

# Stockage de l'énergie avec l'air liquide



Tipe 2020-2021  
Enjeux sociétaux  
N ° d'inscription 34546

# *PLAN*

I/Introduction

II/Cycle de charge

1/Introduction du cycle

2/Modélisation du cycle

3/Analyse du cycle

III/Stockage

1/Modélisation du réservoir

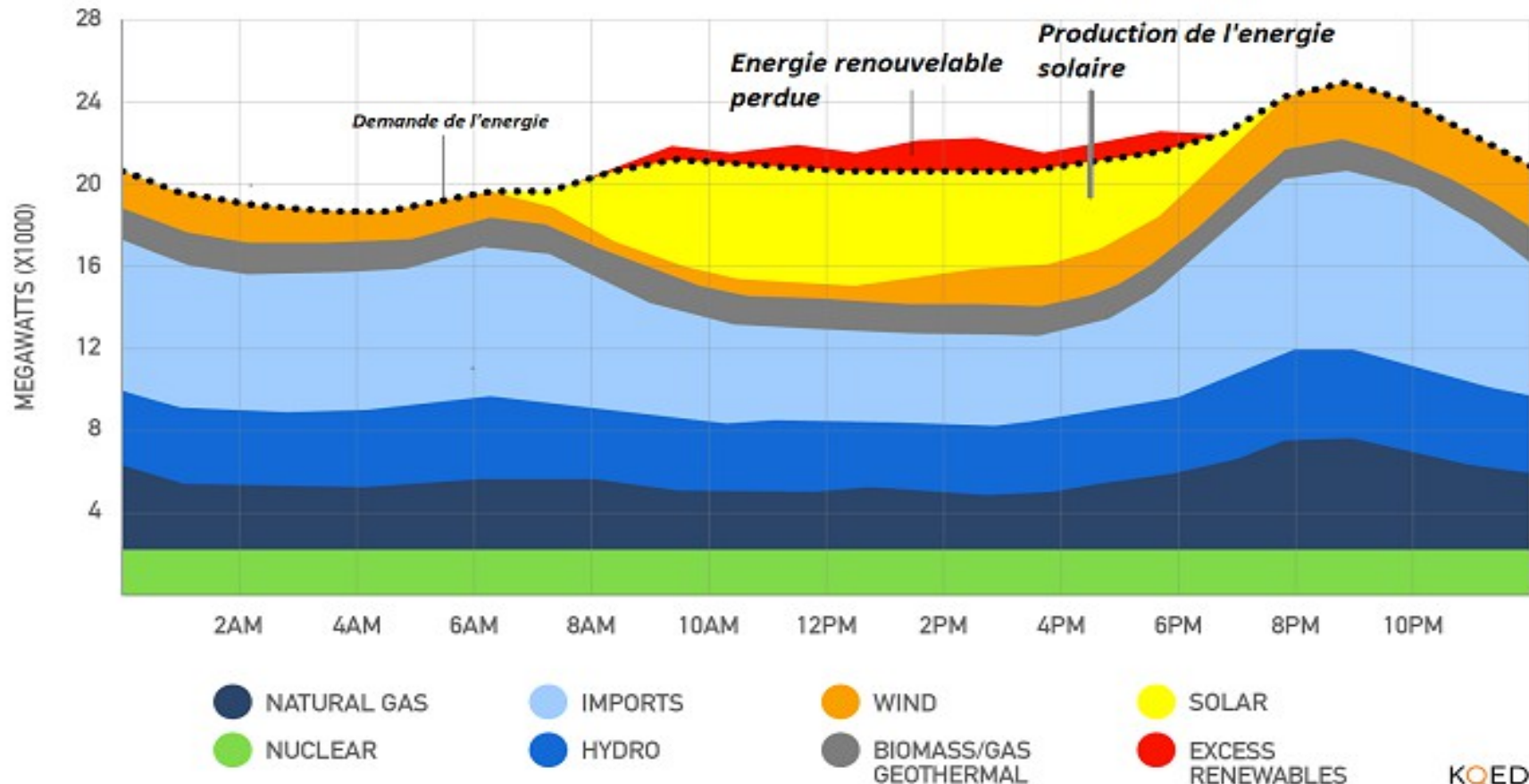
2/Résolution numérique et analyse des résultats

IV/Extraction de l'énergie de l'air liquide

V/Conclusion

# I/Introduction

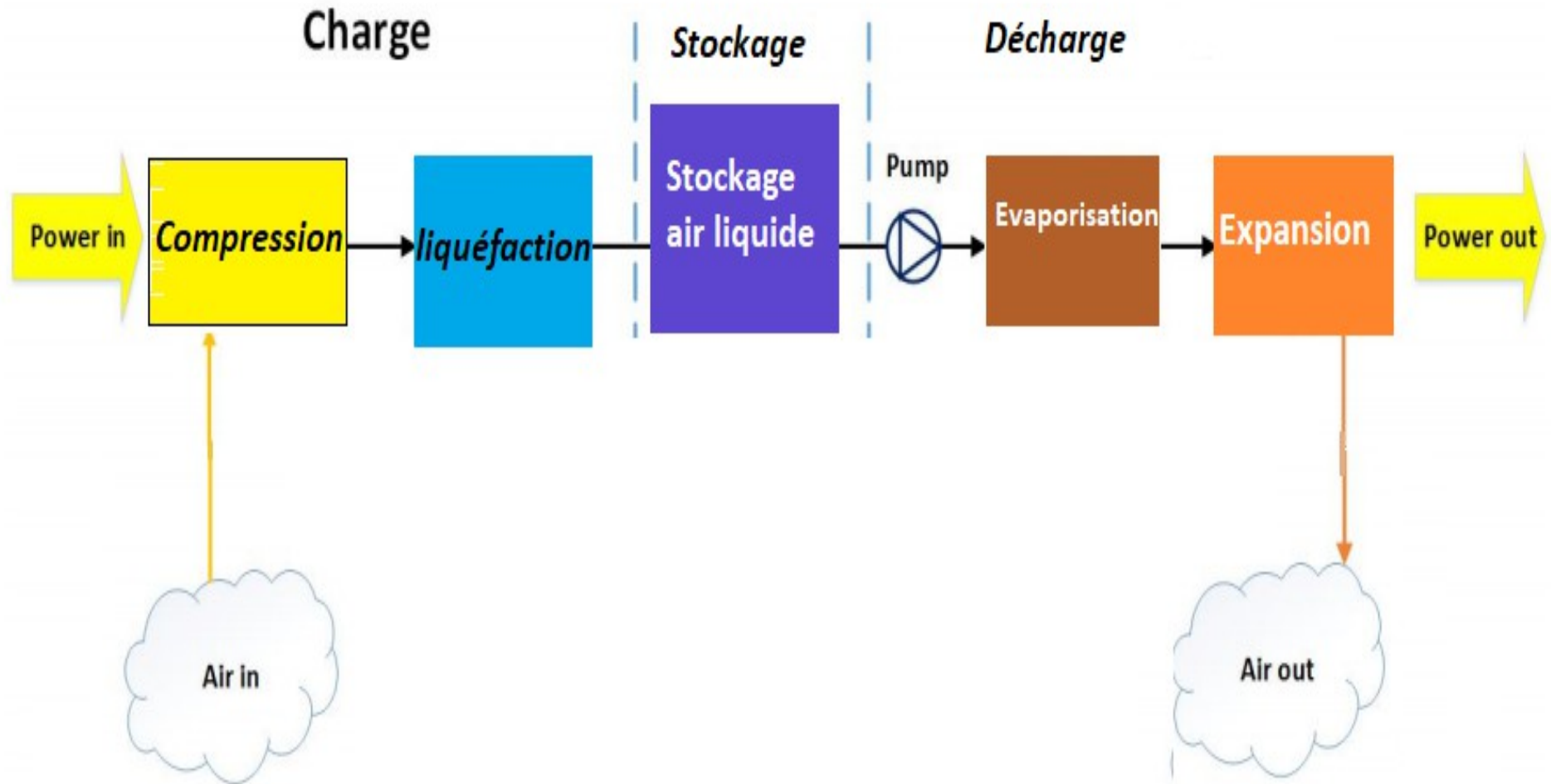
Graphique sur la demande et l'offre d'électricité en Californie.



