the actual data and the estimated data at a certain time.

Estimated data description: Two sets of values estimated by Euler and Runge-Kutta method with the step **h = 1s**. The initial condition for CO_{2Air} and CO_{2Top} is the sample 92^{th} in the dataset, convert from ppm to mgm^{-3} we get

$$CO_{2Air\_initial} = CO_{2Top\_initial} = 777.4233127627641(mgm^{-3})$$

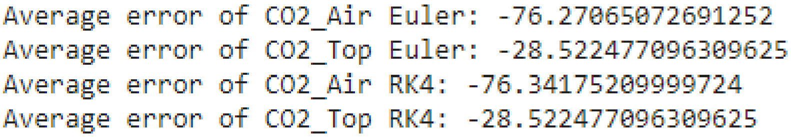
Comment on results

It can be seen from the Figure 14 that the actual data of CO_{2Air} concentration below the greenhouse compartment goes up at the beginning but fluctuates a lot for the last 24 hours. The result of Euler estimation also sees an upward trend at the beginning but reaches a steady state of approximately 1100 (mgm^{-3}) (larger than the CO_{2Out} concentration by 50 (mgm^{-3})) quite soon for about 20 minutes, this trend can be seen from the Figure 13. Moreover, we can see a similar trend which both reach a steady state despite the fact that the difference in values in the first 60 minutes. In Figure 21, the error of Euler estimation variates in the range approximate [-200;200], the trend is similar and depends on the real values, but the longer the algorithms run, the larger and more unstable the error, which is not reliable to predict at least for a day.

Figure 15 and Figure 16 show the upward trend of the CO_{2Top} concetration in the first 10 minutes then reaches the steady state of approximate 1050 (mgm^{-3}) for the last 24 hours, which is approximately equal to the CO_{2Out} concentration outside the greenhouse.

The results of estimation of CO_2 concentration using Runge-Kutta 4 order method is not much different from the case of Euler method in long-run, it also gives a similar trends of value and error, which is shown in Figure 17, 18, 19, 20, 21.

The average error of the first 24 hours of the 2 methods is not much different, only the error of the CO_{2Air} concentration is different for about 0.01 (mgm^{-3})



```
Average error of CO2_Air Euler: -76.27065072691252
Average error of CO2_Top Euler: -28.522477096309625
Average error of CO2_Air RK4: -76.34175209999724
Average error of CO2_Top RK4: -28.522477096309625
```

Figure 12: Average error of CO_2 concentration for the first 24 hours of the 2 methods

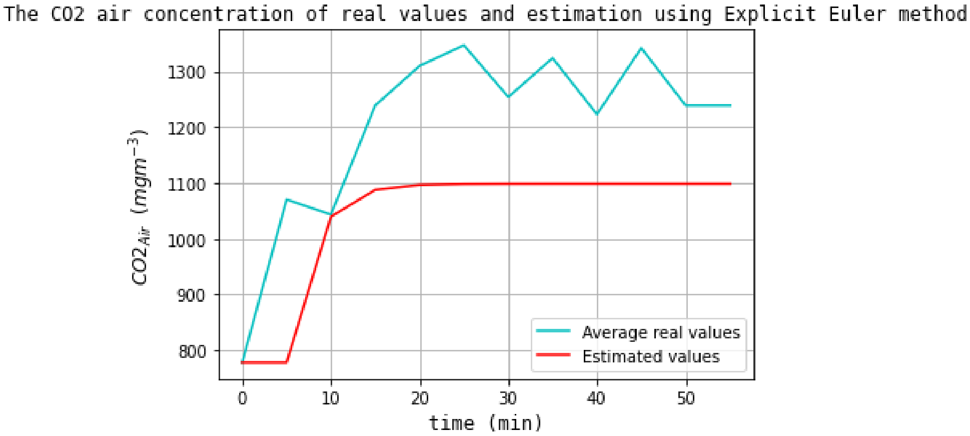


Figure 13: CO_2 concentration below the screen from 0 - 60 minutes using Explicit Euler method

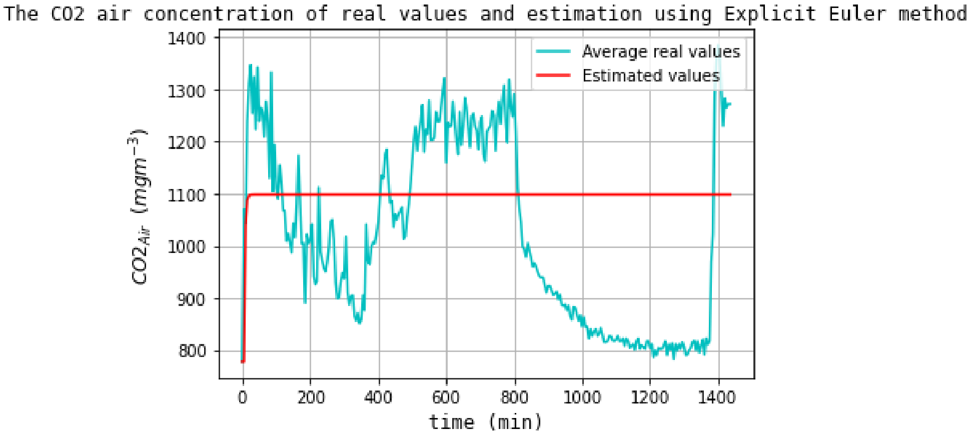


Figure 14: CO_2 concentration below the screen for the first 24 hours using Explicit Euler method

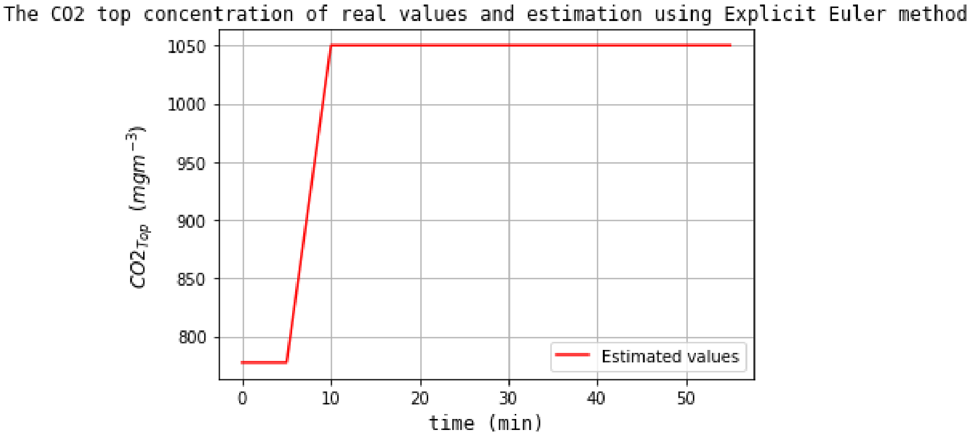


Figure 15: CO_2 concentration above the screen for the first 60 minutes using Explicit Euler method

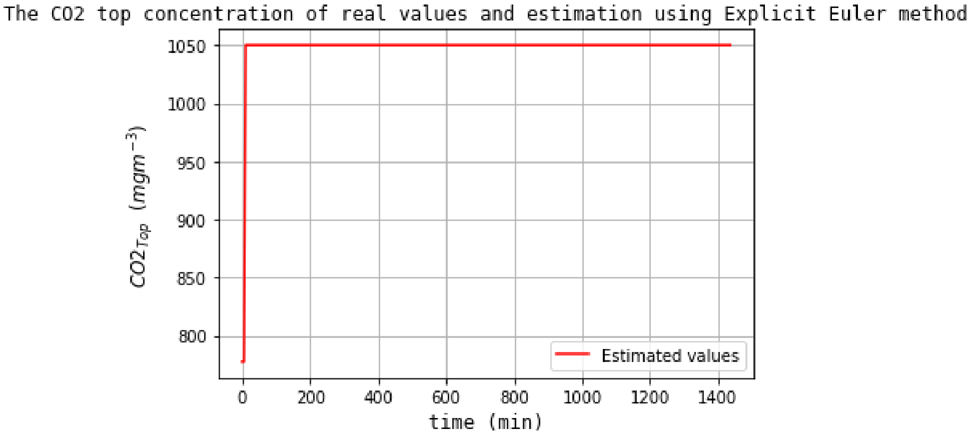


Figure 16: CO_2 concentration above the screen for the first 24 hours using Explicit Euler method

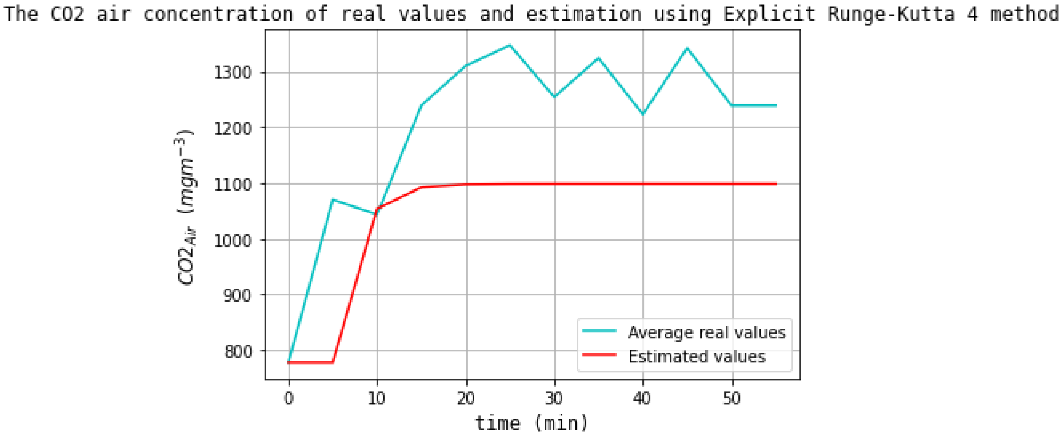


Figure 17: CO_2 concentration below the screen for the first 60 minutes using Runge-Kutta 4 order method

