

# Laboratorio Virtual en Ingeniería Química

**Mariano Martín, Borja Hernández, Antonio Sánchez, Edgar Martín-Hernández,  
Manuel Taifouris, Guillermo Galán**

Department of Chemical Engineering. University of Salamanca. Pza. Caídos 1-5, 37008  
Salamanca (Spain)

## 1.-Introducción

La situación de pandemia por COVID-19 que seguimos padeciendo ha afectado a todas las actividades. En particular, el confinamiento dio como resultado en el cierre de actividades comerciales y no esenciales y, durante cierto tiempo, escuelas y universidades de todo el mundo. Si bien la mayor parte de la enseñanza se trasladó a aulas virtuales utilizando plataformas como google meet, zoom, teams de Microsoft o Blackboard, los laboratorios no pueden sustituirse fácilmente. Se requieren laboratorios para enseñar grados técnicos y de ciencias en general. Se han presentado esfuerzos en disciplinas de ingeniería como la ingeniería eléctrica virtualizando el laboratorio usando MATLAB (Henker y Kelber 2021), laboratorios de control (Galán et al. ,2016, AACHen 2021). Entre ellos nos centramos en los laboratorios de ingeniería química. Está claro que el trabajo práctico en las unidades a escala piloto como reactores, columnas de destilación, etc. no puede ser sustituido por ningún modelo, ya que el manipular estos equipos también es parte de la formación del alumnado. Sin embargo, también es cierto que es posible desarrollar modelos de los diseños experimentales para que puedan ser utilizados y así generar datos experimentales computacionales para que los estudiantes evalúen los principios. Ya hay algunos trabajos disponibles en las disciplinas de transferencia de calor y fluidos (<http://ce-iitb.vlabs.ac.in/>) en transferencia de masa (<http://vmt-iitg.vlabs.ac.in/>, <http://www.learncheme.com/virtual-laboratories>), incluido el uso de simuladores de procesos (Williams et al., 2003) e ingeniería de reactores (<http://www.learncheme.com/virtual-laboratories>; Bell y Fogler, 1996). Sin embargo, los modelos están cerrados para los instructores y solo unos pocos representan unidades reales disponibles en el laboratorio debido a la complejidad en la representación de la termodinámica de los fluidos involucrados. La idea es tener un código, abierto para el instructor y cerrado para el alumno, para que pueda usarse para generar datos. Es importante no solo

comprender el sistema, sino también las fuentes de error para que el modelo no genere la misma respuesta exacta cada vez que se ejecute con el objetivo de reproducir la variabilidad en los resultados experimentales que un estudiante puede obtener en el laboratorio real. Además, el segundo objetivo de este trabajo es presentar herramientas que permitan al alumno evaluar el efecto de las condiciones de funcionamiento en el desempeño de cualquier unidad en particular. El objetivo es que el alumno piense en el efecto de los diferentes parámetros en el rendimiento o rendimiento de una unidad, apoyándose en un modelo que muestre el rendimiento, el perfil de la unidad.

Este proyecto ha desarrollado una herramienta basada en software abierto, con todos los modelos escritos en Python, para apoyar las enseñanzas de laboratorio en ingeniería química. Se presentan dos tipos diferentes de ejemplos, los destinados a ser utilizados para evaluar los principios, tipo de modelos unidireccionales, mientras que el segundo tipo apunta a ser utilizados como sustitutos virtuales (gemelos virtuales) de unidades de laboratorio para que el alumno pueda obtener datos y utilizarlos para el análisis de las unidades.

## **2.-Unidades virtuales**

Si bien ésta es una herramienta lista para ser utilizada, se puede ampliar con otros casos y ejemplos. Te presentamos aquí el Virtual ChemeLab 1.0. Su objetivo es cubrir la mayoría de las materias básicas del plan de estudios de Ingeniería Química, como transferencia de calor, mecánica de fluidos, termodinámica, ingeniería de reactores y operaciones unitarias. Consideramos dos tipos diferentes de prácticas: 1) aquellas orientadas a la comprensión de los conceptos básicos y los fenómenos utilizando la herramienta, donde el alumno tiene como objetivo ejecutar experimentos computacionales y analizar los resultados hacia la comprensión de la unidad y 2) aquellas que tienen como objetivo generar los datos experimentales que un estudiante estaría obteniendo en el laboratorio y usar esos datos como experimentales para realizar el análisis. La Tabla 1 muestra los dos objetivos y los temas tratados en cada uno. Téngase en cuenta que la herramienta se puede ampliar fácilmente con ejemplos adicionales. Los códigos se muestran en el apéndice de esta memoria.

Tabla 1.- Prácticas disponibles en el Virtual ChemLab 1.0

Temática	Evaluación de Principios	Gemelos virtuales
Transferencia de calor	Evaporación Intercambiador de doble tubo	Determinación de la conductividad
Mecánica de fluidos		Determinación del fanning
Termodinámica		Turbina de vapor Turbina de gas
Ingeniería de reactores	Cinética: Fermentador	Cinética
Operaciones Unitarias		Torre de Humidificación Destilación batch/por lotes

Cada una de estas prácticas se presenta considerando los **principios** que pretende evaluar, el **objetivo** para el que está escrito el modelo que simula la práctica así como el guion del trabajo a realizar por el estudiante. Se muestra la **tarea del instructor**, en cuanto a las decisiones que puede tomar y lo que debe solicitar al estudiante, el **tiempo** que se dedica a la práctica así como finalmente el análisis que puede servir como **fundamento teórico** de la práctica.

## 2.1.-Unidades para la evaluación de los fundamentos

Entre estas unidades hemos considerado reactores químicos, análisis cinético, evaporadores.

### 2.1.1.-Ingeniería de la reacción química: Reactores.

Principios: Esta sección contiene un ejemplo para evaluar la cinética de reacciones complejas, como la producción de etanol por fermentación.

Objetivo de los modelos: Evaluar el efecto de la temperatura y la composición de la materia prima en el rendimiento hacia productos. En particular, para el caso del etanol, es interesante evaluar el efecto de la inhibición que limita el rendimiento a etanol, el principal tema que tiene importantes implicaciones en la producción de bioetanol debido a la gran cantidad de agua que se debe eliminar para lograr etanol con una composición válida como combustible. Se pretende estudiar el efecto de temperaturas entre 30 °C y 40 °C y de concentraciones de glucosa (de 25 a 250 g/L) y de xilosa (de 25 a 250 g/L) en el rendimiento a etanol. Además para todos estos casos se debe contrastar el efecto de la inhibición para lo que el instructor debe cambiar el modelo.

Tarea del instructor: Solicitar un informe en el que se grafique y explique el efecto de la concentración de los sustratos en el rendimiento. Fije la inhibición a 0 en el modelo para repetir el estudio.

Duración: 2 clases de 2 horas.

Análisis:

**Síntesis de etanol:** Consideramos la producción de etanol a partir de materias primas lignocelulósicas. En este caso existen dos tipos de sustratos de azúcar, glucosa y xilosa, que sirven como materia prima. La concentración de especies juega un papel en la selección de los modelos que se deben tener en cuenta para el modelo (Martín, 2019). La cinética se caracteriza por la inhibición debida a la presencia de etanol en el líquido y la inhibición debida al sustrato, típicamente modelada mediante ecuaciones de tipo Monod. Basado en el modelo y el trabajo experimental de Krishman et al (1999) y Bertielsson et al (2008)

Cells

$$\mu_g = \frac{\mu_{m,g} \cdot S}{K_{s,g} + S + S^2 / K_{i,g}} \left( 1 - \left( \frac{P}{P_m} \right)^{\beta_g} \right)$$

$$\mu_x = \frac{\mu_{m,x} \cdot S}{K_{s,x} + S + S^2 / K_{i,x}} \left( 1 - \left( \frac{P}{P_m} \right)^{\beta_x} \right)$$

$$\frac{1}{X} \frac{dX}{dt} = \frac{G}{G+X} \mu_g + \frac{X}{G+X} \mu_x$$

Sustrate

$$-\frac{dS}{dt} = \frac{1}{Y_{X/S}} \frac{dX}{dt} + mX = \frac{1}{Y_{P/S}} \frac{dP}{dt}$$

$$-\frac{dS}{dt} = \frac{1}{Y_{P/S}} \frac{dP}{dt}$$

$$-\frac{dxylo}{dt} = \frac{1}{Y_{P/S}} (v_{E,x} X)$$

$$-\frac{dglu}{dt} = \frac{1}{Y_{P/S}} (v_{E,g} X)$$

Product:

$$v_{E,g} = \frac{v_{m,g} \cdot S}{K_{s,g} + S + S^2 / K_{i,g}} \left( 1 - \left( \frac{P}{P_m} \right)^{\gamma_g} \right)$$

$$v_{E,x} = \frac{v_{m,x} \cdot S}{K_{s,x} + S + S^2 / K_{i,x}} \left( 1 - \left( \frac{P}{P_m} \right)^{\gamma_x} \right)$$

$$\frac{1}{X} \frac{dP}{dt} = (v_{E,x} + v_{E,g})$$

With

$$\mu_{m,g} = 0.152 \cdot X^{-0.461}$$

$$\mu_{m,x} = 0.075 \cdot X^{-0.438}$$

$$v_{m,g} = 1.887 \cdot X^{-0.434}$$

$$v_{m,x} = 0.075 \cdot X^{-0.233}$$

(1)

Tabla 2. Parámetros del modelo

Parameter	Glucose Fermentation	Xylose fermentation
$\mu_m$ (h <sup>-1</sup> )	0.662	0.190
$v_m$ (h <sup>-1</sup> )	2.005	0.250
$K_S$ (g/L)	0.565	3.400
$K_S'$ (g/L)	1.342	3.400
$K_i$ (g/L)	283.700	18.100
$K_i'$ (g/L)	4890.000	81.300
$P_m$ (g/L)	95.4 for $P \leq 95.4$ g/L	
	1210.9 for $95.4 \leq P \leq 129$ g/L	510.040
$P_m'$ (g/L)	103 for $P \leq 103$ g/L	
	136.4 for $103 \leq P \leq 136.4$ g/L	60.200
$\beta$	1.29 for $P \leq 95.4$ g/L	
	0.25 for $95.4 \leq P \leq 129$ g/L	1.036
$\gamma$	1.42 for $P \leq 95.4$ g/L	
$m$ (h <sup>-1</sup> )	0.097	0.067
$Y_{P/S}$ (g/g)	0.470	0.400
$Y_{X/S}$ (g/g)	0.115	0.162

Se considera que el tamaño de los inóculos influye en las tasas máximas de crecimiento siempre que X sea inferior a 5 g / L

### 2.1.2.-Operaciones unitarias

#### **Intercambiador de calor de doble tubo.**

Principios: Se utiliza un intercambiador de calor de doble tubo para evaluar la transferencia de calor entre un vapor condensante y un líquido a calentar

Objetivo de los modelos: Evaluar el efecto del sentido de flujo ( en paralelo o en contracorriente) así como el caudal de agua a calentar y el tamaño de las tuberías dado por sus diámetros, en el coeficiente de transmisión de calor, Figura 1. Para ello se cuenta con un líquido a 90°C a ser calentado empleando vapor condensante a 233 °C que solo cambia de fase. Como variables podemos cambiar los flujos de ambos, con la precaución de que el fluido frío no supere la temperatura del vapor, que dará error en la solución. Flujos de vapor de 0.1-0.25 kg/s se pueden emplear así como flujos de líquido de 1 -5 kg/s. Los diámetros del tubo interno debe ser menor que el externo y se pueden tomar de tablas. El espesor del tubo interno se considera constante pero puede modificarse. El caudal y el tamaño del tubo marcan el régimen de flujo que son los que determinan el coeficiente de película del fluido interno.

Tarea del instructor: Solicitar un informe que represente y explique el efecto de las variables como el sentido de flujo, el caudal de agua así como de los diámetros de la tubería interna y externa en la eficiencia de la transferencia de calor.

Duración: 1 clase de 2 horas.

#### Análisis:

Como entrada se fija el flujo de vapor saturado a 233 °C, que será el fluido caliente calefactor. Con él determinamos, para un flujo de agua líquida a 90°C, la temperatura de salida

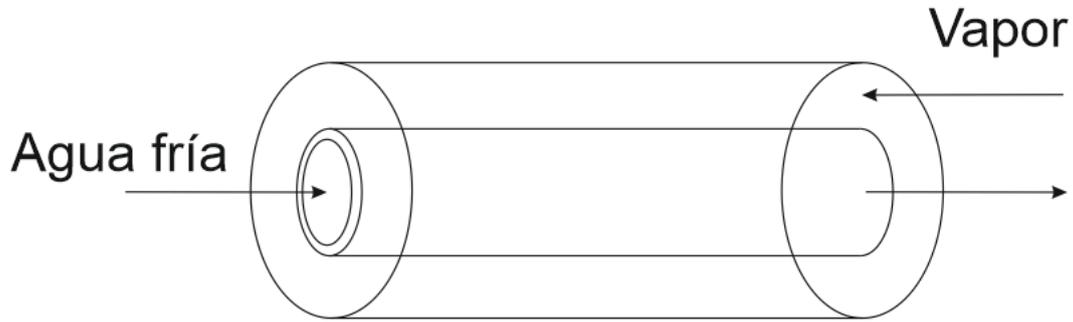


Figura 1.- Esquema del intercambiador de doble tubo

$$\begin{aligned}
 Q_{\text{aportado}} &= AU\Delta T_{lm} \\
 Flow_{\text{Vapor}} &= 0.1 - 0.25 \text{ kg / s} \\
 \lambda &= 1800; \text{ #kJ / kg} \\
 T_1 &= 233^\circ \text{ C;} \\
 T_2 &= 233^\circ \text{ C} \\
 t_1 &= 90^\circ \text{ C;} \\
 Q_{\text{aportado}} &= Flow_{\text{Vapor}} \cdot \lambda; \\
 t_2 &= \frac{Q_{\text{aportado}}}{(Flow_{in} \cdot Cp)} + t_1; \\
 L &= 0.265 \text{ m} \\
 \Delta T_{lm} &= \frac{(T_1 - t_2) - (T_2 - t_1)}{\ln \frac{(T_1 - t_2)}{(T_2 - t_1)}}
 \end{aligned} \tag{2}$$

$T_1, T_2$  = temperatura del fluido caliente, entrada y salida.

$t_1, t_2$  = temperatura del fluido frío, entrada y salida

El coeficiente global de transferencia de calor viene dado por la siguiente ecuación

$$\frac{1}{U} = \frac{1}{h_o} + \frac{1}{h_{od}} + \frac{d_o \ln \left( \frac{d_o}{d_i} \right)}{2k_w} + \frac{d_o}{d_i} \cdot \frac{1}{h_{id}} + \frac{d_o}{d_i} \cdot \frac{1}{h_i} \tag{3}$$

$U_o$  = Coeficiente global de transmisión de calor basado en área del tubo,  $\text{W/m}^2 \cdot \text{K}$

$h_o$  = Coeficiente de película externo,  $\text{W/m}^2 \cdot \text{K}$

$h_i$  = Coeficiente de película interno,  $\text{W/m}^2 \cdot \text{K}$

$h_{od}$  = Coeficiente de suciedad externo,  $\text{W/m}^2 \cdot \text{K}$

$h_{id}$  = Coeficiente de suciedad interno,  $\text{W/m}^2 \cdot \text{K}$

$k_w$  = Conductividad térmica del tubo,  $\text{W/m} \cdot \text{K}$

$d_i$  = diámetro del tubo interior, m.

$d_o$  = diámetro del tubo exterior, m.

Para el fluido a calentar en el tubo interior, la transferencia de calor depende de régimen de flujo que a su vez es función del caudal y del tamaño de la tubería.

<p>Flujo laminar (<math>100 &lt; Re &lt; 2100</math>):</p> $Nu = 1.86 \left[ (Re)(Pr) \left( \frac{d_e}{L} \right) \right]^{0.33} \left( \frac{\mu_B}{\mu_w} \right)^{0.14}$	<p>Flujo turbulento (<math>Re &gt; 10000</math>):</p> <p>No viscosos:</p> $Nu = 0.023(Re)^{0.8} (Pr)^{0.4} \left( \frac{\mu_B}{\mu_w} \right)^{0.14}$ <p>Viscosos:</p> $Nu = 0.027(Re)^{0.8} (Pr)^{0.4} \left( \frac{\mu_B}{\mu_w} \right)^{0.14}$ <p>Gases</p> $Nu = 0.0231(Pr)^{0.4} \left( \frac{\mu_B}{\mu_w} \right)^{0.14}$
<p>Donde <math>Re = \frac{\rho u_t d_e}{\mu} = \frac{G_t d_e}{\mu}</math>    <math>Nu = \frac{h_i d_e}{k_f}</math>    <math>Pr = \frac{c_p \mu}{k_f}</math></p>	

Cp = Capacidad calorífica, (J/kg.K); de = diámetro externo del tubo, (m); hi = coeficiente de película del lado del tubo, (W/m².K); Kf = conductividad del líquido, (W/m.K); L = longitud del tubo, (m); Nu = Nusselt Re = Reynolds; Pr = Prandtl; μB = viscosidad del fluido, (Pa.s); μw = viscosidad del fluido en la pared, (Pa.s)

Supondremos que las resistencias por suciedad son

$$r_i = 1.7 \cdot (1/10^4); \text{ W/m}^2 \cdot \text{K}$$

$$h_{od} = 1.3 \cdot (1/10^4); \text{ W/m}^2 \cdot \text{K}$$

Y la conductividad a través de la tubería

$$k_w = 16.3; \text{ W/m} \cdot \text{K}$$

El coeficiente de película del vapor condensante se considera constante

$$h_o = 12000.0; \text{ W/m}^2 \cdot \text{K}$$

$$h_i = Nu_i \cdot K_f / d_i$$

Tras calcular el coeficiente de película del lado del fluido a calentar, podemos calcular el coeficiente global de transferencia de calor

### **Evaporador**

Principios: Se utiliza un evaporador para concentrar una solución salina. La termodinámica de la solución salina determina las condiciones de funcionamiento de la cámara de evaporación.