Problématique: Comment peut-on stocker de l'énergie à l'aide de l'air liquide? Quelles sont les performances: efficacité et autonomie du système de stockage de l'énergie avec l'air liquide ? 3

# I/Introduction

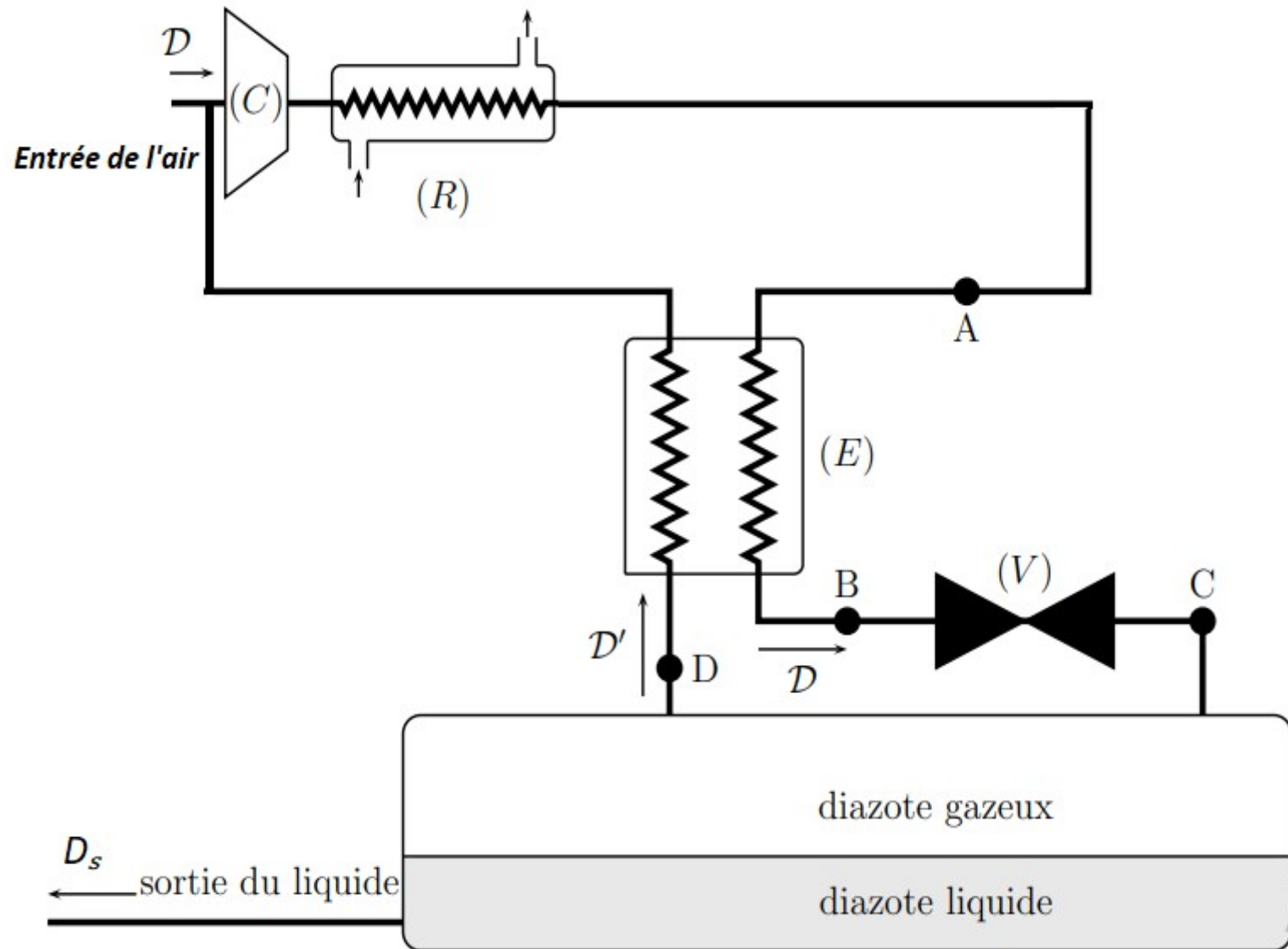
Les trois phases de fonctionnement du système sont: charge, stockage et décharge



## II/Cycle de charge

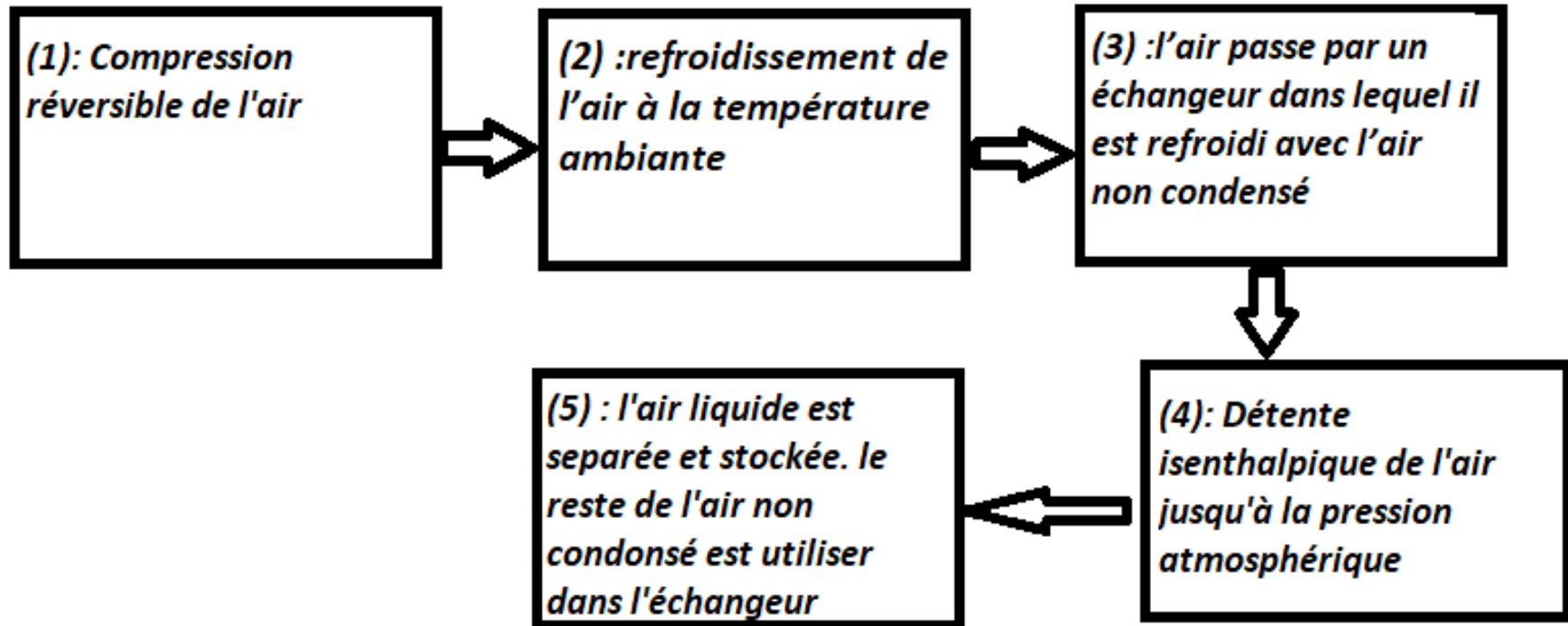
### 1/Introduction du cycle

## Procédé de Linde-Hampson de liquéfaction de l'air



## II/Cycle de charge

### 2/ Modélisation du cycle



## II/Cycle de charge

### 2/ Modélisation du cycle

-Soit  $D$ : Le débit massique de l'air et  $y$ : La fraction de l'air liquide

Le premier et le deuxième principe de la thermodynamique donne :

$$D \cdot (h_2 - h_1) - D \cdot T_{ext}(s_2 - s_1) = \dot{W}_c \quad (1)$$

$$T_{ext} \cdot D \cdot (s_3 - s_2) = -\dot{Q}_{ex} \quad (2)$$

$$D \cdot (h_4 - h_3) + D' \cdot (h_1 - h_5) = 0 \quad (3)$$

$$h_5 = h_4 = y \cdot h_{liq} + (1 - y) \cdot h_{vap} \quad (4)$$

$$D = D_s + D' \text{ et } D_s = yD, D' = (1 - y)D \quad (5)$$



## II/Cycle de charge

### 3/Analyse du cycle

Bilan : les équations (1)-(5)