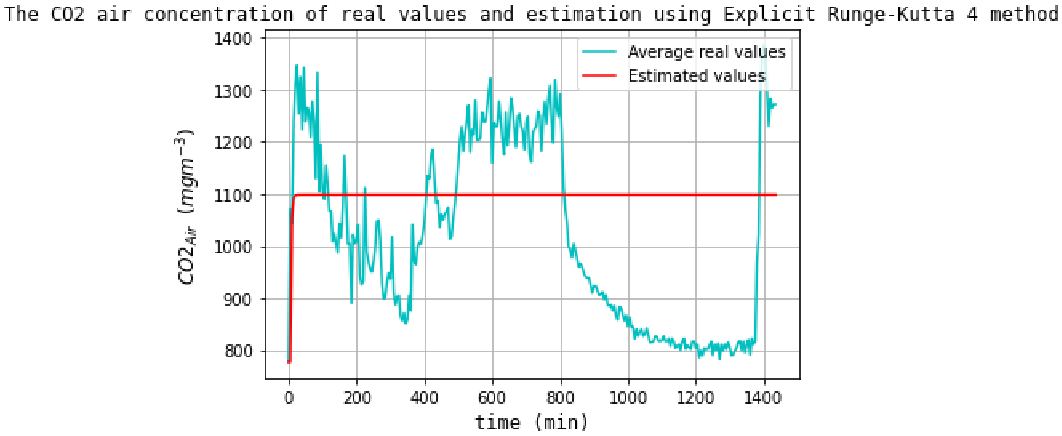


Figure 18: CO_2 concentration below the screen for the first 24 hours using Runge-Kutta 4 order method

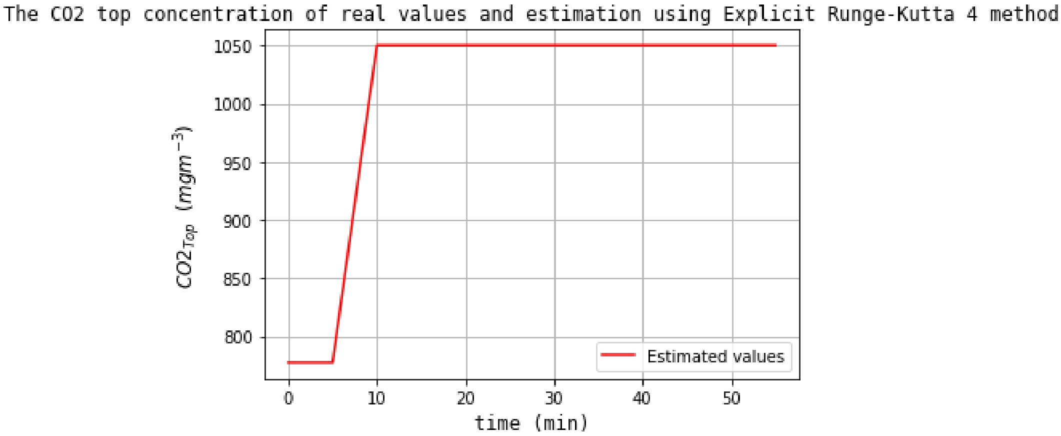


Figure 19: CO_2 concentration above the screen for the first 60 minutes using Runge-Kutta 4 order method

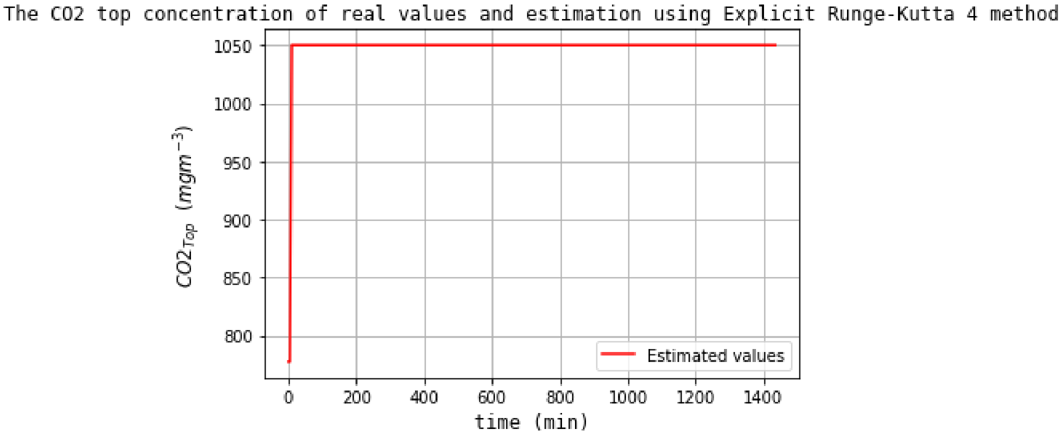


Figure 20: CO_2 concentration above the screen for the first 24 hours using Runge-Kutta 4 order method

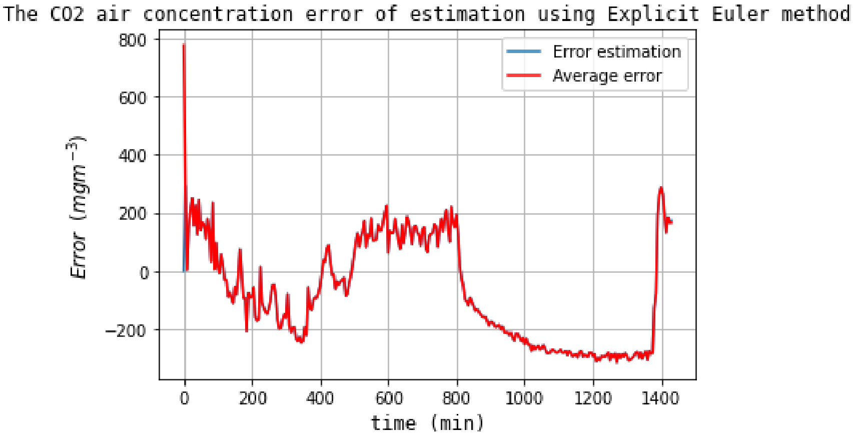


Figure 21: Error of CO_2 concentration estimation using Runge-Kutta 4 order method

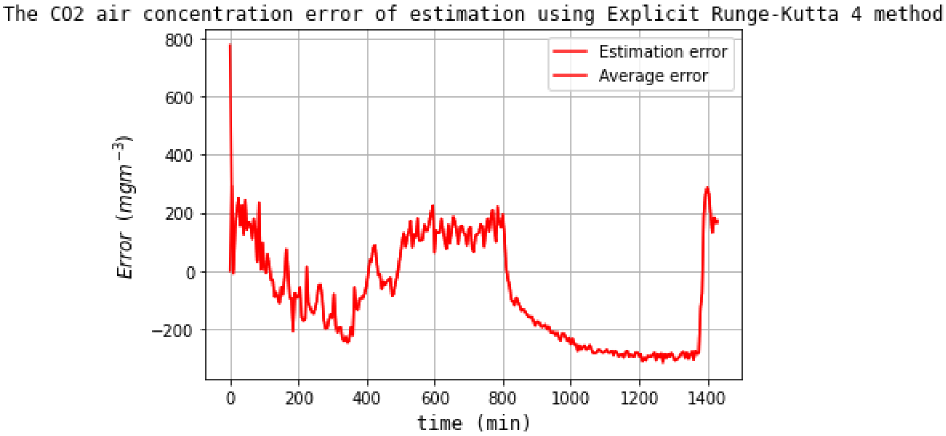


Figure 22: Error of CO_2 concentration estimation using Runge-Kutta 4 order method

3.4 Exercise 5

Do the same thing as in Ex. 2 - Ex. 4 for the vapor pressure VPAir and VPTop as described in chapters 2 and 8 in [Van11]. Present details in the report

3.4.1 Problem A

Restate the model for the VP_{Air} and VP_{Top} which stands for Vapor Pressure in lower and upper compartments of greenhouses in detail.

3.4.1.a Vapor pressure fluxes system

In this section, we will take VP which is vapor pressure into account as a vital factor to describe and analyze its rate of change in the greenhouse. The assumptions that were used in this part is similar to those mentioned from the CO_2 concentration model in problem A, Exercise 2.

As usual, the lower compartment is the main entrance of the flow of vapor pressure. However, the system of vapor pressure fluxes is more complicated than the CO_2 concentration's since there are more connections among different parts of the greenhouse. The sources of VP that exchange with the lower compartment are:

- canopy (plants)
- outlet air of the pad
- fogging system
- direct air heaters

From the lower compartment, a portion of vapor pressure will be lost due to the fluxes which are:

- flow transferred to the upper part
- exchanged with the outdoor air
- exit the greenhouse caused by the pad and fan system
- exit the greenhouse caused by the mechanical cooling sytem

The upper compartment's flux is less complicated, in which the amount of vapor pressure is mainly received from the exchange with the lower compartment and will release to outside or to internal cover layer. A schematic of the vapor pressure cycle that is remade by our group is shown in the Figure 18.