Objetivo de los modelos: Evaluar el efecto de la velocidad de alimentación, la temperatura y la composición en las necesidades de vapor. Evalúe el efecto de la presión en la cámara de evaporación sobre la temperatura. Evaluar el efecto de la concentración final y la presión del vapor sobre el consumo de vapor. Estudiar el efecto de la presión del vapor sobre el consumo y la economía.

La práctica tiene como objetivo estudiar el comportamiento del evaporador para concentrar una disolución de NaOH. Tenemos como variables el caudal, temperatura y concentración de la alimentación así como las necesidades de vapor, la composición del líquido producto y la presión de la cámara de evaporación. Para un caudal fijo de alimentación y una composición objetivo de producto, 0.4w/w, varíese la concentración de entrada de 0.05-0.25 w/w y evalúe el comportamiento del evaporador para distintas presiones de la cámara de evaporación desde 150-300 mmHg. La temperatura de la alimentación y el caudal pueden empezar fijándose a 25°C y 10000kg/h, Figura 2.

Tarea del instructor: Solicitar un informe que represente y explique el efecto de la concentración del producto, la presión del vapor y la presión sobre el punto de ebullición en la cámara de evaporación y el consumo de vapor.

Duración: 1 clase de 2 horas.

Análisis:

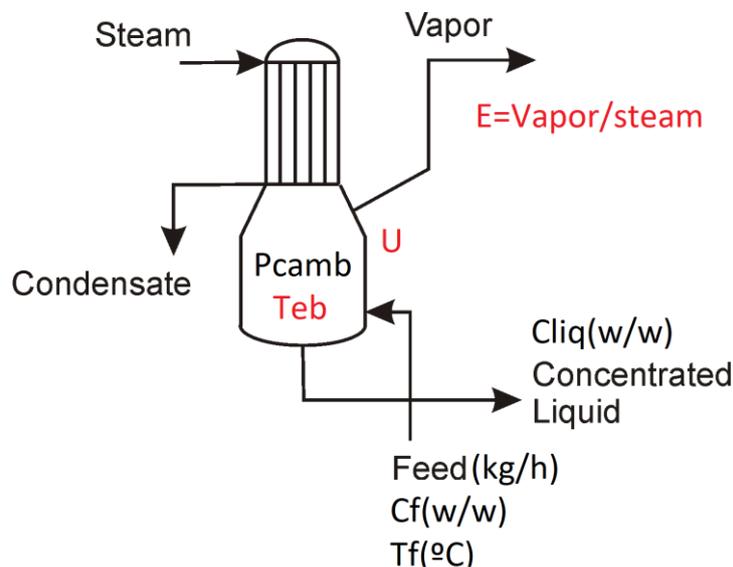


Figura 2.- Esquema del evaporador simple

Formulamos un balance de masa al soluto para determinar el caudal de líquido concentrado, L, y el agua evaporada, E:

$$F \cdot x_f = L \cdot x_L \quad (4)$$

$$F = L + E \quad (5)$$

A continuación, se formula un balance de energía al sistema de la siguiente manera:

$$W \cdot H_w + F \cdot h_f = W \cdot h_w + E \cdot H_e + L \cdot h_L \quad (6)$$

Las entalpías de las corrientes líquidas, la alimentación y el líquido concentrado,  $h_f$  y  $h_L$  respectivamente se correlacionan en función de su temperatura y composición y se encuentran disponibles en Martín (2019). Se ha de considerar el efecto del incremento ebulloscópico en el punto de ebullición. El efecto de la composición de la corriente sobre el punto de ebullición se determina mediante el diagrama de Dühring. Las propiedades del vapor saturado se correlacionan a partir de valores tabulados.

Así, el balance energético arroja:

$$\begin{aligned} &\lambda_{w,5atm} \\ &h_f(20^\circ C, 20\%) = \\ &H_e(H_{vsat, Teb_{PChamber}} + 0.46 \cdot \Delta e_W) \\ &h_L(Teb_{PChamber} + \Delta e_{W\%}, W\%) \end{aligned} \quad (7)$$

El coeficiente global de transferencia de calor se calcula como sigue

$$Q = W \cdot \lambda_w = U \cdot A \cdot \Delta T_{util} \Rightarrow U = \frac{W \cdot \lambda_w}{A \cdot \Delta T_{util}} \quad (8)$$

La economía del vapor,  $Ec$ , del proceso se calcula con la eq. (9):

$$Ec = \frac{E}{W} \quad (9)$$

## 2.2.-Gemelos digitales

En esta sección presentamos unidades como columnas de destilación discontinua, diseños de flujo de fluidos para la evaluación de la caída de presión, conductividad térmica, torres de enfriamiento, reactores para determinar la cinética. La idea básica es sustituir las unidades disponibles en los laboratorios por gemelos digitales que puedan ser utilizados por el estudiante para ejecutar experimentos computacionales. Además de modelar las unidades, la incertidumbre se incluirá en los resultados del modelo para mostrar la variabilidad para acercarse a un experimento en el laboratorio. Los números aleatorios se utilizan dentro de un error para incluir esa variabilidad.

### 2.2.1.-Propiedades de los materiales

#### Conductividad.

Principios: Calcule la conductividad de un metal que se calienta con una fuente de energía eléctrica. La mitad de la unidad está aislada y el resto no. Los extremos de la varilla se enfrían mediante un flujo de agua.

Objetivo de los modelos: evaluar la conductividad térmica de un metal a partir de datos de temperatura a lo largo de una varilla de metal. El modelo incluye las ecuaciones de Fourier y la correlación para incorporar el efecto de la temperatura sobre la conductividad.

Se trata de determinar la conductividad de una barra de metal, cuyo material ha sido elegido por el instructor así como la correlación que determina su efecto con la temperatura. Para distintos valores de energía proporcionada por una resistencia, 250W, 500W, 1000W, y para caudales de refrigeración de los dos extremos de la barra, uno aislado y otro no aislado (10-20 g/s), el gemelo digital determina el perfil de temperaturas a lo largo de la barra tanto en el lado aislado como a lo largo del no aislado. Empleando este perfil, y las ecuaciones de transferencia de calor por conductividad, se pide determinar la conductividad del material.

Tarea del instructor: seleccionar el material y la correlación para la conductividad. Establecer el error "experimental" de las medidas de temperatura y las pérdidas asumidas.

Duración: 2 clases de 2 horas.

#### Análisis

Para el tronco de cono que se muestra en la Figura 3 se plantea un balance de energía por conducción.

$$q = -kA \left( \frac{dT}{dx} \right)$$
$$A = - \left( \frac{RM - Rm}{L} \right) x + Rm \quad (10)$$
$$k = f(T) = aT^2 + bT + c$$

Suponemos un polinomio de segundo grado para incluir el efecto de la temperatura sobre k donde por defecto  $c=52.5 \text{ W/mK}$ ,  $b=0.02228 \text{ W/mK}^2$ . El centro de la varilla se calienta y los extremos se enfrían con corrientes de agua. El flujo de calor hacia cada lado se calcula mediante el incremento de temperatura

del agua. El lado izquierdo está aislado mientras que el lado derecho está abierto a la atmósfera. Se asignan 7 termopares a una distancia de 5 cm entre sí desde el inicio de la varilla, con un radio de  $R_M$  hasta el extremo con un radio de  $R_m$ . La ecuación diferencial de la eq. (10) se resuelve para el espacio entre termopares.

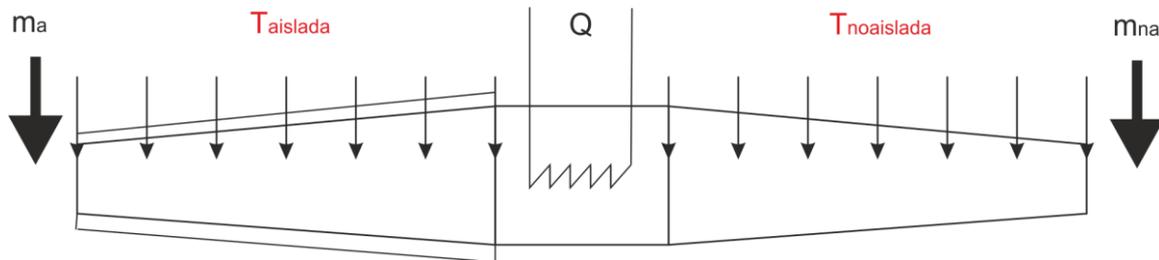


Figura 3.- Esquema del tronco cono

La temperatura leída por los termopares muestra un error. Los principios del equilibrio dan como resultado el hecho de que el flujo de calor no es el mismo en ambos tamaños, ya que el área aislada no recibe tanto para compensar las pérdidas.

### 2.2.2.-Mecánica de fluidos

#### **Pérdida de carga en tuberías**

Principios: Estudie el flujo a través de las tuberías. Se asume que el diseño contiene tubos rectos, medios codos y codos de 180°. Se pueden configurar diferentes instalaciones. Hay un error en la rugosidad de la tubería.

Objetivo de los modelos: Para distintos caudales de agua, empleando el gemelo digital proporciona la pérdida de carga de la instalación, determinar el coeficiente de Fanning.

Tarea del instructor: definir la estructura del diseño, la longitud y el diámetro de la tubería, el tiempo y el número de codos y la correlación para estimar el abanico y el error "experimental" en la caída de presión que se mide.

Duración: 1 clase de 2 horas.

Análisis:

La Figura 4 muestra un esquema general de la instalación donde los codos, tipo y número, son parámetros a decidir por parte del instructor. Por defecto las propiedades son las correspondientes al agua (densidad y viscosidad) y la rugosidad media ( $\epsilon$ ) es 0.002.



Figura 4.- Esquema de la instalación

$$\frac{P_2}{\gamma} - \frac{P_1}{\gamma} = hf_i$$

$$hf_i = f_i \frac{L_i}{d_i} \frac{v_i^2}{2g};$$

$$\frac{1}{\sqrt{f_i}} = -4.0 \log_{10} \left( \frac{\varepsilon / d_i}{3.7} + \frac{1.256}{\text{Re} \sqrt{f_i}} \right) \quad (11)$$

$$\text{Re} = \frac{\rho v_i d_i}{\mu}$$

### 2.2.3-Termodinámica.

#### **Ciclo Rankine. Turbina de Vapor.**

Principios: Esta sección tiene como objetivo evaluar el rendimiento de una turbina de vapor dentro de plantas de energía tanto las que emplean carbón como la energía nuclear o la energía solar concentrada.

Objetivo de los modelos: El objetivo consiste en evaluar la curva de eficiencia de la turbina bajo diferentes cargas en un ciclo Rankine con regeneración. Para ello se podrá ir variando la cantidad de agua del circuito (carga) desde 50 kg/s hasta 100 kg/s y se irá obteniendo las diferentes temperaturas del circuito. A partir de estas se tendrá que determinar el rendimiento de las turbinas y como va cambiando con diferentes cargas, para evaluar la relación entre la eficiencia y la carga. La relación que se define como carga hace referencia a la primera turbina y se define como:

$$\text{Carga} = \frac{\text{Flujo}_2}{\text{Flujo max}} \quad (12)$$

La carga máxima es de 100 kg/s

Para la práctica, se fijarán las siguientes presiones, aunque una vez obtenidos los resultados, se pueden variar para analizar cómo cambian las condiciones del sistema en función de las presiones de las turbinas.

Presión a la entrada de la turbina=100 bar

Presión a la salida de la turbina de alta presión=11 bar

Presión a la salida de la turbina de baja presión= 0.08bar

Datos adicionales:

Consultar la documentación técnica para saber cómo funciona el sistema

Consultar la siguiente bibliografía para obtener las correlaciones para las entalpías y entropías del sistema:

Erick León, Mariano Martín, Optimal production of power in a combined cycle from manure-based biogas, Energy Conversion and Management, 114, 2016, 89-99

Tarea del instructor: seleccionar la curva de eficiencia de la literatura. Establecer el error "experimental"

Duración: 1 clase de 3 horas.

Descripción del ciclo: Un ciclo Rankine consta de cuatro etapas, calentamiento del fluido, evaporación y sobrecalentamiento, expansión isentrópica en la turbina, condensación del vapor de escape y bombeo del líquido. Por lo general, se utiliza en plantas de energía de origen fósil, plantas nucleares y como parte de ciclos combinados. El modelo aquí representado consiste en calentar el líquido comprimido (6). La presión de funcionamiento en la turbina de alta presión varía de 40 a 126 bar para plantas termosolares. Se utilizan tres intercambiadores de calor para calentar primero, para evaporar (3) y finalmente para sobrecalentar el vapor (4). El vapor de alta temperatura y alta presión se expande en las diferentes turbinas. Aquí se modelan turbinas de alta y baja presión. La presión final de extracción de la turbina está en el rango de 0.05-0.19 bar, típicamente alrededor de 0.08 bar, y las turbinas permiten una fracción de líquido sin problemas mecánicos, alrededor del 8% o menos (5). La condensación se realiza con agua de refrigeración. Los principales problemas están relacionados con la termodinámica de la corriente para evaluar el rendimiento de las expansiones hacia la producción de energía. Si bien idealmente las expansiones en la turbina son isentrópicas, existe una eficiencia isentrópica asociada debido a la irreversibilidad que reduce la producción de energía, línea de puntos. Además, hay una serie de

expansiones en el medio y finalmente en las turbinas de baja presión. Las extracciones se utilizan para calentar la fase condensada antes de utilizar la energía de la fuente caliente. Se utilizan hasta 7 extracciones a nivel industrial (Martín y Martín, 2017). Las extracciones se refieren al ciclo de regeneración, Figura 5 (Montes et al., 2009).

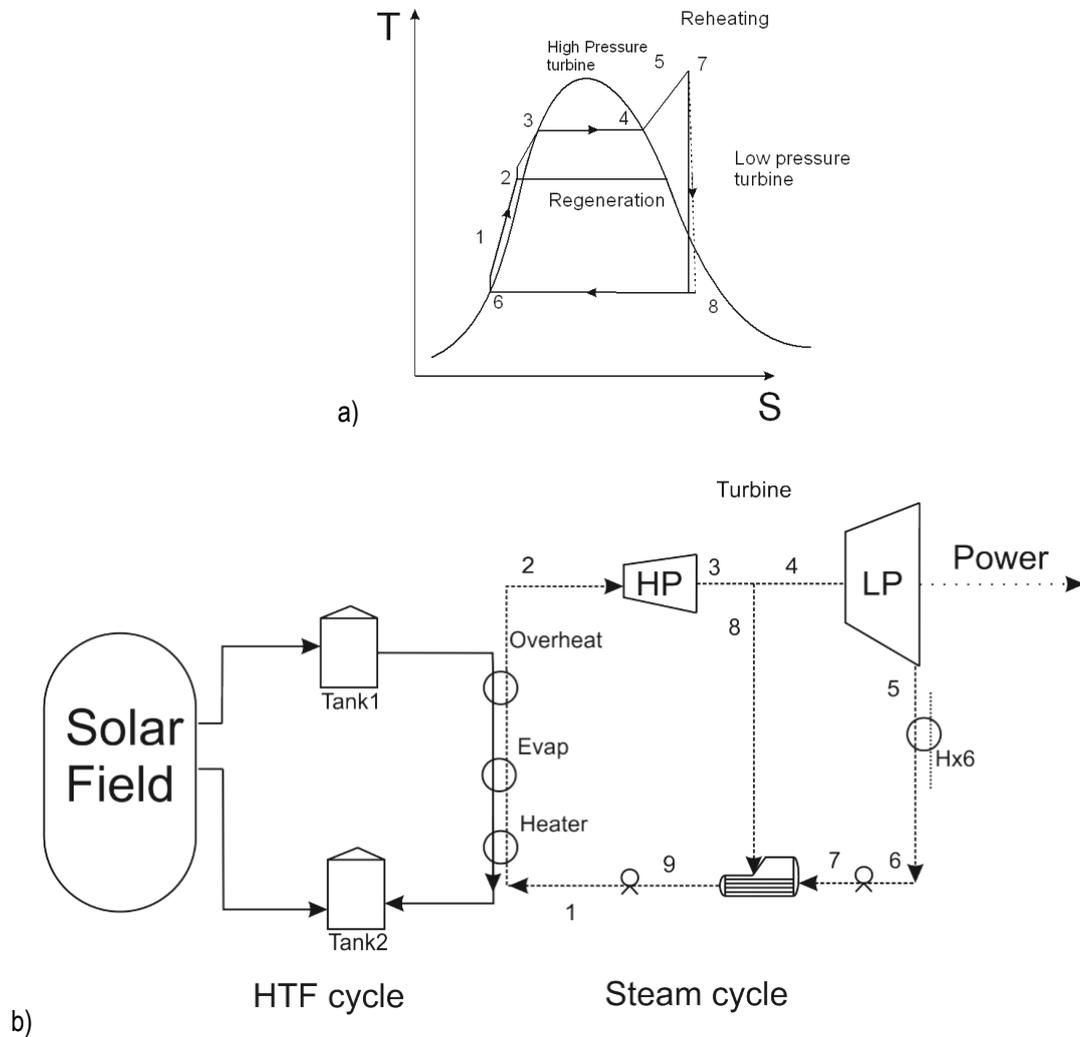


Figura 5.- Ciclo Rankine.