-Fraction de l'air liquide :  $y = \frac{h_3 - h_1}{h_{liq} - h_1}$

-Energie consommée par liquéfaction d'un kg d'air:

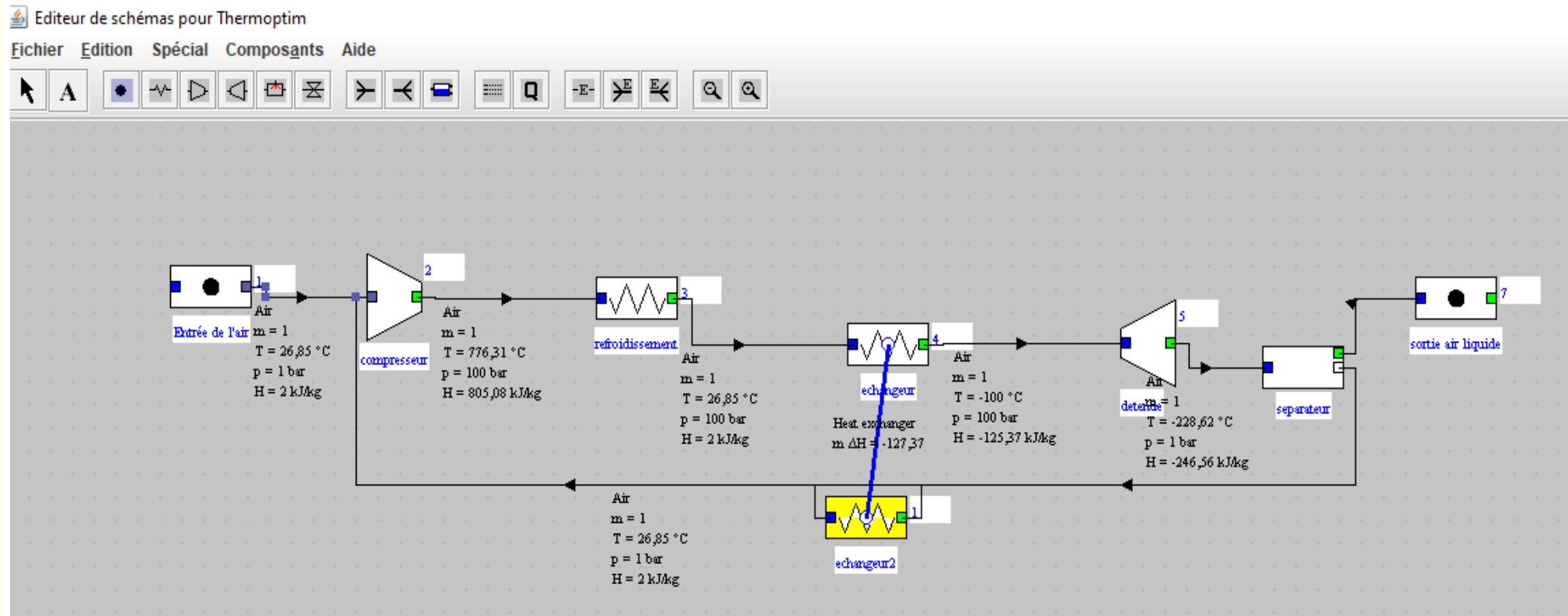
$$\frac{\dot{W}_{total}}{D_s} = \frac{(\dot{W}_c + \dot{Q}_{ex})}{y \cdot D} = \frac{h_2 - h_1 + T_{ext}(s_1 - s_3)}{y}$$



