



**UNIVERSIDAD SIMÓN BOLÍVAR**  
**DECANATO DE ESTUDIOS PROFESIONALES**  
**COORDINACIÓN DE INGENIERÍA QUÍMICA**

**MÓDULO DE SIMULACIÓN DE UNA TORRE DE DESTILACIÓN**  
**MULTICOMPONENTE POR MÉTODOS RIGUROSOS**

Por:

Luis Jesús Díaz Manzo

**PROYECTO DE GRADO**

Presentado ante la Ilustre Universidad Simón Bolívar  
como requisito para optar al título de  
Ingeniero Químico

**Sartenejas, Septiembre de 2015**



**UNIVERSIDAD SIMÓN BOLÍVAR**  
**DECANATO DE ESTUDIOS PROFESIONALES**  
**COORDINACIÓN DE INGENIERÍA QUÍMICA**

**MÓDULO DE SIMULACIÓN DE UNA TORRE DE DESTILACIÓN  
MULTICOMPONENTE POR MÉTODOS RIGUROSOS**

Por:

Luis Jesús Díaz Manzo

Realizado con la asesoría de:

Prof. Sabrina Di Scipio

Prof. Alessandro Trigilio

**PROYECTO DE GRADO**

Presentado ante la Ilustre Universidad Simón Bolívar  
como requisito para optar al título de  
Ingeniero Químico

**Sartenejas, Septiembre de 2015**



**UNIVERSIDAD SIMÓN BOLÍVAR**  
**DECANATO DE ESTUDIOS PROFESIONALES**  
**COORDINACIÓN DE INGENIERÍA QUÍMICA**

**ACTA DE EVALUACIÓN**



**UNIVERSIDAD SIMÓN BOLÍVAR**  
**DECANATO DE ESTUDIOS PROFESIONALES**  
**COORDINACIÓN DE INGENIERÍA QUÍMICA**

**MÓDULO DE SIMULACIÓN DE UNA TORRE DE DESTILACIÓN  
MULTICOMPONENTE POR MÉTODOS RIGUROSOS**

**PROYECTO DE GRADO**

Presentado por: Luis Jesús Díaz Manzo Carnet 07-40838

Realizado con la asesoría de: Prof. Sabrina Di Scipio y Prof. Alessandro Trigilio

**RESUMEN**

La destilación es una parte fundamental en el procesamiento de hidrocarburos, en particular para la refinación de petróleo. Una de las herramientas de diseño en la industria son los simuladores de columnas de destilación, que facilitan la obtención de resultados confiables con algoritmos computarizados los cuales se encargan de procesar los cálculos complejos. A pesar de estar basados en algoritmos disponibles en la literatura, la mayor parte de los Ingenieros Químicos dependen de programas comerciales de código cerrado para sus necesidades, cuando estos pueden desarrollarse en la academia y servir a la formación de los estudiantes para enseñar acerca de su funcionamiento y de sus limitaciones. En este trabajo se planteó el objetivo de implementar diversos métodos de resolución de torres de destilación, entre los cuales se escogieron los de: Fenske, Underwood y Gilliland (FUG), Hengstebeck, Lewis y Matheson, Punto de Burbuja de Wang y Henke y Newton Raphson de Naphtali Sandholm. El lenguaje de programación seleccionado fue MATLAB®. Se utilizó programación orientada a objetos con el propósito de diseñar un programa modular y funcional, de código reutilizable y de robusto encapsulamiento para prevenir la diseminación de los errores de programación. Se desarrolló una interfaz gráfica de usuario amigable. La validez de los resultados obtenidos se realizó mediante una comparación de los perfiles de temperaturas, composiciones y flujos molares de vapores y líquidos entre ambos métodos rigurosos, y además con el programa comercial PRO/II®, arrojando el método Newton Raphson diferencias con el simulador inferiores al 2% en todos los casos de ejemplo y Punto de Burbuja una desviación de hasta 33% en el flujo de vapor o de líquido en una etapa.

**Palabras clave:** Simulación, destilación, separación, multicomponente, métodos rigurosos, cortos, Newton Raphson, *Bubble Point*, FUG, Hengstebeck, Lewis y Matheson.

## **AGRADECIMIENTOS**

Agradezco muy especialmente a mi familia. A mi mamá por todo el apoyo anímico, la inyección de confianza que me ha ayudado a continuar luchando, a mi papá que me ha dado el soporte y me enseñó del esfuerzo y la perseverancia.

Le estoy profundamente agradecido a mi tutora la profesora Sabrina Di Scipio por su guía oportuna, la disposición permanente a responder y orientar, su valiosa asistencia, por confiar en mi capacidad de hacerlo mejor en cada sucesiva versión. Además por ayudarme en la redacción de este documento.

A mi tutor el profesor Alessandro Trigilio por darme todas las respuestas a las preguntas que tuve y por incentivarme a tomar este camino de aventurarme a aprender y dominar herramientas computacionales que ya me había atraído en múltiples ocasiones y a las cuales me aseguraré de sacarle mucho provecho en mi vida.

A todos mis profesores en mi formación académica, en particular al profesor Jaime Riera y al profesor Freddy Figueira por enseñarme el área de termodinámica que ganó mi mayor interés.

A todos aquellos que de alguna manera han contribuido o influido en el desarrollo de este trabajo.

## ÍNDICE GENERAL

MÓDULO DE SIMULACIÓN DE UNA TORRE DE DESTILACIÓN MULTICOMPONENTE POR MÉTODOS RIGUROSOS .....	i
ACTA DE EVALUACIÓN .....	iii
RESUMEN .....	iv
AGRADECIMIENTOS .....	v
ÍNDICE GENERAL .....	vi
ÍNDICE DE TABLAS .....	ix
ÍNDICE DE FIGURAS .....	x
LISTA DE SÍMBOLOS Y UNIDADES .....	xiii
<i>Latinos</i> .....	<i>xiii</i>
<i>Griegos</i> .....	<i>xiv</i>
<i>Subíndices y Superíndices</i> .....	<i>xiv</i>
<i>Abreviaturas</i> .....	<i>xv</i>
INTRODUCCIÓN .....	1
CAPÍTULO 1 MARCO TEÓRICO.....	8
2.1 Ecuaciones de estado para sustancias puras .....	8
2.2 Extensión de las EdE cúbicas a sistemas multicomponentes .....	10
2.3 Cálculo de equilibrio de fases líquido-vapor a partir de las EdE cúbicas .....	11
2.4 Cálculos de entalpía y balances de energía .....	12
2.5 Destilación.....	13
2.6 Métodos de cálculo en destilación .....	14
2.6.1 Métodos cortos o aproximados .....	14

2.6.1.1 Método de Fenske, Underwood y Gilliland (FUG) .....	14
2.6.1.2 Método gráfico de McCabe y Thiele .....	19
2.6.1.3 Extensión de Hengstebeck del método de McCabe y Thiele.....	24
2.6.1.4 Otros métodos aproximados .....	26
2.6.1.5 Métodos rigurosos .....	30
2.7 <i>Análisis de Grados de Libertad para sistemas de separación en cascada</i> .....	47
CAPÍTULO 2 METODOLOGÍA .....	49
3.1 <i>Programación Orientada a Objetos</i> .....	49
3.1.2 Descripción de los objetos .....	50
3.2. <i>Algoritmos de cálculo termodinámicos</i> .....	51
3.3 <i>Métodos de cálculo de torres de destilación</i> .....	57
3.4 <i>Interfaz gráfica de usuario</i> .....	60
CAPÍTULO 3 RESULTADOS Y DISCUSIÓN .....	62
4.1 <i>Planificación y desarrollo de los métodos de cálculo</i> .....	62
4.2 <i>Ejemplo 1</i> .....	64
4.2 <i>Ejemplo 2</i> .....	73
CONCLUSIONES Y RECOMENDACIONES .....	82
REFERENCIAS .....	84
APÉNDICE A CÁLCULO DE EJEMPLO DE LA INTERFAZ GRÁFICA .....	89
APÉNDICE B EJEMPLO DE USO DE LA INTERFAZ GRÁFICA .....	105
APÉNDICE C ARCHIVO EXPORTADO DE DATOS DEL SIMULADOR COMERCIAL PRO/II .....	115
APÉNDICE D ARCHIVO DE SALIDA GENERADO POR ALGORITMO DE LEWIS MATHESON .....	117

APÉNDICE E HISTORIAL DE CÁLCULO DEL EJEMPLO 2 .....	126
---	-----



## ÍNDICE DE TABLAS

Tabla 2.1 Parámetros de las ecuaciones cúbicas de estado (Prausnitz <i>et al.</i> , 2005).....	11
Tabla 2.2 Constantes asociadas a la función de cohesión de las EdE cúbicas (Prausnitz <i>et al.</i> , 2005).....	11
Tabla 2.3. Implicaciones de la condición térmica de la alimentación a los flujos molares.....	23
Tabla 4.1 Comparación de la distribución de los componentes para el ejemplo 1 entre los diferentes métodos del ejemplo 1 .....	66
Tabla 4.2 Calor del condensador y del rehervidor de la columna obtenido por los métodos rigurosos de Bubble Point, Newton Raphson y usando el simulador comercial PRO/II .....	73
Tabla 4.3. Condiciones de las corrientes de alimentación a separar. ....	74
Tabla 4.4. Resultados de la simulación con los métodos rigurosos de BubblePoint, Newton Raphson y los obtenidos con el uso de Inside-Out en un simulador comercial .....	75

## ÍNDICE DE FIGURAS

Figura 2.1 Algoritmo de cálculo para destilación multicomponente por el método corto de FUG (basado en Seader <i>et al.</i> , 2011).....	16
Figura 2.2 Esquema general de una columna de fraccionamiento .....	20
Figura 2.3. Influencia de $q$ en la recta de operación de la zona de agotamiento (Asuaje, 2009) ..	23
Figura 2.4. Representación en el diagrama $y$ - $x$ de los casos ideales de reflujo total (a) y de reflujo mínimo (b) (Asuaje, 2009) .....	24
Figura 2.5. Representación esquemática de una etapa de equilibrio (Seader <i>et al.</i> , 2011).....	30
Figura 2.6. Representación esquemática de un arreglo en cascada de etapas teóricas, un fraccionador (basada en Seader <i>et al.</i> , 2011).....	32
Figura 2.7. Algoritmo de simulación del método de Punto de Burbuja (BP) propuesto por Wang y Henke (adaptado para este estudio basado en Seader <i>et al.</i> 2008) .....	38
Figura 2.8. Algoritmo de Naphtali Sandholm para la resolución de torres de fraccionamiento por el método de Newton Raphson de Corrección Simultánea (Seader <i>et. al.</i> , 2011).....	43
Figura 3.1 Jerarquía de los objetos de cálculo de propiedades termodinámicas.....	51
Figura 3.2 Algoritmo para hallar la temperatura de burbuja de una mezcla líquida.....	53
Figura 3.3 Algoritmo para hallar la temperatura de rocío de una mezcla vapor.....	54
Figura 3.4 Algoritmo para hallar la fracción vaporizada de una mezcla.....	55
Figura 3.5 Diagrama de los objetos de cálculo de métodos de destilación.....	57
Figura 3.6 Interfaz gráfica de usuario de DESTILMATLABUSB.....	60
Figura 4.1. Datos del ejemplo 1.....	65
Figura 4.2 Trazado del número mínimo de etapas para el ejemplo por el método de Hengstebeck para aproximación pseudobinaria a McCabe-Thiele .....	67
Figura 4.3 Resultado del método de Hengstebeck a reflujo real ( $L/D = 2$ ).....	68
Figura 4.4 Flujos molares de vapor para los algoritmos de LM, BP, NR e Inside-Out de PRO/II® .....	69
Figura 4.5 Flujos molares de líquido para los algoritmos de LM, BP, NR e Inside-Out de PRO/II® .....	70
Figura 4.6 Perfil de temperaturas por etapa teórica para los algoritmos de LM, BP, NR y de Inside-Out de Pro/II® .....	71