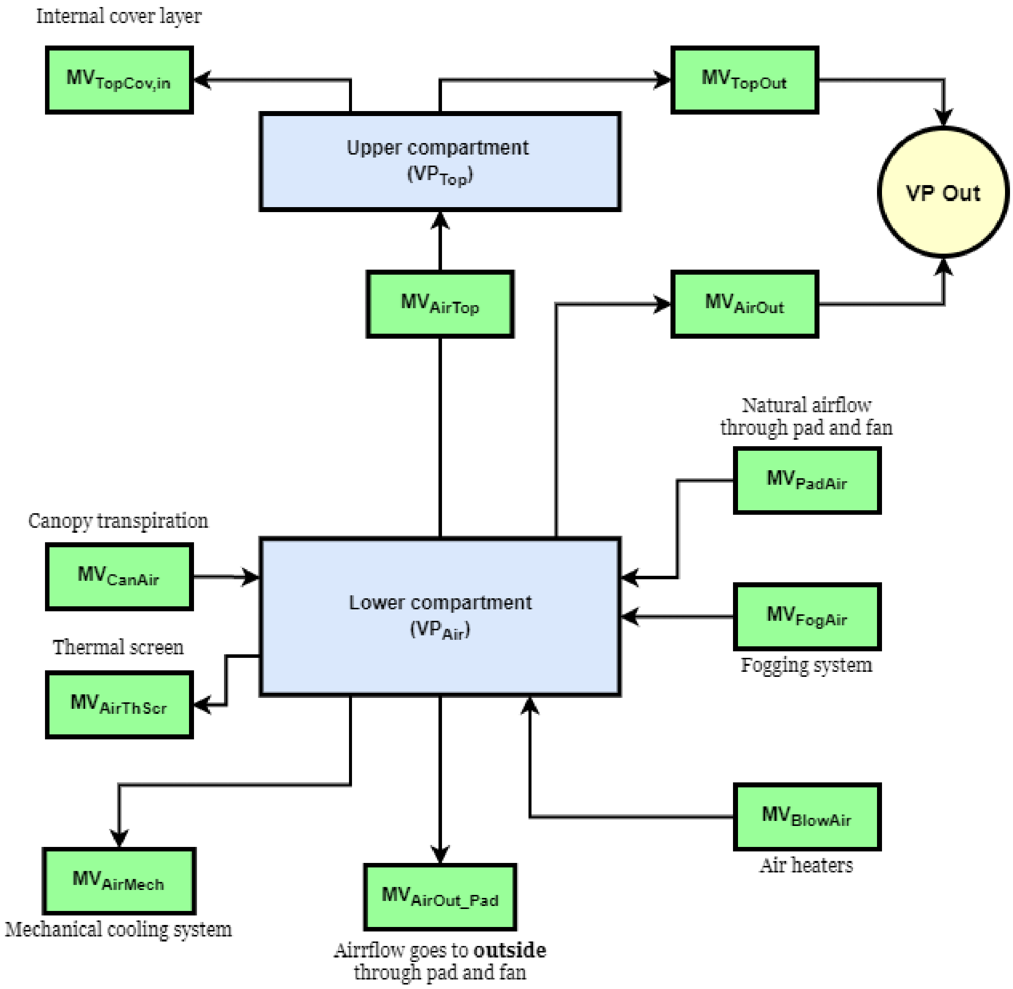


Figure 23: Main CO_2 fluxes in the system

From the figure 18, there will be two differential equations the fluctuation of vapor pressure in the lower and upper compartment of the greenhouse:

$$\begin{cases} cap_{VP_{Air}} \dot{VP}_{Air} = MV_{AirCan} + MV_{PadAir} + MVFogAir + MV_{BlowAir} \\ \qquad\qquad - MV_{AirThScr} - MV_{AirTop} - MV_{AirOut} - MV_{AirOut\_Pad} - MV_{AirMech} \\ cap_{VP_{Top}} \dot{VP}_{Top} = MV_{AirTop} - MV_{TopCov,in} - MV_{TopOut} \end{cases} \tag{27}$$

There are some notations on these equations to understand the system more easily:

- A,B: source of vapor pressure flow
- cap_A: the capacity to store vapor pressure in A (kg m J^{-1})
- VP_A: the vapor pressure in A (Pa)

- MV_{AB}: the net mass vapor flux from A to B (kg m^{-2} s^{-1})
- $V\dot{P}_A$: the rate of change of vapor pressure in A (Pa s^{-1}) (we take the unit of the RHS of the equation divide to the unit of cap_A, i.e $\frac{kg \cdot m^{-2} \cdot s^{-1}}{kg \cdot m \cdot J^{-1}}$ = J m^{-3} s^{-1} = Pa s^{-1}

From the system of ODEs, we will specifically break them down into different **components** including the system parameters. Our step to implement the system in terms of theory and coding listed will be similar to the steps we took in the section CO_2 concentration model.

The only difference in this exercise 5 is we will analyze the rate of change of vapor pressure and its new parameters and more complicated scheme. Besides, the specific values will be **discussed later** in the **next Problem B** (where we assume the reasonable values assigned for each input parameters). Before move on to the equations analysis, we will discuss about some general vapor pressure equations.

Please take note that there are some values on the CO_2 model that will be reused in this section, so we will not describe their notations gain.

3.4.1.b General equation for the amount of vapor pressure

The amount of vapor pressure except for (VP_{Air} and VP_{Top}) can be calculated generally as follow:

$$VP_A = 610.78 \times e^{\frac{T_A+273.15}{(T_A+273.15+238.3)\times 17.2694}} \tag{28}$$

where T_A: the temperature in Kelvin (K)

3.4.1.c General equation for the capicity of vapor pressure

The capacity of vapor pressure can be calculated generally as follow:

$$cap_{VP_A} = \frac{M_{Water} h_A}{RT_A} \tag{29}$$

where
h_A: the height of the A compartment (m)
T_A: Temperature of the A compartment (K)

3.4.1.d Vapor pressure flux from Location to Object

The vapour exchange coefficient between **the air** and **an object** is linearly related to the convective heat exchange coefficient between the air and the object [Van11]. Therefore, the vapour flux from the air to an object by condensation is described by:

$$MV_{AB} = \begin{cases} 0 & VP_A < VP_B \\ 6.4 \cdot 10^{-9} HEC_{12}(VP_A - VPB) & VP_A > VP_B \end{cases} \qquad [kg \cdot m^{-2} s^{-1}] \tag{30}$$

The notations of the above equation are listed below:

- MV_{12}: the vapor flux from air of location A to object B (kg m^{-2} s^{-1})
- $6.4 \cdot 10^{-9}$: the conversion factor relating to HEC (W m^{-2} K^{-1})
- HEC_{12}: the heat exchange coefficient between the air of location A to B (W m^{-2} K^{-1})

- VP_A, VP_B: the saturated vapor pressure at A and B, respectively (Pa)

Because the model should consist of only differentiable equations, equation (29) was smoothed using a differentiable 'switch function' to yield:

$$MV_{12} = \frac{1}{1 + e^{s_{MV_{12}}(VP_1 - VP2)}} 6.4 \cdot 10^{-9} HEC_{12}(VP_1 - VP_2) \qquad [kg \cdot m^{-2} s^{-1}] \tag{31}$$

where $s_{MV_{12}}$: the slope of the differentiable switch function for vapour pressure differences.

3.4.1.e Vapor pressure flux from Location to Location

Besides of the flux between air of location and object, there are also the other fluxes from **one location** to **another location** similar to those of CO_2 concentration model and they are described in a general form:

$$MV_{12} = \frac{M_{Water}}{R} f_{12}(\frac{VP_1}{T_1} - \frac{VP_2}{T_2}) \qquad [kg \cdot m^{-2} s^{-1}] \tag{32}$$

where
M_{Water}: the molar mass of water (kg $kmol^{-1}$)
R: the molar gas constant = 8.314 (J mol^{-1} K^{-1}
f_{12}: the air flux from location 1 to location 2 (m^3 m^{-2} s^{-1}
T_1, T_2: the temperature at location 1 and location 2, respectively (K)

Notice that in the equation (31), our formula is quite different from what it should have been according to the Chapter 8 of the book [Van11]. This is because we have been assumed from that start that all temperature will be used as **Kelvin (K)** units, hence, the constant 273.15 will be cut down.

3.4.1.f AirThScr - Greenhouse Lower Compartment to Thermal Screen

According to Chapter 8 of [Van11], the vapor flux from the greenhouse air compartment to the thermal screen is described analogously to the equation (29). However, we chose to use the equation (30) derived from the equation (29) as it would be more easily for us to implement the algorithm and functions:

$$MV_{AirThScr} = \frac{1}{1 + e^{s_{MV_{AirThScr}}(VP_{Air} - VPThScr)}} 6.4 \cdot 10^{-9} HEC_{AirThScr}(VP_{Air} - VP_{ThScr}) \tag{33}$$

where
$HEC_{AirThScr}$: the heat exchange coefficient between air and thermal screen (W m^{-2} K^{-1})
$s_{MV_{AirThScr}}$: the slope of the differentiable switch function for vapour pressure differences.
VP_{Air}, VP_{ThScr}[2]: the saturated vapor pressure at air and thermal screen, respectively (Pa)

The formula for $HEC_{AirThScr}$ is given below (De Zwart, 1996):

$$HEC_{AirThScr} = 1.7 U_{ThScr} |T_{Air} - T_{ThScr}|^{0.33} \tag{34}$$

[2] The VP_{ThScr} is calculated using equation (28)

