

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



HỆ CƠ SỞ DỮ LIỆU (CO2013)

Báo cáo bài tập lớn 2

Học kì 211 - Nhóm L01

GV ra đề và HD: Trương Quỳnh Chi
Sinh viên: Trần Long Vĩ - 1814804
A - 181xxxx

TP. HỒ CHÍ MINH, THÁNG 11/2020



Mục lục

1	Giới thiệu đề tài	2
2	Kiến thức chuẩn bị về hệ động lực	2
2.1	Giới thiệu về phương trình vi phân	2
2.1.1	Vài mô hình đơn giản	2
2.1.2	Các khái niệm	3
2.1.3	Định nghĩa hệ phương trình vi phân cấp 1 tổng quát	4
2.2	Định lý tồn tại và duy nhất nghiệm	4
2.2.1	Bài toán Cauchy	4
2.2.2	Sự tồn tại duy nhất nghiệm	4
2.3	Ví dụ về hệ phương trình vi phân bậc nhất	5
2.3.1	Bài toán rắn sẵn môi	5
2.3.2	Chuyển động của 2 vật bởi lực hấp dẫn	6
2.4	Giải thuật Explicit Euler và Runge Kutta bậc 4	6
2.4.1	Euler Explicit	6
2.4.2	Runge Kutta bậc 4 Explicit	7
2.5	Hiện thực hai ví dụ ở phần 2.3 bằng Euler Explicit và Runge Kutta bậc 4 Explicit	8
2.5.1	Bài toán rắn sẵn môi	8
2.5.2	Bài toán chuyển động của 2 vật bởi lực hấp dẫn	9
3	Bài 2: Mô hình hệ động lực đối với nồng độ khí CO_2 trong nhà kính	9
3.1	Sự trao đổi khí CO_2	9
3.2	Mô hình Hệ động lực và giả thiết	11
3.3	Phân tích các thông số của hệ động lực	13
3.4	Giá trị của các tham số và giải thích	23
3.5	Giả định về các giá trị	25
3.6	Tính toán xấp xỉ giá trị của nồng độ CO_2 trong nhà kính và mô phỏng	25
3.6.1	Giải thuật tính xấp xỉ hệ động lực	25
3.6.2	Mô phỏng hệ động lực và so sánh với dữ liệu thực	26
4	Mô hình hệ động lực đối với áp suất hơi nước trong nhà kính	28
4.1	Hệ động lực và giả thiết	28
4.2	Các thông số ảnh hưởng tới áp suất hơi nước	28
4.3	Các tham số của hệ động lực và giải thích	36
4.4	Các giả định về giá trị	37
4.5	Tính toán xấp xỉ giá trị của áp suất hơi nước trong nhà kính và mô phỏng	38
4.5.1	Giải thuật tính xấp xỉ hệ động lực	38
4.5.2	Mô phỏng hệ động lực và so sánh với dữ liệu thực	38
	Tài liệu tham khảo	40

1 Giới thiệu đề tài

Ngày nay, nhiều giống hoa màu được trồng trái mùa trong các nhà kính hiện đại với các hệ thống điều khiển tự động hoặc bán tự động có chức năng điều chỉnh các yếu tố thời tiết và khí hậu bên trong nhà kính nhằm tạo môi trường tốt nhất cho các vụ mùa.

Một số thành phần khí hậu chính bên trong nhà kính bao gồm: nhiệt độ, áp suất hơi nước và đặc biệt là nồng độ khí CO_2 ảnh hưởng trực tiếp đến năng suất của cây trồng. Các thành phần này thông thường bị ảnh hưởng bởi một hay nhiều vật thể có trong nhà kính, cấu trúc nhà kính và sự chuyển động của dòng không khí bên trong nhà kính. Như được mô tả trong Hình ??,

2 Kiến thức chuẩn bị về hệ động lực

2.1 Giới thiệu về phương trình vi phân

Phương trình vi phân hay phương trình sai phân là một phương trình toán học nhằm biểu diễn mối quan hệ giữa một hàm chưa được biết (một hoặc nhiều biến) với đạo hàm của nó (có bậc khác nhau). Phương trình sai phân đóng vai trò cực kì quan trọng trong kỹ thuật, vật lý, kinh tế và một số ngành khác.

Trong rất nhiều lĩnh vực ứng dụng, chuyển động của một hệ được mô hình hoá bởi các phương trình vi phân, tức là phương trình có chứa các đạo hàm của ẩn hàm cần tìm. Chẳng hạn, trong cơ học cổ điển (định luật Newton), trong thiên văn học (sự chuyển động của các hành tinh), trong hoá học (các phản ứng hoá học), trong sinh học (sự phát triển của dân số), trong điện tử... Trong hầu hết các lĩnh vực như thế, bài toán chung nhất là mô tả nghiệm của các phương trình này (cả về định tính lẫn về định lượng).

2.1.1 Vài mô hình đơn giản

Sự rơi tự do. Xét một vật có khối lượng m được thả rơi tự do trong khí quyển gần mặt đất. Theo định luật II Newton, chuyển động của vật đó có thể mô tả bởi phương trình

$$F = ma$$

trong đó F là hợp lực tác động lên vật và a là gia tốc chuyển động. Hợp lực F có thể giả thiết chỉ bao gồm lực hấp dẫn (tỷ lệ với khối lượng của vật và hướng xuống) và lực cản (tỷ lệ với vận tốc chuyển động và hướng lên trên). Ngoài ra, do gia tốc chuyển động $a = \frac{dv}{dt}$ nên (1.1) có thể viết dưới dạng

$$m \frac{dv}{dt} = mg - \gamma v$$

trong đó $g \approx 9,8 m/s^2$ là gia tốc trọng trường, còn γ là hệ số cản.

Vậy vận tốc v của vật rơi tự do thỏa mãn phương trình trên với sự xuất hiện của đạo hàm của v . Những phương trình như vậy ta sẽ gọi là phương trình vi phân.

Dung dịch hóa học. Giả sử tại thời điểm ban đầu $t = t_0$ một thùng chứa x_0 kg muối hòa tan trong 1000 lít nước. Ta cho chảy vào thùng một loại nước muối nồng độ a (kg/lít) với lưu lượng r (lít/phút) và khuấy đều. Đồng thời, cho hỗn hợp đó chảy ra khỏi thùng cũng với tốc độ như trên. Gọi $x = x(t)$ là lượng muối trong thùng tại thời điểm bất kỳ. Rõ ràng tỉ lệ thay đổi lượng muối trong thùng dx/dt bằng hiệu của tỉ lệ muối chảy vào ar (kg/phút) trừ đi tỉ lệ muối chảy ra tại thời điểm đang xét $rx/1000$ (kg/phút).

Vậy ta có phương trình vi phân

$$\frac{dx}{dt} = ar - \frac{rx}{1000}$$

Với dữ kiện ban đầu

$$x(t_0) = x_0$$

2.1.2 Các khái niệm

Phương trình vi phân là phương trình có dạng

$$F(x, y, y', y'', \dots, y^{(n)}) = 0$$

trong đó $y = y(x)$ là ẩn hàm cần tìm và nhất thiết phải có sự tham gia của đạo hàm (đến cấp nào đó) của ẩn.

Trong trường hợp ẩn hàm cần tìm là hàm nhiều biến (xuất hiện các đạo hàm riêng) thì phương trình vi phân còn gọi là *phương trình đạo hàm riêng*. Để phân biệt, người ta thường gọi phương trình với ẩn hàm là hàm một biến là *phương trình vi phân* thường và là đối tượng chính của giáo trình này.

Thông thường ta xét các phương trình với ẩn hàm là hàm số một biến thực $y = y(x)$ xác định trên khoảng mở $I \subset \mathbb{R}$; khi đó hàm F trong đẳng thức trên xác định trong một tập mở G của $\mathbb{R} \times \mathbb{R}^{n+1}$. Trong trường hợp ẩn hàm cần tìm là vector-hàm (hàm với giá trị vector) $y(x) = (y_1(x), \dots, y_m(x))^T \in \mathbb{R}^m$, F là một ánh xạ nhận giá trị trong \mathbb{R}^m và phương trình trên được hiểu là hệ phương trình vi phân.

Ta nói một phương trình vi phân có cấp n nếu n là cấp lớn nhất của đạo hàm của ẩn xuất hiện trong phương trình.

Phương trình vi phân thường cấp I có dạng tổng quát

$$F(x, y, y') = 0$$

trong đó $F(x, y, z)$ được giả thiết liên tục cùng với các đạo hàm riêng của nó trên miền $G \times \mathbb{R}^3$. Với một số giả thiết thích hợp (xem định lý hàm ẩn), phương trình vi phân cấp I có thể viết được dưới dạng sau (gọi là dạng giải ra được đối với đạo hàm)

$$y' = f(x, y)$$

với f liên tục trong một miền $D \times \mathbb{R}^2$.

Ví dụ: Các phương trình:

$$\begin{aligned} e^y + y'^2 \cos x &= 1 \\ (y''')^2 - 2xy &= \ln x \\ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} &= 0 \end{aligned}$$

lần lượt là phương trình vi phân thường cấp I, cấp III và phương trình đạo hàm riêng cấp II.

Xét phương trình $F(x, y, y', y'', \dots, y^{(n)}) = 0$, hàm giá trị vector $\phi : I \rightarrow \mathbb{R}^n$ (với $I = (a, b)$ là khoảng nào đó của \mathbb{R}) là nghiệm của phương trình F nếu nó có các đạo hàm liên tục đến cấp n trên I và thoả mãn

$$F(x, \phi(x), \phi'(x), \phi''(x), \dots, \phi^{(n)}(x)) = 0, \quad \forall x \in I$$

Trong trường hợp phương trình vi phân cấp I, nghiệm là một hàm thực một biến $y = \phi(x)$ mà khi thay vào $F(x, y, y') = 0$ hoặc $y' = f(x, y)$, ta được một đẳng thức đúng.

2.1.3 Định nghĩa hệ phương trình vi phân cấp 1 tổng quát

Hệ phương trình vi phân tổng quát là hệ gồm các phương trình chứa biến độc lập, các hàm (nghiệm) cần tìm và nhất thiết phải chứa các đạo hàm của chúng theo biến độc lập. Nếu chỉ xuất hiện các đạo hàm cấp I của các ẩn, ta nói hệ đó là hệ phương trình vi phân cấp I. Ta nói một hệ gồm n phương trình vi phân cấp I là có dạng chuẩn tắc (dạng giải ra được đối với đạo hàm) nếu có thể viết dưới dạng:

$$\begin{cases} \frac{dy_1}{dx} = f_1(x, y_1, \dots, y_n) \\ \frac{dy_2}{dx} = f_2(x, y_1, \dots, y_n) \\ \dots\dots\dots \\ \frac{dy_n}{dx} = f_n(x, y_1, \dots, y_n) \end{cases}$$

trong đó x là biến độc lập, y_1, \dots, y_n là các ẩn cần tìm.

Hệ phương trình chuẩn tắc trên có thể viết lại dưới dạng thu gọn như sau

$$y' = f(x, y)$$

trong đó, $y = (y_1, \dots, y_n)^T$, $y' = (y'_1, \dots, y'_n)^T$ và $f = (f_1, \dots, f_n)^T$.

2.2 Định lý tồn tại và duy nhất nghiệm

2.2.1 Bài toán Cauchy

Ta nhận xét rằng nghiệm của một phương trình vi phân nói chung phụ thuộc vào một hay nhiều hằng số tùy ý nào đó. Để xác định một nghiệm cụ thể, ta cần thêm một hay vài dữ kiện nào đó về nghiệm (tùy theo cấp của phương trình vi phân). Chẳng hạn, $y = \frac{x^3}{3} + C$ là nghiệm (tổng quát) của phương trình $y' = x^2$. Để thấy $y = \frac{x^3}{3} + 1$ là nghiệm (duy nhất) thỏa điều kiện $y(0) = 1$.

Ta xét bài toán sau đây đặt ra đối với phương trình $F(x, y, y') = 0$, gọi là bài toán Cauchy (hay bài toán giá trị ban đầu):

Bài toán: Tìm nghiệm $y(x)$ thỏa $\begin{cases} y' = f(x, y) \\ y(x_0) = y_0 \end{cases}$

trong đó $(x_0, y_0) \in D$ được gọi là điều kiện ban đầu. Câu hỏi tự nhiên đặt ra là bài toán trên có hay không và có bao nhiêu lời giải. Ta lưu ý rằng không phải lúc nào bài toán Cauchy cũng có nghiệm, và khi có nghiệm cũng không nhất thiết có duy nhất nghiệm. Chẳng hạn, phương trình $y' = x^2$, $y(0) = 0$ có duy nhất một nghiệm là $y = \frac{x^3}{3}$. Phương trình $xy' = y$, $y(0) = 1$ không có nghiệm nào; còn phương trình $y' = y^{1/3}$, $y(0) = 0$ có ít nhất 2 nghiệm là $y \equiv 0$ và $y^2 = \frac{8}{27}x^3$.

2.2.2 Sự tồn tại duy nhất nghiệm

Định nghĩa: Cho hàm $f(x, y)$ xác định trên miền $D \subset \mathbb{R}^2$. Ta nói f thỏa điều kiện Lipschitz theo biến y trên D nếu tồn tại hằng số dương L (gọi là hằng số Lipschitz) sao cho:

$$|f(x, y_1) - f(x, y_2)| = L|y_1 - y_2|, \quad \forall (x, y_1), (x, y_2) \in D$$

Định lý: (Định lý tồn tại và duy nhất nghiệm) Giả sử hàm số $f(x, y)$ liên tục và thỏa điều kiện Lipschitz theo biến y trên hình chữ nhật

$$D = \{(x, y) \in \mathbb{R}^2 \mid |x - x_0| \leq a, |y - y_0| \leq b\}$$

Khi đó nghiệm của bài toán Cauchy là tồn tại và duy nhất trong đoạn $I := [x_0 - h, x_0 + h]$, với $h := \min(a, \frac{b}{M})$ và $M := \max_{(x,y) \in D} |F(x, y)|$.

2.3 Ví dụ về hệ phương trình vi phân bậc nhất

2.3.1 Bài toán rắn sắn môi

Sự phát triển của hai quần thể sinh vật (chẳng hạn, $x = x(t)$ là số con mèo và $y = y(t)$ là số con chuột) theo thời gian được mô tả bởi (hệ) phương trình Volterra-Lotka sau đây

$$y' = y(\alpha - \beta x), x' = x(\gamma y - \delta)$$

với α, β, γ và δ là những hằng số đặc trưng cho sự tăng trưởng của các quần thể.

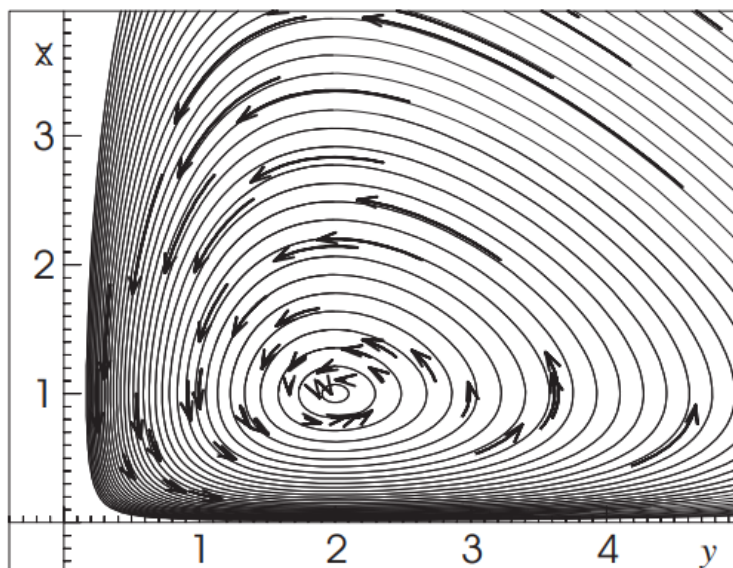
Để tìm nghiệm của phương trình này ta có thể xem y như là hàm theo x , phương trình có thể viết dưới dạng

$$\frac{dy}{dx} = \frac{y(\alpha - \beta x)}{x(\gamma y - \delta)} \quad \vee \quad \frac{(\gamma y - \delta)}{y} dy = \frac{\gamma y - \delta}{x} dx$$

Nghiệm của phương trình này cho bởi

$$\gamma y - \delta \ln y = \alpha \ln x - \beta x + C$$

trong đó C là hằng số tùy ý.