Figura 4.7 Perfil de composiciones para cada uno de los componentes por etapa teórica calculado con los algoritmos de LM, BP y NR .....	72
Figura 4.8. Esquema y parámetros de diseño del ejemplo 2 .....	74
Figura 4.9 Perfil de temperaturas en cada etapa teórica para los métodos de BubblePoint, Newton-Raphson de Naphtali-Sandholm y el de Inside-Out .....	76
Figura 4.10. Perfil de flujo molar de vapor que abandona cada etapa teórica, según el resultado arrojado por los algoritmos de BP, NR y de InsideOut .....	77
Figura 4.11. Flujo molar de líquido para cada etapa teórica, obtenida mediante los métodos de BP, NR y de IO.....	78
Figura 4.12. Convergencia de los perfiles de flujo molar de vapor según el método de punto de burbuja.....	79
Figura.4.13. Convergencia de los perfiles de flujo molar de líquido utilizando el método de punto de burbuja .....	79
Figura 4.14. Convergencia de los perfiles de temperatura en cada plato utilizando el método BubblePoint. ....	80
Figura 4.15. Convergencia de los perfiles de flujo de vapor en cada etapa mediante el método de Newton Raphson .....	80
Figura 4.16. Convergencia de los perfiles de flujo de líquido en cada etapa mediante el método de Newton Raphson .....	81
Figura 4.17. Convergencia de los perfiles de temperatura en cada plato utilizando el método Newton-Raphson .....	81
Fig. A.1 Primer paso: selección del método.....	105
Fig. A.2. Segundo paso: Selección del modelo termodinámico para la corriente de alimentación principal y para la resolución de la columna en cascada.....	106
Fig. A.3 Paso 3: Selección de la lista de componentes requeridos en la columna .....	107
Fig. A.4 Paso 4: Selección del componente Benceno presente en el ejemplo del Capítulo 4.....	107
Fig. A.5 Paso 5. Activar el botón "Añadir" para ingresar el Benceno a la lista de sustancias de la alimentación .....	108
Fig. A.6. Paso 6. Introducir una sustancia hipotética definida por el usuario .....	109
Fig. A.7. Paso 7. Introducir las propiedades críticas de una sustancia definida por el usuario...	109
Fig. A.8 Paso 8. Remover una sustancia de la lista de sustancias .....	110

Fig. A.9 Paso 9. Introducir los parámetros de interacción binaria requeridos por los pares de sustancias definidos .....	110
Fig. A.10. Paso 10: Introducir las variables intensivas que definan la fase de la corriente entre las variables resaltadas en la casilla correspondiente.....	111
Fig. A.11 Paso 11: Introducir flujos o fracciones molares en la tabla seleccionada y flujo total en la casilla "Mol totales" y/o normalizar .....	111
Fig. A.12 Paso 12. Introducir la etapa en la cual se ubica la corriente de alimentación y calcular la corriente .....	112
Fig. A.13 Paso 13: Revisar en el panel superior que haya sido añadida la corriente deseada exitosamente .....	113
Fig. A.14 Paso 14. Introducir el número de etapas teóricas de dato al método riguroso de la torre de destilación .....	113
Fig. A.15 Paso 15. Introducir las especificaciones de la torre de destilación .....	114
Fig. A.16. Paso 16. Estimar el balance de masa o calcular los resultados del algoritmo de Bubble Point.....	114

## LISTA DE SÍMBOLOS Y UNIDADES

### Latinos

$C$	Número de especies químicas [Adim.]
$D$	Flujo molar de destilado, [kg-mol/s]
$F$	Flujo molar de alimentación, [kg-mol/s]
$K$	Coeficiente de distribución en el equilibrio líquido-vapor, [Adim.]
$L$	Flujo molar de líquido, [kg-mol/s]
$N$	Número de etapas teóricas de equilibrio [Adim.]
$N_D$	Número de grados de libertad o variables independientes [Adim.]
$N_E$	Número de ecuaciones independientes [Adim.]
$N_{min}$	Número de etapas mínimas de equilibrio [Adim.]
$N_V$	Número de variables totales [Adim.]
$P$	Presión, [kPa]
$p^{sat}$	Presión de vapor, [kPa]
$R$	Relación de reflujo externa, $L/D$
$R_{min}$	Relación mínima de reflujo externa [Adim.]
$T$	Temperatura, [K]
$V$	Flujo molar de vapor, [kg-mol/s]
$W$	Flujo molar de residuo en el fondo, [kg-mol/s]
$a$	Parámetro de atracción de las ecuaciones de estado cúbicas [kPa·m <sup>-6</sup> ·kmol <sup>-2</sup> ]
$b$	Covolumen o parámetro de repulsión en ecuaciones de estado cúbicas [m <sup>3</sup> /mol]
$c$	Densidad en unidades molares, [kg-mol/m <sup>3</sup> ]
$d$	Flujo molar de destilado de un componente, [kg-mol/s]
$f$	Flujo molar de un componente en la alimentación, [kg-mol/s]
$l_i$	Flujo molar de líquido de un componente $i$ en el interior de la columna, [kg-mol/s]
$q$	Condición térmica de la alimentación [Adim.]
$m$	Parámetro de la función de cohesión [Adim.]
$v_i$	Flujo molar de vapor un componente $i$ en el interior de la columna, [kg-mol/s]
$w$	Flujo molar de residuo de un componente en el fondo, [kg-mol/s]
$x$	Fracción molar en el líquido [Adim.]

$y$	Fracción molar en el vapor [Adim.]
$z$	Fracción molar en la alimentación [Adim.]

## Griegos

$\alpha$	Volatilidad relativa [Adim.]
$\rho$	Densidad, [kg/m <sup>3</sup> ]
$\kappa$	Raíz de la composición de líquido en el desarrollo de Smoker, $0 < x < 1$ [Adim.]
$\Theta$	Parámetro en la ecuación de Underwood [Adim.]
$\Psi$	Factor en la correlación de Gilliland. [Adim.]
$\psi$	Constante de las ecuaciones de estado cúbicas
$\Omega$	Constante de las ecuaciones de estado cúbicas
$\omega$	Factor acéntrico de Pitzer [Adim.]

## Subíndices y Superíndices

$a$	Variable asociada al parámetro de atracción de ecuaciones de estado cúbicas
$c$	Propiedad crítica
$hk$	Componente clave pesado
$i$	Índice del componente $i$
$j$	Índice del componente $j$
$k$	Índice de la etapa teórica
$F$	Referente a la alimentación o de la etapa de alimentación
$L$	Líquido
$lk$	Componente clave liviano
$r$	Propiedad reducida
$V$	Vapor

- \* Composición de equilibrio
- Desplazamiento del eje de las abscisas en el desarrollo de Smoker

## Abreviaturas

<i>DIPPR</i>	Design Institute for Physical Properties (Instituto de Diseño de Propiedades Físicas)
<i>EdE</i>	Ecuación de estado
<i>FUG</i>	Método corto de cálculo de destilación compuesto por ecuaciones de Fenske, Underwood y Gilliland.
<i>HK, LK</i>	Compuesto clave pesado y clave liviano, respectivamente
<i>MESH</i>	Ecuaciones de balances de materia ( <i>M</i> ), condición de equilibrio de fases ( <i>E</i> ), restricción de suma de fracciones molares ( <i>S</i> ) y balance de energía ( <i>H</i> ).
<i>PR</i>	EdE cúbica de Peng y Robinson (1976)
<i>PRG</i>	EdE cúbica de Gasem (2001)
<i>RK</i>	EdE cúbica de Redlich Kwong (1949)
<i>SRK</i>	EdE cúbica de Soave (1972)
<i>VdW</i>	EdE cúbica de Van der Waals

## INTRODUCCIÓN

El petróleo ha sido una de las más importantes fuentes de energía y su gran versatilidad lo llevó en el último siglo a ser utilizado en la manufactura de una amplia gama de productos que contienen una mezcla de múltiples sustancias como disolventes, lubricantes, gasolina, gasóleo, nafta o queroseno. Un objetivo fundamental del procesamiento industrial del petróleo radica en su separación en otras mezclas y compuestos que constituyen los productos de valor comercial. El proceso de separación industrial utilizado para el fraccionamiento de petróleo es la destilación (Navarro, 2003).

La destilación es uno de los procesos de separación más antiguos y que tiene las más variadas aplicaciones a nivel industrial, desde la industria alimentaria hasta su uso mundial en la producción de los ya mencionados derivados del petróleo. En líneas generales, la destilación es una operación unitaria que permite la separación de una mezcla aprovechando la diferencia en volatilidades de sus componentes, para lo cual se promueve un equilibrio de fases entre un líquido y su vapor que, en muchas condiciones, forman mezclas de composiciones diferentes (Treybal, 1982).

El proceso de separación por destilación consiste en promover la transferencia de materia entre corrientes de fases líquido y vapor, tal que ocurra el enriquecimiento de los componentes más volátiles en los vapores y sean arrastradas las sustancias pesadas por el líquido, utilizando como agente de separación al calor suministrado. El contacto físico entre una fase líquida y una fase vapor es llamado etapa de equilibrio y cuando se promueve el contacto entre muchas de ellas en un arreglo en cascada el equipo que conforman es llamado una columna de fraccionamiento. La preferencia del proceso de destilación consiste en la versatilidad con que se puede tratar y separar mezclas multicomponentes, sin más limitaciones en la pureza alcanzable que las que sean económica y físicamente convenientes (Doherty *et al.*, 2008).

Los primeros cálculos para el diseño de destiladores consistieron en deducciones de los balances de equilibrio como equipos ideales en conjunto con aproximaciones o reglas empíricas para compensar en exceso el número de platos requeridos en una columna para cumplir con una separación similar o superior a la deseada. Fue así como se introdujo el uso de métodos gráficos y de métodos de cálculo cortos que aproximan el desempeño real si se opera bajo condiciones que cumplan con las suposiciones utilizadas.



El álgebra computacional permitió el uso de los métodos rigurosos, que incrementan la cantidad de ecuaciones que pueden ser resueltas, por lo que el conjunto completo de ecuaciones de balances de masa (M), condición de equilibrio (E), restricciones de suma (S) de fracciones molares y entalpía (H), llamadas MESH, por sus siglas en inglés, para todos los platos teóricos pueden ser resueltos simultáneamente. Una mayor precisión en el cálculo de la dinámica interna real de la separación permite obtener diseños más fidedignos con un consecuente ahorro de costos de inversión, debidos a la compensación por la incertidumbre de los métodos cortos (Navarro, 2003).

Entre las características de la mayor parte de los simuladores comerciales se han integrado en estándares internacionales desarrollados con base en la metodología de la programación orientada a objetos, la cual provee grandes ventajas con respecto a la estructura funcional de programación. Los algoritmos organizados en objetos permiten crear niveles de abstracción en estructuras independientes, de desarrollo encapsulado, las cuales son mucho más fáciles de probar, modificar, mantener y entre los propósitos más importantes está el de reusar, lo que permite la implementación continua de mejoras y modificaciones en vez de la reconstrucción de todos los fundamentos al momento de realizar una modificación (Leonard, 2005).

La mayor parte de la demanda de la industria por la utilización de métodos rigurosos, para el diseño de torres de destilación, es cubierta por unos pocos simuladores comerciales. Esto acarrea que el costo de las licencias de uso sea elevado y que se encuentre restringido el acceso del usuario a la lógica interna del programa para la manipulación, adaptación o mejora de los algoritmos utilizados (Navarro, 2003).