3.4.1.g TopCov,in - Greenhouse Upper Compartment to Internal Cover Layer

In a similar fashion to the $VP_{AirThScr}$, we will use the equation (30) to calculate the mass vapor flux from the greenhouse upper compartment to the internal cover layer:

$$MV_{TopCov,in} = \frac{1}{1+e^{s_{MV_{TopCov,in}}(VP_{Top}-VPCov,in)}} 6.4\cdot 10^{-9} HEC_{TopCov,in}(VP_{Top} - VP_{Cov,in}) \tag{35}$$

where
$HEC_{TopCov,in}$: the heat exchange coefficient between upper compartment and internal cover layer (W m^{-2} K^{-1})
$s_{MV_{TopCov,in}}$: the slope of the differentiable switch function for vapour pressure differences.
$VP_{Air}, VP_{Cov,in}$[3]: the saturated vapor pressure at upper compartment and thermal screen, respectively (Pa)

The formula for $HEC_{TopCov,in}$ is given below (Roy et al., 2002):

$$HEC_{TopCov,in} = c_{HECin}(T_{Top} - T_{Cov,in})^{0.33}\frac{A_{Cov}}{A_{Flr}} \tag{36}$$

3.4.1.h AirTop - Greenhouse Lower Compartment to Upper Compartment

According to Chapter 8 of [Van11], the vapor flux from one location to other location, specifically from lower compartment to upper one is described analogously to equation (31), whereby its accompanying air fluxes is f_{ThScr}:

$$MV_{AirTop} = \frac{M_{Water}}{R} f_{ThScr}(\frac{VP_{Air}}{T_{Air}} - \frac{VP_{Top}}{T_{Top}}) \tag{37}$$

3.4.1.i AirOut - Greenhouse Lower Compartment to Outside Space

In a similar fashion to the VP_{AirTop}, the vapor flux from lower compartment to outside space is described analogously to equation (31), whereby its accompanying air fluxes is $f_{VentSide} + f_{VentForced}$:

$$MV_{AirOut} = \frac{M_{Water}}{R} f_{ThScr}(\frac{VP_{Air}}{T_{Air}} - \frac{VP_{Out}}{T_{Out}}) \tag{38}$$

3.4.1.j TopOut - Greenhouse Upper Compartment to Outside Space

In a similar fashion to the previous equations, the vapor flux from upper compartment to outside is described analogously to equation (31), whereby its accompanying air fluxes is $f_{VentRoof}$:

$$MV_{TopOut} = \frac{M_{Water}}{R} f_{VentRoof}(\frac{VP_{Top}}{T_{Top}} - \frac{VP_{Out}}{T_{Out}}) \tag{39}$$

3.4.1.k CanAir - Exchange between Leaves and Greenhouse Lower Compartment

According to Chapter 8 of [Van11], the vapor flux between canopy and greenhouse upper compartment is described as follow:

[3] The $VP_{Cov,in}$ is calculated using equation (28)

$$MV_{CanAir} = VEC_{CanAir}(VP_{Can} - VP_{Air}) \tag{40}$$

where
VEC_{CanAir}: the vapour exchange coefficient between the canopy and air (kg Pa s^{-1})

The formula for VEC_{CanAir} is given below (Stanghellini, 1987):

$$VEC_{CanAir} = \frac{2\rho_{Air}c_{p,Air}LAI}{\triangle H\gamma(r_b + r_s)} \tag{41}$$

where
ρ_{Air}: the density of the greenhouse air (kg m^{-3})
$c_{p,Air}$: the specific heat capacity of the greenhouse air (J K^{-1} kg^{-1})
$\triangle H$: the lantent heat of evaporation of water
γ: the psychometric constant (Pa K_{-1})
r_b: the boundary layer resistance of the canopy for vapor transport (s m^{-1})
r_s: the stomatal resistance of the canopy for the vapor transport (s m^{-1})

We will analyze specifically the stomata resistance of the canopy through many more equations below:

$$VEC_{CanAir} = \frac{2\rho_{Air}c_{p,Air}LAI}{\triangle H\gamma(r_b + r_s)} \tag{42}$$

where
ρ_{Air}: the density of the greenhouse air (kg m^{-3})
$c_{p,Air}$: the specific heat capacity of the greenhouse air (J K^{-1} kg^{-1})
$\triangle H$: the lantent heat of evaporation of water
γ: the psychometric constant (Pa K_{-1})
r_b: the boundary layer resistance of the canopy for vapor transport (s m^{-1})
r_s: the stomatal resistance of the canopy for the vapor transport (s m^{-1})

From the VEC_{CanAir}, we will break it down into different parts which are:

$$r_s = rf(R_{Can}) \cdot rf(CO_{2Air_{ppm}}) \cdot rf(VP_{Can} - VPAir) \tag{43}$$

where
$rf(R_{Can})$: the resistance factor for high radiation levels
R_{Can}: the global radiation above the canopy (W m^{-2})
$rf(CO_{2Air})$: the resistance factor for high CO_2 levels
$rf(VP_{Can} - VP_{Air})$: the resistance factor for large vapor pressure differences

$$rf(R_{Can}) = \frac{R_{Can} + c_{evap1}}{R_{Can} + c_{evap2}} \tag{44}$$

$$rf(R_{CO_{2Air}}) = 1 + c_{evap3}(\eta_{mg_ppm}CO_{2Air} - 200)^2 \tag{45}$$

$$rf(VP_{Can} - VP_{Air}) = 1 + c_{evap4}(VP_{Can} - VP_{Air})^2 \tag{46}$$

where
$c_{evap1}(Wm^{-2}), c_{evap2}(Wm^{-2}), c_{evap3}(ppm^{-2}), c_{evap4}(Pa^{-2})$: empirically determined paramaters
η_{mg_ppm}: the conversion factor from $mgm^{-3}CO_2$ to ppm

3.4.1.l BlowAir - Air Heaters to Greenhouse Lower Compartment

According to Chapter 8 of [Van11], the vapor flux from direct air heater to the lower compartment is described as follow:

$$MV_{BlowAir} = \eta_{HeatVap} H_{BlowAir}/10^6 \tag{47}$$

where
$H_{BlowAir}$: the heat flux from the heat blower to the greenhouse air (W m^{-2}), which is described as:

$$H_{BlowAir} = U_{Blow} P_{Blow}/A_{Flr} \tag{48}$$

In the $MV_{BlowAir}$ equation, it is quite different compared to the original formula (8.53) from [Van11] as the former unit is mg m^{-2} s^{-1}, thus we need to devide for 10^6 so this equation would have the same unit kg m^{-2} s^{-1} with the other flux equation.

3.4.1.m PadAir - Pad and Fan System to Greenhouse Lower Compartment

According to Chapter 8 of [Van11], the vapor flux from pad and fan system to the lower compartment is described as follow:

$$MV_{PadAir} = \rho_{Air} f_{Pad} (\eta_{Pad}(x_{Pad} - x_{Out}) + x_{Out}) \tag{49}$$

where
f_{Pad}: the ventilation flux due to the pad and fan system (m^3 m^{-2} s^{-1}
η_{Pad}: the efficiency of the pad and fan system
x_{Pad}: the water vapor contnet of the pad (kg water kg^{-1} air)
x_{Out}: the water vapor content of the outdoor air (kg water kg^{-1} air)

The ventilation flux due to the pad and fan system is described by:

$$f_{Pad} = U_{Pad} \phi_{Pad}/A_{Flr} \tag{50}$$

where
U_{Pad}: the control valve of the pad and fan system
ϕ_{Pad}: the capacity of the air flux through the pad (m^3 s^{-1}

3.4.1.n PadAir - Pad and Fan System to Greenhouse Lower Compartment

According to Chapter 8 of [Van11], the vapor flux from pad and fan system to the lower compartment is described as follow:

$$MV_{PadAir} = \rho_{Air} f_{Pad} (\eta_{Pad}(x_{Pad} - x_{Out}) + x_{Out}) \tag{51}$$

where
f_{Pad}: the ventilation flux due to the pad and fan system (m^3 m^{-2} s^{-1})
η_{Pad}: the efficiency of the pad and fan system
x_{Pad}: the water vapor contnet of the pad (kg water kg^{-1} air)
x_{Out}: the water vapor content of the outdoor air (kg water kg^{-1} air)

The ventilation flux due to the pad and fan system is described by:

$$f_{Pad} = U_{Pad}\phi_{Pad}/A_{Flr} \tag{52}$$

where
U_{Pad}: the control valve of the pad and fan system
ϕ_{Pad}: the capacity of the air flux through the pad ($m^3\ s^{-1}$)

3.4.1.o AirOutPad - Greenhouse Lower Compartment to OutSide Space through Pad and Fan System

According to Chapter 8 of [Van11], the vapor flux from lower compartment to the outside space when using the pad and fan system is described as:

$$MV_{AirOut\_Pad} = f_{Pad}\frac{M_{Water}}{R}(\frac{VP_{Air}}{T_{Air}}) \tag{53}$$

3.4.1.p FogAir - Fogging System to Greenhouse Lower Compartment

According to Chapter 8 of [Van11], the vapor flux from the fogging system to the greenhouse lower compartment is described as:

$$MV_{FogAir} = U_{Fog}\phi_{Fog}/A_{Flr} \tag{54}$$

where
U_{Fog}: the control valve of the fogging system
ϕ_{Fog}: the capacity of the fogging system (kg water s^{-1})

3.4.1.q AirMech - Greenhouse Lower Compartment to the Surface of Mechanical Cooling System

In a similar fashion to the $VP_{AirThScr}$ and $VP_{TopCov,in}$, we will use the equation (30) to calculate the mass vapor flux from the greenhouse lower compartment to the surface of mechanical cooling system:

$$MV_{AirMech} = \frac{1}{1+e^{s_{MV_{AirMech}}(VP_{Air}-VPMech)}}6.4\cdot10^{-9}HEC_{AirMech}(VP_{Air}-VP_{Mech}) \tag{55}$$

where
$HEC_{AirMech}$: the heat exchange coefficient between the surface of the mechanical cooling unit and the greenhouse air (W $m^{-2}\ K^{-1}$)