Hình 1: Nghiệm của phương trình khi $\alpha = \beta = \gamma = 1, \delta = 2$

2.3.2 Chuyển động của 2 vật bởi lực hấp dẫn

Xét 2 vật K , O có khối lượng lần lượt là m_0 , m_K . Vật O bị gắn chặt (cố định) tại gốc tọa độ Ox . Điểm K ban đầu cách O một đoạn $x_K > 0$, với vận tốc ban đầu $v_K = 0$. Ta có hệ PTVP mô tả tọa độ của K trong khoảng trục dương của Ox và vận tốc v tương ứng.

$$\begin{cases} \frac{dx}{dt} = v \\ F = ma \Rightarrow -G \frac{m_0 m_K}{x^2} = m_K \frac{dv}{dt} \end{cases} \Rightarrow \begin{cases} \frac{dx}{dt} = v \\ \frac{dv}{dt} = -G \frac{m_0}{x^2} \end{cases}$$

$$\Rightarrow \frac{dv}{dt} = -G \frac{m_0}{x^2} \Rightarrow v dv = -G \frac{m_0}{x^2} dx$$

$$\int_{v_0}^v v dv = \int_{x_0}^x -G \frac{m_0}{x^2} dx$$

$$\Rightarrow \frac{v^2}{2} - \frac{v_0^2}{2} = G m_0 \left(\frac{1}{x} - \frac{1}{x_0} \right) \Rightarrow v = -\sqrt{2Gm_0} \left(\frac{1}{x} - \frac{1}{x_0} \right)$$

2.4 Giải thuật Explicit Euler và Runge Kutta bậc 4

2.4.1 Euler Explicit

Trong toán học và khoa học máy tính, phương pháp Euler là một phương pháp số bậc một để giải các phương trình vi phân thường (ODEs) với giá trị ban đầu cho trước. Nó là phương pháp hiện (explicit) cơ bản nhất cho việc tính tích phân số của các phương trình vi phân thường và là phương pháp Runge-Kutta đơn giản nhất.

Phương pháp Euler là một phương pháp bậc một, có nghĩa là sai số cục bộ (sai số mỗi bước) tỷ lệ thuận với bình phương của kích thước bước, và sai số tổng thể (sai số tại một thời điểm nào đó) tỷ lệ thuận với kích thước bước. Phương pháp Euler thường phục vụ như là cơ sở để xây dựng các phương pháp phức tạp hơn, ví dụ như, phương pháp Dự đoán- Hiệu chỉnh.

Xem xét bài toán tính toán hình dạng của một đường cong chưa biết bắt đầu tại một điểm cho trước và thỏa mãn một phương trình vi phân nào đó. Ở đây, phương trình vi phân có thể được coi như là một công thức mà nhờ nó độ dốc của đường tiếp tuyến với đường cong có thể được tính toán tại điểm bất kỳ nào trên đường cong, một khi vị trí của điểm đó đã được tính toán.

Ý tưởng là trong khi đường cong ban đầu chưa biết, điểm khởi đầu của nó được biểu thị bởi A_0 , là đã biết (xem hình phía trên bên phải). Thì, từ phương trình vi phân, độ dốc của đường cong tại A_0 có thể được tính toán, và vì vậy, có thể tìm được đường tiếp tuyến.

Đi một bước nhỏ dọc theo đường tiếp tuyến đến một điểm A_1 . Dọc bước nhỏ này, độ dốc không thay đổi quá nhiều, vì vậy A_1 gần với đường cong.

Phương trình vi phân bậc nhất:

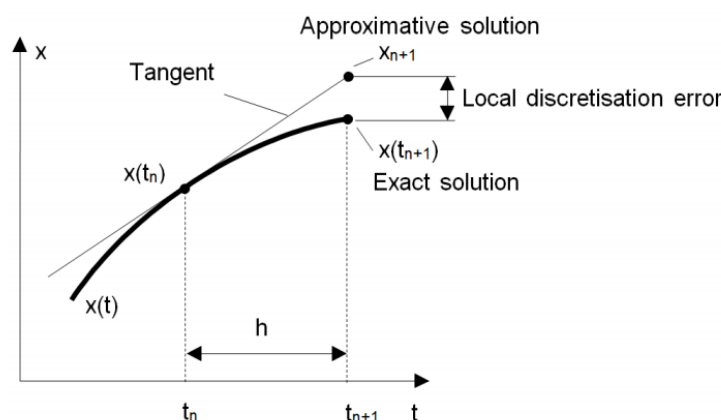
$$\begin{cases} \dot{x} = f(t, x), \\ x(t_0) = x_0 \end{cases}$$

Phân tích Taylor:

$$\begin{aligned} x(t+h) &= x(t) + \dot{x}(t)h + O(h^2), \\ &= x(t) + f(t, x(t))h + O(h^2) \end{aligned}$$

Công thức Euler tiến:

$$\begin{aligned} x_{n+1} &= x_n + f(t_n, x_n)h, \\ \text{Where : } x_n &:= x(t_n) \quad n = 0, 1, 2, \dots \end{aligned}$$



Hình 2: Ý nghĩa hình học của Euler tiến

Nếu chúng ta coi A_1 vẫn còn nằm trên đường cong, cùng một lý luận như đối với điểm A_0 có thể tính được độ dốc và tiếp tuyến của đường cong tại A_1 . Sau vài bước, một đường cong đa giác $A_0A_1A_2A_3 \dots$ được tìm ra. Nói chung, đường cong này không phân kỳ quá xa khỏi đường cong chưa biết ban đầu, và sai số giữa hai đường cong có thể được làm nhỏ nếu kích thước bước là đủ nhỏ và khoảng thời gian tính toán là hữu hạn.

2.4.2 Runge Kutta bậc 4 Explicit

Trong giải tích số, các phương pháp Runge-Kutta là một họ của các phương pháp lặp ẩn (implicit) và hiện (explicit), trong đó bao gồm thường trình nổi tiếng được gọi là các phương pháp Euler, được sử dụng trong việc rời rạc hóa thời gian để tìm lời giải gần đúng cho các phương trình vi phân thường.

Thành viên được biết đến rộng rãi nhất của họ Runge-Kutta là "RK4", "phương pháp Runge-Kutta cổ điển" hoặc đơn giản là "phương pháp Runge-Kutta".

Phương pháp Explicit Runge Kutta bậc 4: Gồm 5 bước tính toán

Bước 1: Tính toán độ dốc tại x_n

$$k_1 = f(t_n, x_n)$$

Bước 2: tính toán độ dốc tại điểm giữa từ k_1

$$k_2 = f(t_n + \frac{1}{2}h, x_n + \frac{1}{2}hk_1)$$

Bước 3: tính toán độ dốc tại điểm giữa từ k_2

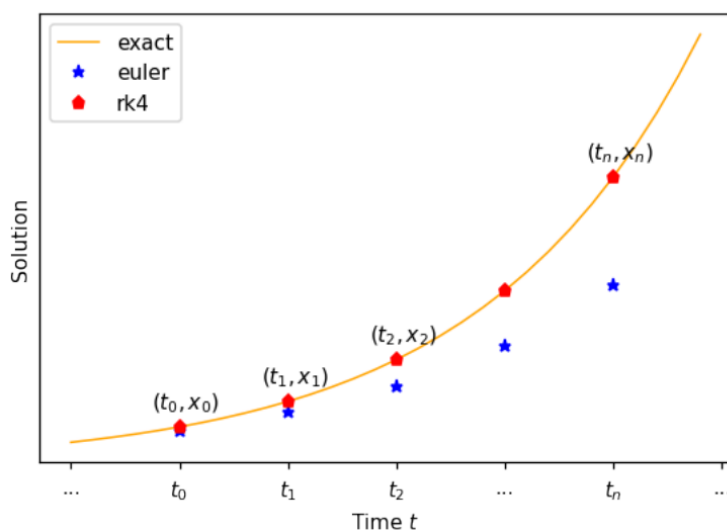
$$k_3 = f(t_n + \frac{1}{2}h, x_n + \frac{1}{2}hk_2)$$

Bước 4: tính toán độ dốc tại điểm cuối từ k_2

$$k_4 = f(t_n + h, x_n + hk_2)$$

Bước 5: Tính x_{n+1}

$$x_{n+1} = x_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$



Hình 3: Chuyển tiếp Euler và Runge Kutta bậc 4

2.5 Hiện thực hai ví dụ ở phần 2.3 bằng Euler Explicit và Runge Kutta bậc 4 Explicit

2.5.1 Bài toán rắn sần môi

Ta có thể diễn tả bài toán qua hệ động lực dưới đây:

$$\begin{aligned} y' &= y(\alpha - \beta x) \\ x' &= x(\gamma y - \delta) \end{aligned}$$

Khi $\alpha = \beta = \gamma = 1, \delta = 2$, ta được hệ dưới đây:

$$\begin{aligned} y' &= y(1 - x) \\ x' &= x(y - 2) \end{aligned}$$

• Phương pháp Euler Explicit:

Giả sử tại t_0 , $x = y = 2$ và $h = 0.5$, ta có:

• Phương pháp Runge Kutta bậc 4 Explicit:

Giả sử tại t_0 , $x = y = 2$ và $h = 0.5$, ta có:

```

data = [2, 2]
h = 0.5
def df(data):
    dx = data[0]*(data[1] - 2)
    dy = data[1]*(1 - data[0])
    return [dx, dy]
def t_h(data):
    x_t_h = data[0] + h*df(data)[0]
    y_t_h = data[1] + h*df(data)[1]
    data = [x_t_h, y_t_h]
    return data

for i in range(5):
    print('h = ' + str(h + i*0.5))
    data = t_h(data)
    print(data)

```

```

###\euler.py'
h = 0.5
[2.0, 1.0]
h = 1.0
[1.0, 0.5]
h = 1.5
[0.25, 0.5]
h = 2.0
[0.0625, 0.6875]
h = 2.5
[0.021484375, 1.009765625]
PS E:\PROJECTS\python\###>

```

Hình 4: Sử dụng Euler Explicit bằng Python để hiện thực bài toán

```

h = 1
data = [2, 2]
def f(t_n, data):
    dx = data[0]*(data[1] - 2)
    dy = data[1]*(1 - data[0])
    return [dx, dy]
k1 = f(0, data)
data_temp = list(data)
for i in range(len(data_temp)):
    data_temp[i] += k1[i]*h/2
k2 = f(h/2, data_temp)
print(len(data_temp))
data_temp = list(data)
for i in range(len(data_temp)):
    data_temp[i] += k2[i]*h/2
k3 = f(h, data_temp)
data_temp = list(data)
for i in range(len(data_temp)):
    data_temp[i] += k3[i]*h
k4 = f(h, data_temp)
for i in range(len(data)):
    data[i] += h*(k1[i] + 2*k2[i] + 2*k3[i] + k4[i])
print(data)

```

```

rs\tranl\.vscode\extensions\ms-python.python-202
###\kutta.py'
h = 0.5
[1.6390584309895835, 1.3023478190104167]
h = 1.0
[1.0880144982013262, 1.0916462414437622]
h = 1.5
[0.6922427664403061, 1.1646090610241102]
h = 2.0
[0.4846965265391469, 1.4394364171976979]
h = 2.5
[0.40866414089946673, 1.9072947664329933]
PS E:\PROJECTS\python\###>

```

Hình 5: Sử dụng RK4 Explicit bằng Python để hiện thực bài toán

2.5.2 Bài toán chuyển động của 2 vật bởi lực hấp dẫn

Ta có thể diễn tả bài toán qua hệ động lực dưới đây:

$$\begin{aligned}x' &= v \\v' &= -G \frac{m_0}{x^2}\end{aligned}$$

• Phương pháp Euler Explicit:

Giả sử tại $m_0 = 1$ và $h = 1$, ta có:

• Phương pháp Runge Kutta bậc 4 Explicit:

Giả sử tại $m_0 = 1$ và $h = 1$, ta có:

3 Bài 2: Mô hình hệ động lực đối với nồng độ khí 2 trong nhà kính

3.1 Sự trao đổi khí 2

Để tổng quát hóa bài toán để có thể áp dụng mô hình trong thực tế, mô hình nhà kính với màn chắn nhiệt (thermal screen) được xét trong đề tài này. Màn chắn nhiệt hay còn gọi là tấm chắn nhiệt được làm từ nhiều loại vật liệu khác nhau như kim loại hoặc nhựa dẻo có tính đàn hồi. Màn chắn nhiệt được sử dụng để bảo vệ cây trồng khỏi thiệt hại gây ra bởi ánh nắng trực tiếp

```

1 G = 6.67e-11
2 m = 1
3 data = [2, 2]
4 h = 1
5 def df(data):
6     dx = data[1]
7     dv = -G*m/(data[0]**2)
8     return [dx, dv]
9 def t_h(data):
10    x_t_h = data[0] + h*df(data)[0]
11    v_t_h = data[1] + h*df(data)[1]
12    data = [x_t_h, v_t_h]
13    return data
14 for i in range(5):
15     print('h = ' + str(h + i))
16     data = t_h(data)
17     print(data)

```

```

MH\euler.py'
h = 1
[4, 1.999999999983325]
h = 2
[5.999999999983325, 1.9999999999791562]
h = 3
[7.999999999962482, 1.9999999999773835]
h = 4
[9.999999999939785, 1.9999999999762612]
h = 5
[11.999999999916846, 1.9999999999755942]
PS E:\PROJECTS\python\MH>

```

Hình 6: Sử dụng Euler Explicit bằng Python để hiện thực bài toán

```

1 G = 6.67e-11
2 m = 1
3 data = [2, 2]
4 h = 1
5 def f(t_n, data):
6     dx = data[1]
7     dy = -G*m/(data[0]**2)
8     return [dx, dy]
9 def kutta(h):
10    k1 = f(0, data)
11    data_temp = list(data)
12    for i in range(len(data_temp)):
13        data_temp[i] += k1[i]*h/2
14    k2 = f(h/2, data_temp)
15    data_temp = list(data)
16    for i in range(len(data_temp)):
17        data_temp[i] += k2[i]*h/2
18    k3 = f(h, data_temp)
19    data_temp = list(data)
20    for i in range(len(data_temp)):
21        data_temp[i] += k3[i]*h
22    k4 = f(h, data_temp)
23    for i in range(len(data)):
24        data[i] += h/6*(k1[i] + 2*k2[i] + 2*k3[i] + k4[i])

```

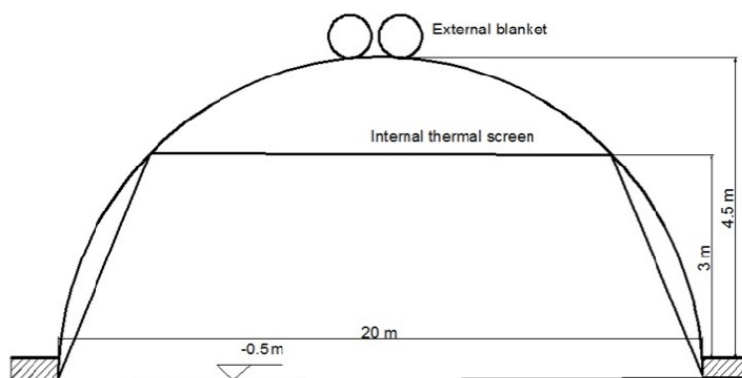
```

MH\kutta.py'
h = 1
[3.9999999999947584, 1.9999999999915854]
h = 2
[5.999999999984752, 1.999999999988832]
h = 3
[7.9999999999727915, 1.9999999999874132]
h = 4
[9.999999999959757, 1.9999999999865794]
h = 5
[11.999999999946841, 1.999999999986236]
PS E:\PROJECTS\python\MH>

```

Hình 7: Sử dụng RK4 Explicit bằng Python để hiện thực bài toán

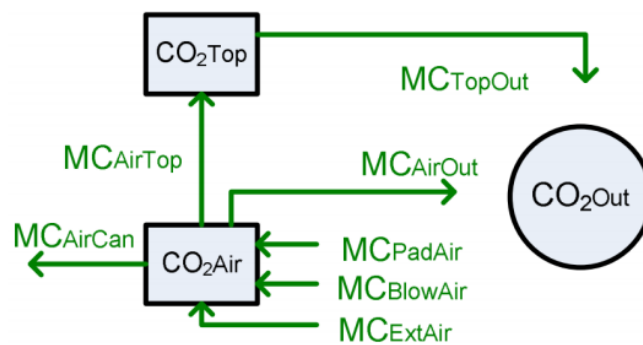
từ mặt trời cũng như chống rét cho cây trồng vào mùa đông ở những nơi có khí hậu ôn đới. Màn



Hình 8: Màn chắn nhiệt có tác dụng che chắn cho cây, điều hòa tiểu khí hậu

chắn nhiệt chia nhà kính thành hai gian khác nhau gồm gian nhà kính dưới màn chắn nhiệt và gian nhà kính phía trên màn chắn nhiệt. Vì gian trên thường hẹp hơn gian dưới nhà kính (xem Hình 8) nên dẫn đến nồng độ khí 2 trong không khí ở gian trên và gian dưới nhà kính cũng khác nhau. Sơ đồ tóm tắt sự lưu thông của lượng khí 2 trong nhà kính khi đó được thể hiện ở Hình 9.