Dado que la información utilizada por un simulador consiste en modelos de predicción del equilibrio de fase, disponibles en la literatura y algoritmos de convergencia, la implementación de módulos de simulación computacional de procesos de destilación por métodos cortos y rigurosos es un área hacia la cual se puede realizar un trabajo con la intención de proveer herramientas para la validación rápida de los proyectos en sus etapas iniciales de ingeniería básica.

Entre los trabajos que se han realizado a nivel internacional en el área de la programación de módulos de simulación de métodos termodinámicos y métodos para resolver operaciones de fraccionamiento, se destacan los siguientes:

- Boston y Britt (1978), Boston (1980) y Russel (1983) desarrollaron un sistema de resolución de torres de destilación mediante un acercamiento dentro-fuera (*Inside-Out*) en el cual un conjunto de modelos termodinámicos aproximados son calculados en una

iteración externa con el uso de las correlaciones complejas, entonces se utilizan los modelos simples en una iteración interna para resolver con ellos los cálculos de equilibrio de fases y simplificar el proceso iterativo de corrección simultanea (con cualquier método riguroso, pero usualmente el de Newton Raphson de Naphtali y Sandholm), de manera que no se tenga el costo computacional de eliminar y recalcular en cada iteración los modelos termodinámicos complejos.

- Skogestad y Moriari (1987) realizaron un estudio en métodos de simulación y control de torres de destilación y generalizaron varias correlaciones de cálculo para el número de etapas teóricas y la localización de la alimentación, con lo que mejoraron sus estimaciones, así como también extendieron los métodos cortos a sistemas multicomponentes con el uso de su comparación a sistemas pseudobinarios.
- Carlberg y Westerberg (1988) realizaron una extensión del método corto de Underwood para sistemas de columnas complejas con torres auxiliares de enriquecimiento o de agotamiento.
- El conjunto de métodos *Inside-Out* ha sido mejorado para permitir el uso de columnas complejas con salidas laterales, múltiples alimentaciones, bombeos e intercambiadores interetapas, destilación extractiva o reactiva y sistemas de múltiples separadores, así como mejora de la velocidad de convergencia a través de factores de amortiguación (*damping-factor*). En ello han trabajado, entre otros, Simandl y Svrcek (1991), Parekh y Mathias (1998), Monroy-Loperena (2003), Li *et al.* (2006), Ivakpour y Kasiri (2008).
- Reyes (1998), en la Universidad de Alicante, realizó un estudio de los métodos de cálculo para diseño de columnas de separación de mezclas multicomponentes, tanto cortos como rigurosos, con el propósito de eliminar algunos de los cálculos iterativos necesarios, realizó una propuesta de métodos para obtener el reflujo mínimo y propuso un método de diseño una columna de destilación a partir de métodos de interpolación que permiten el cálculo aproximado del equilibrio líquido-vapor
- Bausa *et. al* (1998), Watzdorf *et al.* (2004) y Brüggermann y Marquardt (2004) presentaron una serie de artículos vinculados con la extensión de los métodos cortos clásicos de destilación y de algunos métodos cortos para casos especiales. Su desarrollo general de métodos cortos son aplicables para columnas complejas, así como también reducen las limitaciones para columnas simples.
- Tlacuatl (2004), en el Instituto Politécnico Nacional de México, desarrolló la

programación de un procedimiento de cálculo para la simulación semirigurosa por los métodos de Punto de Burbuja de Wang y Henke, sumas de flujos molares (sum-rates) y de *Inside-Out*, para la resolución de separaciones de mezclas altamente no ideales.

- Bandyopadhyay (2006) realizó la extensión del método corto de Smoker para el cálculo del número de etapas en destilación binaria para un caso general que no requiere el uso de la suposición de flujo molar interetapas constante, en cuyo caso se obtienen curvas de operación en el diagrama de McCabe y Thiele, en lugar de rectas.
- Martín (2011), Universidad de Valladolid, realizó la implementación de un módulo de código abierto para el cálculo de ecuaciones termodinámicas, basado en programación orientada a objetos en MATLAB®, que realiza el cálculo de algunas propiedades termodinámicas y de cálculos de equilibrio de fases sólido-líquido-vapor, con el objetivo de impulsar la enseñanza en la academia de los modelos termodinámicos
- Barboza y Díaz (2013), en la Universidad de Cartagena, realizaron un programa para la simulación por el método riguroso de Punto de Burbuja de Wang y Henke para el modelaje del comportamiento en régimen permanente y en estado transitorio dinámico de una torre de destilación real a escala piloto.

Ivakpour y Kasiri (2014) desarrollaron un método de inicialización de las variables de goteo de métodos rigurosos de destilación basándose en modificaciones del método de grupo de Kremser, que mejoran hasta en 30% los tiempos de convergencia de métodos rigurosos.

En la Universidad Simón Bolívar la implementación de algoritmos para la simulación de métodos cortos y semi-rigurosos tiene una importante cantidad de publicaciones y se ha venido estudiando a lo largo de las últimas décadas. Los siguientes trabajos son antecedentes del presente proyecto, los cuales consisten en programación de métodos de diseño de torres de destilación.

- En 1995 Del Nogal realizó la implementación en Visual Basic® de métodos cortos y semi-rigurosos por algoritmos iterativos para la estimación de  $N_{\min}$  y  $R_{\min}$  de destiladores multicomponentes. Se fundamentó en los métodos de Fenske, Shiras, Underwood y Lewis-Matheson.
- En 1999, Carreira y Rodríguez desarrollaron un programa que permite realizar el diseño y evaluación de columnas de destilación multicomponentes bajo ambiente Windows®.

- En 1999, Van der Dys programó la aplicación USBDistilBin, que resuelve problemas de destilación binaria por los métodos gráficos desarrollados por McCabe y Thiele y por Ponchon y Savarit
- En 2003 Navarro utilizó Visual Basic 6 para la resolución de columnas de destilación por el método riguroso *Inside-Out* en un módulo compatible con estándares CAPE-OPEN. Se utilizaron los métodos cortos de Fenske, Underwood y Gilliland para generar semillas a utilizar por el método riguroso.
- En 2009 Asuaje realizó el diseño e implementación de un programa de simulación para evaluar columnas de destilación en Visual Basic® y Excel®. Fueron utilizados los métodos de McCabe y Thiele, Underwood riguroso y Hengstebeck.
- En 2011, Majzoub desarrolló en ambiente Matlab® los métodos gráficos de diseño de torres de destilación binaria McCabe y Thiele y Ponchon y Savarit.
- En 2012, Porras desarrolló en ambiente Matlab® los métodos cortos de Fenske, Underwood, Shiras y Gilliland

En su mayoría las aplicaciones fueron realizadas fundamentadas en lenguajes de programación dispares y el desarrollo realizado con base en el método de programación funcional, lo cual generó como consecuencia un estancamiento en el desarrollo de cada una de las herramientas y la incapacidad para su reutilización. Además muchos de los lenguajes de programación utilizados no se encuentran en el programa de estudios de la carrera de Ingeniería Química por lo cual no ha sido sencillo integrarlos como herramienta de apoyo pedagógico de Operaciones Unitarias para la enseñanza de los métodos de cálculo de destilación (Leonard, 2005).

Por lo tanto, se plantea en este proyecto la integración bajo el entorno de programación dado por Mathworks Matlab® de las herramientas de simulación y cálculo de métodos cortos y rigurosos de destilación y de los modelos termodinámicos de predicción del equilibrio de fase requeridos para su implementación, basándose en programación orientada a objetos (recientemente implementada por Matlab®). Se eligió por su presencia en el plan de estudios la carrera de Ingeniería Química para poder facilitar su uso como herramienta pedagógica y se establecieron los siguientes objetivos:

## Objetivo General

- Desarrollar un módulo de programa de simulación de una torre de destilación continua mediante el uso de métodos rigurosos de resolución simultánea del sistema acoplado de ecuaciones MESH, el cual se presente mediante una interfaz gráfica para el usuario.

## Objetivos específicos

- Programar un módulo de resolución de diferentes métodos aproximados de Destilación Multicomponente: Lewis-Matheson, Smoker, McCabe-Thiele, Hengstebeck.
- Programar el método riguroso de Punto de Burbuja (*BP*) a partir de las suposiciones iniciales estimadas utilizando el lenguaje de programación integrado de Matlab®.
- Programar un módulo de resolución del método riguroso *Newton Raphson* utilizando el lenguaje de programación integrado de Matlab®.
- Generar una interfaz gráfica para la orientación del usuario y la recopilación de los datos requeridos para los cálculos.
- Obtener perfiles de composición, flujos y temperaturas para cada uno de los platos de una columna de destilación diseñada
- Comparar los perfiles con los resultados obtenidos mediante la utilización del simulador SimSci Pro/II®.

Partiendo del programa realizado en este trabajo se puede establecer una base para el continuo desarrollo por el alumnado de herramientas de cálculo necesarias para la simulación de procesos industriales. Se puede trabajar en un conjunto de diferentes herramientas para desembocar en la implementación de un entorno de trabajo completo en simulación de procesos que permita la sustitución de los simuladores comerciales por herramientas generadas por la universidad, de código abierto y con las cuales se pueda brindar un servicio a la empresa venezolana en función de un medio potencial de autofinanciamiento.

El presente libro está organizado en las siguientes partes. El capítulo 1 contiene los fundamentos teóricos de la literatura que son base para la programación de métodos de simulación en destilación. El capítulo 2 presenta la metodología y la explicación de la implementación de los módulos de simulación en el presente proyecto. El capítulo 3 presenta la discusión de los resultados

y también dos obtenidos en el código programado entre sí y con respecto a los del programa comercial PRO/II<sup>®</sup>. A continuación se presentan las conclusiones y recomendaciones. En los anexos se presentan algunos detalles adicionales del modelo de cálculo y algunos de los algoritmos básicos de cálculo de la termodinámica.

# CAPÍTULO 1

## MARCO TEÓRICO

En este capítulo se introducirán las herramientas en el cálculo de predicciones de equilibrio mediante el uso de ecuaciones de estado cúbicas. Se procederán a explicar los fundamentos del proceso de destilación, posteriormente se mencionarán y explicarán los distintos métodos de cálculo que han sido desarrollados para la destilación, pasando por los métodos cortos y finalmente llegando hasta los métodos rigurosos de *Bubble Point* y de Newton Raphson.

### 2.1 Ecuaciones de estado para sustancias puras

Las ecuaciones de estado (*EdE*) son un modelo de representación de los estados de agregación de la materia mediante el uso de sus propiedades mensurables como la temperatura, la presión o el volumen. Las ecuaciones más utilizadas en ingeniería son las cúbicas, que derivan de la ecuación propuesta por Van der Waals, como resultado de una modificación de la ecuación de gases ideales. Entre las ecuaciones cúbicas se tienen las propuestas por Peng y Robinson (1976) y la de Soave (1972), que fueron modificaciones a la presentada por Redlich y Kwong (1949). La ecuación propuesta por van der Waals, ecuación (1), como modificación de la ecuación de gases ideales contempló la introducción de parámetros de cohesión entre las moléculas (*a*) y de repulsión (*b*, covolumen) (Figueira, 2005).

$$P = P_{repulsión} - P_{atracción} = \frac{RT}{v - b} - \frac{a}{v^2} \quad (1)$$

donde

*v*: volumen intensivo, [m<sup>3</sup>/kg]

*P*: Presión a que se encuentra sometida la sustancia o mezcla, [kPa]

*T*: Temperatura del sistema, [K]

*R*: Constante universal de los gases ideales, [kJ·kmol<sup>-1</sup>K<sup>-1</sup>]

*a*: parámetro asociado al término de atracción de las ecuaciones cúbicas

*b*: covolumen intensivo, [m<sup>3</sup>/kg]

Las parámetros  $a$  y  $b$  son evaluados con las propiedades críticas de las sustancias puras, además del factor acéntrico ( $\omega$ ), en ciertas ocasiones, o también puede depender de constantes específicas para cada sustancia.