A mechanical cooling system can be used to decrease both the sensible and latent heat in the greenhouse. It was assumed that the temperature of the surface of the mechanical cooling unit is an input of the system and that the total cooling capacity of the mechanical cooling installation (used for heat and vapour removal) depends on the coefficient of performance (COP) and the installed electrical capacity. Therefore, the formula for $HEC_{AirMech}$ is given below):

$$HEC_{AirMech} = \frac{(U_{MechCool}COP_{MechCool}P_{MechCool}/A_{Flr})}{T_{Air}-T_{MechCool}+6.4\cdot10^{-9}(VP_{Air}-VP_{MechCool})} \tag{56}$$

where
$U_{MechCool}$: the control valve of the machnical cooling mechanism
$COP_{MechCool}$: the coefficient of the performance of the mechanical cooling system
$P_{MechCool}$: the electrical capacity of the mechanical cooling system (W)

3.4.2 Problem B

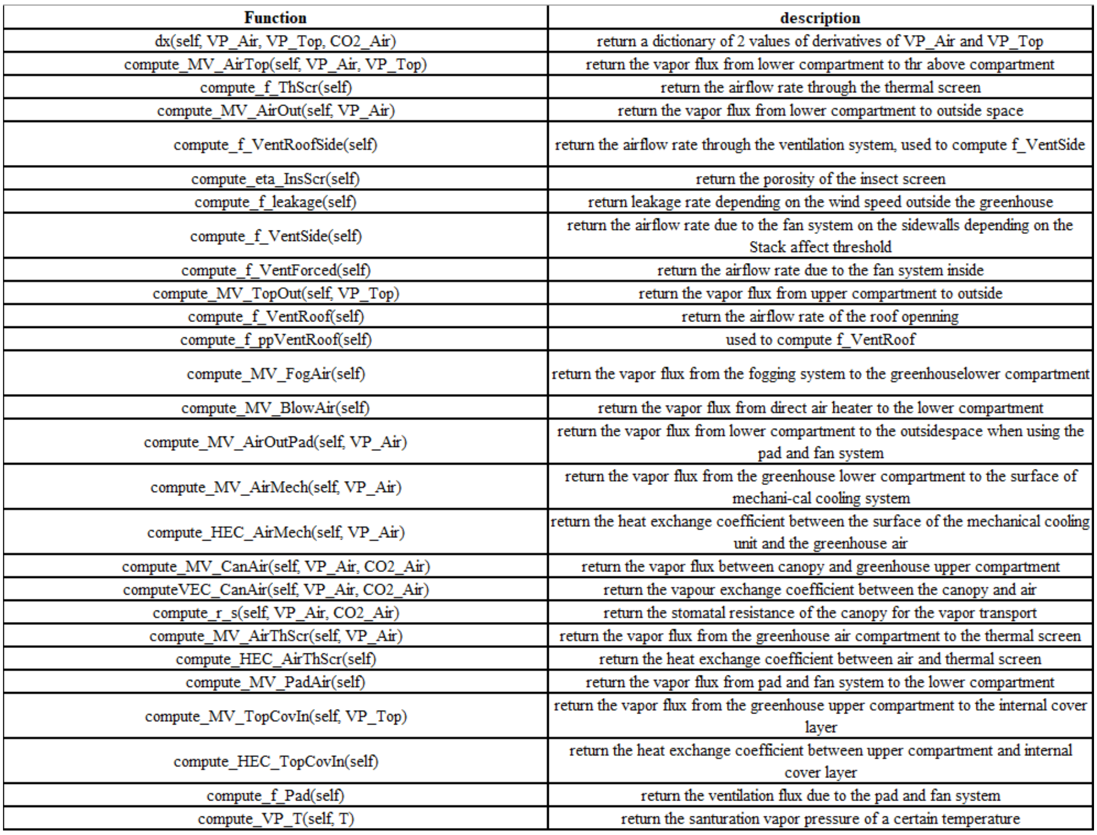
Write programs that calculate the net VP flux from one place to an other place using the formulas provided in the assignment. Then write a program that returnsthe right side of the system's formulas divided by cap_{VPAir} and cap_{VPTop} respectively and named this function dx.

The VP program organization is similar to the CO_2 system program, which have 2 classes

- greenhouseParameters - holds all the parameters constant values of the greenhouse.
- greenhouseVPsystem - holds functions that process system formulas, compute VP fluxes and the rates of change of the VP.

The algorithm of function dx of VP is similar to CO_2 concentration, except the input parameters include the value CO_{2Air} at the same time calculating a certain rate of change of VP.

More detailed description of each function is described in Figure 24.



Function	description
dx(self, VP_Air, VP_Top, CO2_Air)	return a dictionary of 2 values of derivatives of VP_Air and VP_Top
compute_MV_AirTop(self, VP_Air, VP_Top)	return the vapor flux from lower compartment to thr above compartment
compute_f_ThScr(self)	return the airflow rate through the thermal screen
compute_MV_AirOut(self, VP_Air)	return the vapor flux from lower compartment to outside space
compute_f_VentRoofSide(self)	return the airflow rate through the ventilation system, used to compute f_VentSide
compute_eta_InsScr(self)	return the porosity of the insect screen
compute_f_leakage(self)	return leakage rate depending on the wind speed outside the greenhouse
compute_f_VentSide(self)	return the airflow rate due to the fan system on the sidewalls depending on the Stack affect threshold
compute_f_VentForced(self)	return the airflow rate due to the fan system inside
compute_MV_TopOut(self, VP_Top)	return the vapor flux from upper compartment to outside
compute_f_VentRoof(self)	return the airflow rate of the roof openning
compute_f_ppVentRoof(self)	used to compute f_VentRoof
compute_MV_FogAir(self)	return the vapor flux from the fogging system to the greenhouselower compartment
compute_MV_BlowAir(self)	return the vapor flux from direct air heater to the lower compartment
compute_MV_AirOutPad(self, VP_Air)	return the vapor flux from lower compartment to the outsidespace when using the pad and fan system
compute_MV_AirMech(self, VP_Air)	return the vapor flux from the greenhouse lower compartment to the surface of mechani-cal cooling system
compute_HEC_AirMech(self, VP_Air)	return the heat exchange coefficient between the surface of the mechanical cooling unit and the greenhouse air
compute_MV_CanAir(self, VP_Air, CO2_Air)	return the vapor flux between canopy and greenhouse upper compartment
computeVEC_CanAir(self, VP_Air, CO2_Air)	return the vapour exchange coefficient between the canopy and air
compute_r_s(self, VP_Air, CO2_Air)	return the stomatal resistance of the canopy for the vapor transport
compute_MV_AirThScr(self, VP_Air)	return the vapor flux from the greenhouse air compartment to the thermal screen
compute_HEC_AirThScr(self)	return the heat exchange coefficient between air and thermal screen
compute_MV_PadAir(self)	return the vapor flux from pad and fan system to the lower compartment
compute_MV_TopCovIn(self, VP_Top)	return the vapor flux from the greenhouse upper compartment to the internal cover layer
compute_HEC_TopCovIn(self)	return the heat exchange coefficient between upper compartment and internal cover layer
compute_f_Pad(self)	return the ventilation flux due to the pad and fan system
compute_VP_T(self, T)	return the santuration vapor pressure of a certain temperature

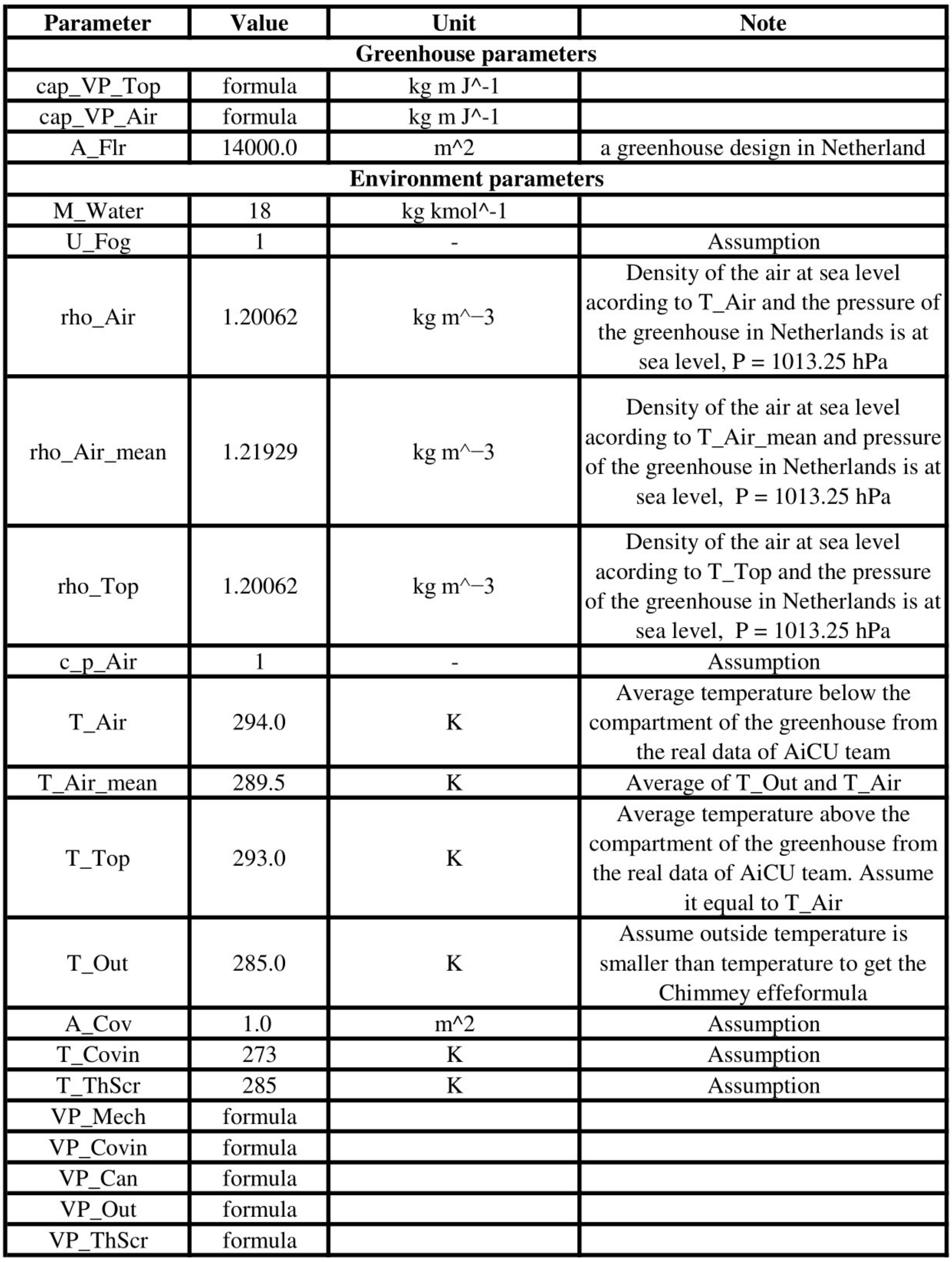
Figure 24: The description of functions of the VP system