Đối với gian dưới của nhà kính, lượng khí 2 chủ yếu được đưa vào từ các nguồn như luồng gió tự nhiên thông qua hệ thống thông gió và thoát ra ngoài bởi hệ thống quạt (xem Hình 10 và Hình 11). Ngoài ra, lượng khí 2 ở gian này cũng nhận được bởi các máy sưởi không khí (xem Hình 12) trong quá trình đốt nóng tạo nhiệt lượng và bởi bên thứ ba chuyên cung cấp khí 2. Một phần lượng khí 2 ở gian dưới nhà kính cũng thất thoát lên gian trên nhà kính dưới sự điều hướng của sự chênh lệch nhiệt độ và mật độ không khí giữa hai gian cũng như một lượng lớn khí 2 cũng sẽ được hấp thụ vào trong cây trồng để thực hiện quá trình quang hợp. Đối với gian trên của nhà kính, lượng khí 2 chủ yếu nhận từ sự trao đổi 2 với gian dưới và có thể thoát ra bên ngoài thông qua các ô thông gió ở trên mái nhà kính (nếu có) như ở Hình 15.



Hình 9: Dòng chuyển động của khí 2 bên trong và bên ngoài nhà kính

3.2 Mô hình Hệ động lực và giả thiết

Dựa trên sơ đồ quan sát được ở Hình 9, sự thay đổi của nồng độ khí 2 ở gian dưới và gian trên bên trong nhà kính được biểu diễn qua hệ gồm hai phương trình sau đây.

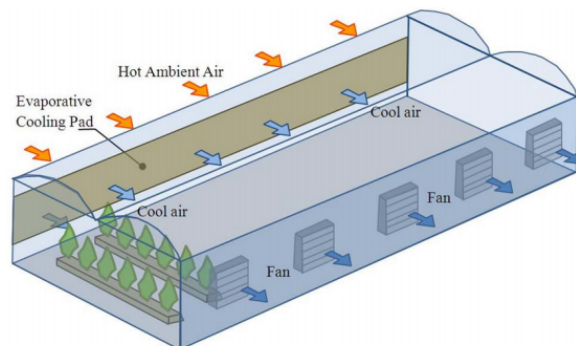
$$\begin{aligned} \text{cap}_{2_{Air}} \dot{2}_{Air} = & MC_{BlowAir} + MC_{ExtAir} + MC_{PadAir} \\ & - MC_{AirCan} - MC_{AirTop} - MC_{AirOut}, \end{aligned} \quad (1)$$

$$\text{cap}_{2_{Top}} \dot{2}_{Top} = MC_{AirTop} - MC_{TopOut}. \quad (2)$$

Trong hệ phương trình vi phân này, một số giả thiết đã được xét đến như lượng khí 2 trong không khí ở gian dưới và trên của nhà kính không bị ảnh hưởng bởi nguồn nào khác ngoại trừ những nguồn đã được thể hiện trong sơ đồ ở hình 9. Hơn nữa, nhà kính là một môi trường hoàn hảo theo nghĩa nồng độ 2 là phân bố đều nhau ở gian dưới và ở gian trên. Các kí hiệu cap_A , 2_A , $\dot{2}_A$ và MC_{AB} lần lượt là khả năng chứa khí 2 trong A (m), nồng độ khí 2 trong A (mg m^{-3}), tốc độ thay đổi nồng độ khí 2 trong A ($\text{mg m}^{-3} \text{s}^{-1}$) và lưu lượng khí 2 đi từ A vào B ($\text{mg m}^{-2} \text{s}^{-1}$), trong đó Air và Top đại diện cho gian dưới và gian trên, Blow đại diện cho máy sưởi, Ext đại diện cho bên thứ ba cung cấp khí 2, Pad đại diện cho hệ thống thông gió, Can đại diện cho tán lá cây trồng và Out đại diện cho không gian bên ngoài nhà kính.

Dưới đây là source code Python dùng để tính hệ động lực đã được mô tả ở trên:

```
1 # Caculating the differential of Co2 Air and Co2 Top
2 # =====> return array includes 2 elements [dx_Co2Air, dx_Co2Top]
```



Hình 10: Dòng chuyển động của khí 2 thông qua hệ thống gió và quạt



Hình 11: Hệ thống gió (trái) và quạt (phải)



Hình 12: Hệ thống tưới

```

3 def derivative(CO2_Air, CO2_Top):
4     global temp
5     temp -= 1
6     mc_air_top = MC_AirTop(CO2_Air, CO2_Top)
7     dCO2Air = (MC_BlowAir() + MC_ExtAir() + MC_PadAir(CO2_Air))

```

```
8         - MC_AirCan(CO2_Air) - mc_air_top - MC_AirOut(CO2_Air)) / CAP_Co2Air
9     dCO2Top = (mc_air_top - MC_TopOut(CO2_Top)) / CAP_Co2Top
10     return [dCO2Air, dCO2Top]
```

Hàm *derivative()* nhận vào hai đối số là giá trị của 2_{Air} và 2_{Top} tại thời điểm t , hàm sẽ tính đạo hàm của nồng độ khí 2 ở gian dưới **dCO2Air** và gian trên bên trong nhà kính **dCO2Top** và trả về một list bao gồm hai giá trị đạo hàm vừa tính được.

3.3 Phân tích các thông số của hệ động lực

Hệ động lực dùng để tính toán sự thay đổi khí 2 trong nhà kính phụ thuộc vào các công thức MC_{AB} .

Trước hết, xét lượng 2 đi từ máy sưởi vào trong gian dưới nhà kính như công thức sau.

$$MC_{BlowAir} = \frac{\eta_{Heat2} U_{Blow} P_{Blow}}{A_{Flr}} \quad (3)$$

Trong đó, η_{Heat2} là lượng khí 2 sinh ra khi 1 Joule nhiệt lượng (cảm nhận được) được sinh ra bởi máy sưởi ($mg \{2\} J^{-1}$). Tham số U_{Blow} thể hiện mức cho phép lượng khí 2 sinh ra bởi máy sưởi đi vào nhà kính có thể điều chỉnh được trong khoảng $[0, 1]$ và không có đơn vị. Hệ số P_{Blow} là khả năng sinh ra 2 của máy sưởi (W) và A_{Flr} là diện tích nhà kính (m^2).

Dưới đây là source code Python dùng để tính $MC_{BlowAir}$:

```
1 '''Calculate MC_BlowAir'''
2 # Luong co2 di tu may suoi vao ben trong gian duoi nha kinh
3 def MC_BlowAir():
4     blow_air = (eta_heatCO2 * U_Blow * P_Blow) / A_Flr
5     return blow_air
```

Hàm $MC_{BlowAir}()$ sẽ tính giá trị của $MC_{BlowAir}$ dựa trên các tham số đã được xác định, từ đó trả về kết quả.

Tương tự, lượng khí 2 được bơm vào nhà kính bởi bên thứ ba chuyên cung cấp khí 2 được cho bởi công thức sau.

$$MC_{ExtAir} = \frac{U_{Ext2} \phi_{Ext2}}{A_{Flr}} \quad (4)$$

Các kí hiệu U_{Ext2} và ϕ_{Ext2} lần lượt là tham số điều chỉnh tốc độ bơm khí 2 vào trong nhà kính (không có đơn vị) và khả năng bơm 2 của bên thứ ba ($mg s^{-1}$).

Đây là phần hiện thực của công thức trên:

```
1 '''Calculate MC_ExtAir'''
2 # Luong khi CO2 duoc bom vao nha kinh boi ben thu ba chuyen cung cap khi CO2
3 def MC_ExtAir():
4     ext_air = (U_Ext * fi_Ext) / A_Flr
5     if temp == 0:
6         MC_ExtAirData.append(ext_air)
7     return ext_air
```

Hàm $MC_{ExtAir}()$ nhận các tham số tương ứng đã xác định từ trước và trả về giá trị của MC_{ExtAir} .

Mặt khác, lượng khí 2 đi vào nhà kính thông qua hệ thống thông gió dựa trên sự chênh lệch của nồng độ khí 2 bên trong và bên ngoài nhà kính và khả năng cho dòng không khí đi qua của tấm thông gió như hình 11. Hơn nữa, khả năng cho dòng không khí đi qua của tấm thông gió có thể điều chỉnh được. Công thức sau dùng để tính MC_{PadAir} .

$$MC_{PadAir} = f_{Pad}(2_{Out} - 2_{Air}) = \frac{U_{Pad}\phi_{Pad}}{A_{Flr}}(2_{Out} - 2_{Air}) \quad (5)$$

Tốc độ của đi qua tấm thông gió f_{Pad} ($m s^{-1}$) được tính bởi tích của tham số U_{Pad} , thể hiện mức cho phép lượng khí 2 đi qua tấm thông gió điều chỉnh được trong khoảng $[0, 1]$ và không có đơn vị, và ϕ_{Pad} , là khả năng cho phép khí 2 đi qua của tấm thông gió ($m^3 s^{-1}$) chia cho diện tích nhà kính.

Dưới đây là phần hiện thực phương trình MC_{PadAir} :

```
1 '''Calculate CO2_PadAir'''
2 # Luong khi CO2 di vao nha kinh thong qua he thong thong gio
3 def MC_PadAir(CO2_Air):
4     f_Pad = U_Pad * fi_Pad / A_Flr # Toc do CO2 di qua tam thong gio (m^3 s^{-1})
5     pad_air = f_Pad*(CO2_Out - CO2_Air)
6     return pad_air
```

Hàm $MC_{PadAir}()$ nhận vào đối số là giá trị của 2_{Air} tại thời điểm t , hàm sẽ tính toán giá trị của f_{Pad} , sau đó tính và trả về giá trị MC_{PadAir} dựa vào giá trị của f_{Pad} vừa tính được và các tham số khác đã được xác định.

Đối với lượng khí 2 đi từ gian dưới lên gian trên nhà kính phức tạp hơn và phụ thuộc vào độ chênh lệch nhiệt độ và độ chênh lệch mật độ của hai gian nhà kính thông qua màn chắn nhiệt.

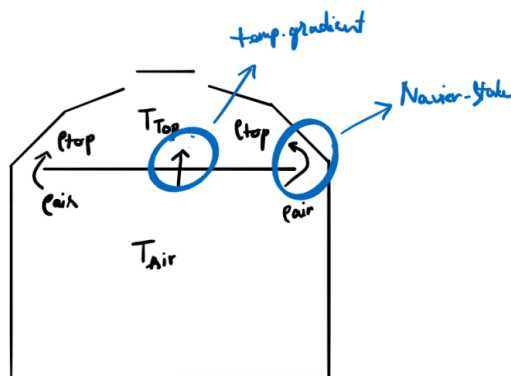
$$MC_{AirTop} = f_{ThScr}(2_{Air} - 2_{Top}) \quad (6)$$

trong đó tốc độ lưu thông khí 2 qua màn chắn nhiệt f_{ThScr} ($m s^{-1}$) là tổng của hai tốc độ gồm tốc độ thẩm thấu qua màn chắn nhiệt và tốc độ tại những nơi không bị chắn bởi màn chắn nhiệt (xem Hình 13).

$$f_{ThScr} = U_{ThScr}K_{ThScr}|T_{Air} - T_{Top}|^{\frac{2}{3}} + (1 - U_{ThScr}) \left[\frac{g(1 - U_{ThScr})}{2\rho_{Air}^{Mean}} |\rho_{Air} - \rho_{Top}| \right]^{\frac{1}{2}} \quad (7)$$

Ở những nơi có màn chắn nhiệt với độ phủ $U_{ThScr} \in [0, 1]$ (không có đơn vị), tốc độ phụ thuộc vào sự chênh lệch giữa nhiệt độ bên trên T_{Top} (K) và bên dưới T_{Air} (K) màn chắn và khả năng cho không khí thẩm thấu của màn chắn là K_{ThScr} ($m K^{-\frac{2}{3}} s^{-1}$). Ở những nơi không có màn chắn nhiệt với độ phủ $1 - U_{ThScr}$, tốc độ được cho bởi một phương trình Navier-Stokes phụ thuộc vào sự chênh lệch của mật độ không khí $T\rho_{Air}$ và $T\rho_{Top}$ cùng tính bằng đơn vị ($kg m^{-3}$). Hệ số $\frac{2}{3}$ trong công thức (7) đến từ thực nghiệm trong công trình của [Bal89]. Trong công trình đó, tác giả đã sử dụng dữ liệu đo đạc được về tốc độ trao đổi không khí thông qua các màn chắn làm từ 12 loại chất liệu khác nhau và sử dụng chúng để huấn luyện mô hình $K_{ThScr}|T_{Air} - T_{Top}|^m$ với m là tham số điều chỉnh được để tìm ra m gần bằng 0.66 là $\frac{2}{3}$.