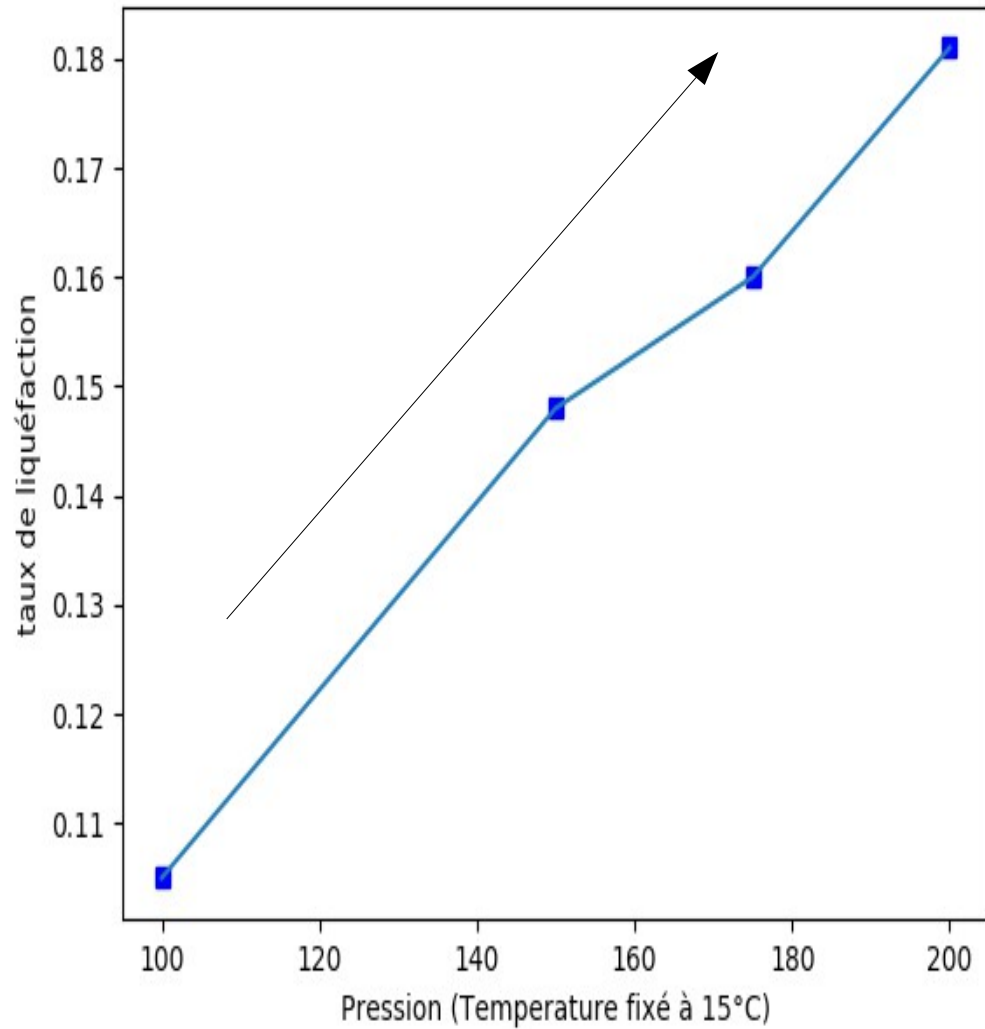
## II/Cycle de charge

### 3/Analyse du cycle

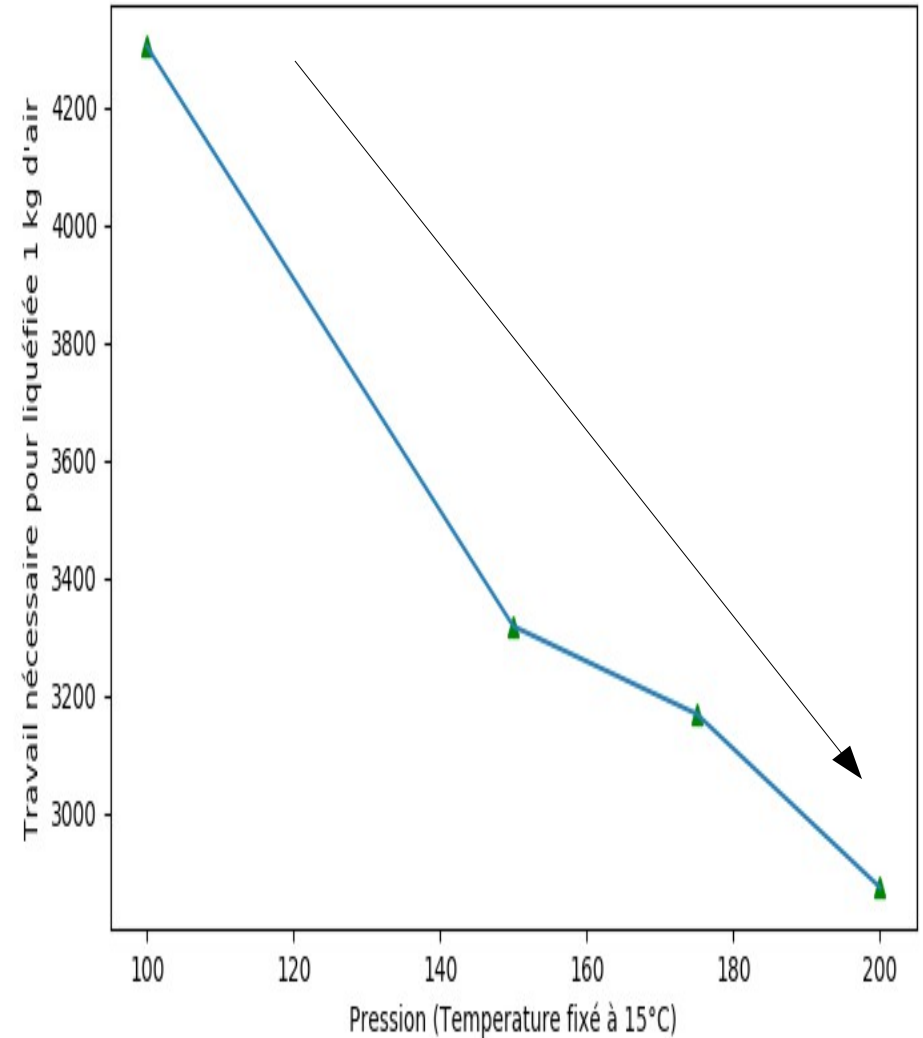
Logiciel de simulation utilisée est Thermoptim-UNIT



Profil d'évolution du taux de liquéfaction en fonction de la Pression



Profil d'évolution du travail nécessaire pour liquéfier un kg d'air en fonction de la Pression

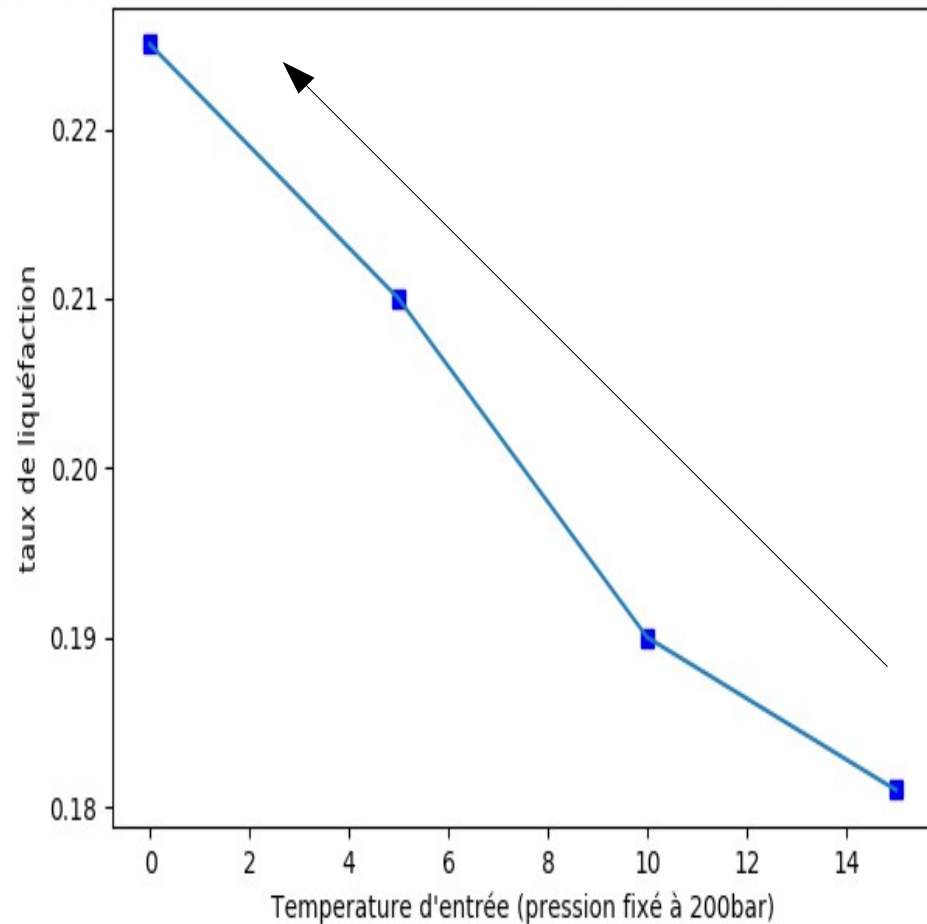


Code Annexe p22

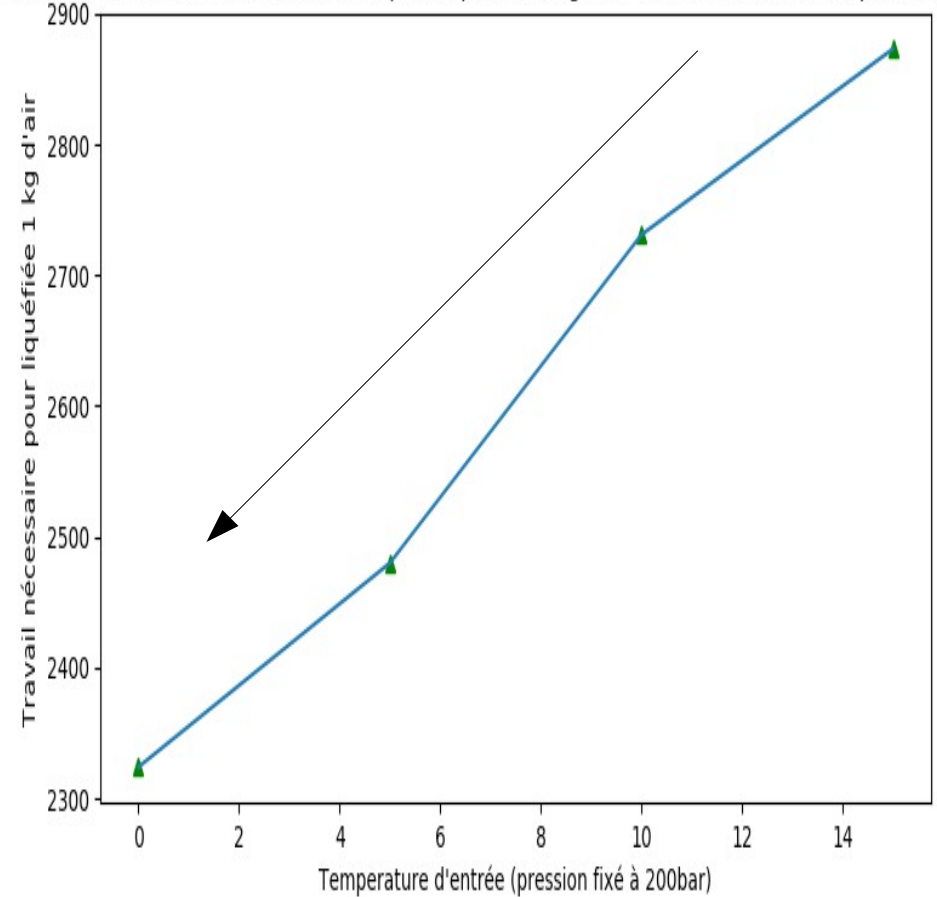
## II/Cycle de charge

### 3/Analyse du cycle

Profil d'évolution du taux de liquéfaction en fonction de la température d'entrée



Profil d'évolution du travail nécessaire pour liquéfier un kg d'air en fonction de la Température d'entrée