3.4.3 Problem C

Assuming constant temperature difference and constant air density difference, study from [Van11] and related citations to find specific and reasonable values foreach input parameter of the function dx including the difference in temperatureand in air density except for the variables VP_{Air} and VP_{Top}.

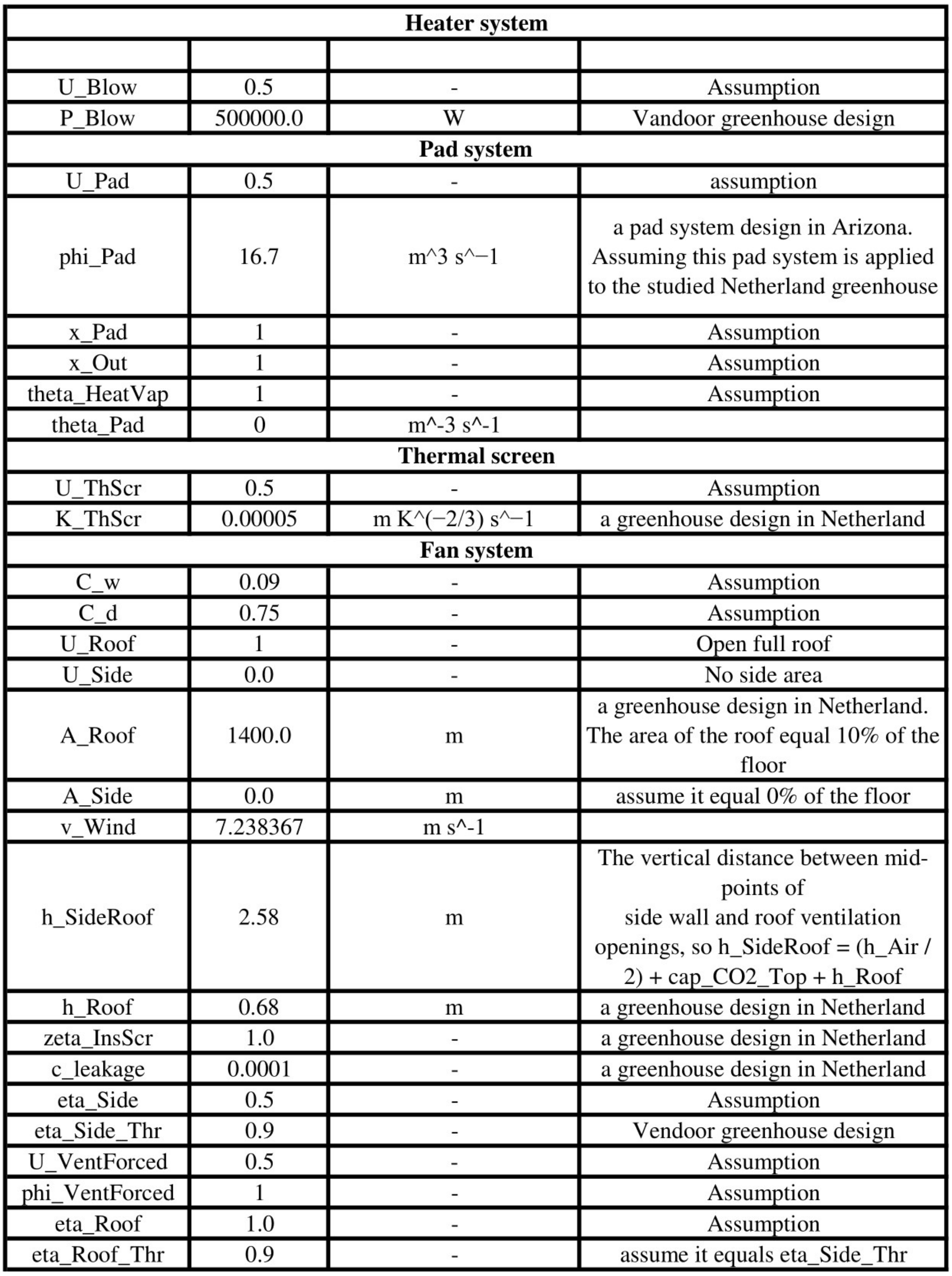
Similarly to the CO_2 model, because of the lack on some greenhouse paramaters information on the data of team AiCU, so we assume to use the parameters value of a certain greenhouse in Netherlands, and try to modified reasonably and estimate to reduce the error with the real amount of vapor pressure data in order to be useful to predict in the future.

According to the assumptions of the greenhouse system we have stated in Exercise 2, team AiCU data, [Van11], assignment description etc, the parameters' values holds as the following table. Note that we also resued some parameters from the CO_2 model to maintain the same condition of the greenhouse.



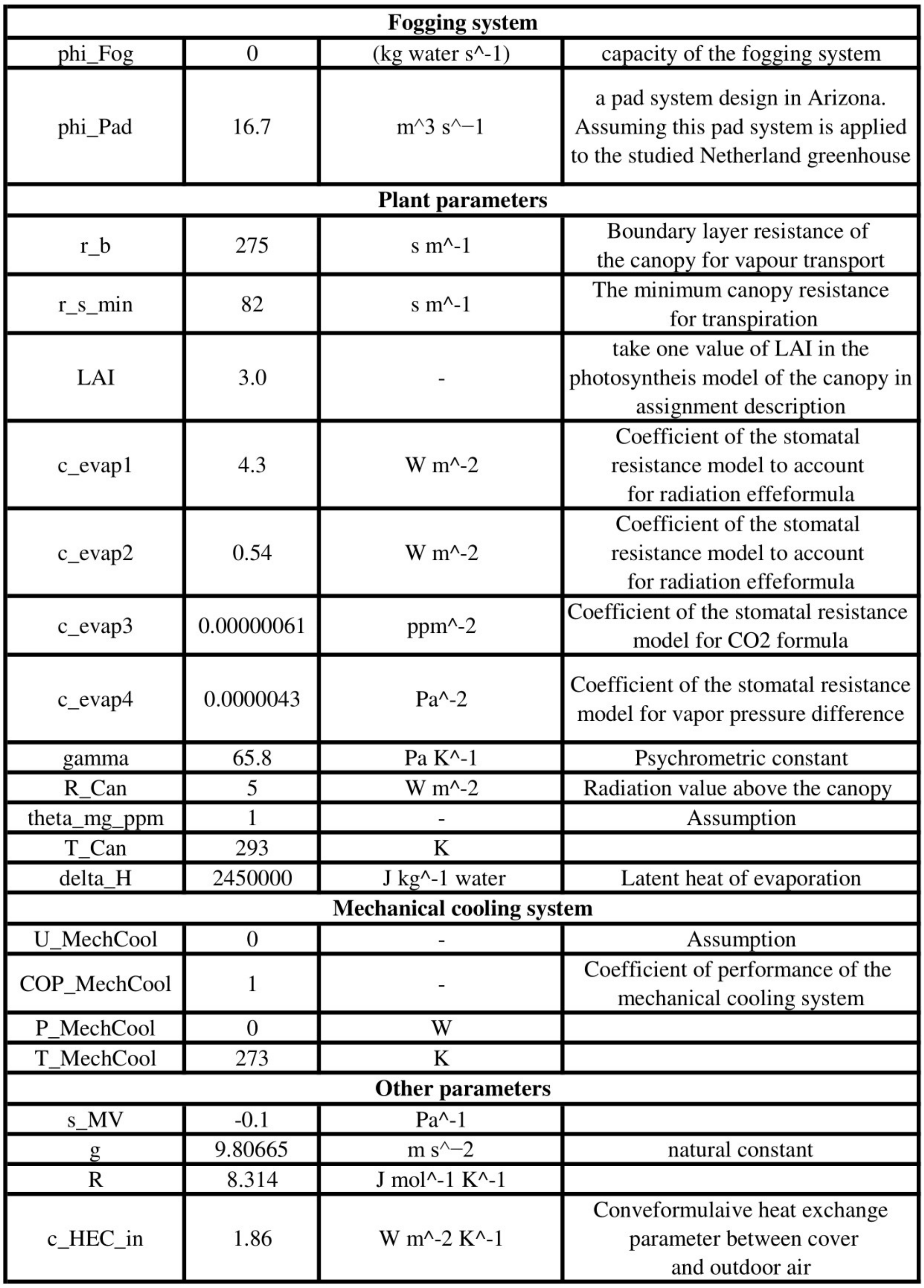
Parameter	Value	Unit	Note
Greenhouse parameters			
cap_VP_Top	formula	kg m J^-1	
cap_VP_Air	formula	kg m J^-1	
A_Flr	14000.0	m^2	a greenhouse design in Netherland
Environment parameters			
M_Water	18	kg kmol^-1	
U_Fog	1	-	Assumption
rho_Air	1.20062	kg m^−3	Density of the air at sea level acording to T_Air and the pressure of the greenhouse in Netherlands is at sea level, P = 1013.25 hPa
rho_Air_mean	1.21929	kg m^−3	Density of the air at sea level acording to T_Air_mean and pressure of the greenhouse in Netherlands is at sea level, P = 1013.25 hPa
rho_Top	1.20062	kg m^−3	Density of the air at sea level acording to T_Top and the pressure of the greenhouse in Netherlands is at sea level, P = 1013.25 hPa
c_p_Air	1	-	Assumption
T_Air	294.0	K	Average temperature below the compartment of the greenhouse from the real data of AiCU team
T_Air_mean	289.5	K	Average of T_Out and T_Air
T_Top	293.0	K	Average temperature above the compartment of the greenhouse from the real data of AiCU team. Assume it equal to T_Air
T_Out	285.0	K	Assume outside temperature is smaller than temperature to get the Chimmey effeformula
A_Cov	1.0	m^2	Assumption
T_Covin	273	K	Assumption
T_ThScr	285	K	Assumption
VP_Mech	formula		
VP_Covin	formula		
VP_Can	formula		
VP_Out	formula		
VP_ThScr	formula		