La ecuación propuesta por van der Waals (VW) aunque es cualitativamente correcta no puede predecir con exactitud las propiedades de sustancias puras, por lo cual se diseñan otras 200 ecuaciones de estado antes de que fuera presentada la modificación realizada por Redlich y Kwong, Ecuación (2) (Figueira, 2005).

$$P = \frac{RT}{v-b} - \frac{a}{T^{1/2}v(v+b)} \quad (2)$$

Con la ecuación de Redlich Kwong (RK) se consiguió una aplicación industrial exitosa, lo que es el desencadenante de que fuese tomada como base para manipularla y estudiarla buscando conseguir explorar sus limitaciones y la versatilidad de su uso para la predicción de propiedades termodinámicas de las sustancias puras (Figueira, 2005).

En particular las modificaciones de Soave (1972) y de Peng y Robinson (1976) realizaron una modificación incluyendo un parámetro de cohesión  $m$  en función del factor acéntrico. Generalizando las diferencias en las varias ecuaciones se puede obtener una forma generalizada que engloba a todas las formas cúbicas en las Ecs. (3), (4) y (5) (Prausnitz *et al.*, 2001) (Smith *et al.*, 2005).

$$Z = \frac{v}{v-b} - \frac{\left( a_c \alpha_a(T) / RT \right) v}{\left( v^2 + \delta v + \varepsilon \right)} \quad (3)$$

$$a_c = \psi_a \frac{R^2 T_c^2}{P_c} \quad (4)$$

$$b = \Omega \frac{RT_c}{P_c} \quad (5)$$

donde:

$Z$ : Factor de compresibilidad, relación entre el volumen molar del gas ideal y el volumen real [Adim]

$a_c$ : Factor crítico de las ecuaciones cúbicas relacionado al parámetro de atracción



$\alpha_a$ : Correlación específica de cada *EdE* en función de parámetros asociados a la sustancia

El subíndice  $c$  representa una propiedad de una sustancia pura en el punto crítico. Las variables  $\varepsilon$ ,  $\delta$ ,  $\Omega$ ,  $\psi_a$  son parámetros de cada ecuación cúbica, cambios en los valores dados a estos parámetros así como a  $\alpha_a$  sirven para representar la gran mayoría de los distintos desarrollos de ecuaciones cúbicas de la literatura (Prausnitz *et al.*, 2001).

La representación de las ecuaciones cúbicas mediante una forma generalizada presenta la gran ventaja que se mostrará en el capítulo 3 de permitir la integración de las rutinas de cálculo para gran número de ecuaciones cúbicas en un objeto abstracto que realiza la resolución independientemente de los valores particulares para la *EdE*.

Las diferentes funciones de cohesión para las *EdE* cúbicas de *SRK*, *PR* y *PRG* se presentan en la Tabla 2.1 y los valores de las constantes asociadas en la Tabla 2.2.

## 2.2 Extensión de las *EdE* cúbicas a sistemas multicomponentes

El modelo de ecuaciones de estado cúbicas dado hasta ahora sólo considera el caso de las sustancias puras. Para que una *EdE* sea utilizada en la estimación de propiedades de mezclas, los parámetros del modelo deben ser determinados utilizando un conjunto de reglas de mezclado, siendo de estas la más común la regla de mezclado de van der Waals (Burman y Ström, 2011). Los parámetros  $b$  y  $a_c$  de la ecuación (3) que dependen de las propiedades de la sustancia pura ahora deben depender de las propiedades de cada una de las sustancias en la mezcla.

Las reglas de mezclado de van der Waals proponen que las interacciones en una mezcla entre moléculas sean cuadráticas en composición tanto para el término atractivo, ecuación (4), como para el covolumen, ecuación (5). Dichas reglas suelen presentarse introduciendo una modificación sencilla, como lo es el parámetro de interacción binaria  $k_{ij}$  para el cálculo del término de atracción (Figueira, 2010).

$$a_c = \sum_i^c \sum_j^c x_i x_j a_{ij}; \quad a_{ij} = \sqrt{a_i a_j} (1 - k_{ij}) \quad (6)$$

$$b = \sum_i^c \sum_j^c x_i x_j b_{ij}; \quad b_{ij} = \frac{b_i + b_j}{2} \quad (7)$$

donde:

$x_i, x_j$ : Fracción molar de los compuestos  $i, j$  en la mezcla. [Adim]

Tabla 2.1 Parámetros de las ecuaciones cúbicas de estado (Prausnitz *et al.*, 2005)

$\alpha\{T_r\}$	$\delta$	$\varepsilon$	$\psi_a$	$\Omega$	Referencia
$(1+(c_1+c_2\omega+c_3\omega^2)(1-(T_r)^{c_4}))^{c_5}$	$b$	0	0,42748	0.086640	Soave (1972)
$(1+(c_1+c_2\omega+c_3\omega^2)(1-(T_r)^{c_4}))^{c_5}$	$2b$	$-b^2$	0,457235	0,077796	Peng y Robinson (1976)
$\exp((c_1 + c_2 T_r)(1 - T_r^{c_3 + c_4 \omega + c_5 \omega^2}))$	$2b$	$-b^2$	0,457235	0,077796	Gasem (2001)

Tabla 2.2 Constantes asociadas a la función de cohesión de las EdE cúbicas (Prausnitz *et al.*, 2005)

Ecuación	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$
SRK	0,48	1,574	-0,176	1/2	2
PR	0,374640	1,54230	-0,26992	1/2	2
PRG	2,00	0,836	0,134	0,508	-0,0467

### 2.3 Cálculo de equilibrio de fases líquido-vapor a partir de las EdE cúbicas

Para realizar estimaciones de equilibrio de fases líquido vapor se requiere resolver la ecuación de estado mediante la aplicación del método de isofugacidad. Para que exista equilibrio de fases en una sustancia o mezcla, tanto las temperaturas, las presiones y las energías libres de Gibbs deben ser iguales en cada una de las fases. En particular para el equilibrio líquido-vapor se tiene la ecuación (8) (Smith *et al.*, 2005).

$$g_V\{T^V, P^V\} = g_L\{T^L, P^L\} \quad (8)$$

El criterio de igualdades de energías libres de Gibbs puede ser sustituido por el criterio de

igualdad de fugacidades, lo que se puede escribir como en la ecuación (9).

$$f_v = f_L \quad (9)$$

El coeficiente de fugacidad de un componente  $i$  en una mezcla, generalizado para cualquiera de las ecuaciones cúbicas presentadas anteriormente se calcula a partir de la ecuación (10) (Prausnitz, Poling y O'Connell, 2001) (Figueira, 2005).

$$\ln \left( \frac{f_i}{y_i P} \right) = \frac{b_i}{b} (Z - 1) - \ln \left( Z \left( 1 - \frac{b}{v} \right) \right) + \frac{a_c}{b} \frac{RT}{RT} \left( 2 \frac{\sum_i^C x_i a_{ij}}{a_c} - \frac{b_i}{b} \right) I_Z \Big|_{\infty}^Z \quad (10)$$

donde:

$f_i$ : Fugacidad del componente  $i$  en la mezcla líquida, [kPa]

$C$ : Número de componentes en la mezcla, [Adim]

Y la solución a la integral  $I_z$  se expresa a partir de la ecuación (11), con la cual se puede evaluar la integral de la ecuación (10) para cada una de las ecuaciones cúbicas planteadas en la planteadas en forma genérica como en la ecuación (3).

$$I_z = \begin{cases} \frac{1}{\sqrt{\Delta}} \ln \left( \frac{2Z + bP/RT \left( \frac{\delta}{b} - \sqrt{\Delta} \right)}{2Z + bP/RT \left( \frac{\delta}{b} + \sqrt{\Delta} \right)} \right), & \text{si } \Delta > 0 \\ - \frac{2bP/RT}{2Z + \delta P/RT} & \Delta = \left( \frac{\delta}{b} \right)^2 - 4 \left( \frac{\varepsilon}{b^2} \right) \\ \frac{2}{\sqrt{-\Delta}} \tan^{-1} \left( \frac{2Z + \delta P/RT}{bP/RT \sqrt{-\Delta}} \right), & \text{si } \Delta < 0 \end{cases} \quad (11)$$

## 2.4 Cálculos de entalpía y balances de energía

Las propiedades residuales permiten la estimación de la diferencia medida entre el comportamiento de una propiedad termodinámica a condiciones reales y su valor a la misma presión y temperatura si se comportara como un gas ideal. Para el caso de mezclas se puede realizar el mezclado como gas ideal y posteriormente corregir la entalpía residual con el uso de las

propiedades residuales de las ecuaciones cúbicas. En el caso de la ecuación cúbica generalizada presentada en la ecuación (3), la entalpía residual viene dada por la ecuación (12):

$$\frac{H^{gi} - H}{RT} = 1 - Z + \frac{(a_c + D)/RT}{b(\Delta)^{1/2}} \left( 2 \frac{\sum_i^c x_i a_{ij}}{a_c} - \frac{b_i}{b} \right) I_Z \Big|_{\infty}^Z; D = T \frac{d(a_c)}{dT} \quad (12)$$

donde:

$H^{gi}$ : Entalpía de la mezcla de sustancias puras si se comportaran como una mezcla de gases ideales, [kJ/kmol]

$H$ : Entalpía de la mezcla real, [kJ/kmol]

$\frac{H^{gi} - H}{RT}$ : Entalpía residual, diferencia entre la entalpía real y la del gas ideal, [kJ·kmol<sup>-1</sup>·K<sup>-1</sup>]

La integral  $I_Z$  se calcula como en la ecuación (11). Para el cálculo de las entalpías residuales se requiere tener una expresión para el calor específico a condición de gas ideal,  $C_P^0$ , para la cual se utilizó el ajuste de la ecuación (13) (Green y Perry, 2008)

$$C_P^0 = c_1 + c_2 \left[ \frac{c_3/T}{\sinh(c_3/T)} \right]^2 + c_4 \left[ \frac{c_5/T}{\cosh(c_5/T)} \right]^2 \quad (13)$$

donde:

$C_P^0$  es el calor específico a presión constante de una sustancia como gas ideal, [kJ/(kmol·K)]

$c_1, c_2, c_3, c_4$  y  $c_5$ : constantes de ajuste para cada sustancia, presentadas en la base de datos *DIPPR* (Green y Perry, 2008)

## 2.5 Destilación

La destilación es una de las operaciones de separación preferidas a nivel industrial para la purificación de mezclas que se encuentran en fase líquida o vapor. El principio de separación es la diferencia en volatilidades de las sustancias puras presentes en la mezcla y el agente de separación es el calor.

Al proveer calor a una mezcla compuesta por sustancias de distinta volatilidad estas tienden

a cambiar de fase formando vapores preferencialmente ricos en los compuestos que poseen menor punto de ebullición normal (más livianos) y se forma un residuo líquido compuesto mayoritariamente por sustancias de más elevado punto de ebullición normal (Treybal, 1982)

La destilación consiste en varias etapas: evaporación, condensación y recolección de destilado y residuo de fondo. Puede involucrar la separación de una mezcla compuesta por dos componentes, en cuyo caso se le denomina binaria; podría comprender tres o más sustancias conocidas, que se conoce con el nombre de destilación multicomponente, o puede separarse una mezcla de sustancias desconocidas, siendo destilación compleja (Kister, 1992).