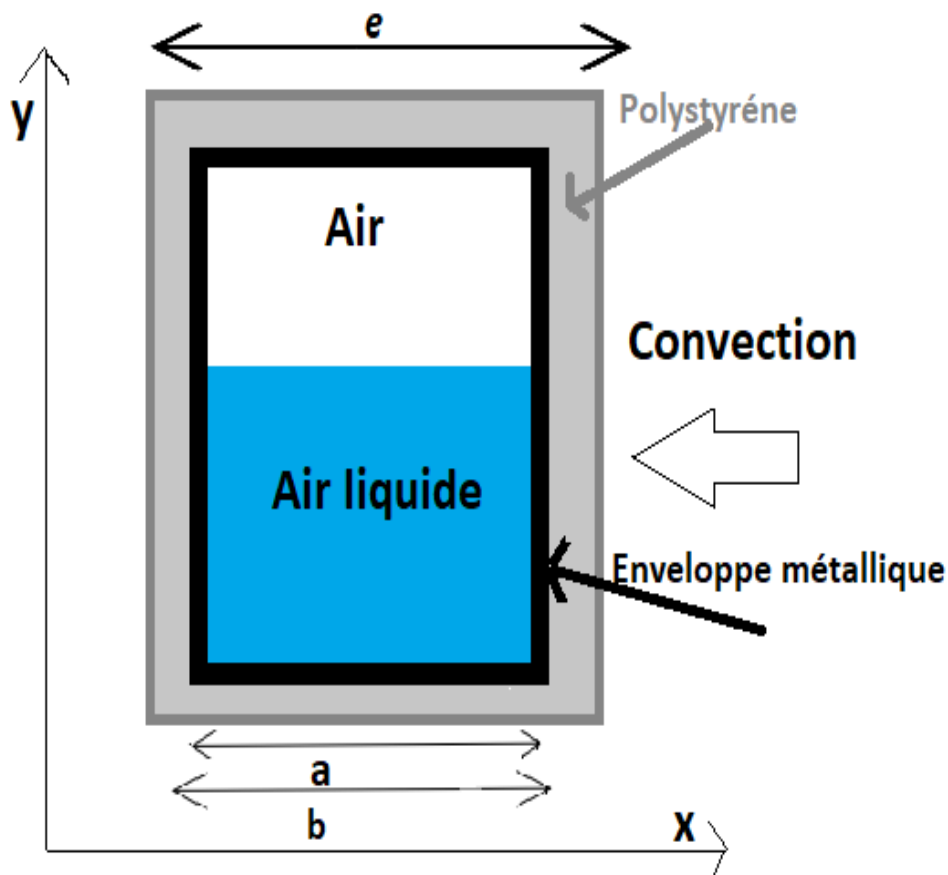


Code Annexe p22

### III/Stockage

#### 1/Modélisation du réservoir

L'air liquide est maintenant stockée dans un réservoir où il est maintenu jusqu'à ce qu'il y ait une demande d'électricité



Hypothèses:

D'après la loi de Fourier  $\vec{j} = -\lambda \overrightarrow{\text{grad}}(T)$

Equation de chaleur :

$$-\rho \cdot c \cdot \frac{\partial T}{\partial t} = \text{div}(\vec{j}) + \frac{4h}{d}(T_{\text{ext}} - T)$$

$\rho$  : Masse volumique,  $c$  : la capacité thermique,

$\vec{j}$  : Courant Thermique,  $T$  : Température du fluide,

$h$  : coefficient de convection thermique

D'après la loi de Fourier  $\vec{j} = -\lambda \overrightarrow{\text{grad}}(T)$

$\lambda$  : Conductivité thermique

$$-\frac{\partial T}{\partial t} = -\frac{\lambda}{\rho \cdot c} \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + \frac{4h}{\rho \cdot c \cdot d}(T_{\text{ext}} - T)$$

### III/Stockage

#### 1/Modélisation du réservoir

La convection est négligée dans la deuxième couche et dans le réservoir, ceci donne :

$$-\frac{\partial T}{\partial t} = -\frac{\lambda_3}{\rho_3 * c_3} \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + \frac{4h}{\rho_3 * c_3 * (e - b)} (T_{ext} - T)$$
$$\frac{\partial T}{\partial t} = \frac{\lambda_1}{\rho_1 * c_1} \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right)$$
$$\frac{\partial T}{\partial t} = \frac{\lambda}{\rho * c} \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right)$$

The diagram illustrates a cross-section of a storage tank. It consists of three main layers: a central blue region labeled 'Air liquide', a middle gray region labeled 'Polystyrène', and an outer white region labeled 'Air'. The total width of the tank is denoted by 'e'. The width of the 'Air liquide' region is 'a', and the width of the 'Polystyrène' region is 'b'. A coordinate system with 'x' and 'y' axes is shown. Arrows indicate 'Convection' and 'Enveloppe métallique'.

## III/Stockage

### 2/Résolution numérique

On résout l'équation de la chaleur à 2 dimensions en utilisant la méthode des différences finies.

On définit tout d'abord les pas :

$$x_n = x_0 + n \cdot \Delta x$$

$$y_m = y_0 + m \cdot \Delta y$$

$$t_i = t_0 + i \cdot \Delta t$$

De telle façon que :

$$T(x_{n+1}, y_m, t_i) = T^i_{n+1,m}$$

$$T(x_n, y_{m+1}, t_i) = T^i_{n,m+1}$$

$$T(x_n, y_m, t_{i+1}) = T^{i+1}_{n,m}$$

à l'aide d'un développement taylor d'ordre 1 en  $t_i$  :

$$T(x_n, y_m, t_i + \Delta t) = T(x_n, y_m, t_i) + \Delta t \frac{\partial T}{\partial t} + O(\Delta t) \longrightarrow \frac{\partial T}{\partial t} = \frac{T^{i+1}_{n,m} - T^i_{n,m}}{\Delta t}$$

à l'aide d'un développement taylor d'ordre 2 en  $x_n$  :

$$\begin{aligned} T(x_n + \Delta x, y_m, t_i) &= T(x_n, y_m, t_i) + \Delta x \frac{\partial T}{\partial x} + \Delta x^2 \frac{1}{2} \frac{\partial^2 T}{\partial x^2} + O(\Delta x^2) \\ T(x_n - \Delta x, y_m, t_i) &= T(x_n, y_m, t_i) - \Delta x \frac{\partial T}{\partial x} + \Delta x^2 \frac{1}{2} \frac{\partial^2 T}{\partial x^2} + O(\Delta x^2) \end{aligned} \longrightarrow \begin{aligned} \frac{\partial^2 T}{\partial x^2} &= \frac{T^i_{n+1,m} - 2T^i_{n,m} + T^i_{n-1,m}}{(\Delta x)^2} \\ \frac{\partial^2 T}{\partial y^2} &= \frac{T^i_{n+1,m} - 2T^i_{n,m} + T^i_{n-1,m}}{(\Delta y)^2} \end{aligned}$$

### III/Stockage

#### 2/Résolution numérique

La méthode des différences finies donne finalement :

$$\frac{T_{n,m}^{i+1} - T_{n,m}^i}{\Delta t} = D_{2,3} \left( \frac{T_{n+1,m}^i - 2T_{n,m}^i + T_{n-1,m}^i}{(\Delta x)^2} + \frac{T_{n+1,m}^i - 2T_{n,m}^i + T_{n-1,m}^i}{(\Delta y)^2} \right) - \beta (T_{ext} - T_{n,m}^i)$$

$$\text{Avec } \beta = \frac{4h}{\rho_1 * c_1 * (e - b)} \text{ et } D_1 = \frac{\lambda_1}{\rho_1 * c_1}$$