Análisis: El vapor es producido por la fuente caliente, ya sea una caldera o un fluido caloportador (HFT) como en el caso de las plantas de energía solar concentrada. Los balances de masa y energía de cada uno de los intercambiadores de calor se realizan de la siguiente manera, ecs. (13) - (14).

$$Q_{avail} = \begin{cases} Fossil \\ CSP \end{cases} \quad (13)$$

$$Q_{Avail} = Q_{toCycle} = \dot{m}_{(wa)} \cdot (H_{Out}(p, T_{S,out}) - H_{in}(p, T_{S,in})) \quad (14)$$

Las entalpías del vapor deben incluirse como modelos en función de la presión y la temperatura

como los desarrollados por Martín y Martín (2013). Es importante asegurar los gradientes de temperatura y evitar los cruces de temperatura en el caso de utilizar un gas de combustión o un HTF como agente calefactor eq. (4):

$$\begin{aligned} T_{S,out} + \delta &\leq T_{HTF,in} \\ T_{S,in} + \delta &\leq T_{HTF,out} \\ \delta &\in [5,10] \end{aligned} \quad (15)$$

Se emplea la ecuación de Antoine para verificar el estado de saturación de las corrientes a una determinada presión eq. (16):

$$P_{turb1} \cdot 760 = e^{\left( A_{(W_a)} - \frac{B_{(W_a)}}{C_{(W_a)} + T_{Evaporator}} \right)} \quad (16)$$

Se supone que la expansión del vapor en las diferentes turbinas tiene una eficiencia isentrópica,  $\eta_s$  eq. (17). Por lo tanto, la corriente que sale de la turbina de alta presión se puede calcular usando eq. (17) - (18)

$$\eta_{real} = \frac{H_{steam,out} - H_{steam,in}}{H_{steam,(isentropy)} - H_{steam,in}} \quad (17)$$

Donde

$$H_{steam,(isentropy)} = f(p_{out}, T_{out}^*) \quad (18)$$

$T^*$  representa la temperatura isentrópica tras la expansión. Así, la entropía,  $s$ , se mantiene constante:

$$s_{steam} = f(p_{in}, T_{in}) = f(p_{out}, T_{out}^*) \quad (19)$$

Después de cada expansión se emplea la correlación de Antoine (16) para validar que se trata de vapor sobrecalentado.

Si la turbina no funciona a plena carga, su eficiencia disminuye. Eq. (20) muestra el efecto de la carga sobre la eficiencia (Jude et al., 2009)

$$\varphi = (-1.0176 \cdot (\text{Load})^4 + 2.4443 \cdot (\text{Load})^3 - 2.1812 \cdot (\text{Load})^2 + 1.0535 \cdot \text{Load} + 0.701) \quad (20)$$

Se pueden igualmente emplear otras correlaciones y se deja al instructor la posibilidad de cambiarlas.

$$\eta_{real} = \eta_s \varphi \quad (21)$$

La potencia obtenida en cada expansión se determina con el eq. (22).

$$W = \dot{m}_{(W_a)} \cdot (H_{\text{steam,in}} - H_{\text{steam,out}}) \quad (22)$$

### Ciclo Brayton. Turbina de gas.

Principios: esta sección tiene como objetivo evaluar el rendimiento de una turbina de gas en ciclos combinados.

Objetivo de los modelos: Evaluar la curva de eficiencia de la turbina bajo diferentes cargas en un ciclo Brayton.

Se dispone de un ciclo Brayton para obtener energía de una fuente de calor. Para el ciclo Brayton se dispone de aire con una composición del 75.5% en Nitrógeno y 24.5% oxígeno en masa. El foco caliente suministra una potencia de 90.72 kW. El objetivo de la práctica es optimizar la relación de compresión de los compresores y la cantidad de aire del ciclo para obtener la máxima cantidad de potencia en la turbina. Para ello, se puede ir variando la relación entre los valores 1 a 10 y la masa de aire entre 50 y 150 kg/s. Los rendimientos politrópicos son de 1.4 (compresores) y 1.3 (turbina) y el rendimiento de los compresores y de la turbina de 0.8. A partir de las temperaturas suministradas por el simulador, determinar los distintos valores de energía producida por la turbina para distintos valores de relación de compresión y masa de gas, hasta determinar la configuración óptima. La compresión es multietapa (3 compresiones) mientras que la expansión es en una única etapa (una turbina). Además se adjuntan los datos correspondientes. El coeficiente de capacidad calorífica viene dado por la ecuación (23) y sus coeficientes están incluidos en la Tabla 3.

$$c_p = A + BT + CT^2 + DT^3 \quad (23)$$

Tabla 3. Coeficientes para el cálculo de la capacidad calorífica

	A	B	C	D
O <sub>2</sub>	19.795	7.34E-02	-5.60E-05	1.72E-08
N <sub>2</sub>	28.106	-3.68E-06	1.75E-05	-1.07E-08
CO <sub>2</sub>	31.15	-1.36E-02	2.68E-05	-1.17E-08

Tarea del instructor: establecer la eficiencia del compresor y la expansión

Duración: 1 clase de 3 horas.

Descripción del ciclo: El ciclo Brayton se utiliza en centrales eléctricas de turbinas de gas. Cuando se usa una combustión para calentar el gas, se usa un ciclo abierto, Figura 6a, para que el gas de

combustión, una vez expandido, no sea recirculado y comprimido. En caso de que el gas se caliente utilizando una fuente externa como la solar, se implementa un ciclo cerrado, Figura 6b, para que el fluido se recircule cerrando el ciclo. Últimamente se han alcanzado relaciones de presión de hasta 40 (General Electric). Para reducir el consumo de energía, la compresión se realiza en una configuración multietapa con enfriamiento intermedio. Del mismo modo, el uso de una expansión de varias etapas con recalentamiento también mejora la eficiencia del ciclo.

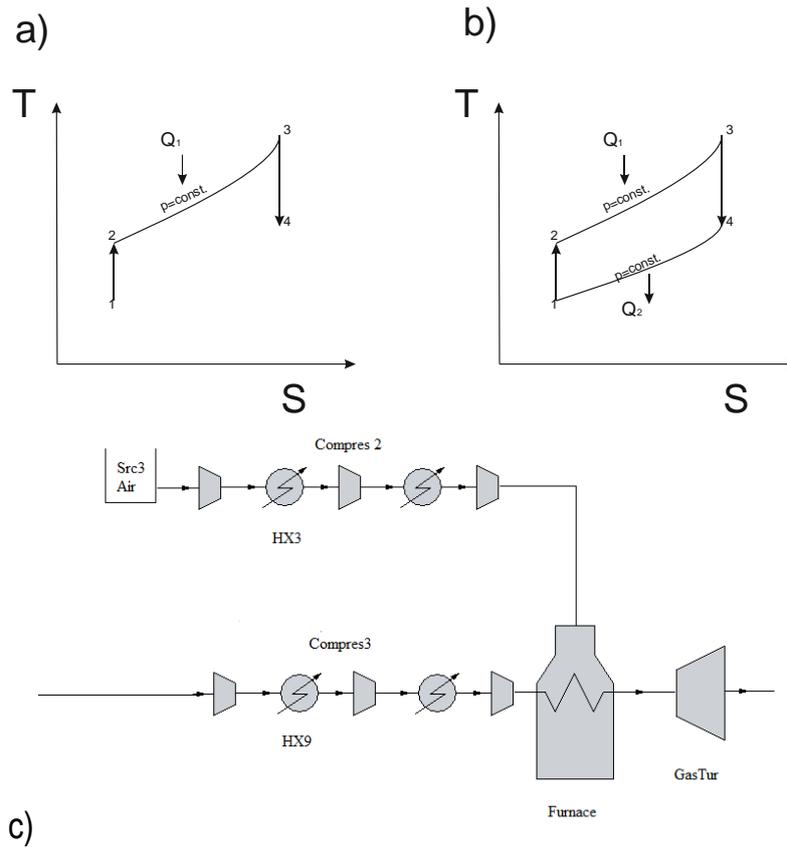


Figura 6.- Ciclo Brayton.

Análisis: Para modelar la turbina de gas asumimos tres etapas que incluyen

- 1) compresión del gas en 1-3 etapas con enfriamiento intermedio, eq. (24) - (25)

$$T_{out/compressor} = T_{in/compressor} + T_{in/compressor} \left( \left( \frac{P_{out/compressor}}{P_{in/compressor}} \right)^{\frac{z-1}{z}} - 1 \right) \frac{1}{\eta_c} \quad (24)$$

$$W_{(Compressor)} = n(F) \cdot \frac{R \cdot z \cdot (T_{in/compressor})}{((MW)(z-1))} \frac{1}{\eta_c} \left( \left( \frac{P_{out/compressor}}{P_{in/compressor}} \right)^{\frac{1}{n} \frac{z-1}{z}} - 1 \right) \quad (25)$$

Donde T corresponde a la temperatura, P a la presión, F al flujo másico, n al número de etapas de compresión, z al coeficiente politrópico (1,4), MW al peso molecular de la corriente comprimida y  $\eta_c$  a la eficiencia de la compresión (0,85). Después de cada etapa de compresión, pero antes del calentamiento, la corriente se enfría. Las necesidades totales de enfriamiento se calculan de la siguiente manera, eq. (26):

$$Q(HX) = (n-1) \sum_i f c_i \cdot \int_{T_{out/compressor}}^{T_{in/compressor}} cp dt \quad (26)$$

2) El gas que sale de la etapa de compresión se calienta con la energía del colector; asumimos que está a una temperatura lo suficientemente alta como para calentar el gas que sale de la etapa de compresión, eq. (27)

$$\begin{aligned} Q_{(collector)} &= DNI_i \cdot A_{field} \cdot \eta_{field} \\ A_{field} &= N_{Heliostat} \cdot A_{Heliostat} \cdot \eta_{Heliostat} \\ H_{Tower} &= 82.6 + 0.2552 \cdot 10^{-3} \cdot Q_{(collector)} \text{ (kW)} \\ Q_{Prod} &= Q_{feed} + Q_{(collector)} \end{aligned} \quad (27)$$

El gas caliente se expande asumiendo una sola etapa, que se modela como eqs. (28) - (29):

$$T_{out/turbine} = T_{in/turbine} + T_{in/turbine} \eta_t \left( \left( \frac{P_{out/turbine}}{P_{in/turbine}} \right)^{\frac{z-1}{z}} - 1 \right) \quad (28)$$

$$W_{(turbine)} = (F) \frac{R \cdot z \cdot (T_{in/turbine})}{((MW)(z-1))} \eta_t \left( \left( \frac{P_{out/turbine}}{P_{in/turbine}} \right)^{\frac{z-1}{z}} - 1 \right) \quad (29)$$

La definición de variables es similar al caso de los compresores. Para un ciclo cerrado  $P_{out/turbine}$  es la misma que  $P_{in/compressor}$  y  $P_{out/compressor}$  es igual a  $P_{in/turbine}$ .

#### 2.2.4.-Cinética de reacción.

Principios: Evaluar la cinética de reacción en un reactor discontinuo en fase líquida para una reacción dada.

Objetivos de los modelos: Determinar los coeficientes estequiométricos, los órdenes de reacción y los parámetros propios de la ecuación de Arrhenius para una reacción genérica  $aA+bB \rightarrow \text{Productos}$ .

Se dispone de un reactor discontinuo en fase líquida donde se pueden modificar las concentraciones iniciales de las especies A y B, la temperatura de trabajo y el tiempo final del experimento. Como en una experimentación real, los estudiantes obtendrán los perfiles de concentración frente al tiempo para distintos tiempos antes del valor final acotado por el estudiante.

Tarea del instructor: fijar a priori los coeficientes estequiométricos de la reacción, los órdenes de reacción, así como, los valores de los parámetros de la ecuación de Arrhenius.

Duración: dos clases de tres horas.

Análisis: La expresión genérica de la cinética de la reacción  $aA+bB \rightarrow \text{Productos}$  se expresa como:

$$r_A = k [C_A]^{n_A} [C_B]^{n_B} \quad (30)$$

Donde la relación entre la constante cinética y la temperatura puede ser explicada usando la ecuación de Arrhenius:

$$k = A \exp\left(\frac{-E_a}{RT}\right) \quad (31)$$

Para determinar los diferentes parámetros, la reacción se lleva a cabo en un reactor discontinuo. Asumiendo mezcla perfecta, fase líquida y operación isoterma, la operación del sistema puede ser descrita a través de la siguiente ecuación:

$$\frac{dC_A}{dt} = -r_A \quad (32)$$

Combinando la expresión cinética y la ecuación que describe el funcionamiento del reactor:

$$\frac{dC_A}{dt} = -k [C_A]^{n_A} [C_B]^{n_B} \quad (33)$$

Para el componente B, esta expresión se transforma en:

$$\frac{dC_B}{dt} = -\left(\frac{b}{a}\right) \left(k [C_A]^{n_A} [C_B]^{n_B}\right) \quad (34)$$

Usando esta ecuación es posible determinar los perfiles de concentración a lo largo del tiempo, con unas determinadas concentraciones iniciales dadas de A y B, la temperatura (para fijar la constante cinética) y el tiempo final de cada experimento.

El instructor fijara de antemano, los parámetros de la ecuación de Arrhenius, los coeficientes estequiométricos y el orden de reacción. Los estudiantes, utilizando el perfil de concentración obtenido y a través de estas ecuaciones, deberán determinar estos parámetros.

Los coeficientes estequiométricos son simples de determinar. Si se fija el coeficiente de la especie A a 1, el coeficiente de la especie B puede ser determinado como el cociente de la especie A y la B.

Para el orden de reacción, se asumo que este puede tomar los valores típicos de 0, 1 o 2.

En la reacción están involucradas dos especies. Por lo tanto, para determinar los órdenes de reacción de cada una, es necesario introducir un exceso de uno de los componentes para considerar que la concentración de esta especie permanece constante con el tiempo.

$$C_B \gg C_A \rightarrow \frac{dC_A}{dt} = -k_{\text{exp}} [C_A]^{n_A} \quad ; \quad k_{\text{exp}} = k [C_B]^{n_B} \quad (35)$$

Por lo tanto, se puede analizar el orden de reacción del componente A.

Si se asumo orden 0:

$$\frac{dC_A}{dt} = -k_{\text{exp}} \rightarrow C_A - C_A^0 = -k_{\text{exp}} t \rightarrow C_A = C_A^0 - k_{\text{exp}} t \quad (36)$$

Por lo tanto, si la concentración de A es representada frente al tiempo, se obtendría una línea recta con una ordenada igual a la concentración inicial y como pendiente  $-k_{\text{exp}}$ .

Si el orden 1 es analizado:

$$\frac{dC_A}{dt} = -k_{\text{exp}} [C_A] \rightarrow -\ln \left( \frac{C_A}{C_A^0} \right) = k_{\text{exp}} t \quad (37)$$

En este caso, si el logaritmo de la concentración de la especie A es representada frente al tiempo, se obtiene una línea recta con una ordenada igual a 0 y como  $k_{\text{exp}}$  pendiente.

Finalmente, si se analiza el orden 2:

$$\frac{dC_A}{dt} = -k_{\text{exp}} [C_A]^2 \rightarrow \frac{1}{C_A} - \frac{1}{C_A^0} = k_{\text{exp}} t \rightarrow \frac{1}{C_A} = k_{\text{exp}} t + \frac{1}{C_A^0} \quad (38)$$

Si la inversa de la concentración se representa frente al inverso del tiempo, se obtendría una línea recta con pendiente  $k_{\text{exp}}$  y como ordenada la inversa de la concentración inicial.

El mismo procedimiento se repite para determinar el orden de reacción de B. Para calcular los parámetros de la ecuación de Arrhenius, se llevan a cabo diferentes experimentos a diferentes temperaturas. Como los órdenes de reacción son conocidos, el valor de la contante cinética se puede

obtener. Un conjunto de temperaturas y constante cinéticas pueden ser obtenidos. Con estos datos es posible determinar la energía de activación y el factor pre-exponencial:

$$k = A \exp\left(\frac{-E_a}{RT}\right) \rightarrow \ln(k) = \ln(A) - \frac{E_a}{R} \frac{1}{T} \quad (39)$$

Por lo tanto, la ordenada en el origen es el logaritmo natural del factor pre-exponencial y la pendiente el cociente entre la energía de activación y la constante de los gases R.