Figure 25: VP system parameters



Heater system			
U_Blow	0.5	-	Assumption
P_Blow	500000.0	W	Vandoor greenhouse design
Pad system			
U_Pad	0.5	-	assumption
phi_Pad	16.7	m^3 s^−1	a pad system design in Arizona. Assuming this pad system is applied to the studied Netherland greenhouse
x_Pad	1	-	Assumption
x_Out	1	-	Assumption
theta_HeatVap	1	-	Assumption
theta_Pad	0	m^-3 s^-1	
Thermal screen			
U_ThScr	0.5	-	Assumption
K_ThScr	0.00005	m K^(−2/3) s^−1	a greenhouse design in Netherland
Fan system			
C_w	0.09	-	Assumption
C_d	0.75	-	Assumption
U_Roof	1	-	Open full roof
U_Side	0.0	-	No side area
A_Roof	1400.0	m	a greenhouse design in Netherland. The area of the roof equal 10% of the floor
A_Side	0.0	m	assume it equal 0% of the floor
v_Wind	7.238367	m s^-1	
h_SideRoof	2.58	m	The vertical distance between mid-points of side wall and roof ventilation openings, so h_SideRoof = (h_Air / 2) + cap_CO2_Top + h_Roof
h_Roof	0.68	m	a greenhouse design in Netherland
zeta_InsScr	1.0	-	a greenhouse design in Netherland
c_leakage	0.0001	-	a greenhouse design in Netherland
eta_Side	0.5	-	Assumption
eta_Side_Thr	0.9	-	Vendoor greenhouse design
U_VentForced	0.5	-	Assumption
phi_VentForced	1	-	Assumption
eta_Roof	1.0	-	Assumption
eta_Roof_Thr	0.9	-	assume it equals eta_Side_Thr

Figure 26: VP system parameters



Fogging system			
phi_Fog	0	(kg water s^-1)	capacity of the fogging system
phi_Pad	16.7	m^3 s^−1	a pad system design in Arizona. Assuming this pad system is applied to the studied Netherland greenhouse
Plant parameters			
r_b	275	s m^-1	Boundary layer resistance of the canopy for vapour transport
r_s_min	82	s m^-1	The minimum canopy resistance for transpiration
LAI	3.0	-	take one value of LAI in the photosyntheis model of the canopy in assignment description
c_evap1	4.3	W m^-2	Coefficient of the stomatal resistance model to account for radiation effeformula
c_evap2	0.54	W m^-2	Coefficient of the stomatal resistance model to account for radiation effeformula
c_evap3	0.00000061	ppm^-2	Coefficient of the stomatal resistance model for CO2 formula
c_evap4	0.0000043	Pa^-2	Coefficient of the stomatal resistance model for vapor pressure difference
gamma	65.8	Pa K^-1	Psychrometric constant
R_Can	5	W m^-2	Radiation value above the canopy
theta_mg_ppm	1	-	Assumption
T_Can	293	K	
delta_H	2450000	J kg^-1 water	Latent heat of evaporation
Mechanical cooling system			
U_MechCool	0	-	Assumption
COP_MechCool	1	-	Coefficient of performance of the mechanical cooling system
P_MechCool	0	W	
T_MechCool	273	K	
Other parameters			
s_MV	-0.1	Pa^-1	
g	9.80665	m s^−2	natural constant
R	8.314	J mol^-1 K^-1	
c_HEC_in	1.86	W m^-2 K^-1	Conveformulaive heat exchange parameter between cover and outdoor air

Figure 27: VP system parameters

3.4.4 Problem D

Study explicit Euler and Runge–Kutta of order 4 algorithms for solving first-order differential equations. Write two programs performing the two algorithms and named them euler and rk4 respectively with inputs: a callable function as dx, the values at time t of the two variables VP, the time step h. These solvers return the approximate values of VP_{Air} and VP_{Top} time t+h.

The two functions that perform 2 estimation methods of VP use the same theoretical computation technique. However, they have a slight difference from the functions **euler_CO2()** and **rk4_CO2()** of CO_2 because we have to take $CO2\_Air$ into account and choose to use its estimated value to compute dx functions as well as the results of VP, which means we take all 4 differential equations and estimate at the same time. We will take a closer look at the 2 functions

euler_VP(dx, CO2_Air, VP_Air, VP_Top, h): returns a dictionary of the 2 next value of VP above and below the screen calculated by using the explicit Euler method. Note that the $CO2\_Air$ here is the result of the Euler estimation at a synchronous time t with VP.

rk4_CO2(dx, dx_CO2, CO2_Air, CO2_Top, VP_Air, VP_Top, h): returns the next value of VP above and below the screen calculated by using the explicit Runge-Kutta Four method. Note that there are 3 additional paramters from the previous CO_2 concentration computation to compute the value k_1, k_2, k_3 of CO_2 concentration, which is then passed into the dx of VP with step h in order to calculate the input parameter $CO2\_Air$ then compute k_2, k_3, k_4 value of VP, synchronous at time t, respectively.

Main function: With n values, the main function iterate the 2 functions n times, every time we append result into two list VP_euler and VP_rk4 to plot data.

dx(): The function calculating the rate of change with respect to time of 3 input: $CO2\_Air$, VP_Air and VP_Top.

More details about this two function, please visit the Notebook file.

3.4.5 Problem E

Select specific values of VP_{Air} and VP_{Top} at time t from the data set as initial values to run the solvers and find the approximate values of VP_{Air} and VP_{Top} in the next 5 minutes, 10 minutes, 20 minutes, ... and calculate the difference of the result from the actual data. Comment on the accuracy of the model and present details in the report.

Because we have not found any real dataset that related to the vapour pressure, so we only estimate it synchronously throught 24 hours with CO_2 model by 2 methods and give some comment on the results and the differences between the two.

Estimated data description: Two sets of values estimated by Euler and Runge-Kutta methodwith the step $h = 1s$. The initial condition for VP_{Air} and VP_{Top} depends on the parameters T_{Air} and T_{Top}, respectively. Using the santuration vapor pressure formula we get the value

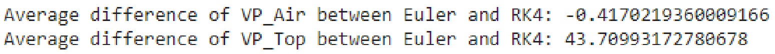
$$VP_{Air\_initial} = VP_{Top\_initial} = 2304.542889027799(Pa)$$

Comment on results

Overall, it can be seen from Figure 29 that the vapour pressure below the thermal screen going upward constantly with time from $2305(Pa)$ going up to over $2309(Pa)$ and see no sign of steady trend on the first 24 hours. In contrast, the vapour pressure above the thermal screen decrease constantly from $2305(Pa)$ going down to below $1800(Pa)$ and see no sign of steady trend on the first 24 hours.

The vapor pressure estimation that use Runge-Kutta 4 Order also see a similar trend. The vapour pressure below the thermal screen going upward from from $2305(Pa)$ going up to over $2310(Pa)$. In contrast, the vapour pressure above the thermal screen decrease from $2305(Pa)$ going down to below $1700(Pa)$.