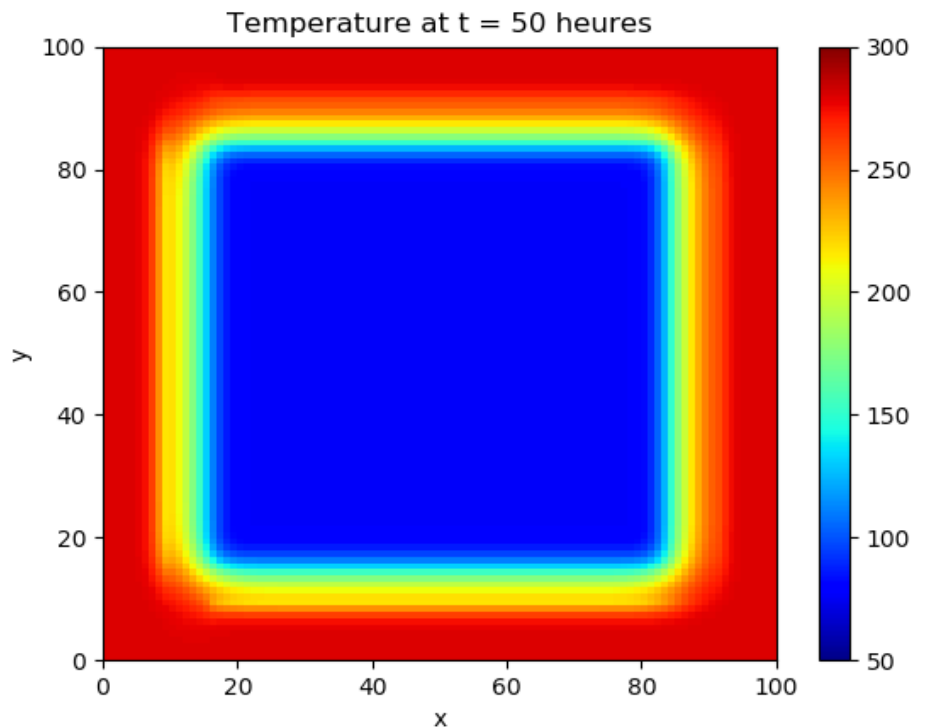
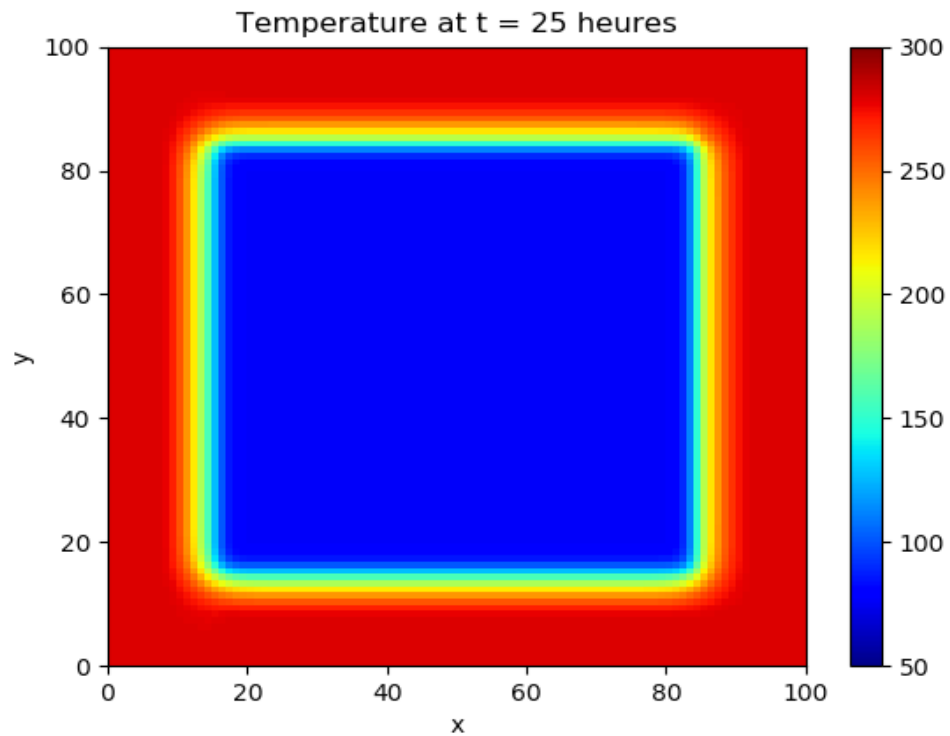
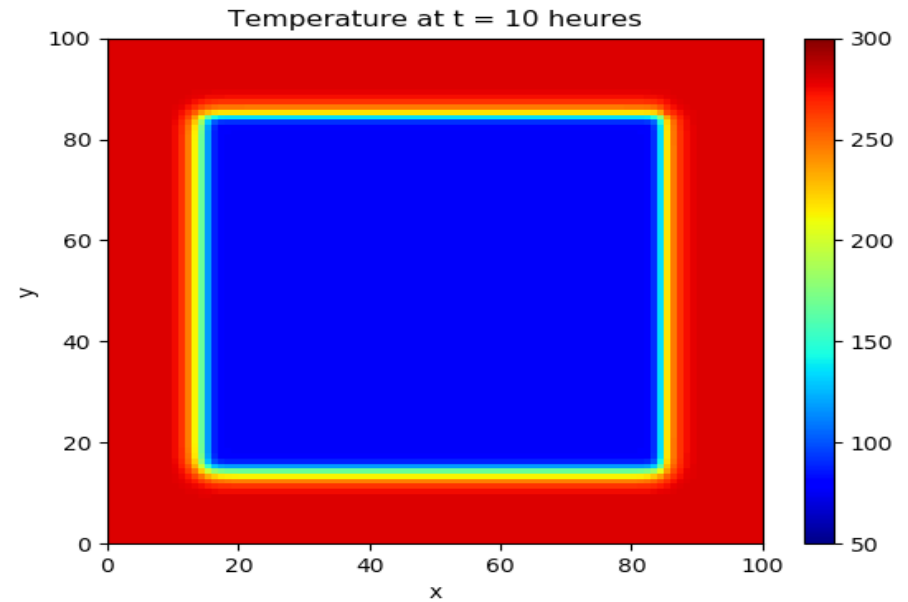
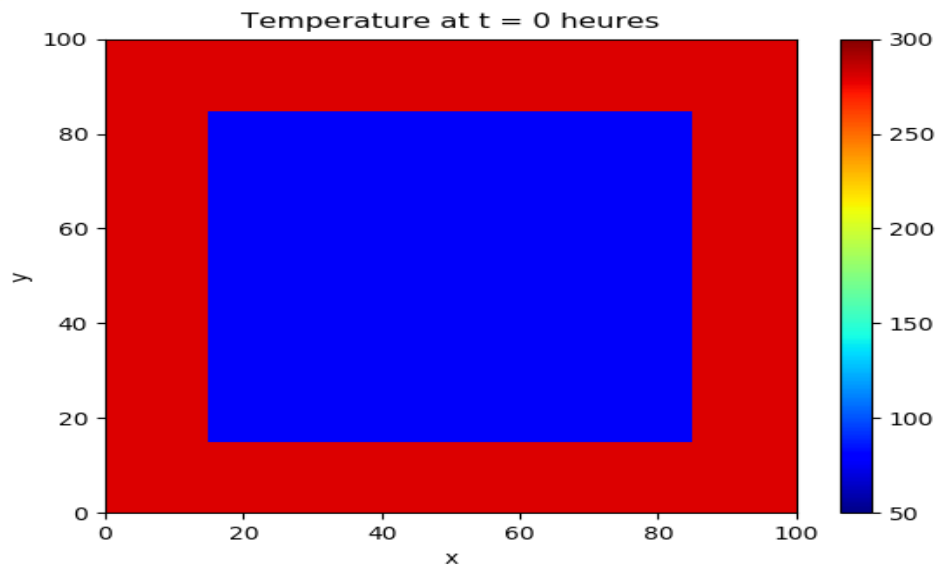
$$\frac{T_{n,m}^{i+1} - T_{n,m}^i}{\Delta t} = D_{2,3} \left( \frac{T_{n+1,m}^i - 2T_{n,m}^i + T_{n-1,m}^i}{(\Delta x)^2} + \frac{T_{n+1,m}^i - 2T_{n,m}^i + T_{n-1,m}^i}{(\Delta y)^2} \right)$$