#### 2.2.4.-Operaciones de separación

##### **Destilación por lotes de una mezcla agua-etanol.**

Principios: evaluar el rendimiento de una destilación por lotes para obtener un producto de una pureza determinada,

Objetivo de los modelos: Reproducir el funcionamiento de la columna de destilación proporcionando como salida la calidad del producto y la cantidad de destilado y residuo para que el alumno evalúe el número de bandejas teóricas de la columna que son una variable que el instructor puede decidir.

El gemelo digital reproduce el funcionamiento de una columna de destilación en discontinuo para la separación de una mezcla etanol-agua. No se conoce el número de etapas con las que está trabajando y las variables con las que podemos evaluar el funcionamiento de dicha columna son la cantidad en kmol de alimentación (200-350 kmol), la composición de la misma en etanol (0.08-0.15 en fracción molar) así como la relación de reflujo (de 1.5 a 3.5) y la pureza que puede alcanzar el destilado (de 0.8 a 0.88). El modelo calculará tanto la cantidad de destilado como de residuo así como la composición de éste. Con estos datos y los fundamentos de la operación de las columnas discontinuas el alumno debe evaluar el número de etapas con las que está operando la columna de modo que se consigan los productos predichos por el gemelo digital.

Tarea del instructor: seleccione la configuración de la columna, el número de platos y el error "experimental".

Duración: 2 clases de 3 horas.

Análisis: Para una columna de destilación dada con un número prefijado de bandejas, se utiliza para separar una mezcla de agua y etanol en condiciones de reflujo constante para obtener un destilado de

cierta pureza en etanol a partir de una materia prima de mezcla con una pureza determinada, Figura 7. El funcionamiento de dicha columna se basa en el hecho de que la composición del residuo cambia con el tiempo durante la operación, de modo que se cumple la siguiente ecuación (Ocon Tojo, 1968)

$$\ln \frac{F}{W} = \int_{x_w}^{x_f} \frac{dx_w}{x_D - x_w} \quad (40)$$

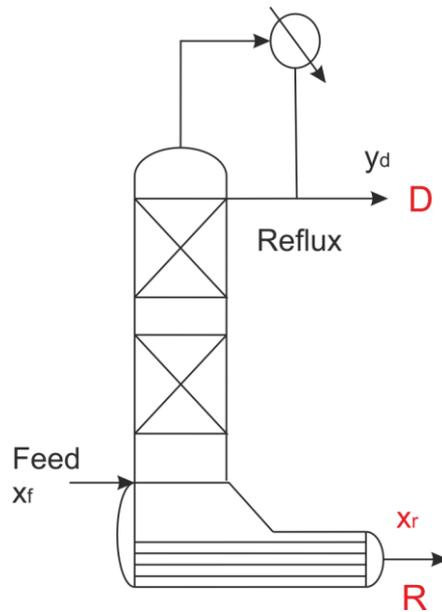


Figura 7.- Esquema de la destilación Batch

Necesitamos evaluar la integral para calcular la relación entre el alimento y el residuo. Par a calcular la composición del residuo, se realiza un estudio de sensibilidad que presenta el efecto de  $x_D$  en los productos para el rango de  $x_f$  y  $x_w$ . Tenga en cuenta que la mezcla de agua y etanol muestra un diagrama X-Y que no es fácil de ajustar a una sola correlación (Martín, 2019)

Los parámetros de entrada del sistema son la alimentación de la columna,  $F$ , la composición y la composición del destilado deseada. A un balance de masa global:

$$F = D + W \quad (41)$$

Y el balance al etanol.

$$F \cdot x_f = D \cdot x_D + W \cdot x_w \quad (42)$$

Nos permitirá calcular el destilado y el residuo así como la composición del residuo.

### **Torre de humidificación.**

Principios: Evaluar la transferencia de masa y calor involucrada en el enfriamiento de un flujo de agua.

Objetivo de los modelos: Simular el comportamiento de una torre de humidificación a partir de la transferencia de masa. El modelo proporcionará el conjunto de temperaturas, seca y húmeda, del flujo de salida de gas así como la temperatura de salida del agua resolviendo los balances de masa y energía del relleno de la columna teniendo como datos de partida el calentamiento del agua, la temperatura del aire de entrada y la humedad y la temperatura del agua de entrada a la torre, así como el flujo de agua y aire. Los resultados que aporta el modelo deben ser empleados para evaluar el coeficiente de transferencia a de materia en la torre.

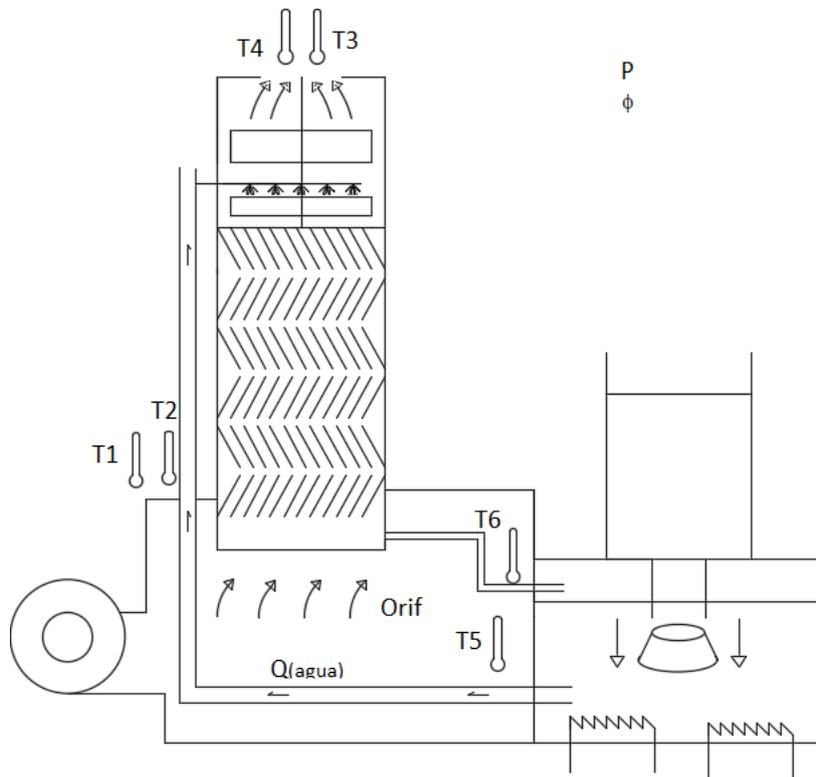


Figura 8. Esquema del equipo.

El gemelo digital reproduce el funcionamiento de una torre de humidificación de tiro forzado, Figura 8. Para ello se debe fijar la energía que aporta la resistencia para calentar el agua (0.5-1 kW), así como la temperatura y humedad del aire ambiente que se puede tomar de datos meteorológicos, la temperatura de entrada del agua caliente a enfriar en la torre (25-35 °C), el caudal de agua ( 15-40 g/s), el diferencial de

orificio que mide el caudal de aire empleado (10-40 mmH<sub>2</sub>O) y la presión total tomada de datos meteorológicos (mmHg).

Con todo ello el gemelo digital calcula las temperaturas húmedas de entrada y salida del aire, la temperatura de salida del aire así como la temperatura del agua. Se pretende que el estudiante determine el NTU y la transferencia de masa (ky) de operación de la torre.

Tarea del instructor: seleccionar la pendiente la pendiente ( $-h_L/k_y$ ), típicamente entre -3 y -10, que caracteriza la transferencia de masa y de calor.

Duración: 2 clases de 3 horas.

Análisis: Realizamos el balance al empaque de la torre de enfriamiento, Figura 9. Es una torre de tiro forzado que utiliza un ventilador para bombear el aire a través de la torre. El agua se alimenta en contracorriente desde la parte superior de la columna.

Calor sensible de la interfase a la masa de aire:

$$h_g a S dZ (t_i - t_g) = V c_h dt_g \quad (43)$$

where:

$$c_h = h_g / k_y \quad (44)$$

$$k_y c_h a S dz (t_i - t_g) = V c_h dt_g \quad (45)$$

$$k_y a S dZ (c_h t_i - c_h t_g) = V c_h dt_g \quad (46)$$

El calor latente de la interfase a la masa de aire:

$$k_y a S dZ (Y_i - Y) \lambda_o = V \lambda_o dy \quad (47)$$

$$k_y a S dZ (\lambda_o Y_i - \lambda_o Y) = V \lambda_o dy \quad (48)$$

El calor total transferido:

$$k_y a S dZ [(c_h t_i + \lambda_o Y_i) - (c_h t_g + \lambda_o Y)] = V [c_h dt_g + \lambda_o dy] \quad (49)$$

que escrito como entalpías queda:

$$k_y a S dZ [H_i - H] = V [dH] \quad (50)$$

de modo que la altura de empaquetamiento:

$$Z = \frac{V}{k_y a S} \int_{H_i}^{H_o} \frac{dH}{H_i - H} \quad (51)$$

Thus,

$$k_y = \frac{V}{Z a S} \int_{H_1}^{H_2} \frac{dH}{H_i - H} = \frac{V}{Z a S} NUT \quad (52)$$

Por un lado, necesitamos calcular el número de unidades de transferencia, NUT, y el caudal de aire que necesitamos utilizar. El problema generalmente se resuelve mediante el método de Mickley que implica iterar la pendiente entre la línea de operación y la línea de equilibrio. Sin embargo, esto es tedioso y requiere mucho tiempo.

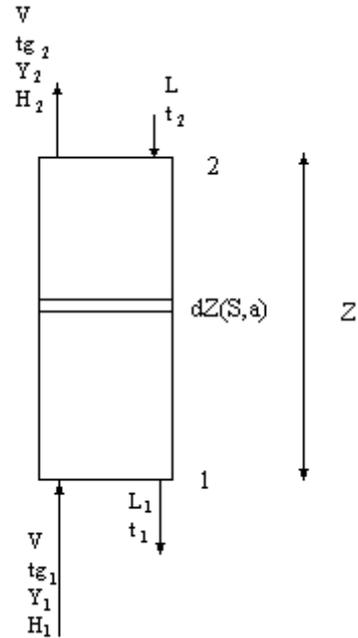


Figura 9.-Esquema del empaquetado de la torre

Primero, determinamos la línea de equilibrio, la entalpía del aire saturado, con base en las condiciones atmosféricas. El punto clave aquí es que necesitamos una ecuación que prediga la línea de equilibrio para calcular los puntos de interfase.

$$T_i \rightarrow P_{v, \text{sat}} \rightarrow Y_{i, \text{sat}} \rightarrow H_i(T_i, Y_i) \quad (53)$$

La línea de operación viene dada por los caudales que atraviesan la columna. La línea de operación se calcula mediante un balance energético entre los flujos de aire y agua.

$$V dH = L dh = L c_L dt_L = L dt_L \quad (54)$$

Usamos  $c_L = 1 \text{ kcal}/(\text{kg}^\circ\text{C})$  para que las ecuaciones queden de forma más simple:

$$\frac{H_2 - H_1}{t_{L2} - t_{L1}} = \frac{L}{V} \quad (55)$$

Las entalpías de entrada y salida de aire se calculan utilizando las temperaturas y humedades experimentales. Se conocen tanto los flujos de aire como de agua y permite calcular la entalpía del aire que sale. Finalmente, la temperatura del aire describe un perfil a lo largo de la columna, se considera el método de Mickley (Geankoplis, 1993). La relación de Lewis relaciona la línea de operación con la línea de equilibrio y resulta de la resistencia a la transferencia de masa y energía entre las fases gaseosa y líquida. Calculamos esto usando un balance energético global.

$$h_i a S dZ (t_L - t_i) = L dh = V dH = k_y a S dZ (H_i - H) \quad (56)$$

$$\frac{H_i - H}{t_i - t_L} = \frac{h_L}{k_y} \quad (57)$$

Este equilibrio vincula el equilibrio y la línea de operación. Es una recta que tiene como pendiente

$\frac{H_i - H}{t_i - t_L}$  y comienza en la línea de operación en  $t_{L1}$ ,  $H_1$ . Normalmente, la pendiente  $(-h_L/k_y)$  está entre -3 y -

10. Este es un valor que caracteriza a la columna y puede ser fijado por el instructor. El perfil de la temperatura del aire se corresponde con la evolución de la temperatura del aire con el cambio de su entalpía. Por lo tanto,

$$\frac{k_y a S dZ (H_i - H)}{k_y c_h a S dZ (t_i - t_g)} = \frac{V dH}{V c_h dt_g} \quad (58)$$

$$\frac{(H_i - H)}{(t_i - t_g)} = \frac{dH}{dt_g} \quad (59)$$

Para resolver esta ecuación diferencial, calculando el perfil de temperatura, el método de Mickley propone discretizar el funcionamiento de la torre de enfriamiento en varias etapas.

Para cada punto de la línea de operación tenemos una temperatura del líquido,  $t_{Lj}$ , y una entalpía del aire  $H_j$ . Esta línea, como se presentó antes, está relacionada con la curva de equilibrio por una línea que tiene una pendiente,  $(-h_L/k_y)$ , que no se conoce. Así, para cada punto  $j$  definimos una línea que va de  $j$  a un punto en la línea de equilibrio  $i$ .

$$H_i - H_j = slope(t_i - t_j) \quad (60)$$

La temperatura húmeda de la corriente de entrada (T2 en Figura 8) se calcula con la humedad del aire:  $T_{wet} = (\Phi_{in} - 1) * P_v(T_{dry}) / 0.5 + T_1$  (61)

La temperatura húmeda de la corriente de salida (T3 en Figura 8) se calcula de manera similar, pero la humedad se calcula a partir de la entalpía del aire. Las temperaturas secas se miden con termopares (T1 de entrada y T 4 de salida) así como las del agua que se está enfriando, T5 el agua caliente y T6 la ya enfriada. Los caudales de agua y de aire se miden mediante un caudalímetro o mediante un orificio

$$VH = (1/29 + Y_e/18) * (0.082 * (273 + T_1)) / (P_{total}/760) \quad (62)$$

$$MasV = 0.0137 * (orif/VH) ** 0.5 \quad (63)$$

Se puede elegir la energía que se aporta al agua mediante la resistencia de 0.5 kW o 1 kW.

### 3.- Ejemplos de la herramienta

La herramienta está alojada en <http://c760378aa416.ngrok.io/VLab/>

La documentación es una versión de este informe



Contacto

## PSEM3 Virtual-Lab IQ Edition

PSEM3 Research Group.  
University of Salamanca.

ACCESO A PSEM3 VIRTUAL-LAB  
Laboratorio virtual de Ingeniería  
Química

ACCESO A LA APLICACIÓN

DOCUMENTACIÓN  
Documentación de PSEM3  
VirtualLab

ACCESO A LA DOCUMENTACIÓN

GRUPO PSEM3  
PSEM3

CRÉDITOS  
Autores

PSEM3 Research Group  
University of Salamanca

Contacto

La herramienta arranca con 10 prácticas, pero es flexible de modo que se pueden incorporar otras.

Para seleccionar la práctica se dispone del siguiente panel.



## Procesos de PSEM3 Virtual-Lab IQ Edition

### TERMODINÁMICA

Simulación de un ciclo Brayton

CICLO BRAYTON

### TERMODINÁMICA

Ciclo Rankine

CICLO RANKINE

### TRANSFERENCIA DE CALOR

Estudio de la conductividad térmica

CONDUCTIVIDAD  
TÉRMICA

### TRANSFERENCIA DE CALOR

Intercambiador de doble tubo

INTERCAMBIADOR  
DOBLE TUBO

### OPERACIONES DE SEPARACIÓN

Modelo de una torre de enfriamiento

TORRE DE  
ENFRIAMIENTO

### OPERACIONES DE SEPARACIÓN

Evaporador simple

EVAPORADOR DE  
EFECTO SIMPLE

### OPERACIONES DE SEPARACIÓN

Destilación batch

DESTILACIÓN BATCH

### REACTORES QUÍMICOS

Fermentador para la producción de etanol

FERMENTADOR  
ETANOL

### REACTORES QUÍMICOS

Reactor discontinuo

REACTOR  
DISCONTINUO

### MECÁNICA DE FLUIDOS

Pérdida de carga de una tubería

PÉRDIDA CARGA  
TUBERÍA

### EN CONSTRUCCIÓN

En construcción

EN CONSTRUCCIÓN

### EN CONSTRUCCIÓN

En construcción

EN CONSTRUCCIÓN

Tomando como ejemplo la destilación Batch, se observa la entrada de datos



Contacto

## Entrada de datos

### PARAMETROS PARA EL ESTUDIO DE LA DESTILACIÓN BATCH

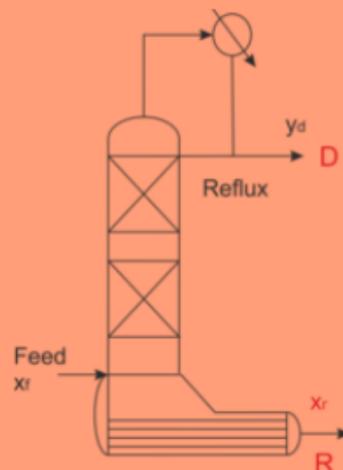
Fracción molar del componente A en el destilado:

Fracción molar del componente A en la alimentación:

Alimentación a la columna (kmol):

Relación de reflujo:

Número de platos:



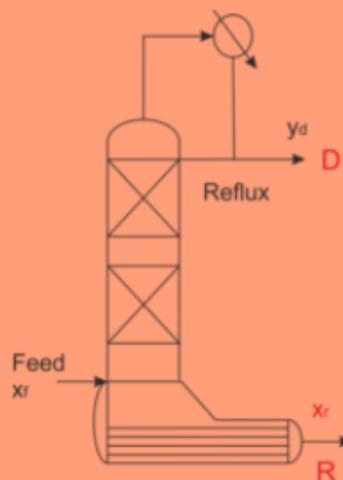
PSEM3 Research Group  
University of Salamanca

Contacto

Al remitir la información el modelo reporta datos como se muestra en la siguiente captura de pantalla

## Resultados

### Destilación batch



#### Parámetros

	Unidades	Valor
Variable		
Fracción molar del componente A en el residuo (x)	NA	0.075773
Flujo residuo	kmol	324.068586
Flujo destilado	kmol	13.532170

Con estos datos, la columna opera con 11 platos.