Hàm mô tả tính toán MC_{AirTop} và các tham số của hàm được thể hiện dưới đây:



Hình 13: Chuyển động của khí 2 qua màn chắn nhiệt

```

1 # Tốc độ lưu thông khí CO2 qua màn chắn nhiệt
2 def f_ThScr():
3     # Mật độ không khí dưới màn chắn nhiệt (kg m-3)
4     p_Air = p_Air0 * math.exp(g * M_Air * h_Elevation / (T_Air * R * 1000))
5     # Mật độ không khí trên màn chắn nhiệt (kg m-3)
6     p_Top = p_Air0 * math.exp(g * M_Air * (h_Elevation + h_Air) / (T_Top * R * 1000))
7     # Mật độ không khí trung bình
8     p_MeanAir = p_Air0 * math.exp(g * M_Air * (h_Elevation + h_Gh) / ((T_Air + T_Top) / 2 *
9         R * 1000))
10
11     thScr = U_ThScr * K_ThScr * (abs(T_Air - T_Top)) ** (2/3)
12     notThScr = (1 - U_ThScr) * (g * (1 - U_ThScr) / (2 * p_MeanAir) * abs(p_Air - p_Top)
13         ) ** 0.5
14     return thScr + notThScr
15
16 # Lượng khí CO2 đi từ gian dưới lên trên nhà kính
17 def MC_AirTop(CO2_Air, CO2_Top):
18     air_top = f_ThScr() * (CO2_Air - CO2_Top)
19     if temp == 0:
20         MC_AirTopData.append(air_top)
21     return air_top

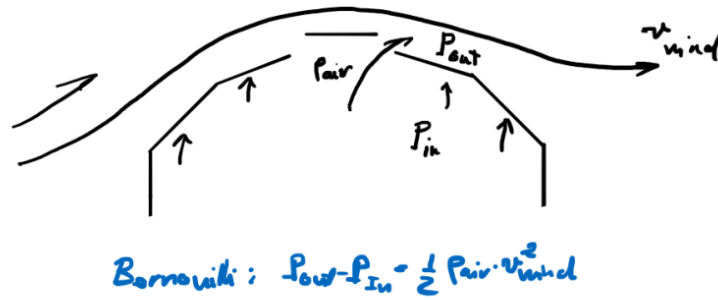
```

Hàm $f_{ThScr}()$ dùng để tính và trả về giá trị của f_{ThScr} bằng cách tính tổng của hai tốc độ gồm tốc độ thẩm thấu qua màn chắn nhiệt (biến **thScr**) và tốc độ tại những nơi không bị chắn bởi màn chắn nhiệt (biến **notThScr**). Dựa vào kết quả của hàm $f_{ThScr}()$, giá trị của 2_{Air} và 2_{Top} tại thời điểm t và các giá trị tham số xác định khác, hàm $MC_{AirTop}()$ sẽ tính ra giá trị của MC_{AirTop} và trả về kết quả tương ứng.

Tương tự, để biểu diễn lượng khí 2 từ bên trong ra bên ngoài nhà kính theo hai hướng từ gian dưới và từ gian trên qua các ô thông gió ta sử dụng các công thức như dưới đây.

$$MC_{AirOut} = (f_{VentSide} + f_{VentForced})(2_{Air} - 2_{Out}) \quad (8)$$

Trong đó, $f_{VentSide}$ là tốc độ gió của hệ thống quạt trên tường bao xung quanh nhà kính và $f_{VentForced}$ là tốc độ gió từ hệ thống quạt bên trong nhà kính, cả hai tốc độ đều được đo với đơn vị ($m s^{-1}$). Tuy nhiên trong trường hợp này, nguyên lý Bernoulli đóng vai trò quan trọng biểu diễn bởi độ chênh lệch áp suất từ phía ngoài nhà kính gây ra bởi luồng gió tự nhiên và áp suất từ phía trong nhà kính gây ra do luồng không khí bên trong (xem Hình 14), hiệu ứng Stack hay còn gọi là hiệu ứng Chimney cũng cần được xét đến (xem Hình 15). Hiệu ứng Stack là hiệu ứng mà vào mùa đông, dòng không khí lạnh từ bên ngoài vào bên trong nhà kính và bị làm nóng dần bởi hệ thống sưởi và có xu hướng đi lên phía trên mái nhà kính và thoát ra trở lại bên ngoài, vào mùa hè thì theo chiều ngược lại.



Hình 14: Dòng không khí qua ô mở mái nhà kính

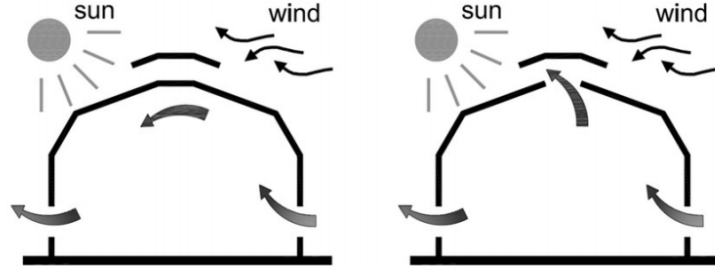
Để tổng quát hóa mô hình cho nhiều loại nhà kính khác nhau, công thức tổng quát dưới đây $f_{VentRoofSide}$ (ms^{-1}) được dùng để thiết lập công thức cho $f_{VentSide}$.

$$f_{VentRoofSide} = \frac{C_d}{A_{Flr}} \left[\frac{U_{Roof}^2 U_{Side}^2 A_{Roof}^2 A_{Side}^2}{U_{Roof}^2 A_{Roof}^2 + U_{Side}^2 A_{Side}^2} \cdot \frac{2gh_{SideRoof}(T_{Air} - T_{Out})}{T_{Air}^{Mean}} + \left(\frac{U_{Roof} A_{Roof} + U_{Side} A_{Side}}{2} \right)^2 C_w v_{Wind}^2 \right]^{\frac{1}{2}} \quad (9)$$

Công thức (10) là tổng của hai thành phần nhân với tỷ lệ giữa hệ số lưu lượng gió C_d không có đơn vị và diện tích nhà kính A_{Flr} (m^2). Thành phần thứ nhất phụ thuộc vào độ chênh lệch nhiệt độ giữa bên ngoài và bên trong nhà kính (ở gian dưới màn chắn nhiệt) đại diện cho hiệu ứng Stack khi diện tích ô thông gió trên mái A_{Roof} (m^2) là khác không. Thành phần thứ hai cho bởi độ chênh lệch áp suất bên trong và bên ngoài nhà kính và được tính bằng tổng diện tích các nơi thông gió trên nhà kính chia hai nhân vận tốc gió tự nhiên v_{Wind} ($m s^{-1}$) và hệ số áp suất gió C_w không có đơn vị. Các hệ số C_d và C_w là các hệ số lý thuyết phụ thuộc vào cấu trúc và hình dáng của nhà kính và có thể ước lượng được thông qua các số liệu đo đạc được trên thực nghiệm.

Ngoài ra, lưới chắn côn trùng gây hại trên các nơi thông gió và hệ số rò rỉ của nhà kính cũng sẽ được xét trong đề tài này. Khi có lưới chắn côn trùng, tốc độ chuyển động của các luồng không khí qua các nơi thông gió sẽ giảm xuống với hệ số

$$\eta_{InsScr} = \zeta_{InsScr}(2 - \zeta_{InsScr}), \quad (10)$$



Hình 15: Không có hiệu ứng Stack (trái) và có hiệu ứng stack (phải)

trong đó ζ_{InsScr} không có đơn vị là độ rở của lưới, nghĩa là tỷ lệ diện tích các lỗ trên lưới và tổng diện tích lưới chắn côn trùng. Với hệ số rò rỉ $cleakage$ không có đơn vị, tốc độ trao đổi không khí thường được xấp xỉ khoảng 50% tốc độ

$$f_{leakage} = \begin{cases} 0.25 \cdot cleakage, & v_{Wind} < 0.25, \\ v_{Wind} \cdot cleakage, & v_{Wind} \geq 0.25 \end{cases} \quad (11)$$

Một cách ngầm hiểu, giả thiết về phân bố đều của sự rò rỉ của nhà kính đã được sử dụng. Gọi η_{Side_Thr} là ngưỡng Stack, nghĩa là nếu η_{Side} là tỷ lệ giữa diện tích các nơi thông gió trên tường bao quanh nhà kính và diện tích của tất cả các nơi thông gió trên nhà kính và η_{Side} vượt ngưỡng Stack thì hiệu ứng Stack không xảy ra và ngược lại. Khi đó, $f_{VentSide}$ được cho bởi công thức sau.

$$f_{VentSide} = \begin{cases} \eta_{InsScr} f''_{VentSide} + 0.5 f_{leakage}, & \eta_{Side} \geq \eta_{Side_Thr}, \\ \eta_{InsScr} \left[U_{ThScr} f''_{VentSide} + (1 - U_{ThScr}) f_{VentRoofSide} \eta_{Side} \right] + 0.5 f_{leakage}, & \eta_{Side} < \eta_{Side_Thr} \end{cases} \quad (12)$$

Trong đó, $f''_{VentSide}$ là $f_{VentRoofSide}$ tính tại $A_{Roof} = 0$. Lưu ý, ở những nơi phủ bởi màn chắn nhiệt, hiệu ứng Stack cũng không xảy ra.

Tốc độ $f_{VentForced}$ bởi hệ thống quạt gió bên trong nhà kính được tính bởi công thức sau.

$$f_{VentForced} = \frac{\eta_{InsScr} U_{VentForced} \phi_{VentForced}}{A_{Flr}}. \quad (13)$$

Kí hiệu $U_{VentForced}$ không có đơn vị thể hiện sự điều chỉnh tốc độ gió mà hệ thống có thể $\phi_{VentForced} (m^3 s^{-1})$ và có giá trị trong khoảng $[0, 1]$.

Phần hiện thực hàm tính MC_{AirOut} và các giá trị liên quan tới nó được thể hiện ở đây:

```

1 def f_VentRoofSide(A_Roof):
2     p1 = (U_Roof*U_Side*A_Roof*A_Side) ** 2
3     p2 = (U_Roof * A_Roof) ** 2 + (U_Side * A_Side)**2
4     p3 = 2*g*h_Side_Roof*(T_Air - T_Out)/T_Mean
5     p4 = ((U_Roof*A_Roof + U_Side*A_Side)/2) ** 2
6 
```

```
7     if A_Roof == 0 and A_Side == 0:
8         return 0
9     return (cd/A_Flr)*((p1/p2*p3 + p4*cw*(v_wind**2)) ** 0.5)
10
11 # toc do gio cua he thong quat tren tuong
12 def f_VentSide():
13     if U_Roof == 0:
14         eta_Roof = 1
15         eta_Side = 1
16     else:
17         eta_Roof = A_Roof * U_Roof / (A_Side * U_Side + A_Roof * U_Roof)
18         eta_Side = (A_Side * U_Side) / (A_Side * U_Side + A_Roof * U_Roof)
19     if eta_Roof < eta_Side_Thr:
20         return zeta_InsScr() * (U_ThScr*f_VentRoofSide(0)+(1-U_ThScr) *
21             f_VentRoofSide(A_Roof) *eta_Side) + 0.5*f_Leakage()
22     else:
23         return zeta_InsScr()*f_VentRoofSide(0)+0.5*f_Leakage()
24
25 # He so giam do luoi chan con trung
26 def zeta_InsScr():
27     return zeta_Insect * (2 - zeta_Insect)
28
29 # Cuong do khong khi ro ri
30 def f_Leakage():
31     if v_wind < 0.25:
32         return 0.25*c_leak
33     else:
34         return v_wind*c_leak
35
36 # toc do boi he thong quat gio trong nha kinh
37 def f_VentForced():
38     return zeta_InsScr()*U_VentForced*fi_VentForced/A_Flr
39
40 # Calculate MC_AirOut
41 def MC_AirOut(CO2_Air):
42     air_out = (f_VentSide() + f_VentForced()) * (CO2_Air - CO2_Out)
43     if temp == 0:
44         MC_AirOutData.append(air_out)
45     return air_out
```

Trong các hàm ở trên, hàm $f_VentRoofSide()$ nhận đối số là diện tích của thông gió trên mái, cùng với các giá trị tham số cho trước để tính chính xác và trả về giá trị của $f_{VentRoofSide}$, hàm $zeta_InsScr()$ dựa vào giá trị của ζ_{InsScr} để tính toán và trả về giá trị của η_{InsScr} , hàm $f_Leakage()$ trả về giá trị cường độ không khí rò rỉ $f_{Leakage}$. Giá trị trả về của ba hàm này được sử dụng bởi hàm $f_VentSide()$ để tính toán và trả về giá trị của $f_{VentSide}$. Tương tự, hàm $f_VentForced()$ sử dụng giá trị trả về của hàm $zeta_InsScr()$ để tính toán ra kết quả của $f_{VentForced}$. Dựa vào kết quả trả về của hai hàm $f_VentSide()$ và hàm $f_VentForced()$, giá trị của MC_{AirOut} được tính toán như trong phương trình (8), cụ thể là được thực hiện thông qua hàm $MC_AirOut()$.

Tương tự như MC_{AirOut} , lượng khí 2 đi từ gian trên nhà kính ra ngoài thông qua ô mở trên

mái nhà kính được tính bởi công thức

$$MC_{TopOut} = f_{VentRoof}(2_{Top} - 2_{Out}). \quad (14)$$

Trong đó, $f_{VentRoof}$ là tốc độ luồng không khí đi qua ô mở mái nhà kính và được cho bởi công thức sau.

$$f_{VentRoof} = \begin{cases} \eta_{InsScr} f''_{VentRoof} + 0.5 f_{leakage}, & \eta_{Roof} \geq \eta_{Roof_Thr}, \\ \eta_{InsScr} \left[U_{ThScr} f''_{VentRoof} + (1 - U_{ThScr}) f_{VentRoofSide} \eta_{Side} \right] + 0.5 f_{leakage}, & \eta_{Roof} < \eta_{Roof_Thr} \end{cases} \quad (15)$$

Tuy nhiên, khác với $f_{VentSide}$ trong công thức (12), khi tỷ lệ η_{Roof} giữa diện tích ô mở trên mái nhà kính và tổng diện tích các ô thông gió trên nhà kính vượt ngưỡng Stack là η_{Roof_Thr} , nghĩa là hiệu ứng Stack không xảy ra, ta không thể sử dụng công thức $f_{VentRoofSide}$ trong (9) với $A_{Side} = 0$ để tính $f''_{VentRoof}$ mà phải sử dụng công thức sau

$$f''_{VentRoof} = \frac{C_d U_{Roof} A_{Roof}}{2 A_{Flr}} \left[\frac{gh_{Roof}(T_{Air} - T_{Out})}{2 T_{Air}^{Mean}} + C_w v_{Wind}^2 \right]^{\frac{1}{2}}. \quad (16)$$

Dựa vào các phương trình được mô tả trên, ta có thể hiện thực hàm tính MC_{TopOut} và các giá trị liên quan qua source code python dưới đây:

```
1 # toc do khong khi di qua o mo mai nha kinh
2 def f_VentRoof():
3     if U_Roof == 0:
4         eta_Roof = 1
5     else:
6         eta_Roof = A_Roof * U_Roof / (A_Side * U_Side + A_Roof * U_Roof)
7
8     if eta_Roof < eta_Roof_Thr:
9         return zeta_InsScr() * (U_Roof*f2_VentRoof() + (1-U_Roof) * f_VentRoofSide(A_Roof)
10            * eta_Roof)+0.5*f_Leakage()
11     else:
12         return zeta_InsScr()*f2_VentRoof()+0.5*f_Leakage()
13
14 def f2_VentRoof():
15     return (cd*U_Roof*A_Roof)/(2*A_Flr) *
16         ((g*h_Vent*(T_Air-T_Out)/(2*T_Mean)+cw*(v_wind**2)) ** 0.5)
17
18 '''Calculate MC_TopOut'''
19 def MC_TopOut(CO2_Top):
20     top_out = f_VentRoof()*(CO2_Top - CO2_Out)
21     if temp == 0:
22         MC_TopOutData.append(top_out)
23     return top_out
```

hàm $f2_VentRoof()$ nhận vào các giá trị tham số tương ứng để trả về giá trị của $f''_{VentRoof}$. Dựa vào kết quả của hàm này và các hàm phụ trợ dùng để tính phương trình (8) đã được giải thích ở trên, ta sẽ tính được giá trị của $f_{VentRoof}$. Cuối cùng là hàm chính dùng để hiện thực MC_{TopOut} , hàm này nhận vào 2 đối số là 2_{Top} và 2_{Out} tại thời điểm t, kết hợp với giá trị $f_{VentRoof}$ đã được tính ở trên.