## **2.6 Métodos de cálculo en destilación**

Los primeros intentos por modelar y predecir los resultados de una destilación son atribuibles a Lord Rayleigh en 1902, que consideró el caso de una mezcla binaria homogénea separada discontinuamente mediante una destilación simple (King, 1980). Posteriormente y a partir de la aplicación industrial de fraccionamiento petrolero, se concentró el interés en los métodos de destilación en donde la carga es alimentada de forma continua. En función de ello se desarrollaron diversos métodos de cálculo.

### **2.6.1 Métodos cortos o aproximados**

Debido a la naturaleza de equilibrio de fases del proceso de destilación se ven involucrados muchos fenómenos físicos y ecuaciones, que involucran cálculos complejos y repetitivos. Antes de la aparición de los algoritmos por computadora los métodos utilizados involucraban la estimación rápida del número de etapas y reflujo requeridos para el diseño del fraccionador en función de un requerimiento específico de separación. Estos métodos de cálculo constan principalmente de los métodos gráficos y otras estimaciones suponiendo condiciones de gases y solución ideales y volatilidades constantes (Towler y Sinnott, 2008).

#### **2.6.1.1 Método de Fenske, Underwood y Gilliland (FUG)**

Entre los diferentes métodos cortos para el diseño de torres de destilación desarrollados para mezclas multicomponentes, uno de los más conocidos es conformado por la compilación de los métodos desarrollados por Fenske , Underwood y Gilliland y es a menudo conocido como FUG

por las iniciales de cada uno. Este está conformado por las contribuciones de las ecuaciones de Fenske para el reflujo total, la ecuación de Underwood para calcular el reflujo mínimo y la correlación de Gilliland que relaciona el desempeño de torres reales con las condiciones de reflujo mínimo y reflujo total, de forma que proporciona un estimado del número de etapas reales para el tasa de reflujo real.

Aunque la mayor exactitud de los métodos cortos se consigue para las mezclas que mejor se aproximan a la idealidad, continúan siendo ampliamente utilizados para obtener estimados iniciales que permitan un primer valor semilla para ser afinado con otros métodos de cálculo de destilación rigurosos.

El algoritmo de resolución recomendado para el método corto de FUG, como se muestra en la Fig 2.1, comienza por la especificación de la alimentación, luego requiere la especificación de la distribución de los componentes claves liviano (LK) y pesado (HK), a partir de lo cual se estima la distribución del resto de los componentes no claves, se determina el perfil de presión de la columna y el tipo de condensador, se utiliza la presión de la columna para realizar la expansión flash adiabática de la corriente de alimentación con lo que se obtiene los coeficientes de distribución de equilibrio líquido-vapor (K) y mediante la ecuación de Fenske se calcula el número mínimo de etapas teóricas y la distribución a reflujo total de los componentes no claves, que de ser distintos a los supuestos originalmente se debe de proceder a repetir con los nuevos valores como estimados iniciales de distribución.

Luego con las ecuaciones de Underwood se puede calcular la tasa de reflujo mínimo ( $R_{\min}$ ) a infinito número de etapas de equilibrio. Se pudiese estimar nuevamente la distribución de los componentes no claves con la ecuación de Shiras (Shiras, 1950), aunque según explican en Seader *et al.* (2011) la distribución de los componentes en torres reales es más cercana a la estimada por reflujo total con la ecuación de Fenske que la estimada a reflujo mínimo.

El algoritmo continúa mediante el cálculo del número de etapas reales estipulando un reflujo mayor al  $R_{\min}$ , lo que es proporcionado por el ajuste dado por la ecuación de Gilliland. Luego se puede estimar la ubicación óptima de la alimentación mediante la ecuación desarrollada por Kirkbride

La ecuación de Fenske para reflujo total permite el cálculo del número de etapas mínimas para la separación de los componentes según la ecuación (14)

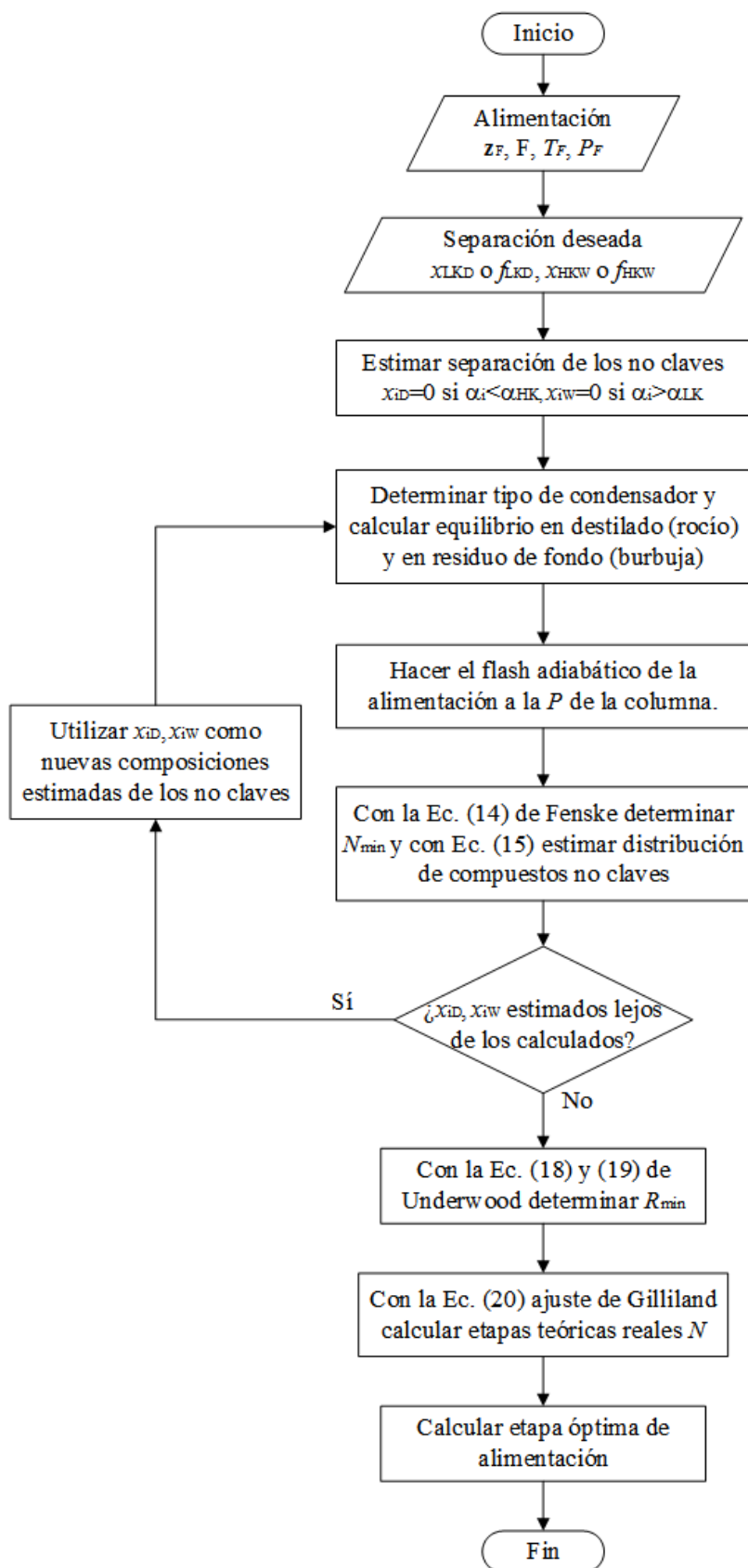


Figura 2.1 Algoritmo de cálculo para destilación multicomponente por el método corto de FUG  
(basado en Seader *et al.*, 2011)

$$N_{\min} = \frac{\log \left[ \left( \frac{Dx_{LK,D}}{Wx_{LK,W}} \right) \left( \frac{Wx_{HK,W}}{Dx_{HK,D}} \right) \right]}{\log(\alpha_{LK})} \quad (14)$$

donde:

$D$ : Flujo molar extraído en el tope de la columna como destilado, [kmol/s]

$W$ : Flujo molar extraído en el fondo de la columna como residuo, [kmol/s]

$_{HK}, _{LK}$ : Se refiere al componente clave pesado o clave liviano de la separación, respectivamente

$N_{\min}$ : Número de etapas mínimas requeridas para la separación especificada, [Adim]

$\alpha_{LK}$ : es la volatilidad relativa del compuesto clave liviano (LK), siendo esta la relación entre los coeficientes de reparto,  $K_i$ , es decir, el coeficiente de reparto es  $K_i = (y_i/x_i)$  y luego la volatilidad relativa de la sustancia es, tomada como es usual en referencia al compuesto clave pesado (HK),  $\alpha_i = (K_i/K_{HK})$ .

Como se muestra en la Fig. 2.1 el cálculo del número de etapas mínimas con la ecuación (14) se realiza sobre la base de una estimación previa de la distribución de los compuestos no claves. La ecuación de Fenske puede luego ser reformulada como en la ecuación (15) para dar con una corrección de la estimación inicial para la distribución de todos los componentes  $i$  no claves en la mezcla

$$\left( \frac{x_i}{x_{HK}} \right)_D = (\alpha_i)^{N_{\min}} \left( \frac{x_i}{x_{HK}} \right)_W \quad (15)$$

El valor de la relatividad relativa que debe usarse en las Ecs. (14) y (15) es el promedio geométrico de la volatilidad relativa en cada una de las etapas de la columna, sin embargo, debido a que el método corto FUG suele ser utilizado como estimado inicial y carece de la precisión de los métodos rigurosos basados en la resolución simultánea de todos los balances de masa, energía y equilibrio, suele considerarse suficiente para los propósitos del método FUG con utilizar el promedio geométrico de las volatilidades relativas de las etapas de tope y fondo:

$$\alpha_i = \bar{\alpha}_i = (\alpha_{i,tope} \cdot \alpha_{i,fondo})^{1/2} \quad (16)$$

Como explica Shiras (1950), en el proceso de destilación multicomponente existen dos casos principales de destilación a reflujo mínimo, que se clasifican por: De tipo 1 son aquellas



separaciones en donde todos los componentes se encuentran completamente distribuidos y aparecen tanto al tope como al fondo de la columna. De tipo 2 son aquellas separaciones en donde algunos de los componentes se encuentran únicamente presentes en uno de los dos productos.