#### **4.-Análisis del uso**

Las prácticas en Ing. Química son de primer semestre por lo que se evaluó el uso de esta herramienta con alumnos que ya habían cursado este año o el pasado las practicas que hemos virtualizado a efectos de comparar el grado de competencias y la similitud del uso del gemelo virtual frente a la práctica real. Se pidió a los alumnos que evaluaran las prácticas y se muestran aquí los resultados, práctica por práctica de la encuesta que se les pasó. El número total de estudiantes de muestra fue 12, que supone un grupo completo asignado a un profesor por asignatura de experimentación en ingeniería química.

<b>Práctica:</b>	
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	
Ayuda a identificar los principios	
Permite evaluar la operación	
Reproduce la operación del equipo	
¿Cómo se compara con el práctica real?	
Sugerencias	

Así los promedios de cada una de las prácticas se muestran a continuación:

<b>Práctica:</b>	<b>Columna humidificación</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	4
Permite evaluar la operación	4
Reproduce la operación del equipo	3.5
¿Cómo se compara con el práctica real?	4
Sugerencias	Eliminación de los resultados de NUT y ky para un mayor trabajo de los alumnos para su propio beneficio y conozcan cómo se realiza más allá de hace click en "submit"

<b>Práctica:</b>	<b>Destilación batch</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	4
Permite evaluar la operación	4
Reproduce la operación del equipo	4
¿Cómo se compara con el práctica real?	4
Sugerencias	

<b>Práctica:</b>	<b>Reactor cinética (discontinuo)</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	2
Ayuda a identificar los principios	4
Permite evaluar la operación	4
Reproduce la operación del equipo	3
¿Cómo se compara con el práctica real?	3
Sugerencias	

<b>Práctica:</b>	<b>Fermentador (discontinuo)</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	4
Permite evaluar la operación	4
Reproduce la operación del equipo	3
¿Cómo se compara con el práctica real?	3
Sugerencias	

<b>Práctica:</b>	<b>Ciclo Brayton</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	4.5
Ayuda a identificar los principios	4.5
Permite evaluar la operación	4.5
Reproduce la operación del equipo	NA
¿Cómo se compara con el práctica real?	NA
Sugerencias	Podría ser curioso que dejara hacer dos cálculos con dos gráficas a la vez para poder comparar situaciones, poniendo gráficas y tablas una junto a la otra. Creo que me ignora cuando le pido cambiar el caudal.

<b>Práctica:</b>	<b>Ciclo Rankine</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	4.5
Permite evaluar la operación	4.5
Reproduce la operación del equipo	5
¿Cómo se compara con el práctica real?	5
Sugerencias	No tenemos una turbina de vapor como tal para evaluar los ciclos, pero veo que la aplicación es muy útil a la hora de resolver los problemas de termotecnia, para comprobar si los resultados están bien o no. Además, que aparezca el diagrama al lado de los resultados lo hace muy visual y viéndolo a medida que se cambian los parámetros hace que sea fácil de entender junto con la explicación de la documentación.

<b>Práctica:</b>	<b>Evaporador</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	3
Permite evaluar la operación	4
Reproduce la operación del equipo	4
¿Cómo se compara con el práctica real?	4
Sugerencias	Eliminación de los resultados de Economía y quizá Temperatura de evaporación para un mayor trabajo de los alumnos para su propio beneficio y conozcan cómo se realiza más allá de hace click en "submit"

<b>Práctica:</b>	<b>Pérdida de Carga en tuberías</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	4
Permite evaluar la operación	4
Reproduce la operación del equipo	5
¿Cómo se compara con el práctica real?	5
Sugerencias	

<b>Práctica:</b>	<b>Transferencia de calor doble tubo</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	5
Permite evaluar la operación	4
Reproduce la operación del equipo	5
¿Cómo se compara con el práctica real?	5
Sugerencias	Creo que hay revisar porque hay interrogaciones en el título de la práctica y en el resultado de hiexp

<b>Práctica:</b>	<b>Conductividad</b>
	<b>De 0 a 5 puntos</b>
¿Es Intuitiva? ¿fácil de usar?	5
Ayuda a identificar los principios	5
Permite evaluar la operación	4
Reproduce la operación del equipo	5
¿Cómo se compara con el práctica real?	5
Sugerencias	

## **5. Discusión y Conclusiones**

Este proyecto ha desarrollado una herramienta para reforzar el aprendizaje de asignaturas prácticas dentro del grado en Ingeniería química. Sin bien puede ser usada como material docente en caso de tener que impartir dichas asignaturas de forma virtual, en condiciones de normalidad se espera que su

uso sea para la consolidación de los conceptos. El trabajo con los equipos es crucial en una industria que no ha parado en pandemia para proporcionar las materias primas tanto para la elaboración del gel hidroalcohólico, las mismas mascarillas, los combustibles y la propia electricidad. El uso de este material para impartir docencia virtual se ha evaluado a través de la experiencia de un grupo de estudiantes.

Se observa que los estudiantes destacaron la utilidad de la herramienta para estudiar el comportamiento de estos equipos y apoyar la adquisición de las competencias de las asignaturas de “Experimentación en Ingeniería Química” I, II . Es intuitiva y fácil de usar amén de presentar la documentación de cada una de las prácticas. Una de las limitaciones es que no hay nada que sustituya el trabajo en laboratorio, en el que la dinámica de la estabilización de la operación de unidades complejas como la columna de destilación, que tarda más de 3 h en funcionar o la torre de humidificación que tarda mínimo 30-45 min es alcanzar el estado estacionario, así como los errores de los instrumentos de medida, ya sean termopares o la medición de concentraciones de disoluciones no se pueden simular de forma eficiente. Un caso extremo son los 3 días que dura una fermentación, que hacen que este tipo de prácticas sean prohibitivas en una docencia reglada. Es importante que el estudiante se dé cuenta que los tiempos para alcanzar estado estacionario son necesarios y la razón por lo que las plantas no pueden arrancar y parar a voluntad. Sin embargo estos laboratorios virtuales son un buen complemento en casos extremos como la pandemia que hemos sufrido, así como para afianzar conocimientos y competencias. En condiciones de vuelta a la normalidad, esta herramienta puede permitir ayudar a preparar las prácticas en sí ya que permite una evaluación rápida de la operación.

## **6.-Referencias**

Aachen (2021) <https://www.irt.rwth-aachen.de/cms/IRT/Studium/Downloads/~jshdg/Virtual-Control-Lab/lidx/1/>

Bell, J. T., & Fogler, H. S. (1996). Vicher: a virtual reality based educational module for chemical reaction engineering. *Computer Applications in Engineering Education*, 4(4), 285–296.

Bertilsson, M., Andersson, J., and Lidesonn, G. (2008). Modeling simultaneous glucose and xylose uptake in *Saccharomyces cerevisiae* from kinetics and gene expression of sugar transporters. *Bioprocess Biosyst. Eng.*, 31, 369-377.

Bird, R.B.; Stewart, W.E.; Lightfoot, E.N., (2002) *Transport Phenomena*. 2nd Edition John Willey and Sons. N.Y.

- Bissett, L. (1977) *Chemical Engineering* , 84(21), 155.
- Cherednichenko, V. M., Dissertation, Karpova, Physico Chemical Institute ,Moscow, U.S.S.R., 1953.
- Geankoplis, C.J. (1993)*Transport Processes and Unit Operations* (3rd Edition) Prentice Hall
- Fogler, S.H. (2000) *Elements of Chemical Reaction Engineering*, Prentice Hall International Series, 3rd Edition, Upper Saddle River, New Jersey
- Galán, D., Heradio, R., de la Torre, L., Dormido, S., Esquembre, F. (2016) Virtual Control Labs Experimentation: The Water Tank System IFAC- PapersOnLine 49 (6) 87-92
- Henker, M., Kelber, K. (2021) Virtualizing Electrical engineering teaching labs.  
[https://es.mathworks.com/company/newsletters/articles/virtualizing-electrical-engineering-teaching-labs.html?s\\_v1=36867&elqem=3353549\\_EM\\_ES\\_DIR\\_21-03\\_MOE-EDU&elqTrackId=195d12ebb7c849a2b813c99ccf8574c2&elq=c7fd107c4b2a4766983b5f6b995279b3&elqaid=36867&elqat=1&elqCampaignId=13478](https://es.mathworks.com/company/newsletters/articles/virtualizing-electrical-engineering-teaching-labs.html?s_v1=36867&elqem=3353549_EM_ES_DIR_21-03_MOE-EDU&elqTrackId=195d12ebb7c849a2b813c99ccf8574c2&elq=c7fd107c4b2a4766983b5f6b995279b3&elqaid=36867&elqat=1&elqCampaignId=13478)
- Krishnan, M.S.; Ho, N.W.Y.; Tsa, G.T. (1999) Fermentation kinetics of ethanol production from glucose and xylose by recombinant *Saccharomyces* 1400 (pLNH33) *Appl. Biochem. Biotech.* 77-710. 373-388
- Lee, S. (2007) *Methanol Synthesis from Syngas*. Chapter 9. *Handbook of alternative Fuel technologies*. Taylor and Francis
- Marechal, F., Heyen, G., Kalitventzeff, B. (1997) Energy Saving in Methanol Synthesis: Use of Heat Integration Techniques and Simulation Tools. *Computers and Chemical Engineering* 21, S511-S516
- Martín, M (2019) *Introduction to software for chemical engineers*. CRC Press. Boca Raton US.
- McCabe, W.L.; Smith, J.C., Harriot, P. (1999) *Unit Operations of Chemical Engineering* 7th Edition
- Ocon, J, Tojo, G (1968) *Problemas de Ingeniería Química*. Tomo 1. Aguilar Madrid
- Ribeiro, A., Santos, J.C., Rodrigues, A .E., (2010). PSA design for stoichiometric adjustment of bio-syngas for methanol production and co-capture of carbon dioxide, *Chemical Engineering Journal* 163,355–363
- <http://ce-iitb.vlabs.ac.in/>
- <http://vmt-iitg.vlabs.ac.in/>
- <http://www.learncheme.com/virtual-laboratories>
- Williams, J.L., Hilliard, M., Smith, C., Hoo, , K.A. Wiesner, T.F., Parker, H.W. and William Lan, W. (2003) *The Virtual Chemical Engineering Unit Operations Laboratory Proceedings of the 2003 American Society for Engineering Education Annual Conference & Exposition*

## Apéndice

### Evaluación de Principios

#### Evaporación-----

```
#Import libraries
import math
import numpy as np
import pandas as pd

#We create an empty list to store the concentrations data

#Parameters
data=pd.read_csv('dataEvap.csv')
data.columns=['Feeddata']
Feeddt=data['Feeddata']
Fd=data['Feeddata'].to_numpy();
Feedrr=Fd.astype(float);
PvapW=Feedrr[0];
Feed=Feedrr[1];
Cf=Feedrr[2];
Tfeed=Feedrr[3];
Cliq1=Feedrr[4];
Pcamb1=Feedrr[5];

#min-1
#Presion del vapor de servicios en atm
#PvapW=3;
#Alimentacion kg/h
#Feed=10000.0;
#Cfeed w%w
#Cf=0.12;
#temperature feed (°C)
#Tfeed=50.0;
#Cliq 2 w%w
#Cliq1=0.4;
#Pressure Cam 2 en mmHg
#Pcamb1=150.0;
#Area (m2)
Area1=90;

Cfeed=Cf*(1+0.1*np.random.uniform(-1,1));
Feed1=Feed*(1+0.05*np.random.uniform(-1,1));

hfeed1=((-0.0413*Tfeed**2-0.8245*Tfeed+816.925)*Cfeed**3+(0.0273*Tfeed**2+1.2025*Tfeed-176.35)*Cfeed**2+(-0.0048*Tfeed**2-0.723*Tfeed-12.433)*Cfeed+(1.0675*Tfeed-1.975))

Tvap=3816.44/(18.3036-math.log(PvapW*760))-227.02;
Teb1=3816.44/(18.3036-math.log(Pcamb1))-227.02;
Liquid1=Feed1*Cfeed/Cliq1;
Evap1=Feed1-Liquid1;
lambdaw=(-0.0000043722*Tvap**3 + 0.00043484*Tvap**2 - 0.58433*Tvap + 597.48);
Tcam1=(-0.000030233*(100*Cliq1)**3 + 0.00021232*(100*Cliq1)**2 - 0.0042815*(100*Cliq1) + 1.0459)*Teb1+(0.00024541*(100*Cliq1)**3 + 0.0060805*(100*Cliq1)**2 - 0.009887*(100*Cliq1) + 0.17863);
hliq1=( -0.0413*Tcam1**2-0.8245*Tcam1+816.925)*Cliq1**3+(0.0273*Tcam1**2+1.2025*Tcam1-176.35)*Cliq1**2+(-0.0048*Tcam1**2-0.723*Tcam1-12.433)*Cliq1+(1.0675*Tcam1-1.975);
Hev1=(-0.000030903*Teb1**3 + 0.00022613*Teb1**2 + 0.42436*Teb1 + 597.42)+0.46*(Tcam1-Teb1);
VaporW=(Liquid1*hliq1+Evap1*Hev1-Feed1*hfeed1)/lambdaw;
Uglobal1=VaporW*lambdaw/(Area1*(Tvap-Tcam1));
Economy=Evap1/VaporW;
print("U_global(W/m2·K)=", Uglobal1 , "Flow_utility(kg/s)=", VaporW,"T_evap_chamber(°C)=",Tcam1,"Economy=", Economy)

df = pd.DataFrame([Uglobal1,VaporW,Tcam1,Economy])
```

```
df.to_csv('ResultsEvaporator.csv')
```

## Intercambiador de doble tubo-----

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

#Parameters
data=pd.read_csv('dataHX.csv')
data.columns=['Feeddata']
Feeddt=data['Feeddata']
Fd=data['Feeddata'].to_numpy();
Feedrr=Fd.astype(float);
Flow_l=Feedrr[0];
type=Feedrr[1];
di=Feedrr[2];
dout=Feedrr[3];

#Flow_l=0.1 #kg/s
Density_l_in= 1000; #kg/m3
Density_l_out= 1000;
Heat_capacity_l_in= 4.18; #kJ/kg.k
Heat_capacity_l_out=4.18;
viscosity_l_in= 8.9e-4; #Pas
viscosity_l_out= 8.9e-4;
Kf_l = 0.6; # W/m.k
T_l_in=90;
lambdaS=1800; #kJ/kg
T_O_in=233;
T_O_out=233;
Flow_O=Flow_l*(1.5/10);

Flow_Oreal=Flow_O*(1+0.025*np.random.uniform(-1,1));
Qaportado=Flow_Oreal*(lambdaS);
Flow_lreal=Flow_l*(1+0.025*np.random.uniform(-1,1));
T_l_outc=Qaportado/(Flow_lreal*Heat_capacity_l_in)+T_l_in;
T_l_out=T_l_outc*(1+0.025*np.random.uniform(-1,1))

# {b = 0 Non viscous liquids
# b = 1 Viscous liquids
# b = 2 Gases}
# {Considering the omission of the viscosity corrector factor}

#Type of flow
# type 1 parallel
#type 2 counter

#type=2;

b=0.0;
#di=0.049;#m
#dout=0.075;#m
dh=dout-di;#m
Long=0.265; #m
#{ Cooling = 1 for cooling,
# Cooling = 0 for heating}
Cooling_l =0.0;

# Working fluid
# Inner tube

ut_l_1 = (Flow_lreal)/(Density_l_in*(np.pi*(1/4)*(di**2)));
ut_l_2=(Flow_lreal)/(Density_l_out*(np.pi*(1/4)*(di**2)));
ut_l = (ut_l_1 + ut_l_2)/2;
```