Cuối cùng, chúng ta cần mô tả lượng khí 2 bị hấp thụ vào trong tán lá thông qua quá trình quang hợp.

$$MC_{AirCan} = M_{CH_2O} h_{C_{Buf}} (P - R). \quad (17)$$

Trong đó M_{CH_2O} là khối lượng mol CH_2O ($mg \mu mol^{-1}$), P là tốc độ quang hợp ($\mu mol \{2\} m^{-2} s^{-1}$), R là tốc độ hô hấp của cây ($\mu mol \{2\} m^{-2} s^{-1}$) và hệ số

$$h_{C_{Buf}} = \begin{cases} 0, & C_{Buf} > C_{Buf}^{Max}, \\ 1, & C_{Buf} \leq C_{Buf}^{Max} \end{cases} \quad (18)$$

thể hiện sự ngưng quá trình quang hợp khi lượng CH_2O là C_{Buf} ($mg \{CH_2O\} m^{-2}$) sinh ra đã vượt sức chứa của cây C_{Buf}^{Max} ($mg \{CH_2O\} m^{-2}$). Thông thường, tốc độ hô hấp của cây không đáng kể so với tốc độ quang hợp của cây và có thể được lược bỏ hoặc được tính vào khoảng 1% của tốc độ quang hợp của cây.

Tốc độ quang hợp ở tán cây P được mô tả bởi

$$P = \frac{J \cdot (2S_{Stom} - \Gamma)}{4(2S_{Stom} + 2\Gamma)} \quad (19)$$

Trong đó, J ($\mu mol \{e^{-}\} m^{-2} s^{-1}$) là tốc độ chuyển động của electron, 4 ($\mu mol \{e^{-}\} \mu mol^{-1} 2$) là số electron trên mỗi phân tử 2 cố định, $2S_{Stom}$ ($\mu mol \{2\} mol^{-1} \{Air\}$) và Γ ($\mu mol \{2\} mol^{-1} \{Air\}$) là điểm bù 2.

Tốc độ hô hấp R được mô tả bởi

$$R = P \cdot \frac{\Gamma}{2S_{Stom}} \quad (20)$$

Tốc độ vận chuyển electron J được tính bởi

$$J = \frac{J^{POT} + \alpha PAR_{Can} - \sqrt{(J^{POT} + \alpha PAR_{Can})^2 - 4\Theta \cdot J^{POT} \cdot \alpha PAR_{Can}}}{2\Theta} \quad (21)$$

Trong đó, J^{POT} ($\mu mol \{e^{-}\} m^{-2} s^{-1}$) là tốc độ tiềm năng của sự vận chuyển electron, PAR_{Can} ($\mu mol \{photons\} m^{-2} s^{-1}$) là PAR bị thẩm thấu, α ($\mu mol \{e^{-}\} \mu mol^{-1} photons$) là hệ số chuyển đổi từ photon thành electron và Θ ($-$) là mức độ cong của tốc độ vận chuyển electron. Tốc độ tiềm năng của sự vận chuyển electron J^{POT} phụ thuộc vào nhiệt độ

$$J^{POT} = J_{25,Can}^{MAX} \cdot e^{E_j \frac{T_{Can,K} - T_{25,K}}{R \cdot T_{Can,K} \cdot T_{25,K}}} \cdot \frac{1 + e^{\frac{S \cdot T_{25,K} - H}{R \cdot T_{25,K}}}}{1 + e^{\frac{S \cdot T_{Can,K} - H}{R \cdot T_{Can,K}}}} \quad (22)$$

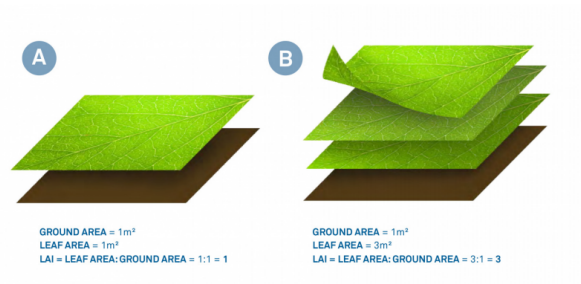
Trong đó, $J_{25,Can}^{MAX}$ ($\mu mol \{e^{-}\} m^{-2} s^{-1}$) là tốc độ vận chuyển electron tối đa ở $25^\circ C$ đối với tán cây, E_j ($J mol^{-1}$) là năng lượng kích hoạt cho J^{POT} , $T_{Can,K}$ (K) là nhiệt độ tán cây, $T_{25,K}$ (K) là nhiệt độ tương quan tại $25^\circ C$, R ($J mol^{-1} K^{-1}$) là hằng số khí, S ($J mol^{-1} K^{-1}$) là hàm trạng thái entropy và H ($J mol^{-1} K^{-1}$) là năng lượng ngừng kích hoạt.

Tốc độ vận chuyển electron tối đa ở $25^\circ C$, $J_{25,Can}^{MAX}$ được tính bằng

$$J_{25,Can}^{MAX} = LAI \cdot J_{25,Leaf}^{MAX} \quad (23)$$

Trong đó, $J_{25,Leaf}^{MAX}$ ($\mu\text{mol } \{e^-\} m^{-2} s^{-1}$) là tốc độ vận chuyển electron tối đa trong lá ở $25^\circ C$, là chỉ số diện tích lá (leaf area index -).

Chỉ số được tính bởi tổng mật độ độ lá trên một đơn vị diện tích đất trong nhà kính. Khi đó, nếu tán lá càng dày thì chỉ số càng cao (xem Hình 16).



Hình 16: Chỉ số diện tích lá

Nồng độ khí 2 bên trong khí khổng, 2_{Stom} phụ thuộc vào độ dẫn của khí khổng và mô thực vật, sức cản của lớp ranh giới, tốc độ quang hợp và sự chênh lệch giữa nồng độ CO₂ trong khí khổng và nồng độ CO₂ của không khí trong nhà kính. Tuy nhiên, nồng độ CO₂ trong khí khổng được giả định là một phần cố định của nồng độ CO₂ trong nhà kính.

$$2_{Stom} = \eta_{2Air_Stom} \cdot 2_{Air} \quad (24)$$

Trong đó, η_{2Air_Stom} (-) là hệ số chuyển đổi từ nồng độ 2 trong nhà kính. Điểm bù 2 Γ ảnh hưởng đến tốc độ quang hợp của lá và phụ thuộc vào nhiệt độ

$$\Gamma = c_{\Gamma} T_{Can} \quad (25)$$

Trong đó, c_{Γ} ($\mu mol \{2\} mol^{-1} \{Air\} K^{-1}$) xác định sự ảnh hưởng của nhiệt độ tán lá đến điểm bù CO₂.

Mối quan hệ giữa nhiệt độ tán lá và điểm bù 2 có thể dùng để tính tốc độ quang hợp của lá. Tuy nhiên, áp dụng công thức (25) với tăng tán lá dẫn đến nhiệt độ tán lá tối ưu thấp không thực tế đối với tốc độ quang hợp khi mức CO₂ và ánh sáng yếu, bởi vì dựa vào công thức 24, $J_{25,Can}^{MAX}$ lớn hơn so với $J_{25,Leaf}^{MAX}$. Để tránh điều này xảy ra, ta sẽ cho Γ phụ thuộc vào tỉ lệ giữa $J_{25,Can}^{MAX}$ và $J_{25,Leaf}^{MAX}$

$$\Gamma = \frac{J_{25,Leaf}^{MAX}}{J_{25,Can}^{MAX}} c_{\Gamma} T_{Can} + 20 c_{\Gamma} \left(1 - \frac{J_{25,Leaf}^{MAX}}{J_{25,Can}^{MAX}}\right) \quad (26)$$

Công thức (26) được sử dụng để đảm bảo rằng đối với tất cả giá trị của $J_{25,Can}^{MAX}$ ở 20°C, điểm bù được tính bằng công thức (26) sẽ bằng với điểm bù được tính bằng công thức (25).

Các công thức từ (17)-(26) được hiện thực qua source code Python sau:

```
1 # Tốc độ quang hợp
2 def P(CO2_Air):
3     return J()*(CO2_Stom(CO2_Air) - tau()) / (4*(CO2_Stom(CO2_Air) + 2*tau()))
4
5 # Tốc độ ho hấp
6 def R_(CO2_Air):
7     return P(CO2_Air) * tau() / CO2_Stom(CO2_Air)
8
9 # Nồng độ co2 trong khí khổng
10 def CO2_Stom(CO2_Air):
11     return eta_CO2AirStom * CO2_Air
12
13 # Điểm bù nồng độ co2 trong khí khổng
14 def tau():
15     T_Can = T_Air + 10
16     return cT*T_Can/LAI + 20*cT*(1-1/LAI)
17
18 # Tốc độ truyền e
19 def J():
20     sum = J_POT() + alpha*PAR_Can
21     product = J_POT() * alpha*PAR_Can
22     return (sum - (sum**2 - 4*theta*product)**0.5) / (2*theta)
23
24 # Tốc độ truyền e tiềm năng
```

```
25 def J_POT():
26     T_Can = T_Air + 10
27     factor1 = math.exp(Ej * (T_Can - T25) / (R * T_Can * T25))
28     factor2 = (1 + math.exp((S*T25 - H)/(R*T25))) / \ (1 + math.exp((S*T_Can -
        H)/(R*T_Can)))
29     return J_MAX25Can() * factor1 * factor2
30
31 # Tốc độ truyền e lớn nhất ở 25°C của tán lá
32 def J_MAX25Can():
33     return LAI * J_MAX25Leaf
34
35 '''Calculate MC_AirCan'''
36 def MC_AirCan(CO2_Air):
37     air_can = CH2O * 1 * (P(CO2_Air) - R_(CO2_Air))
38     if temp == 0:
39         MC_AirCanData.append(air_can)
40     return air_can
```

Hàm $CO2_Stom()$ nhân đối số là 2_{Air} tại thời điểm t để trả về giá trị của nồng độ 2 trong khí khổng. Tương tự, hàm $P()$ dùng để hiện thực phương trình (19), hàm $\tau()$ dùng để tính điểm bù nồng độ co_2 trong khí khổng τ , ba hàm $J()$, $J_POT()$ và $J_MAX25Can()$ lần lượt để tính tốc độ truyền electron, tốc độ truyền electron tiềm năng và lớn nhất ở $25^\circ C$ của tán lá. Hàm $MC_AirCan()$ dựa vào kết quả của các hàm hỗ trợ đã được giải thích ở trên để tính toán và trả về lượng khí 2 bị hấp thụ vào tán lá qua quá trình quang hợp.

3.4 Giá trị của các tham số và giải thích

Trong source code dùng để hiện thực mô hình hệ động lực thể hiện sự thay đổi của nồng độ khí 2 trong nhà kính, có các thông số được xác định bởi [Van11]. Cụ thể:

```
1 CAP_Co2Air = h_Air # capacity of CO2 Air
2 CAP_Co2Top = h_Gh - h_Air # capacity of CO2 Top
3 # -----
4 eta_heatCO2 = 0.057
5 U_Blow = 1 # Muc cho phép lượng co2 sinh ra bởi máy sưởi đi vào nhà kính
6 P_Blow = 0 # Công suất nhiệt của máy sưởi không khí trực tiếp (W)
7 A_Flr = 1.4*(10**4)
8 # -----
9 U_Ext = 0.5 # Tham số điều chỉnh tốc độ bơm khí vào CO2
10 fi_Ext = 7.2*(10**4) # Khả năng bơm co2 của bên thu ba
11 # -----
12 U_Pad = 1 # Giá trị của van điều khiển
13 fi_Pad = 0 # khả năng cho phép khí CO2 qua tam thông gió (m^3 s^-1)
14 CO2_Out = 668 # mg m^-3
15 # -----
16 g = 9.81 # Gia tốc trọng trường (m s^-2)
17 R = 8.314 # J mol^-1 K^-1
18 M_Air = 28.96 # g mol^-1
19 U_ThScr = 0.5 # Độ phụ của màn chắn [0,1]
20 # Khả năng cho không khí tham thấu của màn chắn
21 K_ThScr = 0.05*(10**(-3))
22 # Nhiệt độ gian trên màn chắn (K) (se được lưu từ data sheet)
23 T_Air = 0
```



```
24 T_Top = 0 # Nhiệt độ gian dưới màn chắn (K) (= T_Air + 1)
25 h_Elevation = 0 # Độ cao của nhà kính so với mực nước biển (m)
26 h_Air = 3.8 # chiều cao của compartment bên dưới nhà kính (m)
27 h_Gh = 4.2
28 p_Air0 = 1.20 # Mật độ không khí ở mực nước biển (kg m-3)
29 # -----
30 v_wind = 0 # Tốc độ gió (sẽ được lấy từ datasheet)
31 c_leak = 10**(-4) # Hệ số rò rỉ
32 zeta_Insect = 1
33 eta_Roof_Thr = 0.9 # ngưỡng Stack (ở Roof)
34 eta_Side_Thr = eta_Roof_Thr # ngưỡng Stack (ở Side)
35 U_ShScr = 1
36 C_Gh_d = 0.75
37 C_Gh_w = 0.09
38 eta_ShScrCd = 0
39 eta_ShScrCw = 0
40 cw = C_Gh_w*(1 - eta_ShScrCw*U_ShScr)
41 cd = C_Gh_d*(1 - eta_ShScrCd*U_ShScr)
42 U_Side = 1
43 A_Side = 0*A_Flr
44 U_Roof = 1
45 A_Roof = 0.1*A_Flr # diện tích ở thông gió trên mái
46 # S(thông gió trên tường)/S(tổng tất cả các thông gió)
47 eta_Side = (A_Side*U_Side)/(A_Side*U_Side + A_Roof*U_Roof)
48 # The vertical distance between mid-points of side wall and roof ventilation openings
49 h_Side_Roof = 0
50 T_Out = 0 # Nhiệt độ bên ngoài nhà kính (sẽ được lấy từ datasheet)
51 # Nhiệt độ trung bình giữa không khí dưới màn chắn nhiệt và bên ngoài nhà kính (K)
52 T_Mean = 0 # sẽ được tính từ data sheet
53 U_VentForced = 1 # tốc độ gió của hệ thống quạt [0,1]
54 fi_VentForced = 0 # lưu lượng không khí đi qua quạt gió
55 # -----
56 # diện tích ở mở trên mái nhà kính trên tổng các ở thông gió
57 eta_Roof = A_Roof*U_Roof / (A_Side*U_Side + A_Roof*U_Roof)
58 h_Vent = 0.68 # the vertical dimension of a single ventilation opening
59 # -----
60 eta_ppm_mg = 1/0.554
61 alpha = 0.385 # umol(e-) umol-1(photons)
62 theta = 0.7 # none
63 eta_CO2AirStom = 0.67 # umol(CO2) mol-1(air)
64 cT = 1.7 # umol(CO2) mol-1(air) K-1
65 Ej = 37*(10**3) # J mol-1
66 H = 22*(10**4) # J mol-1
67 CH2O = 30*(10**(-3)) # mg(CH2O) umol-1(CH2O)
68 S = 710 # J mol-1 K-1
69 T25 = 298.15 # K
70 LAI = 2.5 # m2 m-2
71 J_MAX25Leaf = 210 # umol(e-) m-2(leaf) s-1
72 PAR_Can = 100 # là một hằng số bất kỳ
```

3.5 Giả định về các giá trị

Trong Assignment này, nhóm đã đưa ra một vài giả định nhằm khắc phục hạn chế về việc thiếu dữ liệu thực tế, bên cạnh đó cũng để việc hiện thực bài toán tốt hơn:

$$T_{Top} = T_{Air} + 1$$

$$T_{Can} = T_{Air} + 10$$

$$T_{Mean} = \frac{T_{Air} + T_{Out}}{2}$$

$$U_{ThScr} = 0.5$$

$$U_{Roof} = \frac{ventLee - ventWind}{100}$$

Nếu $U_{Roof} < 0.7$ sẽ tăng U_{Roof} thêm 0.3 (cho khớp với dữ liệu thực tế hơn)

3.6 Tính toán xấp xỉ giá trị của nồng độ 2 trong nhà kính và mô phỏng

3.6.1 Giải thuật tính xấp xỉ hệ động lực

Trong Assignment này, chúng ta sẽ dùng giải thuật Euler Explicit và Runge Kutta Explicit bậc 4, (đã được giải thích ở phần đầu).