Para separaciones del tipo 1 en donde todos los componentes se encuentran distribuidos en los productos de tope y de fondo se puede utilizar la ecuación de Shiras (1950), representada por la ecuación (17)

$$\frac{D(x_i)_D}{L_F(x_i)_F} = \frac{(\alpha_i - 1)}{\alpha_{LK} - 1} \cdot \frac{D(x_{LK})_D}{L_F(x_{LK})_F} + \frac{(\alpha_{LK} - \alpha_i)}{\alpha_{LK} - 1} \cdot \frac{D(x_{HK})_D}{L_F(x_{HK})_F} \quad (17)$$

Underwood diseñó un método en el cual se puede resolver el reflujo mínimo,  $R_{min}$ , para separaciones de tipo 2 (Shiras, 1950). El sistema de ecuaciones a resolver consiste en hallar raíces  $\Theta$  que satisfagan la ecuación (18), obteniendo una raíz menos que el número de compuestos que distribuyen a ambos productos de tope y fondo.

$$\sum_i \frac{\alpha_i z_{iF}}{\alpha_i - \Theta} = 1 - q \quad (18)$$

donde:

$$q = \frac{H_V - H_F}{H_V - H_L}: \text{condición térmica de la alimentación, [Adim].}$$

$z_{i,F}$ : Fracción molar de la sustancia  $i$  en la mezcla de alimentación, [Adim]

Para completar el cálculo de  $R_{min}$  con la ecuación de Underwood se utilizan los valores de  $\Theta$  hallados con la ecuación (18) en el sistema de ecuaciones (19), las cuales se resuelven todas simultáneamente para hallar las variables de  $R_{min}$ , la corrección del flujo de destilado  $D$  y la distribución de los componentes  $i$  no claves en el destilado, es decir  $x_i$ .

$$\left\{ \begin{array}{l} D(R_{min} + 1) = \sum_i \frac{\alpha_i (Dx_i)}{\alpha_i - \Theta} \text{ se escribe una vez por cada } \Theta \\ D = \sum_i di \end{array} \right. \quad (19)$$

Una vez obtenidos el número de etapas mínimas y el reflujo mínimo, el siguiente paso es la estimación del número de etapas teóricas para a determinado valor del reflujo real. Gilliland realizó un ajuste gráfico entre los datos de reflujo reales y la cantidad de etapas teóricas. Posteriormente Molokanov *et. al.* (1971) desarrollaron la ecuación (20) que representa los cálculos y relaciona el número de etapas con el reflujo:

$$\frac{N - N_{\min}}{N + 1} = 1 - \exp \left[ \left( \frac{1 + 54,4X}{11 + 117,2X} \right) \left( \frac{X - 1}{X^{0,5}} \right) \right] \quad (20)$$

### 2.6.1.2 Método gráfico de McCabe y Thiele

Entre los métodos de cálculo de una columna de destilación se encuentran los métodos consistentes en el trazado gráfico de etapas en diagramas que representan el equilibrio de fases del sistema, por ejemplo mediante el uso del diagrama  $x$ - $y$ , como es el método de McCabe y Thiele. Fue diseñado cuando sus autores diseñaron un ingenioso modelo gráfico para representar las aproximaciones al problema planteadas por Lewis (Mathias, 2009). Debido a esto presenta un conjunto de limitaciones, que se derivan de las suposiciones implícitas en su uso:

- La mezcla a separar es binaria
- El perfil de flujo molar líquido y vapor se mantiene constante
- La caída de presión a lo largo de la torre es despreciable y el proceso es isobárico
- No existen pérdidas significativas de calor al medio, existe aislamiento térmico o los calores de solución son grandes
- Los calores de vaporización son constantes e independientes de la composición del líquido.
- Las fases de líquido y vapor se comportan como solución ideal y como gas ideal, respectivamente, de manera que no exista efecto en la presión debida a la fugacidad del líquido (que se cumple a presiones de destilación bajas e introduce errores al utilizar presiones elevadas).

Adicionalmente a las suposiciones intrínsecas, una buena práctica en el uso como herramienta del diagrama de McCabe y Thiele implica la resolución de columnas que requieran un número de platos teóricos no muy elevado (<25), debido a que los calores del condensador y rehervidor son sensibles a la selección apropiada de la etapa óptima (Asuaje, 2009)

En la Fig. 2.2 se muestra el esquema de una columna de destilación simple, compuesta de una alimentación y dos productos de salida, en el tope  $D$  y en el fondo  $B$ . En ella se pueden identificar dos zonas distintas, que coinciden en el plato de alimentación (F): existe una zona de rectificación, correspondiente a las etapas 1 a F-1 y una zona de agotamiento entre las etapas F+1 a M.

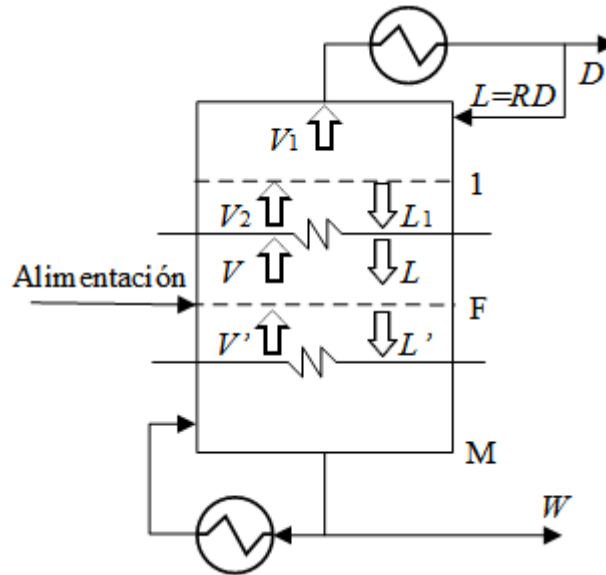


Figura 2.2 Esquema general de una columna de fraccionamiento

El balance de masa en el condensador 1 se muestra en la ecuación (21):

$$V_1 = D + L \quad (21)$$

Dado que el método de McCabe y Thiele supone que el perfil de flujos molares se mantiene constante, el modelo general presentado en la Fig. 2.2 se simplifica para la zona de rectificación como en las ecuaciones (22):

$$V_1 = V_2 = V \quad (22)$$

$$L_1 = L_2 = L$$

La suposición de flujos molares constantes aplica similarmente para los flujos molares de la zona de agotamiento, manteniéndose constantes en la etapa de agotamiento  $V'$  y  $L'$ .

Para el componente más volátil el balance de masa desde el plato 1 hasta un plato cualquiera  $k$  perteneciente a la zona de rectificación queda expresado por la ecuación (23)

$$y_{k+1}V = x_D \cdot D + x_k \cdot L \quad (23)$$

donde:  $V, L$ : flujo de vapor y líquido, respectivamente, en la zona de rectificación

$y_{k+1}$ : composición del compuesto volátil en el vapor en la etapa  $k+1$

$x_D$ : composición en la salida del destilado del componente volátil.

$x_k$ : fracción molar en el líquido de la etapa  $k$  del compuesto volátil

Asimismo, el balance de masa de la zona de agotamiento para la etapa  $k'$  se muestra en las ecuaciones (24) y (25):

$$L' = W + V' \quad (24)$$

$$x'_{k'-1} \cdot L' = x_W \cdot W + y'_{k'} \cdot V' \quad (25)$$

donde:  $V', L'$ : flujo de vapor y líquido, respectivamente, en la zona de despojo

$D$ : residuo, corriente de extracción del fondo de la columna

$x_{k'-1}$ : fracción molar en el líquido de la etapa  $k'$  del compuesto volátil

$y_{k'}$ : composición del compuesto volátil en el vapor en la etapa  $k'$

Al considerar constantes los flujos molares, las ecuaciones (23) y (25) representan unas líneas rectas en el diagrama  $y$ - $x$ , las cuales son llamadas rectas de operación y con ellas se obtienen plato a plato las composiciones de las fases vapor y líquido con las del plato anterior, reescribiendo las expresiones como se indica en las ecuaciones (26) y (27):

$$y_{k+1} = \frac{L}{V} \cdot x_k + \frac{D}{V} \cdot x_D \quad (26)$$

$$y'_{k'+1} = \frac{L'}{V'} \cdot x'_{k'} - \frac{W}{V'} \cdot x'_W \quad (27)$$

Para la etapa de rectificación se cumple que  $V=(L+D)=D(R+1)$ , siendo  $R$  la relación de reflujo en el tope de la columna. Entonces la ecuación (26) se puede escribir como se indica en la ecuación (28)

$$y_{k+1} = \frac{R}{R+1} \cdot x_k + \frac{1}{R+1} \cdot x_D \quad (28)$$

Con la ecuación (28) se puede resolver la composición de todos los platos y trazar la recta de operación hasta la etapa en la cual se defina el plato de alimentación. Allí los flujos de vapor y de líquido cambian con respecto a la condición térmica de la alimentación, representada en la variable  $q$  (Seader *et al.*, 2011). La condición térmica de la alimentación,  $q$  en ecuación (29), puede tener varias implicaciones. Una alimentación en fase mezcla contribuirá con vapor ascendente y líquido descendente a las etapas  $f-1$  y  $f+1$ , respectivamente. Una alimentación que se encuentre en líquido comprimido a las condiciones del plato de alimentación hará experimentar al flujo de vapor  $V'$  una súbita condensación al ser expuesto a la corriente que no está en equilibrio, contribuyendo a incrementar el flujo de líquido hacia la etapa  $f+1$  en un factor mayor al flujo de líquido en la alimentación mientras que disminuye el flujo de vapor ascendente a  $f-1$ . Un flujo de alimentación en vapor sobrecalentado vaporiza parte del líquido proveniente del tope de la columna, suponiendo el caso recíproco al de líquido comprimido. Todas las implicaciones se encuentran sintetizadas en la definición de la condición térmica,  $q$ , en la ecuación (29) y también están sintetizadas en la Tabla 2.3.

$$q = \frac{L' - L}{F} = \frac{H_V - H_F}{H_V - H_L} \quad (29)$$

En la etapa de alimentación se puede realizar un balance de masa y se obtiene la ecuación (30)

$$y_F = \frac{q}{q-1} \cdot x_F - \frac{z_F}{q-1} \quad (30)$$

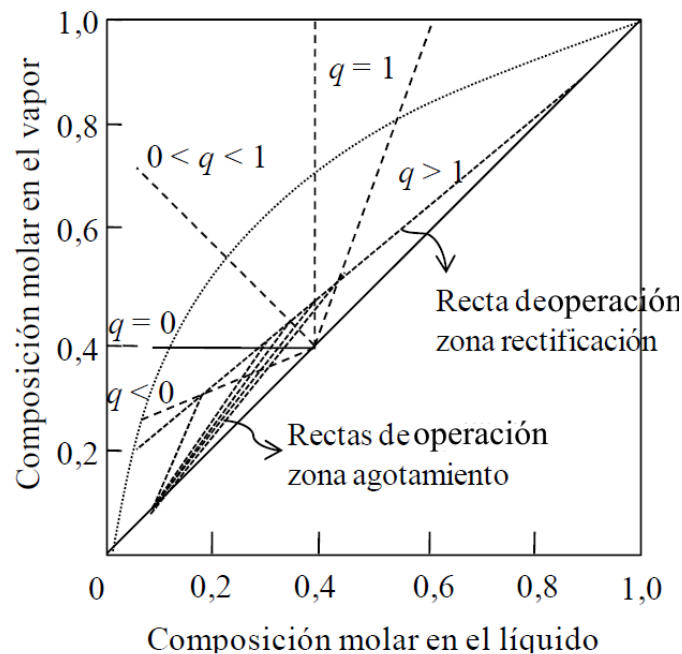
La relación  $L'/V'$  representa la pendiente de la zona de agotamiento, como muestra la ecuación (27). Dicha relación es calculada mediante la condición térmica de la alimentación y la recta de la zona de rectificación, gráficamente uno de sus puntos debe coincidir con la intersección entre ambas y se obtiene  $x_F$  al evaluar  $y_F$  de la ecuación (27) en la ecuación (30), o como se ve en la Fig. 2.3.

En el método de McCabe y Thiele la pendiente de la recta de operación ( $L/V$ ) de rectificación es proporcional a la relación de reflujo  $R$ . En el caso en que se hace recircular a la torre casi todo el vapor que se obtiene de la etapa 1, es decir, en el caso límite de reflujo total, la recta de operación de la zona de rectificación en la Fig. 2.3 incrementa su pendiente alejándose de la curva de equilibrio y se superpone a la diagonal. Es en esta condición en que se calcula el número mínimo de etapas de la columna. Se muestra en la Fig. 2.4(a).