```

Re_l_t=(((Density_l_in+Density_l_out)*0.5)*ut_l*di)/(((viscosity_l_in+viscosity_l_out)*0.5));
Pr_l=((Heat_capacity_l_in*1000)*((viscosity_l_in+viscosity_l_out)*0.5))/Kf_l;

if Cooling_l == 1:
    n = 0.3;
else: n = 0.4;

if ((Re_l_t > 10000.0) and (b == 0.0)):
    Nu_l = 0.023*(Re_l_t**0.8)*(Pr_l**n);
elif ((Re_l_t > 10000.0) and (b == 1.0)):
    Nu_l = 0.027*(Re_l_t**0.8)*(Pr_l**n);
elif ((Re_l_t > 10000.0) & (b == 2.0)):
    Nu_l = 0.0231*(Pr_l**n);
else:
    Nu_l = 1.86*(((Re_l_t)*(Pr_l)*(di/Long))**0.33);
hi=Nu_l*Kf_l/di;

#Heating agent

ho=12000.0;
if type==1:
    LMTD=((T_O_in-T_I_in)-(T_O_out-T_I_out))/np.log((T_O_in-T_I_in)/(T_O_out-T_I_out))
    plt.plot([0,Qaportado],[T_I_in,T_I_out],b-',[0,Qaportado],[T_O_in,T_O_out],r-)
    plt.xlabel('Q(kW)')
    plt.ylabel('T(°C)')
else:
    LMTD=((T_O_out-T_I_in)-(T_O_in-T_I_out))/np.log((T_O_out-T_I_in)/(T_O_in-T_I_out))
    plt.plot([0,Qaportado],[T_I_in,T_I_out],b-',[0,Qaportado],[T_O_out,T_O_in],r-)
    plt.xlabel('Q(kW)')
    plt.ylabel('T(°C)')

#5.- Fouling coefficients
ri = 1.7*(1/10**4);
ro = 1.3*(1/10**4);
kw = 16.3; # W/m.k
#6.-Overall heat transfer coefficient

Uteor = 1/( (1/ho) + (ro) + ((dout*np.log(dout/di))/(2*kw)) + (dout/di)*ri + (dout/di)*(1/hi) ) ;

#6.- Surface area
#Q = U*A*LMTD;
# Duty
#Available area
Adisp=Long*(np.pi*di);
Areareq=Qaportado/(Uteor*LMTD);

# Pressure drop
#7.- Pressure drop in the tube side.
fd = (1.82*(np.log10(Re_l_t) - 1.64))**(-2);
f_t=fd/4;
dPt= f_t*(Long/di)*0.5*((Density_l_in+ Density_l_out)*0.5*(ut_l**2)); # Pa
Uexp= Qaportado/(Adisp*LMTD);
hiexp = (dout/di)*1/((1/Uexp-(1/ho) - (ro) - ((dout*np.log(dout/di))/(2*kw)) - (dout/di)*ri ));
print("U_global(W/m2·K)=", Uexp , "LMTD(°C)=",LMTD, "hi(W/m2·K)=",hiexp)
df = pd.DataFrame([Uexp,LMTD,hiexp])

df.to_csv('ResultsHX.csv')

```

## Fermentador-----

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy import integrate

```

```

data=pd.read_csv('dataEtha.csv')
data.columns=['Feeddata']
Feeddt=data['Feeddata']
Fd=data['Feeddata'].to_numpy();
Feedrr=Fd.astype(float);
Gluco=Feedrr[0];
Xylo=Feedrr[1];
Etha=Feedrr[2];
Celu=Feedrr[3];

#[kJ/mol]
#300°C k en min-1
v_m_g=2.005;
v_m_x=0.250;
K_s_g=0.565;
K_i_g=283.7;
K_s_x=3.4;
K_i_x=18.1;
K_sp_g=1.341;
K_ip_g=4890;
K_sp_x=3.4;
K_ip_x=81.3;
m_g=0.097;
m_x=0.067;
Y_P_S_g=0.470;
Y_P_S_x=0.4;
P_m_x=59.04;
P_mp_x=60.2;
Beta_x=1.036;
gamma_x=0.608;

Xini=[Gluco,Xylo,Etha,Celu];

#Xini=[200,50,0,1];
x=[(),(),(),()];

#We create an empty list to store the concentrations data

tau=np.linspace(0,36,250)
#Equations: We define a function which collect the equations to solve

def kinetics(x,tau):
    Glucose=x[0];
    xylose=x[1];
    ethanol=x[2];
    cells=x[3];

    if ethanol <95.4:
        P_m_g=95.4;
        Beta_g=1.29;
        gamma_g=1.42;
    else:
        P_m_g=129;
        Beta_g=0.25;
        gamma_g=0;

    if ethanol <103:
        P_mp_g=103;
    else:
        P_mp_g=136;

    if cells < 5:

```

```

mu_m_g=0.152*(cells)**(-0.461);
v_m_g=1.887*(cells)**(-0.434);
mu_m_x=0.075*(cells)**(-0.438);
v_m_x=0.16*(cells)**(-0.233);
else:
mu_m_g =0.662;
v_m_g= 2.005;
mu_m_x=0.190;
v_m_x= 0.25;

mu_g=mu_m_g*Glucose*(1-(ethanol/P_m_g)**Beta_g)/(K_s_g+Glucose+Glucose**2/K_i_g);
mu_x =mu_m_x*xylose*(1-(ethanol/P_m_x)**Beta_x)/(K_s_x+xylose+xylose**2/K_i_x);
v_g=v_m_g*Glucose*(1-(ethanol/P_mp_g)**gamma_g)/(K_sp_g+Glucose+Glucose**2/K_ip_g);
v_x=v_m_x*xylose*(1-(ethanol/P_mp_x)**gamma_x)/(K_sp_x+xylose+xylose**2/K_ip_x);

dxdtau=np.zeros(4)
dxdtau[0]=-(1/Y_P_S_g)*v_g*cells
dxdtau[1]=-(1/Y_P_S_x)*v_x*cells
dxdtau[2]=(v_g+v_x)*cells
dxdtau[3]=cells*((Glucose)*mu_g/(Glucose+xylose)+(xylose)*mu_x/(Glucose+xylose))
return dxdtau

#We collect the results in a list calles Res
res_x=integrate.odeint(kinetics,Xini,tau)

plt.plot(tau,res_x[:,0],'k-')
plt.plot(tau,res_x[:,1],'b-')
plt.plot(tau,res_x[:,2],'r-')
plt.plot(tau,res_x[:,3],'g-')

plt.xlabel('t(h)')
plt.ylabel('Concentration(g/L)')
plt.legend('GXEC')

df = pd.DataFrame([res_x[:,0]])

df.to_csv('ResultsEtha.csv')

```

## Gemelos virtuales

### Determinación de la conductividad-----

```

import numpy as np
import pandas as pd
from scipy.optimize import newton_krylov
import matplotlib.pyplot as plt

#Data input
#Instructor configuration
#Conductivity W/mK
aK=52.5;
bK=-0.02228;
cK=0;
# error
error=0.025;
perdidas=0.15;

#Student
data=pd.read_csv('datacond.csv')
data.columns=['Feeddata']
Feeddt=data['Feeddata']
Fd=data['Feeddata'].to_numpy();
Feedrr=Fd.astype(float);
Qttotal=Feedrr[0];

```

```

mais=Feedrr[1];
mnoais=Feedrr[2];

#Student variables
#Qtotal=500; # W
#mais=10;# g/s
#mnoais=10;# g/s

#Model
qperdidass=0.15*Qtotal;
factor=0.55;
qaislada=Qtotal*(1-perdidass)*factor;

cpag=4.18;# J/gK
Tin=20;# °C
Tout_ais=Tin+qaislada/(mais*cpag);# °C

qnoaislada=Qtotal-qaislada-qperdidass;
Tout_noais=Tin+qnoaislada/(mnoais*cpag);

#Dimensiones tronco de cono
RM=0.12;
Rm=0.10;
Long=0.3;
Pend=(RM-Rm)/Long;

L1=[0,0.05,0.1,0.15,0.2,0.25]
L2=[0.05,0.1,0.15,0.2,0.25,0.3]

Variable=[(),(),(),(),(),()];
length=len(L1);
discret=np.arange(1,length,1);
Tmet0=500;

def Eq(Variable):
    Tmet1=Variable[0];
    Tmet2=Variable[1];
    Tmet3=Variable[2];
    Tmet4=Variable[3];
    Tmet5=Variable[4];
    Tmet6=Variable[5];

    return[qaislada+(aK*(Tmet1-Tmet0)+(bK/2)*(Tmet1**2-Tmet0**2)+(cK/3)*(Tmet1**3-Tmet0**3))/((L2[0]-L1[0])/(np.pi*((RM-Pend*L1[0])*(RM-Pend*L2[0])))),
           qaislada+(aK*(Tmet2-Tmet1)+(bK/2)*(Tmet2**2-Tmet1**2)+(cK/3)*(Tmet2**3-Tmet1**3))/((L2[1]-L1[1])/(np.pi*((RM-Pend*L1[1])*(RM-Pend*L2[1])))),
           qaislada+(aK*(Tmet3-Tmet2)+(bK/2)*(Tmet3**2-Tmet2**2)+(cK/3)*(Tmet3**3-Tmet2**3))/((L2[2]-L1[2])/(np.pi*((RM-Pend*L1[2])*(RM-Pend*L2[2])))),
           qaislada+(aK*(Tmet4-Tmet3)+(bK/2)*(Tmet4**2-Tmet3**2)+(cK/3)*(Tmet4**3-Tmet3**3))/((L2[3]-L1[3])/(np.pi*((RM-Pend*L1[3])*(RM-Pend*L2[3])))),
           qaislada+(aK*(Tmet5-Tmet4)+(bK/2)*(Tmet5**2-Tmet4**2)+(cK/3)*(Tmet5**3-Tmet4**3))/((L2[4]-L1[4])/(np.pi*((RM-Pend*L1[4])*(RM-Pend*L2[4])))),
           qaislada+(aK*(Tmet6-Tmet5)+(bK/2)*(Tmet6**2-Tmet5**2)+(cK/3)*(Tmet6**3-Tmet5**3))/((L2[5]-L1[5])/(np.pi*((RM-Pend*L1[5])*(RM-Pend*L2[5]))))] ;

Guess=(500,495,490,485,480,475);

Res = newton_krylov(Eq, Guess, method='lgmres')
#Change the data type to float to manage the results
#Res=np.float_(Res)
#Print the results list
print ('Tisolated(°C)=' ,Res)

```

```

#Store each concentration result in a new variable
Tais1=Res[0];
Tais2=Res[1];
Tais3=Res[2];
Tais4=Res[3];
Tais5=Res[4];
Tais6=Res[5];

plt.figure()

plt.plot([0,0.05,0.1,0.15,0.2,0.25,0.3],[Tmet0,Tais1, Tais2, Tais3, Tais4, Tais5, Tais6],'k-')
plt.xlabel('L(m)')
plt.ylabel('T(K)')

Variable2=[(),(),(),(),(),()];

def Eq2(Variable2):
    Tmetn1=Variable2[0];
    Tmetn2=Variable2[1];
    Tmetn3=Variable2[2];
    Tmetn4=Variable2[3];
    Tmetn5=Variable2[4];
    Tmetn6=Variable2[5];

    return[qnoaislada+(aK*(Tmetn1-Tmet0)+(bK/2)*(Tmetn1**2-Tmet0**2)+(cK/3)*(Tmetn1**3-Tmet0**3))/((L2[0]-L1[0])/(np.pi*((RM-Pend*L1[0])*(RM-Pend*L2[0])))),
           qnoaislada+(aK*(Tmetn2-Tmetn1)+(bK/2)*(Tmetn2**2-Tmetn1**2)+(cK/3)*(Tmetn2**3-Tmetn1**3))/((L2[1]-L1[1])/(np.pi*((RM-Pend*L1[1])*(RM-Pend*L2[1])))),
           qnoaislada+(aK*(Tmetn3-Tmetn2)+(bK/2)*(Tmetn3**2-Tmetn2**2)+(cK/3)*(Tmetn3**3-Tmetn2**3))/((L2[2]-L1[2])/(np.pi*((RM-Pend*L1[2])*(RM-Pend*L2[2])))),
           qnoaislada+(aK*(Tmetn4-Tmetn3)+(bK/2)*(Tmetn4**2-Tmetn3**2)+(cK/3)*(Tmetn4**3-Tmetn3**3))/((L2[3]-L1[3])/(np.pi*((RM-Pend*L1[3])*(RM-Pend*L2[3])))),
           qnoaislada+(aK*(Tmetn5-Tmetn4)+(bK/2)*(Tmetn5**2-Tmetn4**2)+(cK/3)*(Tmetn5**3-Tmetn4**3))/((L2[4]-L1[4])/(np.pi*((RM-Pend*L1[4])*(RM-Pend*L2[4])))),
           qnoaislada+(aK*(Tmetn6-Tmetn5)+(bK/2)*(Tmetn6**2-Tmetn5**2)+(cK/3)*(Tmetn6**3-Tmetn5**3))/((L2[5]-L1[5])/(np.pi*((RM-Pend*L1[5])*(RM-Pend*L2[5]))))]

Guess=(500,495,490,485,480,475);

Res2 = newton_krylov(Eq2, Guess, method='Igmres')
#Change the data type to float to manage the results
#Res=np.float_(Res)
#Print the results list
print ('Tnonisolated(°C)',Res2)

#Store each concentration result in a new variable
Tnais1=Res2[0];
Tnais2=Res2[1];
Tnais3=Res2[2];
Tnais4=Res2[3];
Tnais5=Res2[4];
Tnais6=Res2[5];

plt.plot([0,0.05,0.1,0.15,0.2,0.25,0.3],[Tmet0,Tnais1, Tnais2, Tnais3, Tnais4, Tnais5, Tnais6],'b-')
plt.xlabel('L(m)')
plt.ylabel('T(K)')

plt.figure()

Ttp1noais1=Tnais1*(1+error*np.random.uniform(-1,1));
Ttp1noais2=Tnais2*(1+error*np.random.uniform(-1,1));
Ttp1noais3=Tnais3*(1+error*np.random.uniform(-1,1));
Ttp1noais4=Tnais4*(1+error*np.random.uniform(-1,1));
Ttp1noais5=Tnais5*(1+error*np.random.uniform(-1,1));

```

```

Ttp1noais6=Tnais6*(1+error*np.random.uniform(-1,1));

plt.plot([0,0.05,0.1,0.15,0.2,0.25,0.3],[Tmet0,Ttp1noais1, Ttp1noais2, Ttp1noais3, Ttp1noais4, Ttp1noais5, Ttp1noais6],'ko')
plt.xlabel('L(m)')
plt.ylabel('T(K)')
plt.savefig('Zonanoaislada.png',bbox_inches='tight',dpi=600)

Ttp1ais1=Tais1*(1+error*np.random.uniform(-1,1));
Ttp1ais2=Tais2*(1+error*np.random.uniform(-1,1));
Ttp1ais3=Tais3*(1+error*np.random.uniform(-1,1));
Ttp1ais4=Tais4*(1+error*np.random.uniform(-1,1));
Ttp1ais5=Tais5*(1+error*np.random.uniform(-1,1));
Ttp1ais6=Tais6*(1+error*np.random.uniform(-1,1));

plt.plot([0,0.05,0.1,0.15,0.2,0.25,0.3],[Tmet0,Ttp1ais1, Ttp1ais2, Ttp1ais3, Ttp1ais4, Ttp1ais5, Ttp1ais6],'bo')
plt.xlabel('L(m)')
plt.ylabel('T(K)')
plt.savefig('Zonaaislada.png',bbox_inches='tight',dpi=600)

df = pd.DataFrame([Tmet0,Ttp1noais1, Ttp1noais2, Ttp1noais3, Ttp1noais4, Ttp1noais5, Ttp1noais6],[Tmet0,Ttp1ais1, Ttp1ais2,
Ttp1ais3, Ttp1ais4, Ttp1ais5, Ttp1ais6])

df.to_csv('ResultsCond.csv')

```

## Determinación del fanning-----

```

import numpy as np
import pandas as pd
from scipy.optimize import newton_krylov

#Data input
#instructor
Long_pipe=1# m
Elbowhalf=2;# number of elbow 1/2
Elbow180=2;# number of elbow 180
error=0.05;
dtub=0.02;
viscosity=0.001;#Pas
density=1; #kg/L

#Student
data=pd.read_csv('dataTub.csv')
data.columns=['Flowrate']
Flowdata=data['Flowrate']
Fr=data['Flowrate'].to_numpy();
Flowarr=Fr.astype(float);
Flow=Flowarr[0];
#Flow=1.5;# L/h

#Model

g=9.8;
velocity=(Flow/3600)/(density*(np.pi)*(dtub/2)**2);
epsilon=0.002*(1+error*np.random.uniform(-1,1));
Long=Long_pipe+0.7*Elbowhalf+1.35*Elbow180;
fanning=[]
Reynolds=density*1000*velocity*dtub/viscosity;

def Eq(fanning):

    return [1/fanning**0.5+4*np.log10(epsilon/(dtub/3.7)+1.256/Reynolds*fanning**0.5)]

Guess=(0.01);

Res = newton_krylov(Eq, Guess, method='lgmres');

```

```
#Change the data type to float to manage the results
#Res=np.float_(Res)
#Print the results list
#print ('Res',Res)
```

```
#Store each concentration result in a new variable
ffanning=Res;
```

```
print ('f=',ffanning)
```

```
hf=ffanning*Long*velocity**2/(dtub*2*g);
DeltaP=hf*g*density*1000;#Pa
```

```
print("DeltaP (Pa)=", DeltaP)
```

```
df = pd.DataFrame([DeltaP])
```