Dưới đây là phần source code thể hiện cho hai giải thuật trên:

```
1 # Explicit Euler implementation
2 def euler(dx, data, h, loop):
3     if loop == 0:
4         return data
5     k = dx(data[0], data[1], data[2])
6     for i in range(len(data) - 1):
7         data[i] += k[i] * h
8     return euler(dx, list(data), h, loop - 1)
9
10 # Explicit order-4 Runge-Kutta implementation
11 def runge_Kutta(dx, data, h, loop):
12     if loop == 0:
13         return data
14     k1 = dx(data[0], data[1], data[2])
15     data_temp = list(data)
16     for i in range(len(data_temp) - 1):
17         data_temp[i] += k1[i] * h / 2
18     k2 = dx(data_temp[0], data_temp[1], data_temp[2])
19
20     data_temp = list(data)
21     for i in range(len(data_temp) - 1):
22         data_temp[i] += k2[i] * h / 2
23     k3 = dx(data_temp[0], data_temp[1], data_temp[2])
24
25     data_temp = list(data)
26     for i in range(len(data_temp) - 1):
27         data_temp[i] += k3[i] * h
28     k4 = dx(data_temp[0], data_temp[1], data_temp[2])
29
30     for i in range(len(data) - 1):
31         data[i] += h/6*(k1[i] + 2*k2[i] + 2*k3[i] + k4[i])
32
```

```
33 return runge_Kutta(dx, list(data), h, loop - 1)
```

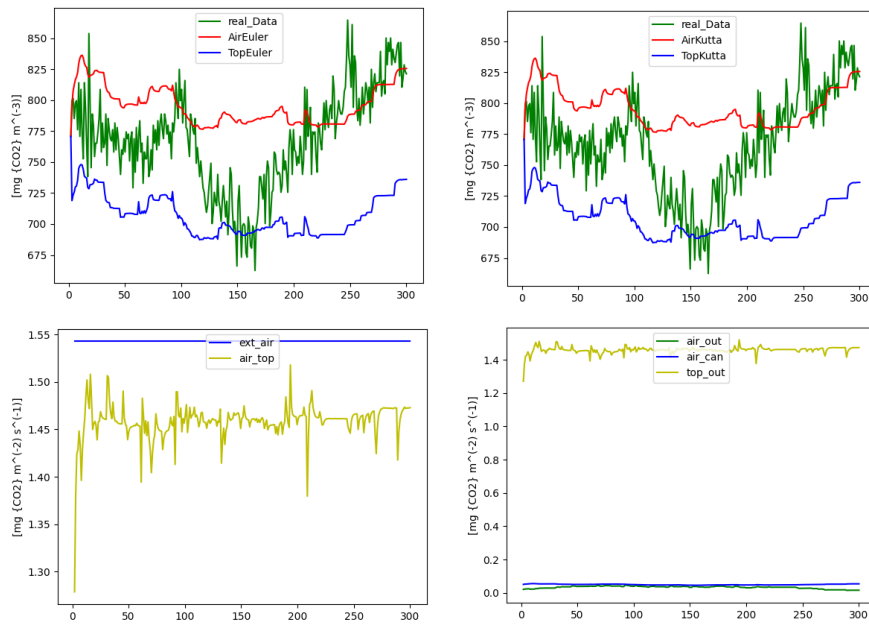
Hai hàm *euler()* và *runge_kutta()* sẽ nhận vào giá trị đạo hàm và giá trị của 2_{Air} cũng như 2_{Top} tại thời điểm t , bên cạnh đó là bước nhảy h . Dựa vào giải thuật của Euler Explicit và Runge Kutta Explicit bậc 4 (đã được phân tích cụ thể ở phần đầu), hai hàm trên sẽ trả về giá trị của 2_{Air} và 2_{Top} tại thời điểm $t + h$.

3.6.2 Mô phỏng hệ động lực và so sánh với dữ liệu thực

Trong Assignment này, nhóm đã lấy dữ liệu thực của một nhà kính trồng dưa leo ở Hà Lan được thu thập vào năm 2018. Dữ liệu sẽ lấy từ 2 file, bao gồm *meteo.csv* là file chứa dữ liệu của các thông số bên ngoài nhà kính, file *Greenhouse_climate.csv* là file chứa dữ liệu của các thông số bên trong nhà kính.

```
1 with open("GH_Cucumber\AutonomousGreenhouseChallengeFirstEdition(2018)\meteo.csv", 'r')
  as csvfile:
2 data = csv.reader(csvfile, delimiter=',')
3     # open file, traverse all air temperature values to calculate a portion of T_Mean
  first
4     for row in data:
5         i += 1
6         if i < 3:
7             continue
8         vWind.append(float(row[10]))
9         TOut.append(float(row[8]) + 273.15)
10        VP_Out.append(float(row[7])/100 * saturated(float(row[8]) + 273.15))
11        if i == stop + 1:
12            break
13    csvfile.close()
14
15 with
  open("GH_Cucumber\AutonomousGreenhouseChallengeFirstEdition(2018)\Sonoma\Greenhouse_climate.csv",
  'r') as csvfile:
16 data = csv.reader(csvfile, delimiter=',')
17 # open file, traverse all air temperature values to calculate final T_Mean
18 i = -1
19 for row in data:
20     i += 1
21     if i < 3:
22         continue
23     if i == 3:
24         startVP = float(row[8])/100 * saturated(float(row[9]) + 273.15)
25         TAir.append(float(row[9]) + 273.15)
26         ventLee.append(float(row[10]))
27         ventWind.append(float(row[11]))
28         realData.append(float(row[8])/100)
29         if i == stop + 1:
30             break
31    csvfile.close()
```

Sau khi tính toán và chạy mô phỏng, ta thu được kết quả của giá trị 2 ở trên và dưới màn chắn nhiệt của hai giải thuật Euler và Runge Kutta bậc 4 so với dữ liệu thực như sau:



Hình 17: Hình ảnh mô phỏng giá trị 2 trên và dưới màn chắn.

Dựa vào hình ảnh mô phỏng ở trên, chúng ta có thể rút ra kết luận rằng: Phương pháp tính toán hệ động lực dựa trên hai giải thuật Euler Explicit và Runge Kutta Explicit bậc 4, mặc dù có nhiều thông số được fixed giá trị để giảm bớt độ khó theo yêu cầu đề bài, tuy nhiên kết quả thu được kết quả khá tiệm cận với dữ liệu thật. Do đó, giá trị tính toán được nhìn chung có tính thực tế cao.

Bên cạnh việc chạy dữ liệu mô phỏng, để giá trị tính toán được có tính thực tế hơn cũng như để so sánh với giá trị thực một cách cụ thể, nhóm đã tính toán ra giá trị của 2_{Air} tại các thời điểm $t + 5, t + 10, t + 15, t + 20, t + 25$ bằng các sử dụng 2 phương pháp tính toán xấp xỉ Euler và RK4 như được trình bày ở bảng dưới đây:

Thời gian ($t = 0$)	GT thực	Euler	$ \Delta_{Euler} $	RK4	$ \Delta_{RK4} $
$t + 5$	799.639	792.6094	7.0296	792.5921	7.0469
$t + 10$	801.444	800.9248	0.5192	800.9099	0.5341
$t + 15$	785.1886	805.1167	19.9281	805.1065	19.9179
$t + 20$	797.8339	810.0681	24.2342	810.0591	24.8705
$t + 25$	799.6389	812.3390	12.7001	812.3329	12.694

4 Mô hình hệ động lực đối với áp suất hơi nước trong nhà kính

4.1 Hệ động lực và giả thiết

Sự thay đổi áp suất hơi nước ở không gian trên và dưới màn chắn nhiệt trong nhà kính được mô tả bởi hệ phương trình vi phân sau:

$$\begin{aligned} cap_{VP_{Air}} \dot{VP}_{Air} = & MV_{CanAir} + MV_{PadAir} + MV_{FogAir} + MV_{BlowAir} \\ & - MV_{AirThScr} - MV_{AirTop} - MV_{AirOut} - MV_{AirOut_Pad} - MV_{AirMech}, \end{aligned} \quad (27)$$

$$cap_{VP_{Top}} \dot{VP}_{Top} = MV_{AirTop} - MV_{TopCov,in} - MV_{TopOut} \quad (28)$$

Trong đó, \dot{VP}_{Air} và \dot{VP}_{Top} ($kg\ m^{-2}\ s^{-1}$) lần lượt là đạo hàm của áp suất hơi ở gian dưới và trên của màn chắn nhiệt trong nhà kính, $cap_{VP_{Air}}$ và $cap_{VP_{Top}}$ lần lượt là độ ẩm tối đa trong không khí ở gian dưới và trên của màn chắn nhiệt trong nhà kính.

Dưới đây là source code python dùng để hiện thực hệ động lực trên:

```
1 def derivative(VP_Air, VP_Top, VP_Out):
2     CAP_VP_Air = M_water * h_Air / (R * 1000 * T_Air) # capacity of VP_Air
3     CAP_VP_Top = M_water * (h_Gh - h_Air) / (R * 1000 * T_Top) # capacity of VP_Top
4     global temp
5     temp -= 1
6     mv_air_top = MV_AirTop(VP_Air, VP_Top)
7     dVP_Air = (MV_CanAir(VP_Air) + MV_PadAir() + MV_FogAir() + MV_BlowAir()
8               - MV_AirThScr(VP_Air) - mv_air_top - MV_AirOut(VP_Air, VP_Out)
9               - MV_AirOut_Pad(VP_Air) - MV_AirMech(VP_Air))/CAP_VP_Air
10    dVP_Top = (mv_air_top - MV_TopCov_in(VP_Top) - MV_TopOut(VP_Top, VP_Out)) /
11              CAP_VP_Top
12    return [dVP_Air, dVP_Top]
```

Hàm *derivative()* trên nhận vào 3 đối số là VP_{Air} , VP_{Top} và VP_{Out} tại thời điểm t , từ đó hàm sẽ trả về 1 list 2 phần tử là giá trị đạo hàm của VP_{Air} và VP_{Top} , từ đó có thể tính được giá trị của chúng tại thời điểm $t + h$.

4.2 Các thông số ảnh hưởng tới áp suất hơi nước

Hơi nước trao đổi giữa không khí trong nhà kính và các yếu tố như tán lá (MV_{CanAir}), không khí đi ra từ hệ thống thông gió (MV_{PadAir}), hệ thống tạo sương (MV_{FogAir}), hệ thống làm nóng không khí trực tiếp ($MV_{BlowAir}$), màn chắn nhiệt ($MV_{AirThScr}$), Không khí ngăn trên màn chắn (MV_{AirTop}), không khí bên ngoài (MV_{AirOut}), lượng không khí ngoài trời qua sự trao đổi của hệ thống thông gió (MV_{AirOut_Pad}) và hệ thống làm mát cơ học ($MC_{AirMech}$), hay hơi nước trao đổi giữa không khí trên màn chắn nhiệt với không khí bên ngoài (MC_{TopOut}), với lớp bao bên trong màn chắn nhiệt ($MV_{TopCov,in}$), đều là những yếu tố ảnh hưởng tới tốc độ biến thiên của áp suất hơi nước và sẽ được mô tả cụ thể sau đây.

Sự thoát hơi nước của lá MV_{CanAir} ($kg\ m^{-2}\ s^{-1}$) được mô tả bởi công thức:

$$MV_{CanAir} = VEC_{CanAir}(VP_{Can} - VP_{Air}) \quad (29)$$

Trong đó, VEC_{CanAir} ($Kg\ m^{-2}\ Pa^{-1}\ s^{-1}$) là hệ số trao đổi hơi giữa tán cây và không khí, VP_{Can} (Pa) là áp suất hơi bão hòa tại nhiệt độ tán lá. Hệ số trao đổi hơi giữa tán cây và không

khí VEC_{CanAir} được mô tả bởi phương trình:

$$VEC_{CanAir} = \frac{2\rho_{Air}c_{p,Air}LAI}{\Delta H\gamma(r_b + r_s)} \quad (30)$$

Trong đó, ρ_{Air} ($Kg\ m^{-3}$) là mật độ của không khí bên trong nhà kính, ΔH là nhiệt tiềm ẩn của sự bay hơi của nước, γ ($Pa\ K^{-1}$) là hằng số biểu diễn mối liên hệ giữa áp suất riêng phần của nước trong không khí với nhiệt độ không khí, r_b ($m\ s^{-1}$) là sức cản của lớp ngoài của tán lá và r_s ($m\ s^{-1}$) là sức cản của khí khổng trong tán lá tới sự vận chuyển hơi nước.

Sức cản của lớp ngoài của tán lá tới sự vận chuyển hơi nước phụ thuộc vào tốc độ gió trong nhà kính và sự chênh lệch nhiệt độ giữa tán cây và không khí xung quanh. Sức cản của khí khổng trong tán lá tới sự vận chuyển hơi nước r_s được mô tả bằng

$$r_s = r_{s,min} \cdot rf(R_{Can}) \cdot rf(CO_{2Air_ppm}) \cdot rf(VP_{Can} - VP_{Air}) \quad (31)$$

với $r_{s,min}$ là sức cản tối thiểu của tán lá và rf là hệ số cản trong trường hợp mức bức xạ cao, mức 2 cao và chênh lệch áp suất lớn. Hệ số cản rf được tính bởi các phương trình sau

$$rf(R_{Can}) = \frac{R_{Can} + c_{evap1}}{R_{Can} + c_{evap2}}, \quad (32)$$

$$rf(CO_{2Air}) = 1 + c_{evap3}(\eta_{mg_ppm}CO_{2Air} - 200)^2 \quad (33)$$

$$rf(VP_{Can} - VP_{Air}) = 1 + c_{evap4}(VP_{Can} - VP_{Air})^2 \quad (34)$$

trong đó, R_{Can} $W\ m^{-2}$ là bức xạ toàn cục trên tán lá, c_{evap1} , c_{evap2} , c_{evap3} và c_{evap4} là các thông số được xác định từ thực nghiệm, η_{mg_ppm} ($ppm\ m^3\ mg^{-1}$) là hệ số chuyển đổi từ $m^{-3}\ mg\ 2$ sang ppm . Trên thực tế, các giá trị c_{evap3} và c_{evap4} phụ thuộc vào ngày và đêm. Tuy nhiên, các thông số này có các phương trình đi kèm không thể xác định được tại thời điểm mặt trời mọc và lặn. Để giải quyết vấn đề này, [Sta93] đã đưa ra phương pháp làm mịn tham số c_{evap3} và c_{evap4} bằng công tắc vi sai để các phương trình (32), (33), (34) xác định. Các thông số thoát hơi nước được làm mịn được mô tả bằng:

$$c_{evap3} = c_{evap3}^{day}(1 - S_{r_s}) + c_{evap3}^{night}S_{r_s} \quad (35)$$

$$c_{evap4} = c_{evap4}^{day}(1 - S_{r_s}) + c_{evap4}^{night}S_{r_s} \quad (36)$$

Theo yêu cầu đề bài, vì bị giới hạn về mặt dữ liệu, nên ta sẽ mặc định $r_s = r_{s,min}$.