In Figure 28, we can see there exist differences when estimate the vapour pressure of the 2 methods. The average difference of VP_{Top} is larger than the average difference of VP_{Air}.



```
Average difference of VP_Air between Euler and RK4: -0.4170219360009166
Average difference of VP_Top between Euler and RK4: 43.70993172780678
```

Figure 28: Average difference of VP value between the 2 methods

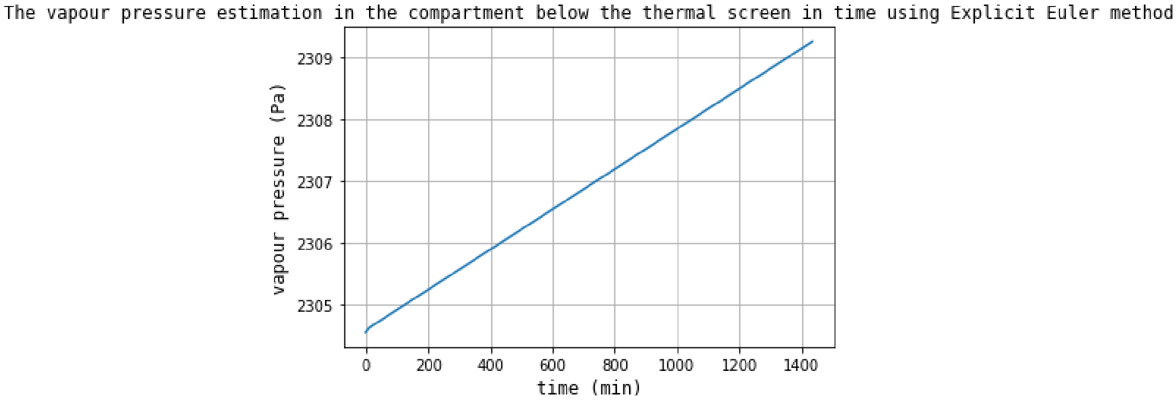


Figure 29: Vapour pressure below the screen for the first 24 hours using Euler method

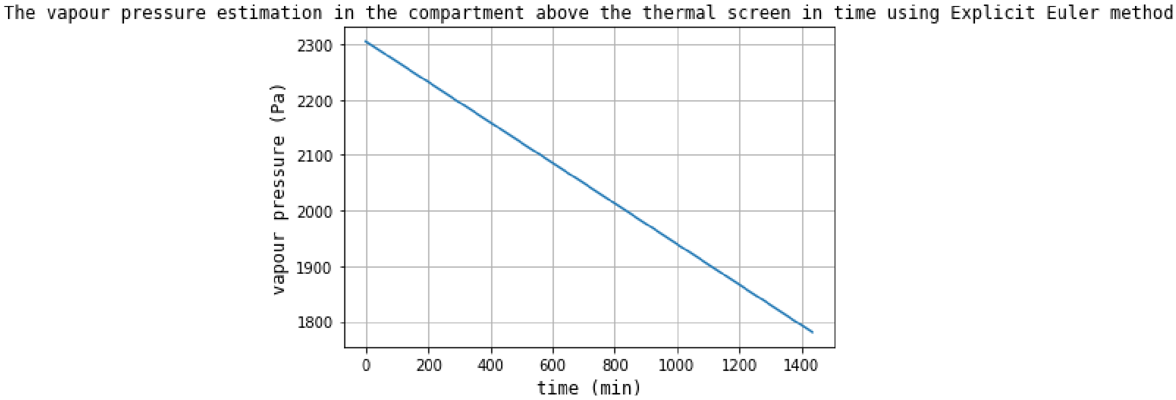


Figure 30: Vapour pressure above the screen for the first 24 hours using Euler method

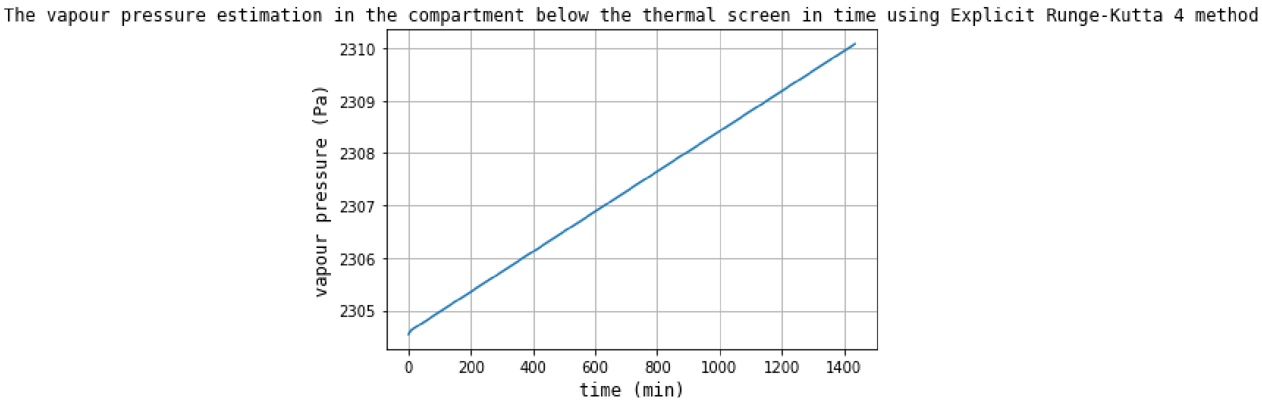


Figure 31: Vapour pressure below the screen for the first 24 hours using Explicit Runge-Kutta 4 method

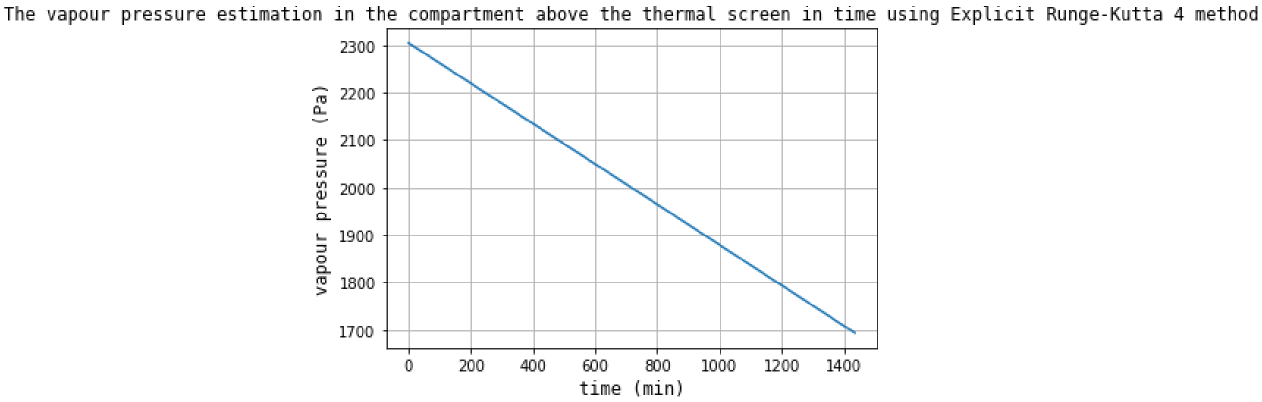


Figure 32: Vapour pressure above the screen for the first 24 hours using Explicit Runge-Kutta 4 method

3.5 Python Jupiter Notebook organization

The organization of the Notebook that helps you to goes through the code.

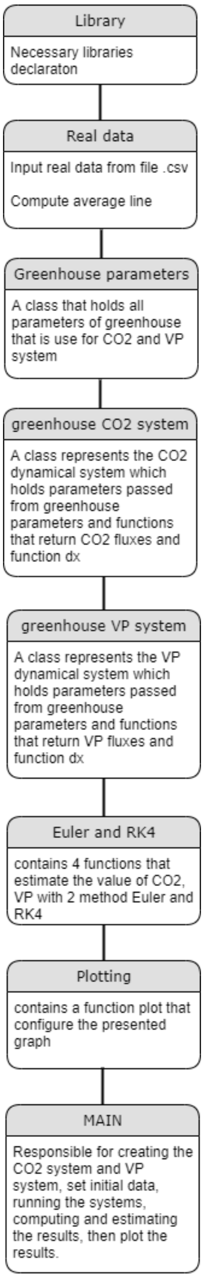

Library
Necessary libraries declaraton
Real data
Input real data from file .csv
Compute average line
Greenhouse parameters
A class that holds all parameters of greenhouse that is use for CO2 and VP system
greenhouse CO2 system
A class represents the CO2 dynamical system which holds parameters passed from greenhouse parameters and functions that return CO2 fluxes and function dx
greenhouse VP system
A class represents the VP dynamical system which holds parameters passed from greenhouse parameters and functions that return VP fluxes and function dx
Euler and RK4
contains 4 functions that estimate the value of CO2, VP with 2 method Euler and RK4
Plotting
contains a function plot that configure the presented graph
MAIN
Responsible for creating the CO2 system and VP system, set initial data, running the systems, computing and estimating the results, then plot the results.

Figure 33: Python Notebook organization

References

[BG08] John Charles Butcher and Nicolette Goodwin. *Numerical methods for ordinary differential equations*. Vol. 2. Wiley Online Library, 2008.

[Ope11] MIT OpenCourseWare. “The Existence and Uniqueness Theorem for Linear Systems”. In: (2011). DOI: https://ocw.mit.edu/courses/mathematics/18-03sc-differential-equations-fall-2011/unit-iv-first-order-systems/matrix-exponentials/MIT18_03SCF11_s35_2text.pdf.

[Van11] Bram H.E. Vanthoor. “A mode-based greenhouse design method”. 2011. ISBN: ISBN 978-90-8585-919-2.

[GFH13] Frank R. Giordano, William P. Fox, and Steven B. Horton. *A First Course in Mathematical Modeling Fifth Edition*. Richard Stratton, 2013. ISBN: 978-1-285-05090-4.

[NN20] T.Thinh Nguyen and A.Khuong Nguyen. *Mathematical Modeling Assignment: Dynamical systems in forecasting Greenhouse Micro-climate*. 2020.

[SCW20] James Stewart, Daniel K Clegg, and Saleem Watson. *Calculus: early transcendentals*. Cengage Learning, 2020.