$$D_{2,3} = \frac{\lambda_{2,3}}{\rho_{2,3} * c_{2,3}}$$



# III/Stockage

## 2/Résolution numérique



### III/Stockage

## 2/Résolution numérique

### Calcul du taux de vaporisation

*Taux d'évaporation est :*

$$V^- = \frac{\Phi_{total}}{V \cdot L \cdot \rho} * 100$$

$\Phi_{total}$  : *Flux total*

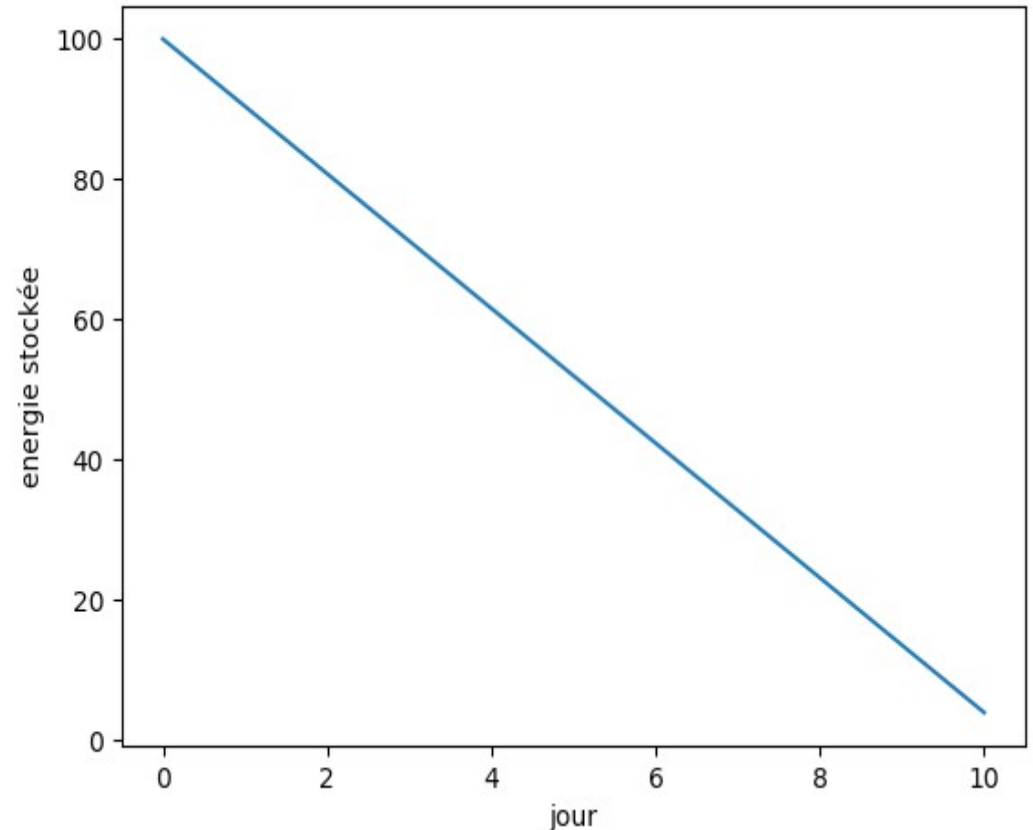
$V$  : *Volume du reservoir de l'air liquide*