Qua đó, ta có thể hiện thực hàm tính tốc độ thoát hơi nước của tán lá bằng Python như sau:

```
1 def saturated(T):
2     return -274.36 + 877.52*math.exp(0.0545*(T-273.15))
3
4 def MV_CanAir(VP_Air):
5     rs = rs_min*1.1
6     # Mat do khong khi duoi man chan nhiet (kg m^{-3})
7     p_Air = p_Air0 * math.exp(g * M_Air * h_Elevation / (T_Air * R * 1000))
8     VEC_CanAir = 2*p_Air*c_p_air*LAI/(dentaH*gamma*(rb+rs))
9     T_Can = T_Air + 2
10    VP_Can = saturated(T_Can)
11    can_air = VEC_CanAir*(VP_Can - VP_Air)
12    if temp == loop:
13        MV_CanAirData.append(can_air)
14    return can_air
```

Hàm $saturated(T)$ là hàm sẽ tính giá trị của áp suất hơi bão hòa tại nhiệt độ $T(K)$. Theo yêu cầu đề bài, ta lấy $r_s = r_{s,min}$, tuy nhiên mặc định thì độ chênh lệch so với thực tế là rất lớn, nên nhóm quyết định lấy $r_s = 1.1 \cdot r_{s,min}$ để giá trị xấp xỉ sát với thực nghiệm hơn. Hàm MV_AirCan được dùng để tính và trả giá trị của MV_{CanAir} qua việc tính các thành phần liên quan trong phương trình (29).

Tiếp theo, dòng hơi nước đi từ tấm thông gió và hệ thống quạt tới vùng không khí bên trong nhà kính MV_{PadAir} được mô tả bởi phương trình

$$MV_{PadAir} = \rho_{Air} f_{Pad} (\eta_{Pad} (x_{Pad} - x_{Out}) + x_{Out}) \quad (37)$$

Trong đó, $f_{Pad} (m s^{-1})$ là tốc độ hơi nước đi qua bề mặt tấm thông gió và quạt (đã được đề cập trong phương trình (5)), $\eta_{Pad} (-)$ là độ hiệu quả của hệ thống thông gió và quạt, x_{Pad} và x_{Air} ($Kg \{water\} Kg^{-1}\{Air^{-1}\}$) lần lượt là hàm lượng hơi nước của tấm thông gió và của không khí bên ngoài.

Dưới đây là phần hiện thực phương trình (37):

```
1 def MV_PadAir():
2     # Mat do khong khi duoi man chan nhiet (kg m-3)
3     p_Air = p_Air0 * math.exp(g * M_Air * h_Elevation / (T_Air * R * 1000))
4     # cuong do khong khi qua tam thong gio (m3 s-1)
5     f_Pad = U_Pad * fi_Pad / A_Flr
6     pad_air = p_Air * f_Pad * (eta_pad * (x_pad - x_out) + x_out)
7     return pad_air
```

Hàm MV_PadAir sẽ hiện thực các tham số thành phần của phương trình (37), từ đó sẽ tính được giá trị của MV_{PadAir} .

Ngược lại, dòng hơi nước đi từ vùng không khí bên trong nhà kính đi ra ngoài qua hệ thống thông gió và quạt MV_{AirOut_pad} được biểu diễn bởi

$$MV_{AirOut_pad} = f_{Pad} \frac{M_{Water}}{R} \left(\frac{VP_{Air}}{T_{Air} + 273.15} \right) \quad (38)$$

trong đó, $M_{Water} = 18$ là khối lượng mol của nước.

Để thể hiện cũng như tính toán giá trị của MV_{AirOut_pad} , ta có thể tham khảo source code dưới đây:

```
1 def MV_AirOut_Pad(VP_Air):
2     f_Pad = U_Pad * fi_Pad / A_Flr # Cuong do khong khi di qua tam khong gio (m3 s-1)
3     airOut_pad = f_Pad * M_water / (R * (10**3)) * (VP_Air / T_Air)
4     return airOut_pad
```

Hàm MV_AirOut_Pad sẽ tính toán các giá trị tham số, sau đó trả về kết quả MV_{AirOut_pad} để hỗ trợ cho việc tính toán hệ đông lực (27)-(27).

Thông lượng hơi nước từ hệ thống tạo sương đến không khí trong nhà kính MV_{FogAir} được

mô tả bởi

$$MV_{FogAir} = \frac{U_{Fog} \phi_{Fog}}{A_{Flr}} \quad (39)$$

Trong đó, U_{Fog} (–) là giá trị van điều khiển của hệ thống tạo sương, nhận giá trị trong khoảng $[0, 1]$, ϕ_{Fog} ($Kg \{Water\} s^{-1}$) là công suất của hệ thống phun sương. Dưới đây là source code Python hiện thực tính toán phương trình trên:

```
1 def MV_FogAir():  
2     fog_air = U_Fog * fi_Fog / A_Flr  
3     return fog_air
```

Cụ thể, hàm MV_{FogAir} sẽ dựa vào 3 tham số là U_{Fog} , ϕ_{Fog} và A_{Flr} để tính toán và trả về giá trị của MV_{FogAir} .

Cường độ hơi nước từ máy thổi nhiệt đến không khí trong nhà kính $MV_{BlowAir}$ được mô tả tỉ lệ với dòng nhiệt từ máy sưởi không khí đến không khí trong nhà kính $H_{BlowAir}$:

$$MV_{BlowAir} = \eta_{HeatVap} H_{BlowAir} \quad (40)$$

Trong đó, $\eta_{HeatVap}$ ($Kg \text{ Vapour } J^{-1}$) là lượng hơi được giải phóng khi 1 Joule năng lượng được tạo ra bởi máy sưởi không khí. Thông lượng nhiệt $H_{BlowAir}$ được tính bởi:

$$H_{BlowAir} = \frac{U_{Blow} P_{Blow}}{A_{Flr}} \quad (41)$$

với các thông số được mô tả cụ thể trong phương trình (3).
Hàm hiện thực phương trình (40) được mô tả dưới đây:

```
1 def MV_BlowAir():  
2     blow_air = (eta_heatVap * U_Blow * P_Blow) / A_Flr  
3     return blow_air
```

Với giá trị của $H_{BlowAir}$ được thể hiện qua phương trình (41), kết hợp với phương trình (40), hàm $MV_{BlowAir}$ sẽ tính được giá trị của $MV_{BlowAir}$ và trả về kết quả tương ứng.

Dựa vào công thức MV_{12} tổng quát, ta có thể biểu diễn thông lượng hơi nước đi từ không khí nhà kính lên màn chắn nhiệt $MV_{AirThScr}$ bằng

$$MV_{AirThScr} = \begin{cases} 0, & VP_{Air} < VP_{ThScr} \\ 6.4 \cdot 10^{-9} HEC_{AirThScr} (VP_{Air} - VP_{ThScr}), & VP_{Air} > VP_{ThScr} \end{cases} \quad (42)$$

Trong đó, $6.4 \cdot 10^{-9}$ là hệ số chuyển đổi từ hệ số trao đổi nhiệt (đơn vị $W m^{-2} K^{-1}$) thành hệ số trao đổi hơi (Đơn vị $Kg m^{-2} s^{-1} Pa^{-1}$), $HEC_{AirThScr}$ ($W m^{-2} K^{-1}$) là hệ số trao đổi nhiệt giữ không khí trong nhà kính và trên màn chắn nhiệt. $HEC_{AirThScr}$ được tính bằng

$$HEC_{AirThScr} = 1.7 U_{ThScr} |T_{Air} - T_{ThScr}|^{0.33} \quad (43)$$

Dưới đây là source code hiện thực phương trình (32)

```

1 def MV_AirThScr(VP_Air):
2     T_ThScr = (T_Air + T_Top) / 2 # Nhiệt độ của màn chắn nhiệt
3     VP_ThScr = saturated(T_ThScr)
4     if (VP_Air < VP_ThScr):
5         if temp == loop:
6             MV_AirThScrData.append(0)
7         return 0
8
9     HEC_AirThScr = 1.7 * U_ThScr * (abs(T_Air - T_ThScr)**0.33)
10    air_thScr = 6.4*(10**(-9))*HEC_AirThScr*(VP_Air - VP_ThScr)
11    if temp == loop:
12        MV_AirThScrData.append(air_thScr)
13    return air_thScr

```

Hàm $MV_AirThScr$ sẽ tính các giá trị thành phần của $MV_{AirThScr}$ như T_{ThScr} (Trong phương trình (43)), $HEC_{AirThScr}$ và VP_{ThScr} . Với tham số đầu vào là giá trị của VP_{Air} tại thời điểm t , hàm sẽ tính và trả về giá trị của lượng hơi nước trao đổi giữa không gian dưới màn chắn nhiệt và màn chắn nhiệt.

Tương tự như vậy, ta có công thức biểu diễn thông lượng hơi nước từ ngăn trên màn chắn nhiệt đến lớp bao bên trong $MV_{TopCov,in}$ là

$$MV_{TopCov,in} = \begin{cases} 0, & VP_{Top} < VP_{Cov,in} \\ 6.4 \cdot 10^{-9} HEC_{TopCov,in} (VP_{Top} - VP_{Cov,in}), & VP_{Top} > VP_{Cov,in} \end{cases} \quad (44)$$

Với $HEC_{TopCov,in}$ ($W \cdot m^{-2} \cdot K^{-1}$) là hệ số trao đổi nhiệt giữ không khí trên màn chắn nhiệt trong nhà kính và lớp bao bên trong, được tính bởi

$$HEC_{TopCov,in} = c_{HECin} (T_{Top} - T_{Cov,in})^{0.33} \frac{A_{Cov}}{A_{Flr}} \quad (45)$$

Phương trình (44) sẽ được hiện thực qua hàm dưới đây:

```

1 '''Calculate MV_TopCov,in'''
2 def MV_TopCov_in(VP_Top):
3     T_Cov_in = (T_Air*2 + T_Top)/3 # Nhiệt độ của lớp phủ
4     VP_Cov_in = saturated(T_Cov_in) # áp suất hơi bão hòa của lớp phủ
5     if VP_Top < VP_Cov_in:
6         if temp == loop:
7             MV_TopCov_inData.append(0)
8         return 0
9
10    HEC_TopCov_in = c_HECin*(abs(T_Top - T_Cov_in)**(0.33))*A_Cov/A_Flr
11    top_cov_in = 6.4*(10**(-9))*HEC_TopCov_in*(VP_Top - VP_Cov_in)
12    if temp == loop:
13        MV_TopCov_inData.append(top_cov_in)
14    return top_cov_in

```

Qua đó, hàm $MV_TopCov_in()$ có tham số đầu vào là VP_{Top} , hàm tính toán các thông số tương ứng với hai phương trình (44)-(45 sau đó trả về kết quả của $MV_{TopCov,in}$.

Dựa trên dạng chung của dòng hơi nước đi từ A đến B được nghiên cứu trong [Van11], dòng hơi nước từ không khí dưới màn chắn đến trên màn chắn trong nhà kính MV_{AirTop} được mô tả bởi công thức

$$MV_{AirTop} = \frac{M_{Water}}{R} f_{ThScr} \left(\frac{VP_{Air}}{T_{Air} + 273.15} - \frac{VP_{Top}}{T_{Top} + 273.15} \right) \quad (46)$$

trong đó, f_{ThScr} là tổng của hai tốc độ gồm tốc độ thẩm thấu qua màn chắn nhiệt và tốc độ tại những nơi không bị chắn bởi màn chắn nhiệt (Đã được mô tả qua phương trình (7)).

Dưới đây là phần hiện thực của phương trình trên:

```
1 def f_ThScr():
2     p_Air = p_Air0 * math.exp(g * M_Air * h_Elevation
3         / (T_Air * R * 1000))
4     p_Top = p_Air0 * math.exp(g * M_Air *
5         (h_Elevation + h_Air) / (T_Top * R * 1000))
6     p_MeanAir = p_Air0 * math.exp(g * M_Air *
7         (h_Elevation + h_Gh) / ((T_Air + T_Top)/2 * R * 1000))
8
9     thScr = U_ThScr * K_ThScr * (abs(T_Air - T_Top)) ** (2/3)
10    notThScr = (1 - U_ThScr) * (g*(1 - U_ThScr) / (2*p_MeanAir)
11        * abs(p_Air - p_Top))**0.5
12    return thScr + notThScr
13
14 def MV_AirTop(VP_Air, VP_Top):
15     air_top = M_water/(R*1000)*f_ThScr()*(VP_Air/T_Air - VP_Top/T_Top)
16     if temp == loop:
17         MV_AirTopData.append(air_top)
18     return air_top
```

Với hàm $f_{ThScr}()$ tương tự lấy từ phương trình (7), hàm $MV_{AirTop}()$ sẽ tính toán và trả về giá trị của cường độ hơi nước đi từ không gian dưới màn chắn nhiệt đến không gian trên màn chắn nhiệt.

Tương tự, dòng hơi nước từ không khí trong nhà kính đến không khí bên ngoài MV_{AirOut} được biểu diễn bằng

$$MV_{AirOut} = \frac{M_{Water}}{R} (f_{VentSide} + f_{VentForced}) \left(\frac{VP_{Air}}{T_{Air} + 273.15} - \frac{VP_{Top}}{T_{Top} + 273.15} \right) \quad (47)$$

Trong đó, $f_{VentSide}$ là tốc độ gió của hệ thống quạt trên tường bao xung quanh nhà kính (Đã được mô tả qua phương trình (12)) và $f_{VentForced}$ là tốc độ gió từ hệ thống quạt bên trong nhà kính (Đã được mô tả qua phương trình (13)).

Dưới đây là phần hiện thực của phương trình tính MV_{AirOut} :

```
1 def f_VentRoofSide(A_Roof):
2     p1 = (U_Roof*U_Side*A_Roof*A_Side) ** 2
3     p2 = (U_Roof * A_Roof) ** 2 + (U_Side * A_Side)**2
4     p3 = 2*g*h_Side_Roof*(T_Air - T_Out)/T_Mean
```

```
5     p4 = ((U_Roof*A_Roof + U_Side*A_Side)/2) ** 2
6
7     if A_Roof == 0 and A_Side == 0:
8         return 0
9     return (cd/A_Flr)*( (p1/p2*p3 + p4*cw*(v_wind**2)) ** 0.5 )
10
11 # toc do gio cua he thong quat tren tuong
12 def f_VentSide():
13     if U_Roof == 0:
14         eta_Roof = 1.1
15     else:
16         eta_Roof = A_Roof * U_Roof / (A_Side * U_Side + A_Roof * U_Roof)
17         eta_Side = (A_Side * U_Side) / (A_Side * U_Side + A_Roof * U_Roof)
18     if eta_Roof < eta_Side_Thr:
19         return zeta_InsScr() * (U_ThScr * f_VentRoofSide(0) + (1 - U_ThScr)
20             * f_VentRoofSide(A_Roof) * eta_Side) + 0.5 * f_Leakage()
21     else:
22         return zeta_InsScr() * f_VentRoofSide(0) + 0.5 * f_Leakage()
23
24 # He so giam do luoi chan con trung
25 def zeta_InsScr():
26     return zeta_Insect * (2 - zeta_Insect)
27
28 # Cuong do khong khi ro ri
29 def f_Leakage():
30     if v_wind < 0.25:
31         return 0.25*c_leak
32     else:
33         return v_wind*c_leak
34
35 def f_VentForced():
36     return zeta_InsScr()*U_VentForced*fi_VentForced/A_Flr
37
38 '''Calculate MV_AirOut'''
39 def MV_AirOut(VP_Air, VP_Out):
40     air_out = M_water/(R*1000)*(f_VentSide() + f_VentForced()*(VP_Air/T_Air -
41         VP_Out/T_Out))
42     if temp == loop:
43         MV_AirOutData.append(air_out)
44     return air_out
```

Hàm $MV_AirOut()$ dựa vào giá trị trả về của các hàm đã được mô tả bởi các phương trình (12)-(13) để tính toán và trả về giá trị của cường độ hơi nước đi từ gian dưới màn chắn ra ngoài nhà kính.