Tabla 2.3. Implicaciones de la condición térmica de la alimentación a los flujos molares

Valor	Condición térmica	Implicaciones
$q < 0$	Vapor Sobrecalentado	$V > V', L > L'$
$q = 0$	Vapor saturado	$V > V', L = L'$
$0 < q < 1$	Mezcla líquido-vapor	$V > V', L < L'$
$q = 1$	Líquido saturado	$V = V', L < L'$
$q > 1$	Líquido sub enfriado	$V < V', L < L'$

Otra condición ideal es la de reflujo mínimo, en la cual se obtiene un número infinito de platos. Al disminuir la relación de reflujo disminuye la pendiente  $L/V$  de la recta de operación de la zona de enriquecimiento, con lo cual la recta de operación de la sección de tope se aleja de la diagonal y se intersecta con la curva de equilibrio, por lo cual se produce un punto de infinitesimalidad en la separación que logran los platos trazados cerca de la alimentación. Esta condición de reflujo mínimo coincide con el tipo 1 de reflujo mínimo de Shiras, ecuación (17), vista en la sección anterior y se muestra en la Fig. 2.4(b).

Figura 2.3. Influencia de  $q$  en la recta de operación de la zona de agotamiento (Asuaje, 2009)

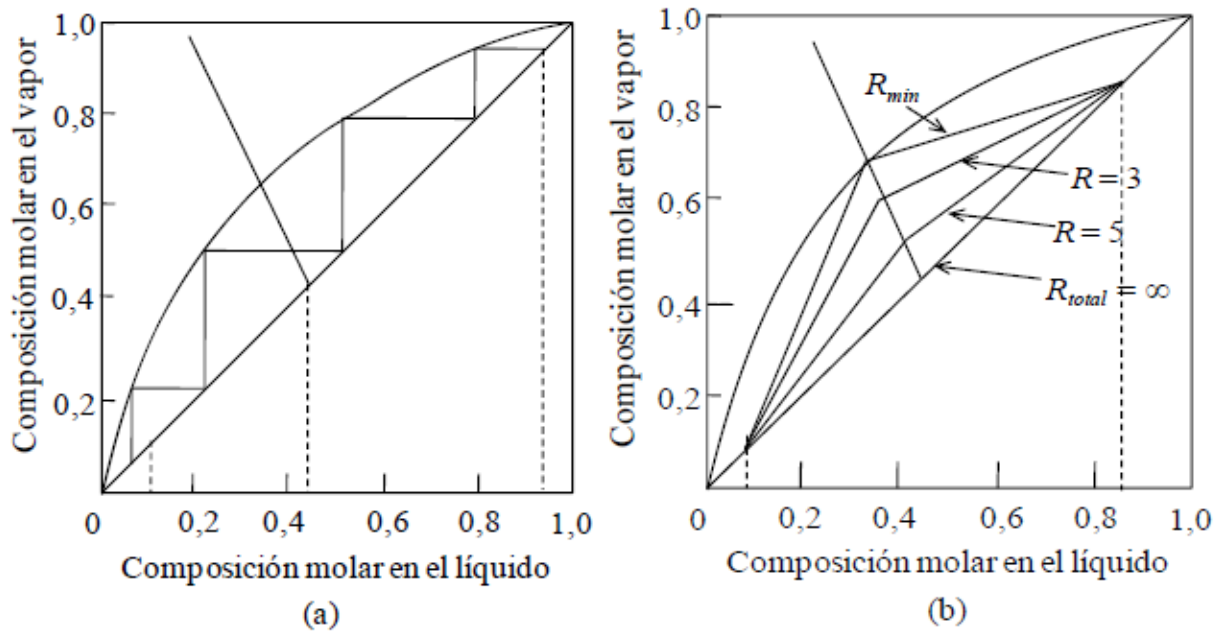


Figura 2.4. Representación en el diagrama y-x de los casos ideales de reflujo total (a) y de reflujo mínimo (b) (Asuaje, 2009)

### 2.6.1.3 Extensión de Hengstebeck del método de McCabe y Thiele

El método de Hengstebeck supone la validez de la extensión del método de McCabe y Thiele a sistemas de mezclas multicomponentes, tal que un par de compuestos claves define el nivel de separación de todos los componentes de la mezcla (Reyes, 1998).

Para un componente  $i$  los balances de masa de Lewis y Sorel, ecuaciones (31), (32), (33) y (34), pueden ser planteados en función de los flujos de los componentes individuales (Towler y Sinnott, 2008).

$$v_{k+1,i} = l_{k,i} + d_i \quad (31)$$

$$v_{k,i} = K_{k,i} \frac{V}{L} l_{k,i} \quad (32)$$

$$l'_{k+1,i} = v_{k',i} + b_i \quad (33)$$

$$v'_{k',i} = K_{k',i} \frac{V'}{L'} l'_{k',i} \quad (34)$$

donde:

$l_{k,i}$ : flujo de líquido del compuesto  $i$  en la etapa  $k$

$v_{k,i}$ : flujo de vapor del componente  $i$  en la etapa  $k$

$d_i$ : flujo del compuesto  $i$  en la salida de tope

$b_i$ : flujo del compuesto  $i$  en la salida de fondo

$K_{k,i}$ : el coeficiente de reparto de fase (equilibrio) del componente  $i$  en la etapa  $k$

El superíndice ' denota la sección de despojo

Al par de componentes claves se le aplica un trazado de etapas en un diagrama  $x$ - $y$  que solo los represente y contenga el equilibrio de una mezcla binaria entre ambos. Para excluir la influencia de los demás compuestos se procede normalizando las fracciones molares por unas del modelo binario equivalente. Dichos cálculos se realizan mediante las Ecs. (35), (36) (Hengstebeck, 1961)

$$x_{e,LK} = \frac{x_{LK}}{x_{LK} + x_{HK}} \quad (35)$$

$$y_{e,LK} = \frac{y_{LK}}{y_{LK} + y_{HK}} \quad (36)$$

Después del trazado de las etapas se supone que todos los componentes más livianos que el LK se encuentran exclusivamente en el producto de tope y todos los más pesados que el HK se encuentran exclusivamente en el producto de fondo. Sin embargo en las etapas intermedias se reescalan a multicomponente los flujos de líquido y de vapor haciendo el cálculo para los componentes no claves de sus respectivos flujos limitantes para las secciones de despojo o de enriquecimiento, como fue propuesto por Jenny 1939 (Towler y Sinnott, 2008)

$$\underline{l}_i = \frac{d_i}{\alpha_i - 1} \quad (37)$$

$$\underline{v}_i = l_i + d_i \quad (38)$$

$$\underline{v}'_i = \frac{\alpha_i b_i}{\alpha_{LK} - \alpha_i} \quad (39)$$



$$\underline{l}'_i = v'_i + b_i \quad (40)$$

donde:

$\underline{l}_i, \underline{v}_i$ : flujo limitante de líquido y vapor del compuesto  $i$  en la sección de rectificación para componentes más livianos que el clave liviano

$\underline{l}'_i, \underline{v}'_i$ : flujo limitante de líquido y vapor del compuesto  $i$  en la sección de despojo para componentes más pesados que el clave liviano

#### 2.6.1.4 Otros métodos aproximados

##### a) Ecuación de Smoker

Smoker (1938) dedujo una solución analítica a los balances de equilibrio bajo la suposición de que la volatilidad relativa permanece constante. Una utilidad primordial del método de Smoker es el cálculo rápido y aproximado del número de etapas requerido para la separación de una mezcla binaria compuesta por isómeros, cuya volatilidad relativa es muy cercana a 1, siendo impráctico utilizar un método de McCabe-Thiele puesto que no podría observarse con precisión el trazado gráfico de las etapas.

Una recta de operación es representada por la ecuación (41):

$$y = Ax + B \quad (41)$$

Como la volatilidad relativa se supone constante, todo el equilibrio es representado en términos de la volatilidad relativa, como se indica en las ecuaciones (42) y (43)

$$y = \frac{\alpha x}{1 + (\alpha - 1)x} \quad (42)$$

$$A(\alpha - 1)x^2 + [A + B(\alpha - 1) - \alpha]x + B = 0 \quad (43)$$

Resolviendo la ecuación (43) se obtienen 3 raíces, de las cuales la raíz  $\kappa$  es la que se sitúa entre 0 y 1, que se puede demostrar sólo habrá una raíz de esas características en destilación binaria (Towler y Sinnott, 2008).

Si se desplaza el eje de las abscisas  $x$  en la cantidad  $\kappa$ , formando una nueva ecuación de

segundo orden en la nueva variable  $x^\circ = x - \kappa$  y con la condición que pase por el origen del sistema de coordenadas (Smoker, 1938) se obtiene de una deducción que el número de etapas es, que puede ser calculado con la ecuación (44).

$$N_R, N_S = \frac{\ln \left( \frac{x^\circ_0 \left[ 1 - \frac{u(\alpha-1)A}{\alpha - u^2 A} \right] x^\circ_k}{x^\circ_k \left[ 1 - \frac{u(\alpha-1)A}{\alpha - u^2 A} \right] x^\circ_0} \right)}{\ln \left( \frac{\alpha}{u^2 A} \right)} \quad (44)$$

donde:

$N_R$ : Número de etapas de la sección de rectificación, desde el condensador a la alimentación

$N_S$ : Número de etapas de la sección de despojo,  $N_S = N - N_R$

$$u = [1 + (\alpha - 1) \kappa]$$

La ecuación (44) debe resolverse para cada sección de la columna, en particular si sólo se tiene una alimentación se resuelve una vez para la sección de rectificación y otra vez para la sección de despojo.

En la sección de rectificación las variables  $x^\circ_k$  y  $x^\circ_0$  y los coeficientes  $A$  y  $B$  de la recta son calculados a partir de las ecuaciones (45), (46), (47) y (48) (Towler y Sinnott, 2008):

$$x^\circ_0 = x_D - \kappa \quad (45)$$

$$x^\circ_k = z_F - \kappa \quad (46)$$

$$A = \frac{R}{R+1} \quad (47)$$

$$B = \frac{x_D}{R+1} \quad (48)$$

Y en la sección de despojo se usan las ecuaciones (49), (50), (51) y (52)

$$x^\circ_0 = z_F - \kappa \quad (49)$$

$$x_k^\circ = x_W - K \quad (50)$$

$$A = \frac{Rz_F^\circ + x_D - (R+1)x_W}{(R+1)(z_F^\circ - x_W)} \quad (51)$$

$$B = \frac{(z_F^\circ - x_D)x_W}{(R+1)(z_F^\circ - x_W)} \quad (52)$$

En todas las ecuaciones se utilizó  $z_F^\circ$  para denotar al punto de intersección de la ecuación (30), la recta de la condición térmica de alimentación, con la curva de equilibrio. Dicho punto será  $z_F$  para el caso de que la alimentación entre en líquido saturado ( $q=1$ ) o vendrá dado por la ecuación (53).

$$z_F^\circ = \frac{B + \frac{z_F}{q-1}}{\frac{q}{q-1} - A} \quad (53)$$

**b) Jafarey, Shinskey y Westerberg**

Dado que la ecuación (15), derivada del desarrollo hecho por Fenske, aplica únicamente a fluidos en reflujo total su uso es una idealización (Skogestad y Morari, 1987), pero un desarrollo similar puede presentarse para columnas con una relación de reflujo finita (Skogestad, 1997), que permita calcular el número de etapas teóricas ( $N$ ) para cada sección del fraccionador suponiendo que el número de etapas de la rectificación,  $N_R$  y el número de etapas del agotamiento,  $N_S$ , sean similares,  $N_R \approx N_S \approx N/2$ . El principio general a partir del cual se han encontrado correlaciones similares al desarrollo de Fenske es expresado por Skogestad (1987) como en la ecuación (54).

$$\frac{\left( \frac{x_{LK}}{x_{HK}} \right)_D}{\left( \frac{x_{LK}}{x_{HK}} \right)_W} = (\bar{\alpha}_i)^N \left[ \frac{\left( \frac{L}{V} \right)_{tope}}{\left( \frac{L}{V} \right)_{fondo}} \right]^{N/2} \quad (54)$$