## Turbina de vapor

```
import numpy as np
import random
from scipy.optimize import fsolve
```

#Entrada de datos. En esta parte he puesto valores aleatorios para ver si el modelo tiene problemas para encontrar solución entre esos valores. Estos serán los valores límites que puedan cambiar los alumnos para determinar la eficiencia de la bomba

#El objetivo es que los alumnos calculen la eficiencia de la turbina  $eff\_turb1$  y la relacionen con la carga  $tal$  como se hace de forma teórica con la siguiente correlación:

```
#  $eff\_turb1 = isoeff*(-1.0176*(load)**4+2.4443*(load)**3-2.1812*(load)**2+1.0535*load+0.701)$ 
```

```
fwa2=random.randrange(10,100)
Pwa2=random.randrange(100,200)
Pwa3=random.randrange(5,20)
Pwa5=random.randrange(1,2)
```

```
print("flujo total del circuito=",fwa2)
print("Presión de la turbina de alta presión=",Pwa2)
print("Presión de la turbina de media presión=",Pwa3)
print("Presión de la turbina de baja presión=",Pwa5)
```

```
def func(x,fwa2,Pwa2,Pwa3,Pwa5):
```

```
#Lectura de las variables desde excel.
#Definición de parámetros de entrada al problema
c_p_ind=4.18
isoeff=0.8
#Condiciones del líquido térmico
THTF1=565
THTF2=290
#punto de alta presión
```

```
fwamax=100
load=fwa2/fwamax
#Puntos de presión intermedia
```

```
#Puntos de baja presión
```

```
#Cálculos preliminares
```

```
#Igualdades
```

```

#Esto slo es una separacion de dos corrientes por lo que se mantienen las condiciones de operación:
Pwa8=Pwa3
#La mezcla va a ser isobarica y por tanto:
Pwa7=Pwa8
Pwa9=Pwa7
Pwa4=Pwa3
Pwa6=Pwa5

#Calor producido por los espejos

#Eficiencia de la turbina
eff_turb1 = isoeff*(-1.0176*(load)**4+2.4443*(load)**3-2.1812*(load)**2+1.0535*load+0.701)
#Caracterizacion del punto 2

#Requerimiento del flujo térmico

#Caracterizacion del punto 2
#Declaracion del las variables

Tiso_3=x[0]
Hiso_3=x[1]
Hwa3=x[2]
Twa3=x[3]
Swa3=x[4]
Twa4=x[5]
Twa8=x[6]
Hwa4=x[7]
Hwa8=x[8]
Swa4=x[9]
Swa8=x[10]
Tebu3=x[11]
Twa2=522
Hwa1=x[12]
fwa3=x[13]
W_turbine1=x[14]
Tebu5=x[15]
fwa8=x[16]
fwa5=x[17]
Twa5=x[18]
fracLiso=x[19]
Hiso_5=x[20]
Hwa5=x[21]
fracL=x[22]
W_turbine2=x[23]
fwa6=x[24]
Twa6=x[25]
Hwa6=x[26]
Q_HX6=x[27]
W_pump1=x[28]
Hwa7=x[29]
fwa7=x[30]
fwa9=x[31]
Hwa9=x[32]
Twa9=x[33]
Pwa1=x[34]
W_pump2=x[35]
Twa1=x[36]
Twa7=x[37]
fwa1=x[38]
fwa4=x[39]
Hwa2=x[40]
Swa2=x[41]
load=x[42]
Q_collector=x[43]
eff_turb1=x[44]
intcp=x[45]

```

fHTF1=x[46]  
fHTF2=x[47]

```
return [  
  Swa2-(0.00000000015719*PWa3 + 0.0000000074013)*(x[0])**3-(-0.0000000010074*PWa3**2 - 0.00000030171*PWa3 -  
  0.000028872)*(x[0])**2-(0.00000094914*PWa3**2 + 0.000029097*PWa3 + 0.0050938)*(x[0])-(0.000041223*PWa3**2 -  
  0.028841*PWa3 + 5.9537),  
  Hiso_3-((-0.0000000000011619*PWa3**2 - 0.000000000087596*PWa3 -  
  0.0000000022611)*(Tiso_3)**4+(0.000000004298*PWa3**2 + 0.0000003276*PWa3 + 0.0000007313)*(Tiso_3)**3+(-  
  0.000005801*PWa3**2 - 0.000046*PWa3 - 0.0005009)*(Tiso_3)**2+(0.0003383*PWa3**2 + 0.02947*PWa3 + 2.195)*(Tiso_3)+(-  
  0.072042*PWa3**2 - 7.7877*PWa3 + 2440.8)),  
  eff_turb1*(Hiso_3-Hwa2)-((Hwa3-Hwa2)),  
  Hwa3 -((-0.0000000000011619*PWa3**2 - 0.000000000087596*PWa3 -  
  0.0000000022611)*(Twa3)**4+(0.000000004298*PWa3**2 + 0.0000003276*PWa3 + 0.0000007313)*(Twa3)**3+(-  
  0.000005801*PWa3**2 - 0.000046*PWa3 - 0.0005009)*(Twa3)**2+(0.0003383*PWa3**2 + 0.02947*PWa3 + 2.195)*(Twa3)+(-  
  0.072042*PWa3**2 - 7.7877*PWa3 + 2440.8)),  
  Swa3-((0.00000000015719*PWa3 + 0.0000000074013)*(Twa3)**3+(-0.0000000010074*PWa3**2 - 0.00000030171*PWa3 -  
  0.000028872)*(Twa3)**2+(0.00000094914*PWa3**2 + 0.000029097*PWa3 + 0.0050938)*(Twa3)+(0.000041223*PWa3**2 -  
  0.028841*PWa3 + 5.9537)),  
  Twa3 -Twa4,  
  Twa3 -Twa8,  
  Hwa3 -Hwa4,  
  Hwa3 -Hwa8,  
  Swa3-Swa4,  
  Swa3-Swa8,  
  PWa3*760-(np.exp(18.3036-3816.44/(227.02+Tebu3))),  
  fwa2-fwa1,  
  fwa2*(Hwa2-Hwa1)-Q_collector ,  
  fwa2 -fwa3,  
  W_turbine1 -fwa2*(Hwa2-Hwa3),  
  PWa5*760 - (np.exp(18.3036-3816.44/(227.02+Tebu5)) ),  
  fwa4+fwa8-fwa3,  
  fwa5 -fwa4,  
  Tebu5 - Twa5,  
  
  Swa4 -((fracLiso)*(1.0372*10**(-12)*(Tebu5)**5 - 8.6494*10**(-10)*(Tebu5)**4 + 2.8965*10**(-7)*(Tebu5)**3 - 5.6730*10**(-  
  5)*(Tebu5)**2 + 1.6802*10**(-2)*(Tebu5) - 2.1997*10**(-2)) + (1-fracLiso)*(-2.0373*10**(-12)*(Tebu5)**5 + 1.8589*10**(-  
  9)*(Tebu5)**4 - 7.1901*10**(-7)*(Tebu5)**3 + 1.6112*10**(-4)*(Tebu5)**2 - 2.8904*10**(-2)*(Tebu5) + 9.1915) ),  
  Hiso_5 -((fracLiso)*(3.6082*10**(-12)*(Tebu5)**6 - 3.4120*10**(-9)*(Tebu5)**5 + 1.2303*10**(-6)*(Tebu5)**4 - 2.0306*10**(-  
  4)*(Tebu5)**3 + 1.5552*10**(-2)*(Tebu5)**2 + 3.7216*(Tebu5) + 3.0035)+(1-fracLiso) *(-6.5690*10**(-12)*(Tebu5)**6 + 6.3049*10**(-  
  9)*(Tebu5)**5 - 2.3080*10**(-6)*(Tebu5)**4 + 3.8339*10**(-4)*(Tebu5)**3 - 3.0632*10**(-2)*(Tebu5)**2 + 2.7553*(Tebu5) +  
  2.4957*10**3) ),  
  eff_turb1 *(Hiso_5-Hwa4)-(Hwa5-Hwa4) ,  
  Hwa5-((fracL)*(3.6082*10**(-12)*(Tebu5)**6 - 3.4120*10**(-9)*(Tebu5)**5 + 1.2303*10**(-6)*(Tebu5)**4 - 2.0306*10**(-  
  4)*(Tebu5)**3 + 1.5552*10**(-2)*(Tebu5)**2 + 3.7216*(Tebu5) + 3.0035)+(1-fracL) *(-6.5690*10**(-12)*(Tebu5)**6 + 6.3049*10**(-  
  9)*(Tebu5)**5 - 2.3080*10**(-6)*(Tebu5)**4 + 3.8339*10**(-4)*(Tebu5)**3 - 3.0632*10**(-2)*(Tebu5)**2 + 2.7553*(Tebu5) +  
  2.4957*10**3) ),  
  W_turbine2- fwa4*(Hwa4-Hwa5) ,  
  
  #Cooling  
  fwa6-fwa5,  
  #Porque esta en equilibrio L-V  
  Twa6-Twa5 ,  
  Hwa6-(1*(3.6082*10**(-12)*(Twa6)**6 - 3.4120*10**(-9)*(Twa6)**5 + 1.2303*10**(-6)*(Twa6)**4 - 2.0306*10**(-4)*(Twa6)**3 +  
  1.5552*10**(-2)*(Twa6)**2 + 3.7216*(Twa6) + 3.0035) ),  
  Q_HX6-(fwa5*( Hwa6-Hwa5)/1000) ,  
  
  #Premixing  
  W_pump1-(fwa6*(Hwa7-Hwa6) ),  
  Hwa7-(Hwa6+0.001*(PWa7-PWa6)*101.325) ,  
  
  #mixing  
  
  #mass balance  
  fwa7 -fwa6,
```

```

fwa7+fwa8 -fwa9 ,

#energy balance
fwa7*Hwa7 + Hwa8*fwa8 - (fwa9*Hwa9 ),
Hwa9 -(4.2921*(Twa9) + 4.1269),

#la quito porque ahora es redundante.
PWa9*760 - (np.exp(18.3036-3816.44/(227.02+Twa9)) ),

#model.C37 = pyo.Constraint(expr =fwa9 == fwa1)
Hwa1-(Hwa9+0.001*(PWa1-PWa9)*101.325 ),
W_pump2 -fwa9*(Hwa1-Hwa9) ,

Twa1 -Twa9 ,
Twa7 -Twa6 ,
PWa1 -PWa2 ,
Hwa2-((-0.00000000000011619*PWa2**2 - 0.000000000087596*PWa2 -
0.00000000022611)*(Twa2)**4+(0.000000004298*PWa2**2 + 0.00000003276*PWa2 + 0.0000007313)*(Twa2)**3+(-
0.0000005801*PWa2**2 - 0.000046*PWa2 - 0.0005009)*(Twa2)**2+(0.0003383*PWa2**2 + 0.02947*PWa2 + 2.195)*(Twa2)+(-
0.072042*PWa2**2 - 7.7877*PWa2 + 2440.8)),
Swa2-((0.00000000015719*PWa2 + 0.0000000074013)*(Twa2)**3+(-0.0000000010074*PWa2**2 - 0.000000030171*PWa2 -
0.0000028872)*(Twa2)**2+(0.000000094914*PWa2**2 + 0.000029097*PWa2 + 0.0050938)*(Twa2)+(0.000041223*PWa2**2 -
0.028841*PWa2 + 5.9537) ),
load-fwa2/fwamax,

eff_turb1-(isoeff*(-1.0176*(load)**4+2.4443*(load)**3-2.1812*(load)**2+1.0535*load+0.701)),
intcp-(1.443*(THTF1-THTF2)+0.000172*0.5*( (THTF1)**2-(THTF2)**2)),
fHTF1-Q_collector/intcp,
fHTF2-fHTF1,
]

x = fsolve(func,
[227,2885,3012,284,7,284,284,3000,3000,7,7 ,184,805,18.78,
9500,41,4,14.6,41,0.16,2200,2344, 0.097,9800, 14.6, 41, 174,
-31.84,16.23,175,14.6,19,796,184,100,169.42,184.57,41.82,18,14.6,
3520,7,0.6,51000,0.8,417,122.28,122.28],
args=(fwa2,PWa2,PWa3,PWa5),

xtol=0.0000000000001)

#[200,3000,3000,284,7,284,284,3000,3000,7,7,184,18.78,805,18,9000,41,4,14,41,0.166,2000,2300,0.097,9800,14.67,41,17,-
31,16,175,14,18.6,796,195 ,100,170,184,41,3500,7,8,9  ]

Tiso_3=x[0]
Hiso_3=x[1]
Hwa3=x[2]
Twa3=x[3]
Swa3=x[4]
Twa4=x[5]
Twa8=x[6]
Hwa4=x[7]
Hwa8=x[8]
Swa4=x[9]
Swa8=x[10]
Tebu3=x[11]
Twa2=522
Hwa1=x[12]
fwa3=x[13]
W_turbine1=x[14]
Tebu5=x[15]
fwa8=x[16]
fwa5=x[17]
Twa5=x[18]
fracLiso=x[19]

```



BN2=-1.36E-02  
 CCO2=-5.60E-05  
 CO2=1.75E-05  
 CN2=2.68E-05  
 DCO2=1.72E-08  
 DO2=-1.07E-08  
 DN2=-1.17E-08  
 k\_p=1.4  
 nu\_p=0.8  
 eps1=1E-4  
 T\_cooldown=25  
 A\_helio=0.120  
 rd\_helio=0.9  
 Rad=0.7  
 Yield\_field=0.6  
 k\_p2=1.3

Number=3  
 Relacion=2  
 Q\_Sun=90.72

#Caracterizacion del punto 2  
 #Declaracion del las variables

flujoCO2HX3C1=x[0]  
 flujoO2HX3C1= x[1]  
 flujoN2HX3C1= x[2]  
 MW\_gas=x[3]  
 TrabajoC=x[4]  
 TemperatureCF=x[5]  
 flujoCO2C1FU=x[6]  
 flujoO2C1FU=x[7]  
 flujoN2C1FU=x[8]  
 Calorcedidomulietapa=x[9]  
 Calorreact=x[10]  
 flujoCO2FUT=x[11]  
 flujoO2FUT=x[12]  
 flujoN2FUT=x[13]  
 Calorproduc=x[14]  
 TemperatureFT=x[15]  
 flujoCO2THX4=x[16]  
 flujoO2THX4=x[17]  
 flujoN2THX4=x[18]  
 TemperatureHX4=x[19]  
 WTurbGas=x[20]  
 Calorcedidfocofrio=x[21]  
 Power=x[22]

```

return [flujoCO2HX3C1-mgas*ComposicionCO2,
        flujoO2HX3C1-mgas*ComposicionO2,
        flujoN2HX3C1-mgas*ComposicionN2,
        mgas-MW_gas*(flujoCO2HX3C1/PmolecularCO2+flujoO2HX3C1/PmolecularO2+flujoN2HX3C1/PmolecularN2),
        TrabajoC*((MW_gas+0.001)*(k_p-1))-
(Number*(1/nu_p)*mgas*(8.314*k_p*(T_cooldown+273))*((Relacion)**((1/(Number+eps1))*((k_p-1)/k_p))-1)/1000),
        TemperatureCF-(((T_cooldown+273)+(1/nu_p)*(T_cooldown+273))*((Relacion)**((1/(Number))*((k_p-1)/k_p))-1))-273),
        flujoCO2HX3C1-flujoCO2C1FU,
        flujoO2HX3C1-flujoO2C1FU,
        flujoN2HX3C1-flujoN2C1FU,
        Calorcedidomulietapa+(Number-1)*(flujoCO2C1FU/PmolecularCO2)*(ACO2*(
25)+(1/2)*BCO2*((TemperatureCF+273)**2-(298)**2)+(1/3)*CCO2*((TemperatureCF+273)**3-(298)**3)+
(1/4)*DCO2*((TemperatureCF+273)**4-(298)**4))+((flujoO2C1FU/PmolecularO2)*(AO2*(
25)+(1/2)*BO2*((TemperatureCF+273)**2-
(298)**2)+(1/3)*CO2*((TemperatureCF+273)**3-
(298)**3)+(1/4)*DO2*((TemperatureCF+273)**4-(298)**4))+((flujoN2C1FU/PmolecularN2)*(AN2*( TemperatureCF-25)+
(1/2)*BN2*((TemperatureCF+273)**2-(298)**2)+(1/3)*CN2*((TemperatureCF+273)**3-(298)**3)+
(1/4)*DN2*((TemperatureCF+273)**4-(298)**4)))/1000),