$L$  : *la chaleur latente de vaporisation moyenne de l'air*

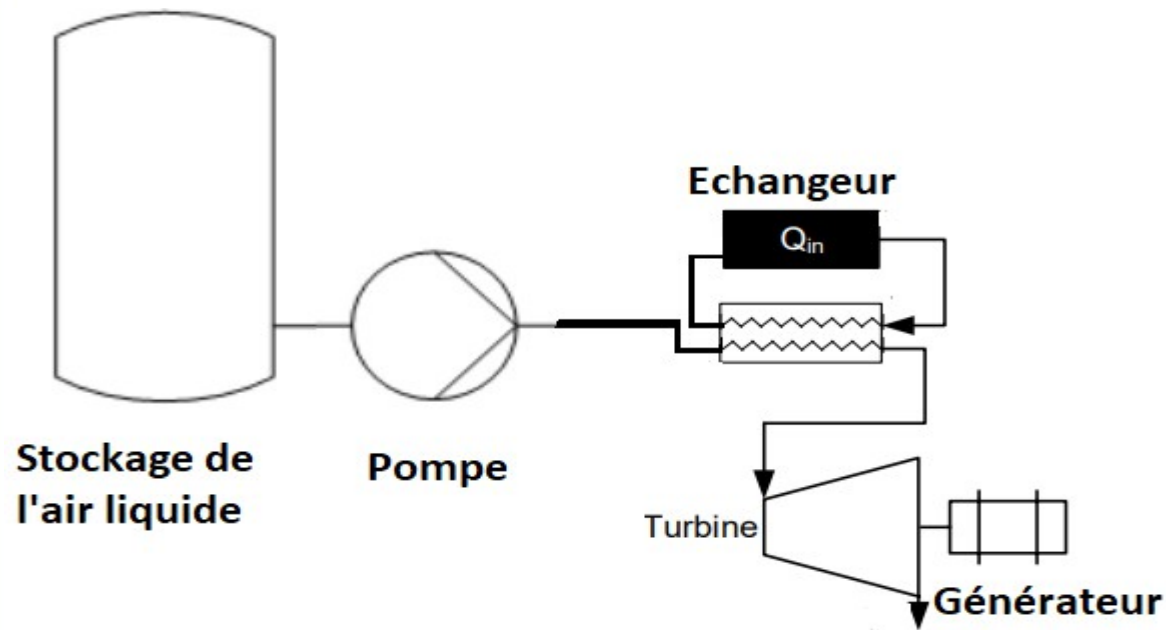
$\rho$  : *Masse volumique de l'air*

A.N :  $V^- = 0.4\% / h$

Profil d'évolution de l'energie stockée en fonction du temps



## IV/Extraction de l'énergie de l'air liquide

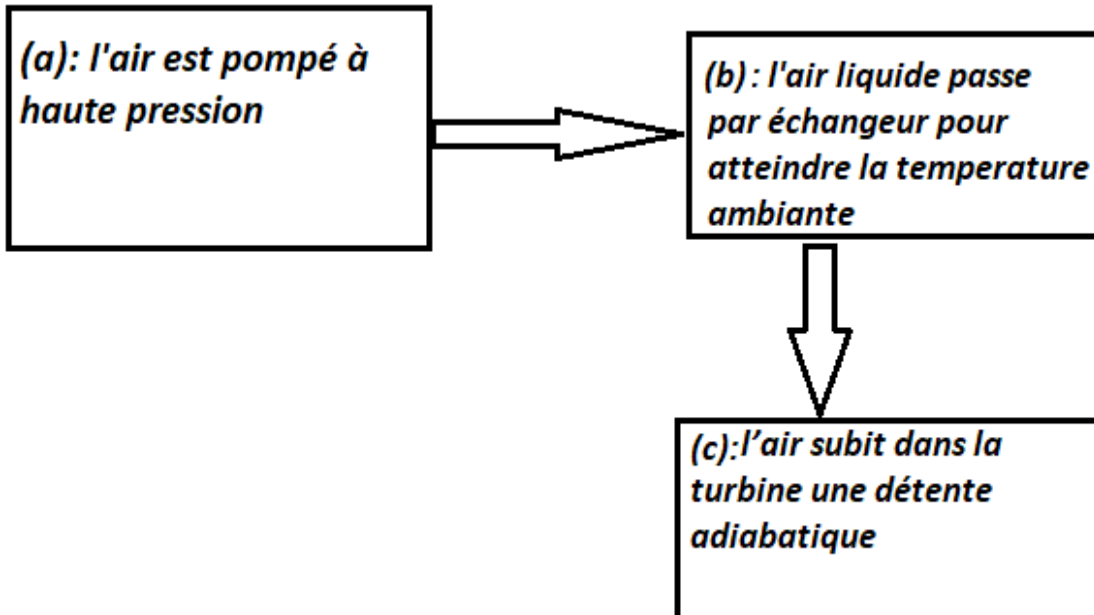


*On suppose qu'on a le même débit dans les deux cycles*

$$\dot{W}_p = D \cdot (h_a - h_0) \quad (1)$$

$$\dot{Q}_{ex} = D \cdot T_{ext}(s_b - s_a) \quad (2)$$

$$\dot{W}_T = D \cdot (h_c - h_b) \quad (3)$$

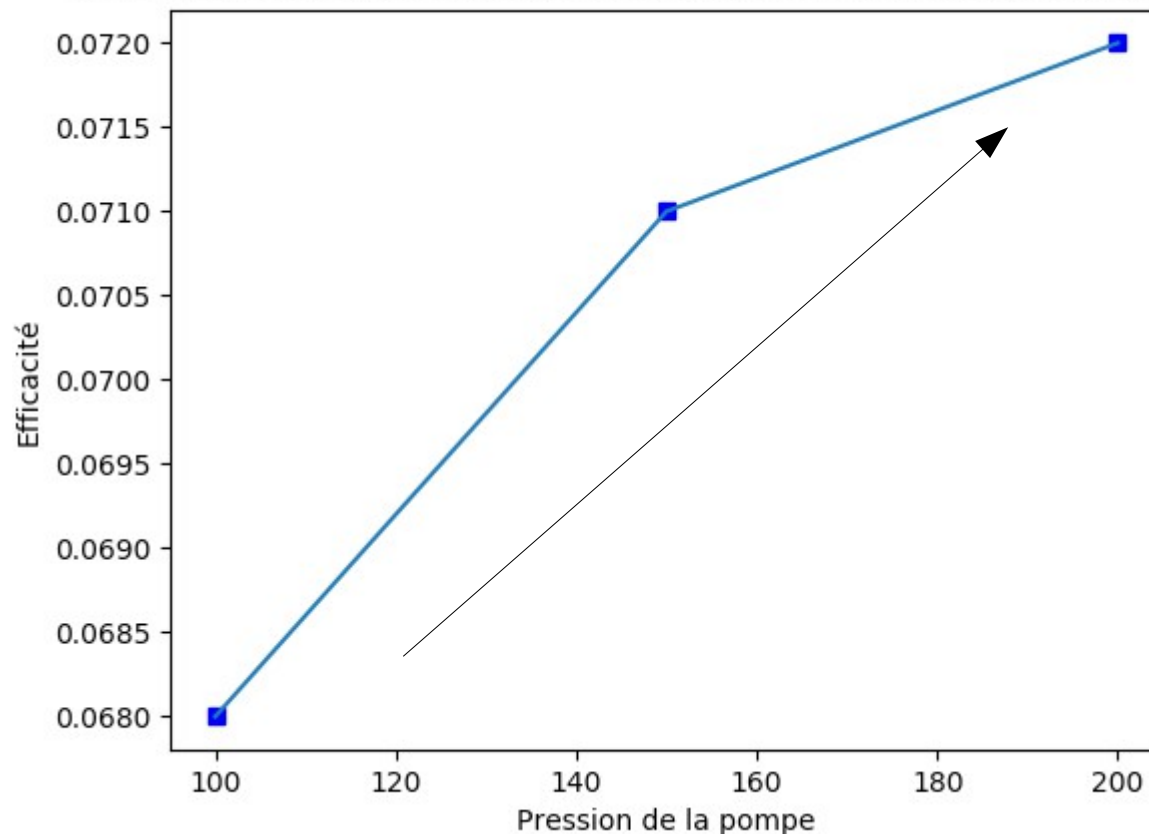


## IV/Extraction de l'énergie de l'air liquide

Efficacité total du système est donnée par:

$$\eta = \frac{\dot{W}_T}{\dot{W}_p + \dot{Q}_{ex2} + \frac{(\dot{W}_c + \dot{Q}_{ex})}{y}}$$

Profil d'évolution de l'efficacité en fonction de la Pression de la pompe



Code annexe page 22

## V/Conclusion

Les valeurs de l'efficacité sont très faibles

Causes :

- Energie consommée par liquéfaction d'un kg d'azote est très haute avec le cycle de Linde. Le cycle de Claude donne 700 KJ/kg qui est 1/3 du cycle de Linde.
- L'énergie utilisée au niveau des échangeur peut être réduite.
- Si on peut stocker la chaleur émise du cycle et la recycler, on peut atteindre une efficacité 0,7 comme décrit dans le papier de recherche Operating Range for a Combined, Building-Scale Liquid Air Energy Storage and Expansion System: Energy and Exergy Analysis

MERCI POUR VOTRE ATTENTION

# Annexe Code 1

```
import matplotlib.pyplot as plt

X=[100,150,175,200]
Y=[0.105,0.148,0.16,0.181]
Z=[4304,3317,3168,2873]

X1=[15,10,5,0]
Y1=[0.181,0.19,0.21,0.225]
Z1=[2873,2731,2480,2324]

plt.plot(X,Y,'bs')
plt.plot(X,Y)
plt.xlabel('Pression (Temperature fixé à 15°C)')
plt.ylabel('taux de liquéfaction')
plt.title('Profil d'évolution du taux de liquéfaction en fonction de la Pression ')

plt.show()
plt.plot(X,Z,'g^')
plt.plot(X,Z)
plt.xlabel('Pression (Temperature fixé à 15°C)')
plt.ylabel("Travail nécessaire pour liquéfiée 1 kg d'air")
plt.title("Profil d'évolution du travail nécessaire pour liquifier un kg d'air en fonction de la Pression")
plt.show()

plt.plot(X1,Y1,'bs')
plt.plot(X1,Y1)
plt.xlabel("Temperature d'entrée (pression fixé à 200bar)")
plt.ylabel('taux de liquéfaction')
plt.title("Profil d'évolution du taux de liquéfaction en fonction de la température d'entrée ")

plt.show()
plt.plot(X1,Z1,'g^')
plt.plot(X1,Z1)
plt.xlabel("Temperature d'entrée (pression fixé à 200bar)")
plt.ylabel("Travail nécessaire pour liquéfiée 1 kg d'air")
plt.title("Profil d'évolution du travail nécessaire pour liquifier un kg d'air en fonction de la Temperature d'entrée")
plt.show()
```



## Annexe Code 2

```
import numpy as np
import matplotlib.pyplot as plt

longeur = 100 #longeur du réservoir
max_iter_time = 100 # Nombres des itérations

delta_x = 1 # pas pour la simulation numérique

alpha1=0.003 #polystéréne
alpha2=0.016 #métal inoxydable
alpha3=0.0036 #azote
beta=0.08
#condition de stabilité
delta_t = (delta_x ** 2)/(4 * alpha2)
#Les valeurs de gamma pour chaque matériel
gamma1 = (alpha1 * delta_t) / (delta_x ** 2)
gamma2 = (alpha2 * delta_t) / (delta_x ** 2)
gamma3 = (alpha3 * delta_t) / (delta_x ** 2)
# Initialization de la matrice
u = np.empty((max_iter_time, longeur, longeur))
#condition initial de l'azote
u_initial_1 = 77
#condition initial de la couche extérieur
u_initial_2 = 280
# conditions initial
u[0,:,15:]=u_initial_1
u[:,0,15:]=u_initial_2
u[:,0,85:]=u_initial_2
u[:,15,0:]=u_initial_2
u[:,85,0:]=u_initial_2
u[:,15,0:]=u_initial_2
#constante pour le calcul
constant=beta*u_initial_2
constant0=beta+1
```

## Annexe Code 2

```
def calcul(u):
    for k in range(0, max_iter_time-1, 1):
        for i in range(longueur-1):
            for j in range(longueur-1):
                # pour chaque endroit on a une différente equation différentiel
                if (0<=i<=10 and 0<=j<=longueur-1) or (90<=i<=longueur-1 and 0<=j<=longueur-1) or (0<=j<=10 and 0<=i<=longueur-1) or (90<=j<=longueur-1 and 0<=i<=longueur-1):
                    u[k + 1, i, j] = gamma1*(u[k][i+1][j] + u[k][i-1][j] + u[k][i][j+1] + u[k][i][j-1] - 4*u[k][i][j]) - constant + constant0*u[k][i][j]
                if (10<=i<=15 and 10<=j<=90) or (85<=i<=90 and 10<=j<=90) or (10<=j<=15 and 10<=j<=90) or (85<=j<=90 and 10<=i<=90) :
                    u[k + 1, i, j] = gamma2*(u[k][i+1][j] + u[k][i-1][j] + u[k][i][j+1] + u[k][i][j-1] - 4*u[k][i][j]) + u[k][i][j]
                if (15<=i<=85 and 15<=j<=85):
                    u[k + 1, i, j] = gamma3*(u[k][i+1][j] + u[k][i-1][j] + u[k][i][j+1] + u[k][i][j-1] - 4*u[k][i][j]) + u[k][i][j]

    return u

def plot(u,k):
    plt.title(f"Temperature at t = "+str(k)+" heures")
    plt.xlabel("x")
    plt.ylabel("y")
    plt.pcolormesh(u, cmap=plt.cm.jet, vmin=50, vmax=300) #Associe à chaque température entre 50 et 300 une couleur spécifique
    plt.colorbar()
    plt.show()
    return plt

calcul(u)
def animate(k):
    plot(u[k], k)
```