Dòng hơi nước từ không khí trên màn chắn trong nhà kính đến không khí bên ngoài MV_{TopOut} cũng được áp dụng công thức tổng quát trong [Van11], vì vậy sẽ được mô tả bằng

$$MV_{TopOut} = \frac{M_{Water}}{R} f_{VentRoof} \left(\frac{VP_{Air}}{T_{Air} + 273.15} - \frac{VP_{Top}}{T_{Top} + 273.15} \right) \quad (48)$$

Trong đó, $f_{VentRoof}$ là tốc độ luồng không khí đi qua ô mở mái nhà kính (Đã được mô tả qua

phương trình (15)).

Dưới đây là hàm hiện thực của phương trình (48):

```

1  '''Calculate MV_TopOut'''
2  def MV_TopOut(VP_Top, VP_Out):
3      top_out = M_water/(R*1000)*f_VentRoof()*(VP_Top/T_Top - VP_Out/T_Out)
4      if temp == loop:
5          MV_TopOutData.append(top_out)
6      return top_out
7
8  # tốc độ không khí đi qua ô mo mai nhà kính
9  def f_VentRoof():
10     if U_Roof == 0:
11         eta_Roof = 1.1
12     else:
13         eta_Roof = A_Roof * U_Roof / (A_Side * U_Side + A_Roof * U_Roof)
14     if eta_Roof < eta_Roof_Thr:
15         return zeta_InsScr() * (U_ThScr * f2_VentRoof() + (1 - U_ThScr)
16             * f_VentRoofSide(A_Roof) * eta_Roof) + 0.5 * f_Leakage()
17     else:
18         return zeta_InsScr() * f2_VentRoof() + 0.5 * f_Leakage()
19
20 def f2_VentRoof():
21     return (cd*U_Roof*A_Roof)/(2*A_Flr)
22     * ((g*h_Vent*(T_Air-T_Out)/(2*T_Mean)+cw*(v_wind**2) ) ** 0.5)

```

Các hàm $f_VentRoof()$ và $f2_VentRoof()$ dựa vào giải thuật của phương trình (15) đã được thể hiện ở phần trước. Dựa vào đó, hàm $MV_TopOut()$ sẽ trả về giá trị của cường độ hơi nước đi từ gian trên màn chắn nhiệt ra ngoài nhà kính.

Cuối cùng, thông số ảnh hưởng tới áp suất hơi nước được gồm dòng hơi từ không khí nhà kính đến bề mặt của hệ thống làm mát cơ học, $MV_{AirMech}$. Dựa trên nghiên cứu công thức tổng quát về tốc độ hơi nước từ không khí sang một vật thể trong [Van11], $MV_{AirMech}$ được biểu diễn bằng

$$MV_{AirMech} = \begin{cases} 0, & VP_{Air} < VP_{MechCool} \\ 6.4 \cdot 10^{-9} HEC_{MechAir} (VP_{Air} - VP_{MechCool}), & VP_{Air} > VP_{MechCool} \end{cases} \quad (49)$$

Trong đó, $VP_{MechCool}$ (Pa) là áp suất hơi bão hòa của cơ cấu làm mát cơ khí, $HEC_{MechAir}$ ($W m^{-2} K^{-1}$) là hệ số trao đổi nhiệt giữa không khí bên trong nhà kính với bề mặt của bộ làm mát cơ học, được tính bởi công thức

$$HEC_{AirMech} = \frac{U_{MechCool} COP_{MechCool} P_{MechCool} / A_{Flr}}{T_{Air} - T_{MechCool} + 6.4 \cdot 10^{-9} \Delta H (VP_{Air} - VP_{MechCool})} \quad (50)$$

Trong đó, $U_{MechCool}$ (—) là giá trị của van điều khiển của thiết bị làm mát cơ học, giá trị nằm trong khoảng $[0, 1]$, $COP_{MechCool}$ (—) là hệ số hiệu suất của hệ thống làm mát cơ học, $P_{MechCool}$ (W) là công suất điện của hệ thống làm mát cơ học và $T_{MechCool}$ ($^{\circ}C$) là nhiệt độ của bề mặt làm mát. Ta có thể hiện thực phương trình (49) bằng Python như sau:

```

1  '''Calculate MV_AirMech'''

```

```
2 def MV_AirMech(VP_Air):
3     T_MechCool = 0
4     VP_MechCool = saturated(T_MechCool)
5
6     if VP_Air < VP_MechCool:
7         return 0
8     numerator = U_MechCool*COP_MechCool*P_MechCool/A_Flr
9     denominator = T_Air-T_MechCool + 6.4*(10**(-9))*dentaH*(VP_Air-VP_MechCool)
10    HEC_MechAir = numerator/denominator
11
12    return 6.4*(10**(-9))*HEC_MechAir*(VP_Air - VP_MechCool)
```

Hàm $MV_AirMech()$ sẽ tính toán các giá trị thành phần của phương trình (49), cụ thể ở đây là tính $HEC_{AirMech}$. Có được các giá trị tham số tương ứng, ta có thể dễ dàng tính được giá trị của $MV_{AirMech}$. Qua đó, hàm sẽ trả về giá trị tương ứng của cường độ hơi nước của gian dưới màn chắn nhiệt bị hút bởi hệ thống làm lạnh.

4.3 Các tham số của hệ động lực và giải thích

Trong source code dùng để hiện thực mô hình hệ động lực thể hiện sự thay đổi của áp suất hơi nước trong nhà kính, có các thông số được xác định bởi [Van11] và được xác định bởi tập dữ liệu lấy từ thực tế. Cụ thể:

```
1 c_p_air = 1*(10**3) # Cong suat nhiet cu the cua khong khi (J.K^{-1}.kg^{-1})
2 LAI = 2.5 # Chi so la (m^2 m^{-2})
3 dentaH = 2.45*(10**6) # Nhiet bay hoi an (J.kg^{-1}water)
4 gamma = 65.8 # (Pa.K^{-1})
5 rs_min = 82.0 # Tro khang nho nhat cua la (s.m^{-1})
6 rb = 275 # Tro khang cua lop ranh gioi cua la
7 #-----
8 U_Pad = 1 # Gia tri cua van dieu khien cua tam thong gio
9 fi_Pad = 0 # Cong suat khong khi di qua tam thong gio (m^3 s^{-1})
10 eta_pad = 0 # Do hieu qua cua he thong thong gio va quat
11 x_pad = 0 # thanh phan nuoc cua he thong thong gio (kg water . kg^{-1} air)
12 x_out = 0 # thanh phan nuoc cua khong khi ngoai (kg water . kg^{-1} air)
13 A_Flr = 1.4*(10**4)
14 #-----
15 U_Fog = 1 # Van dieu khien cua he thong tao suong
16 fi_Fog = 0 # Cong suat hoi nuoc cua he thong tao suong (kg water s^{-1})
17 #-----
18 eta_heatVap = 4.43*(10**(-8)) # Luong hoi nuoc sinh ra khi 1J Q tao ra boi may suoi (kg
    vapour J^{-1})
19 U_Blow = 1 # gia tri van dieu khien cua may suoi
20 P_Blow = 0 # Cong suat nhiet cua may suoi khong khi truc tiep (W)
21 #-----
22 U_ThScr = 0.1 # Gia tri do phu cua man chan nhiet [0,1]
23 #-----
24 g = 9.8 # Gia toc trong truong (m s^{-2})
25 M_Air = 28.96 # g mol^{-1}
26 K_ThScr = 0.05*(10**(-3)) # Kha nang cho khong khi tham thau cua man
27 T_Air = 0 # Nhiet do gian tren man chan (K)
28 T_Top = 0 # nhiet do gian duoi man chan (K) (= T_Air + 1)
29 h_Elevation = 0 # Do cao cua nha kinh so voi muc nuoc bien (m)
```

```
30 h_Air = 3.8          # Chiều cao của compartment bên dưới nhà kính (m)
31 h_Gh = 4.2
32 p_Air0 = 1.20        # Mật độ không khí ở mức nước biển (kg m-3)
33 #-----
34 v_wind = 0           # Tốc độ gió
35 c_leak = 10**(-4)     # Hệ số rò rỉ
36 zeta_Insect = 1
37 eta_Roof_Thr = 0.9    # ngưỡng Stack (ở Roof)
38 eta_Side_Thr = eta_Roof_Thr # ngưỡng Stack (ở Side)
39 U_ShScr = 0.4
40 C_Gh_d = 0.75
41 C_Gh_w = 0.09
42 eta_ShScrCd = 0
43 eta_ShScrCw = 0
44 cw = C_Gh_w*(1 - eta_ShScrCw*U_ShScr)
45 cd = C_Gh_d*(1 - eta_ShScrCd*U_ShScr)
46 U_Side = 1
47 A_Side = 0*A_Flr
48 U_Roof = 1
49 A_Roof = 0.1*A_Flr    # diện tích ở thông gió trên mái
50 eta_Side = (A_Side*U_Side)/(A_Side*U_Side + A_Roof*U_Roof)
51 h_Side_Roof = 0 # The vertical distance between mid-points of side wall and roof
    ventilation openings
52 T_Out = 0
53 T_Mean = 0
54 U_VentForced = 1
55 fi_VentForced = 0 # lưu lượng không khí đi qua quạt gió
56 #-----
57 M_water = 18
58 R = 8.314
59 #-----
60 U_MechCool = 1
61 COP_MechCool = 0
62 P_MechCool = 0
63 #-----
64 c_HECin = 1.86
65 A_Cov = 1.8*(10**4)
66 #-----
67 h_Vent = 0.68          # the vertical dimension of a single ventilation opening
```

4.4 Các giả định về giá trị

Trong Assignment này, nhóm đã đưa ra một vài giả định nhằm khắc phục việc thiếu dữ liệu và để hiện thực bài toán tốt hơn:

$$T_{Top} = T_{Air} + 5$$

$$T_{Can} = T_{Air} + 2$$

$$r_s = 1.1r_{s,min}$$

$$T_{Mean} = \frac{T_{Air} + T_{Top}}{2}$$

$$U_{ThScr} = 0.1$$

$$U_{Roof} = \frac{VentLee + ventWind}{200}$$

Nếu $U_{Roof} < 0.3$ sẽ tăng U_{Roof} thêm 0.7 (cho khớp với dữ liệu thực tế hơn)

$$T_{ThScr} = \frac{T_{Air} + T_{Top}}{2}$$

$$T_{Cov,in} = \frac{2T_{Air} + T_{Top}}{3}$$

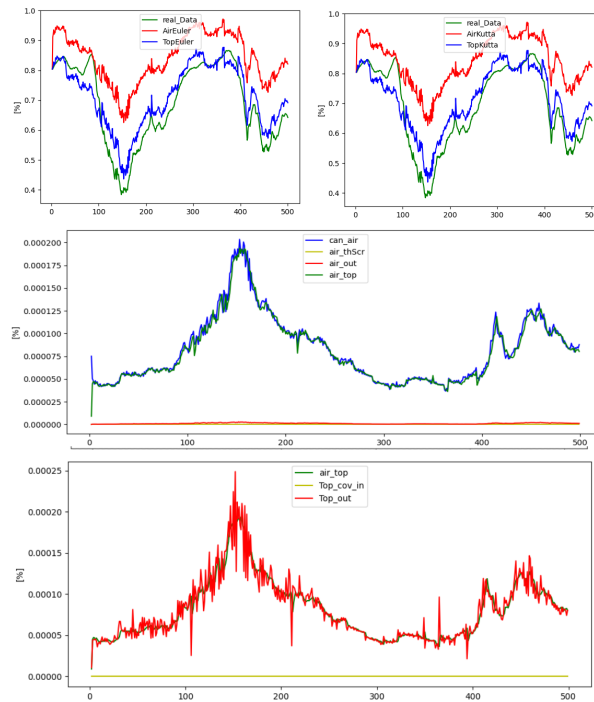
4.5 Tính toán xấp xỉ giá trị của áp suất hơi nước trong nhà kính và mô phỏng

4.5.1 Giải thuật tính xấp xỉ hệ động lực

Trong phần này, chúng ta vẫn sẽ dùng giải thuật Euler Explicit và Runge Kutta Explicit bậc 4 đã được đề cập ở phần 1 và hiện thực ở phần 3.5.1.

4.5.2 Mô phỏng hệ động lực và so sánh với dữ liệu thực

Với dữ liệu được lấy từ 2 file *meteo.csv* và *Greenhouse_climate.csv* (2 file này đã được giải thích ở phần 3.5.2), nhóm đã tính toán và trả về giá trị của áp suất hơi nước tại thời điểm $t + h$, sau đó nhóm đã mô phỏng và thu được kết quả của giá trị áp suất hơi nước ở trên và dưới màn chắn nhiệt dựa trên hai giải thuật Euler và Runge Kutta bậc 4 so với dữ liệu thực như sau:



Hình 18: Hình ảnh mô phỏng giá trị của áp suất hơi nước trên và dưới màn chắn.

Bên cạnh việc chạy dữ liệu mô phỏng, để giá trị tính toán được có tính thực tế hơn cũng như để so sánh với giá trị thực một cách cụ thể, nhóm đã tính toán ra giá trị của VP_{Air} tại các thời điểm $t + 5, t + 10, t + 15, t + 20, t + 25$ bằng các sử dụng 2 phương pháp tính toán xấp xỉ Euler và RK4 như được trình bày ở bảng dưới đây:

Thời gian ($t = 0$)	GT thực	Euler	$ \Delta\text{Euler} $	RK4	$ \Delta\text{RK4} $
$t + 5$	2321.44	2120.3388	201.1012	2119.8686	201.5714
$t + 10$	2321.44	2141.7843	179.6557	2141.7062	179.7338
$t + 15$	2293.3	2148.9042	144.3958	2148.8841	144.4159
$t + 20$	2321.44	2141.5689	179.8771	2141.5806	179.8594
$t + 25$	2307.33	2151.8117	155.5183	2151.8117	155.5123

Dựa vào hình ảnh mô phỏng ở trên, chúng ta có thể rút ra kết luận rằng: Phương pháp tính toán hệ động lực dựa trên hai giải thuật Euler Explicit và Runge Kutta Explicit bậc 4 thu được kết quả khá tiệm cận với dữ liệu thật, tuy nhiên có hơi lệch vì trong quá trình tính toán, có những tham số được fixed vì có những giá trị không đủ dữ liệu để xác định, ví dụ như giá trị $r_s = r_{s,min}$ được fixed so với r_s được phân tích trong [Van11].



Tài liệu tham khảo

- [Bal89] L.J. Balemans. Assessment of criteria for energetic effectiveness of greenhouse screens. 1989.
- [MG] N.J. van de Braak MIGUEL A.F. and 1995 G.P.A. Bot. “Mass flow through materials with pores and openings: II-natural convection”. In: *submitted for publication in International Journal of Heat and Mass Transfer* ().
- [Sta93] C. Stanghellini and J.A. Bunce. *Response of photosynthesis and conductance to light, CO₂, temperature and humidity in tomato plants grown at ambient and elevated CO₂*. 1993.
- [Van11] Bram HE Vanthoor. *A model-based greenhouse design method*. 2011.
- [De96] HF De Zwart. *Analyzing energy-saving options in greenhouse cultivation using a simulation model*. 1996.