```

```

Calorreact-(((flujoco2C1FU/PmolecularCO2)*(ACO2*(
TemperatureCF-25)+(1/2)*BCO2*((TemperatureCF+273)**2-
(298)**2)+(1/3)*CCO2*((TemperatureCF+273)**3-(298)**3)+
(1/4)*DCO2*((TemperatureCF+273)**4-(298)**4)))+(flujoco2C1FU/PmolecularO2)*(AO2*(
TemperatureCF-
25)+(1/2)*BO2*((TemperatureCF+273)**2-
(298)**2)+(1/3)*CO2*((TemperatureCF+273)**3-
(298)**3)+(1/4)*DO2*((TemperatureCF+273)**4-(298)**4)))+(flujon2C1FU/PmolecularN2)*(AN2*(
TemperatureCF-25)+
(1/2)*BN2*((TemperatureCF+273)**2-(298)**2)+(1/3)*CN2*((TemperatureCF+273)**3-(298)**3)+
(1/4)*DN2*((TemperatureCF+273)**4-(298)**4)))/1000),
flujoco2C1FU-flujoco2FUT,
flujoo2C1FU-flujoo2FUT,
flujon2C1FU-flujon2FUT,
Calorproduc-(Calorreact+Q_Sun),
Calorproduc-(
(flujoco2FUT/PmolecularCO2*(ACO2*(
TemperatureFT-25)+
(1/2)*BCO2*((TemperatureFT+273)**2-(298)**2)+
(1/3)*CCO2*((TemperatureFT+273)**3-(298)**3)+
(1/4)*DCO2*((TemperatureFT+273)**4-(298)**4))
+flujoo2FUT/PmolecularO2*(AO2*(
TemperatureFT-25)+
(1/2)*BO2*((TemperatureFT+273)**2-(298)**2)+
(1/3)*CO2*((TemperatureFT+273)**3-(298)**3)+
(1/4)*DO2*((TemperatureFT+273)**4-(298)**4))
+flujon2FUT/PmolecularN2*(AN2*(
TemperatureFT-25)+
(1/2)*BN2*((TemperatureFT+273)**2-(298)**2)+
(1/3)*CN2*((TemperatureFT+273)**3-(298)**3)+
(1/4)*DN2*((TemperatureFT+273)**4-(298)**4))
)/1000),
flujoco2FUT- flujoco2THX4,
flujoo2FUT- flujoo2THX4,
flujon2FUT- flujon2THX4,

TemperatureHX4-(((TemperatureFT+273)+(nu_p)*(TemperatureFT+273)*((1/Relacion)**(0.3/1.3)-1))-273),

WTurbGas*((MW_gas+0.001)*(k_p2-1))*1000-(nu_p*mgas*(8.314*k_p2*(TemperatureFT+273)*((1/Relacion)**((k_p2-
1)/k_p2)-1))),
Calorcedidfocofrio-((flujoco2THX4/PmolecularCO2*(ACO2*(
TemperatureHX4-25)+
(1/2)*BCO2*((TemperatureHX4+273)**2-(298)**2)+
(1/3)*CCO2*((TemperatureHX4+273)**3-(298)**3)+
(1/4)*DCO2*((TemperatureHX4+273)**4-(298)**4))
+flujoo2THX4/PmolecularO2*(AO2*(
TemperatureHX4-25)+
(1/2)*BO2*((TemperatureHX4+273)**2-(298)**2)+
(1/3)*CO2*((TemperatureHX4+273)**3-(298)**3)+
(1/4)*DO2*((TemperatureHX4+273)**4-(298)**4))
+flujon2THX4/PmolecularN2*(AN2*(
TemperatureHX4-25)+
(1/2)*BN2*((TemperatureHX4+273)**2-(298)**2)+
(1/3)*CN2*((TemperatureHX4+273)**3-(298)**3)+
(1/4)*DN2*((TemperatureHX4+273)**4-(298)**4))
)/1000),

WTurbGas-(TrabajoC+Power)

]
x = fsolve(func,
[0,12.25,37.75,28.88,4.89,58,0,12.25,37.75,-
3.2784,1.6392,0,12.25,37.75,92.359,1980.98,0,12.25,37.75,1651,20.57,85.93,15.67
],
args=(mgas,Relacion),

```



```

b=2 #coeficiente estequimetrico
na=1 #orden de reaccion
nb=2 #orden de reaccion
A=1e5 #Factor preexponencial
Ea=30000 #(J/mol)

```

```

#Parameters
R=8.31 #J/mol K
k1=A*np.exp(-Ea/(R*T))

```

```

Conc=[(), ()]

```

```

def kinetic (Conc,t):
    dNdt = np.zeros(2)
    dNdt[0] = -a*k1*Conc[0]**na*Conc[1]**nb
    dNdt[1] = -(b/a)*k1*Conc[0]**na*Conc[1]**nb
    return dNdt

```

```

t=np.linspace(0,tfin,100+1)

```

```

Res=integrate.odeint(kinetic,Cin,t)

```

```

"""
plt.plot(t,Res[:,0], label='A')
plt.plot(t,Res[:,1], label='B')
plt.legend()
plt.xlabel('time(s)')
plt.ylabel('Concentration (mol/l)')
"""

```

```

Res1=pd.DataFrame()

```

```

for i in range(101):
    if Cin[0] < Cin[1]:
        Error = 0.03*Res[i,0]
        Res1.loc[i,'CA'] = m.uniform(Res[i,0]-Error,Res[i,0]+Error)
        Res1.loc[i,'CB'] = m.uniform(Res[i,1]-Error,Res[i,1]+Error)
    else:
        Error = 0.03*Res[i,1]
        Res1.loc[i,'CA'] = m.uniform(Res[i,0]-Error,Res[i,0]+Error)
        Res1.loc[i,'CB'] = m.uniform(Res[i,1]-Error,Res[i,1]+Error)

```

```

Final=pd.DataFrame()

```

```

for i in [0,10,20,30,40,50,60,70,80,90,100]:
    Final.loc[i,'t (s)'] = t[i]
    Final.loc[i,'CA (mol/l)'] = Res1.iloc[i,0]
    Final.loc[i,'CB (mol/l)'] = Res1.iloc[i,1]

```

```

plt.plot('t (s)', 'CA (mol/l)', data=Final, label='A',linestyle = 'None',marker='o', markerfacecolor='blue')
plt.plot('t (s)', 'CB (mol/l)', data=Final, label='B',linestyle = 'None',marker='o', markerfacecolor='green')
plt.legend()
plt.xlabel('tiempo(s)')
plt.ylabel('Concentración (mol/l)')

```

```

plt.savefig('FigureReactor.png',bbox_inches='tight',dpi=500)

```

```

Final.to_csv(r'F:\ANTONIO DOCTORADO\DOCENCIA\INNOVACION DOCENTE\Reactores\Final.csv',index=False)

```

## Torre de Humidificación-----

```

import numpy as np

```

```

import pandas as pd
import matplotlib.pyplot as plt
from scipy.optimize import newton_krylov

#~ Parameters
#Instructor
slope=-1.7;
slope_real=slope*(1+0.05*np.random.uniform(-1,1));

#Student
data=pd.read_csv('dataCT.csv')
data.columns=['Feeddata']
Feeddt=data['Feeddata']
Fd=data['Feeddata'].to_numpy();
Feedrr=Fd.astype(float);
Ptotal=Feedrr[0];
T1=Feedrr[1];
T5=Feedrr[2];
orif=Feedrr[3];
Magua=Feedrr[4];
Heating=Feedrr[5];
Phi_in=Feedrr[6];

#Ptotal=698           #Total pressure (mmHg)
#T1=21.9             #Air Inlet Dry Temperature point 1 (°C)
#T2                 #Air Inlet Wet Temperature point 2 (°C)
#T3                 #Air Outlet Wet Temperature point 3 (°C)
#T4=19.4            #Air Outlet Dry Temperature point 4 (°C)
#T5=33.2            #Water Inlet Temperature point 5 (°C)
#T6                 #Water Outlet Temperature point 6 (°C)

#orif=18             #Orifice diferencial(mmH2O)
#Magua=20            #Water mass g/s
#Heating=1;
#Phi_in=0.75;

##Model
Puntos=10;
#~ Computing H_WetAir(T) curve
T_evaluation=np.arange(1,51,1) #Temperatures to evaluate H

PV=[];
Yvap=[];
H=[];

for j in T_evaluation: #The curve is calculated from 1 to 50 points
    PV_j=-2.73933E-11*j**6+2.85333E-8*j**5+1.27164E-6*j**4+2.36198E-4*j**3+1.00275E-2*j**2+3.38158E-1*j+4.57466;
    Yvap_j=0.62*PV_j/(Ptotal-PV_j);
    H_j=(0.24+0.46*Yvap_j)*j+597.2*Yvap_j;

    PV.append(PV_j);
    Yvap.append(Yvap_j);
    H.append(H_j);

#print('PV=', PV, '\n')
#print('Yvap=', Yvap, '\n')
#print('H=', H, '\n')

a=np.polyfit(T_evaluation,H,4)

#~ AIR INLET STREAM
#~ Inlet Wet Vapor Pressure (point 2)

PWti=-2.73933e-11*(T1)**6+2.85333e-8*(T1)**5+1.27164e-6*(T1)**4+2.36198e-4*(T1)**3+1.00275e-2*(T1)**2+3.38158e-1*(T1)+4.57466;
Phi_in_real=Phi_in*(1+0.05*np.random.uniform(-1,1));

```

```

T2=(Phi_in_real-1)*PWti/0.5+T1;

#~ Inlet humidity (kg/kg_dry_air)
Ye=0.62*Phi_in_real*PWti/(Ptotal-Phi_in_real*PWti)
#~ Inlet stream enthalpy (kcal/kg_a_s)
He=(0.24+0.46*Ye)*T1+597.2*Ye

MasL=Magua/1000;
#~ L/V slope (Operation line)
Heating_real=Heating*(1+0.05*np.random.uniform(-1,0));

T6=T5-Heating_real/(MasL*4.18)

VH=(1/29+Ye/18)*(0.082*(273+T1))/(Ptotal/760)
MasV=0.0137*(orif/VH)**0.5

Hs=(T5-T6)*MasL/MasV+He ;

## Discritization

Points=np.arange(0,Puntos,1);
paso=(T5-T6)/Puntos;

#~ AIR OUTLET STREAM
#~ Outlet Wet Vapor Pressure (point 3)

Hliq=[]
Tliq=[]
for j in Points:
    Tlop=T6+j*paso
    Hop=He+(MasL/MasV)*j*paso;
    Hliq.append(Hop);
    Tliq.append(Tlop);

Heq=[];
Teq=[];
for i in Points:

    def Eq(tempi):
        return [Hliq[i]+slope_real*(tempi-Tliq[i]) -(a[0]*tempo**4+a[1]*tempo**3+a[2]*tempo**2+a[3]*tempo+a[4])]
    Guess=(20);

    Res = newton_krylov(Eq, Guess, method='lgmres')
#Change the data type to float to manage the results
    Res=np.float_(Res)
#Print the results list
#Store each concentration result in a new variable
    tempinter=Res;
    Hinter= a[0]*tempinter**4+a[1]*tempinter**3+a[2]*tempinter**2+a[3]*tempinter+a[4]
    Teq.append(tempinter)
    Heq.append(Hinter)

Points2=np.arange(0,Puntos-1,1);
Tgas=[T1];
for k in Points2:
    Tgas_next=Tgas[k]+(Hliq[k+1]-Hliq[k])*((Teq[k]-Tgas[k])/(Heq[k]-Hliq[k]));
    Tgas.append(Tgas_next)

T4=Tgas[Puntos-1];
PWts=-2.73933e-11*(T4)**6+2.85333e-8*(T4)**5+1.27164e-6*(T4)**4+2.36198e-4*(T4)**3+1.00275e-2*(T4)**2+3.38158e-1*(T4)+4.57466

Ys=(Hs-0.24*T4)/(597.2+0.46*T4);

def Eq(Phi_salida):

```

```

return [Ys-0.62*Phi_salida*PWts/(Ptotal-Phi_salida*PWts)]

Guess=(0.8);

Res2 = newton_krylov(Eq, Guess, method='lgmres')
Res2=np.float_(Res2)
#Print the results list
#Store each concentration result in a new variable
Phi_out=Res2;

T3=(Phi_out-1)*PWts/0.5+T4;

InvHimHe=[]

#Plot the results
for k in Points:
    InvHimHe_k=1/(Heq[k]-Hliq[k])
    InvHimHe.append(InvHimHe_k)

b=np.polyfit(Hliq,InvHimHe,3)

#Calculate NUT
NUTb=(1/4)*b[0]*Hs**4+(1/3)*b[1]*Hs**3+0.5*b[2]*Hs**2+b[3]*Hs;
NUTa=(1/4)*b[0]*He**4+(1/3)*b[1]*He**3+0.5*b[2]*He**2+b[3]*He;
NUT=NUTb-NUTa
z=0.48
#Calculate ky
ky=NUT*MasV/(110*0.15*0.15*z)

print("T1(°C)=", T1 , "T2(°C)=", T2,"T3(°C)=",T3,"T4(°C)=",T4,"T5(°C)=",T5,"T6(°C)=",T6)

print("NUT", NUT , "ky(kg/(m2·s))=", ky)
plt.plot(Tliq,Hliq,'k', Teq,Heq,'k-',Tgas,Hliq,'k.')
plt.xlabel('T(°C)')
plt.ylabel('H(kcal/kg)')
plt.savefig('Columnahumidificacion.png',bbox_inches='tight',dpi=600)

df = pd.DataFrame([T1,T2,T3,T4,T5,T6,NUT,ky,Tliq,Hliq,Teq,Heq,Tgas])

df.to_csv('ResultsCT.csv')

```

## Destilación batch/por lotes-----

```

import numpy as np
import math
import pandas as pd
from scipy.optimize import newton_krylov

#Data input
#plates=11;
#Instructor
plates=5;#From 5 to 11
error=0.05;

#Student
data=pd.read_csv('datadist.csv')
data.columns=['Feeddata']
Feeddt=data['Feeddata']
Fd=data['Feeddata'].to_numpy();
Feedrr=Fd.astype(float);
yd=Feedrr[0];
xf=Feedrr[1];
Feed=Feedrr[2];
Reflux=Feedrr[3];

```

```

#Student
#yd=0.83;
#xf=0.11;
#Feed=350;
#Reflux=3.5; #From 1.5 to 3.5

#Model
LtoD=Reflux*(1+error*np.random.uniform(-1,1));
Pend=(LtoD)/(LtoD+1);

Variable=[(),(),()]

xw=[]
inv=[]
Integral=[]
#Equations: We define a function which collect the equations to solve

def Eq(Variable):

#Variable specification
xres=Variable[0];
Residue=Variable[1];
Distillate=Variable[2];

#%List of Xd to solve the integral
ydest=np.arange(0.75,0.86,0.01);

long=len(ydest);

#Polynomial coefficients for vapor liquid equilibrium
coea2=-16.78846;
coeb2=5.35043771;
coec2=0.07820301;

coea1=0.2774380;
coeb1=0.21006435;
coec1=0.4766639;

discret=np.arange(0,long,1);
trays=np.arange(0,plates,1);
discrint=np.arange(0,long-1,1);
#iteration for computing the integral
for i in discret:
    yeq=[]
    xeq=[]

    yd=ydest[i];
    yeq.append(yd);

    for j in trays:
#Algebraic solution for tray calculation

        if yeq[j]> 0.5044:
            xeq_j=(-coeb1+(coeb1**2-4*(coea1*(coec1-yeq[j])))**0.5)/(2*coea1);
        else:
            xeq_j=(-coeb2+(coeb2**2-4*(coea2*(coec2-yeq[j])))**0.5)/(2*coea2);

        xeq.append(xeq_j)

    if j<plates-1:
        yeq_j1=Pend*xeq[j]+(1-Pend)*yd;
        yeq.append(yeq_j1)

xw_i=xeq[plates-1];

```

```

    inv_i= 1/(ydest[j]-xw_i);
    xw.append(xw_i)
    inv.append(inv_i)

del yeq
del xeq

#Integral computation
difer=[];
for k in discrint:
    difer_k =inv[k]*(xw[k+1]-xw[k]);
    difer.append(difer_k)

coefint=np.polyfit(xw,inv,2);
Integral=(coefint[0]/3)*(xf**3-xres**3)+(coefint[1]/2)*(xf**2-xres**2)+(coefint[2])*(xf-xres);

#List of equations to be solved
return [Residue*math.exp(Integral)-Feed,
        Feed-Distillate-Residue,
        Feed*xf-Distillate*yd-Residue*xres]

Guess=(0.03,315,20);

Res = newton_krylov(Eq, Guess, method='lgmres')
#Change the data type to float to manage the results
#Res=np.float_(Res)
#Print the results list
#print ('Res',Res)

#Store each concentration result in a new variable
xresiduo=Res[0];
FResidue=Res[1];
FDistillate=Res[2];

FResidue_real=FResidue*(1+error*np.random.uniform(-1,1));
FDistillate_real=FDistillate*(1+error*np.random.uniform(-1,1));
xresiduo_real=xresiduo*(1+error*np.random.uniform(-1,1));

#Print the new variables to show the concentration of each component separatly
print("x_residue=", xresiduo, "FResidue (kmol)=", FResidue, "FDistillate(kmol)=", FDistillate)

print("x_residue(exp)=", xresiduo_real, "FResidue(exp) (kmol)=", FResidue_real, "FDistillate_real(kmol)=", FDistillate_real)

df = pd.DataFrame([xresiduo_real,FResidue_real,FDistillate_real])

df.to_csv("ResultsDistillation.csv")

```