Hacer una correlación aproximada entre la separación de la mezcla con respecto a los flujos de tope y fondo fue realizado antes por Jafarey (1979) que corresponde a la ecuación (55) si se alimenta en punto de burbuja y a la ecuación (56) en cualquier otro caso.

$$N = \frac{\ln \left[ \left( \frac{(x_{LK})_D}{1 - (x_{LK})_D} \right) \left( \frac{1 - (x_{LK})_W}{(x_{LK})_W} \right) \right]}{\ln \left[ \frac{(\bar{\alpha}_{LK})}{1 + \frac{1}{Rz_F}} \right]^{1/2}} \quad (55)$$

$$N = \frac{\ln \left[ \left( \frac{(x_{LK})_D}{1 - (x_{LK})_D} \right) \left( \frac{1 - (x_{LK})_W}{(x_{LK})_W} \right) \right]}{\ln \left[ \alpha^2 - \frac{\alpha_{LK}^2 (R + q_F)}{(R + 1)(Rz_F + q_F)} \right]^{1/2}} \quad (56)$$

Una correlación similar fue presentada por Westerberg, en las ecuaciones. (57) y (58).

$$N = \frac{12,29}{(\bar{\alpha}_{LK} - 1)^{0,693} \cdot (1 - \xi)^{0,158}} \quad (57)$$

$$R = \frac{1,38}{(\bar{\alpha}_{LK} - 1)^{0,717} \cdot (1 - \xi)^{0,109}} \quad (58)$$

donde:

$$\xi = \frac{D(x_{LK})_D}{F(z_{LK})_F}: \text{Recuperación fraccional del componente clave liviano}$$

Y otra correlación de la forma general formulada por Skogestad en la ecuación (54), es también la expresada por las ecuaciones (59), (60) y (61) por Shinskey

$$X = \frac{\alpha^{1,68}}{N^{0,32}} \left( \frac{V}{F} \right) \quad (59)$$

$$\text{Si } X < 1,825 \Rightarrow \frac{N_{\min}}{N} = \frac{\ln \left[ \frac{(x_{LK})_D}{1 - (x_{LK})_D} \cdot \frac{1 - (x_{LK})_W}{(x_{LK})_W} \right]}{N \ln(\alpha_{LK})} = 0,286X \quad (60)$$

$$\text{Si } X > 1,825 \Rightarrow \frac{N_{\min}}{N} = \frac{\ln \left[ \frac{(x_{LK})_D}{1 - (x_{LK})_D} \cdot \frac{1 - (x_{LK})_W}{(x_{LK})_W} \right]}{N \ln(\alpha_{LK})} = \frac{1}{\sqrt{\frac{8,35}{X^2} + 1,163}} \quad (61)$$

### 2.6.1.5 Métodos rigurosos

Se puede modelar cada etapa de equilibrio  $k$  de un fraccionador mediante un conjunto de ecuaciones que relacionan las variables que determinan el comportamiento del equipo, estas son las ecuaciones MESH. La componen el balance de materia (M), condición de equilibrio de fases (E), sumatorias de fracciones másicas o molares (S) y el balance de energía (H) que conforman la totalidad de las ecuaciones que pueden ser planteadas.

Para cualquier etapa de equilibrio teórica en una columna de separación con arreglo en cascada la misma puede ser modelada como se muestra en la Fig. 2.5

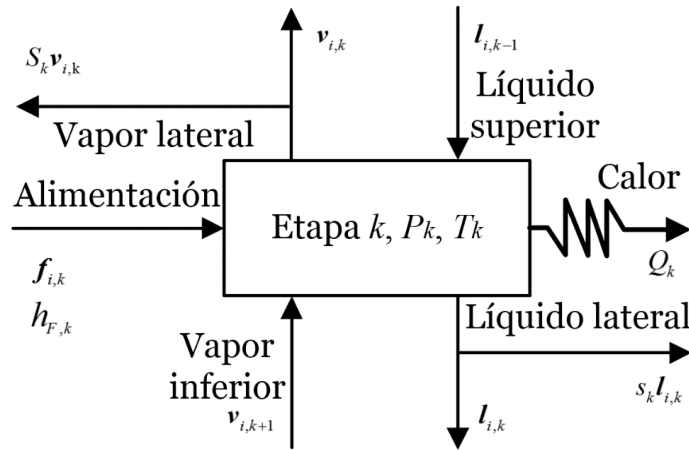


Figura 2.5. Representación esquemática de una etapa de equilibrio (Seader *et al.*, 2011)

Una etapa de equilibrio genérica acarrea como suposiciones las siguientes: 1) se alcanza equilibrio de fases líquido-vapor en la etapa. 2) No ocurren reacciones químicas que cambien el flujo o composición molar. 3) Es despreciable el efecto producido por el arrastre de gotas de líquido en la fase vapor ascendente o la oclusión de burbujas de vapor en la fase líquida descendente. Como se muestra en la Fig. 2.6 una columna de destilación puede verse como sucesión de etapas de equilibrio numeradas desde el tope al fondo.

El conjunto de ecuaciones MESH consiste de las ecuaciones (Seader *et al.*, 2011):\_

- **Ecuaciones M:** Balances de materia para cada componente  $i$  en cada etapa  $k$ , ecuación (62).

$$\mathbf{M}_{i,k} \Rightarrow l_{i,k} (1 + s_k) + v_{i,k} (1 + S_k) - l_{i,k-1} - v_{i,k+1} - f_{i,k} = 0 \quad (62)$$

- **Ecuaciones E:** Condiciones de equilibrio entre fases para cada componente, ecuación (63).

$$\mathbf{E}_{i,k} \Rightarrow K_{i,k} l_{i,k} \frac{\sum_j^C v_{j,k}}{\sum_j^C l_{j,k}} - v_{i,k} = 0 \quad (63)$$

- **Ecuaciones S:** Representa las sumatorias de fracciones molares, ecuaciones (64) y (65).

$$(\mathbf{S}_y)_k \Rightarrow \sum_{i=1}^C y_{i,k} - 1 = \sum_{i=1}^C \left( \frac{v_{i,k}}{\sum_{j=1}^C v_{j,k}} \right) - 1 = 0 \quad (64)$$

$$(\mathbf{S}_x)_k \Rightarrow \sum_{i=1}^C x_{i,k} - 1 = \sum_{i=1}^C \left( \frac{l_{i,k}}{\sum_{j=1}^C l_{j,k}} \right) - 1 = 0 \quad (65)$$

- **Ecuaciones H:** Relaciones de balance de energía para cada etapa genérica, ecuación (66).

$$\begin{aligned} \mathbf{H}_{i,k} &\Rightarrow h_{L,k} (1 + s_k) \sum_{j=1}^C l_{j,k} + h_{V,k} (1 + S_k) \sum_{j=1}^C v_{j,k} \\ &- h_{L,k-1} \sum_{j=1}^C l_{j,k-1} - h_{V,k+1} \sum_{j=1}^C v_{j,k+1} - h_{F,k} \sum_{j=1}^C f_{j,k} - Q_k = 0 \end{aligned} \quad (66)$$

donde:

$s_k$ : fracción del flujo de líquido que es extraída en la etapa  $k$ . [Adim]

$S_k$ : fracción del flujo de vapor que es extraída en la etapa  $k$ , [Adim]

$f_{i,k}$ : Flujo molar del compuesto  $i$  en la corriente de alimentación de la etapa  $k$ , [kmol/s]

$h_{L,k}, h_{V,k}$ : Entalpía intensiva de la fase líquida (L) o vapor (V) en la etapa  $k$ , [kJ/(kmol·s)]

Con las ecuaciones **MESH** se compone un sistema de ecuaciones de  $N(2C+1)$  variables dependientes cuando las variables  $N, f_{i,k}, s_k, S_k, Q_k$  (salvo  $Q_1, Q_N$ ) y alimentaciones especificadas. Con el sistema de ecuaciones se conforma una estructura matricial tridiagonal que representa la matriz jacobiana de diferenciales parciales, una estructura que es resuelta para cada componente  $i$  usualmente con un algoritmo de Thomas para llevar a cero o a un margen de tolerancia las ecuaciones **M**, **E**, **S** y **H**.

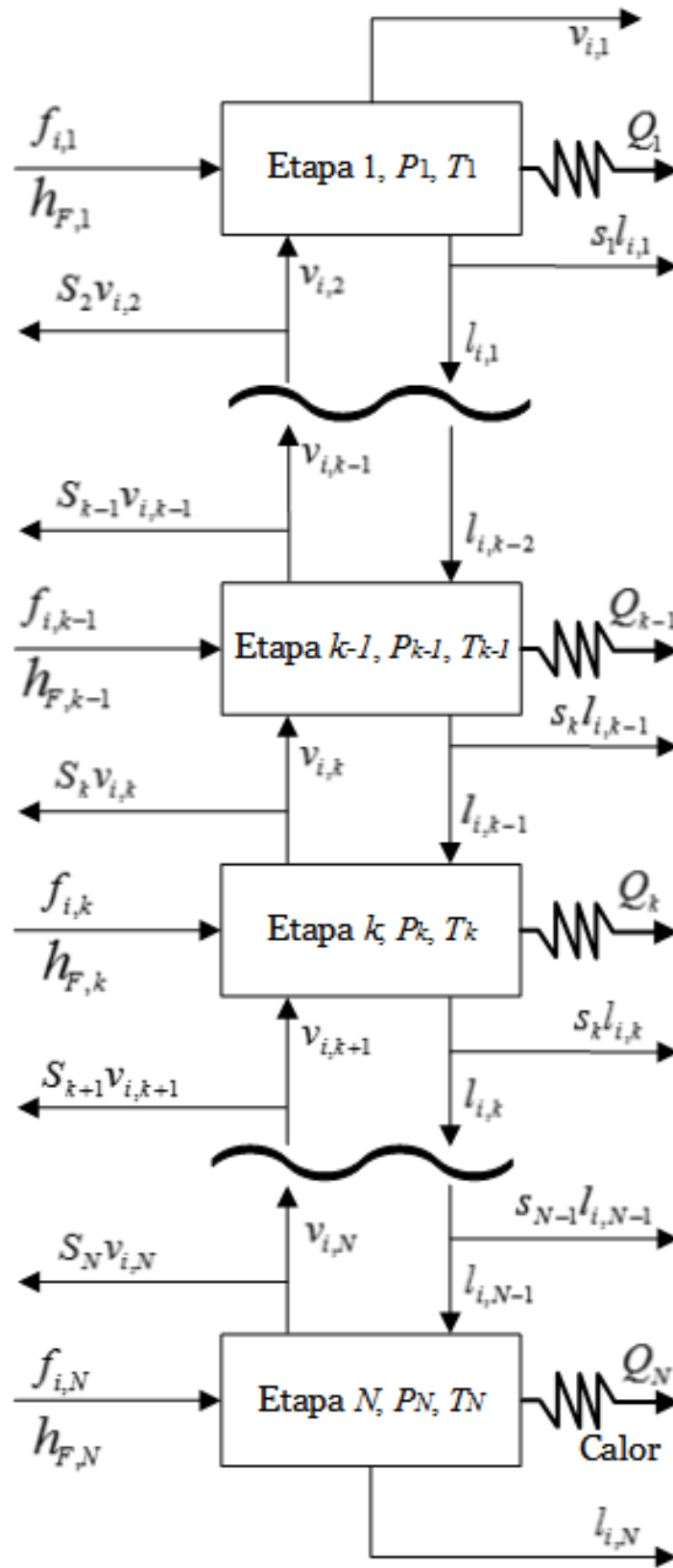


Figura 2.6. Representación esquemática de un arreglo en cascada de etapas teóricas, un fraccionador (basada en Seader *et al.*, 2011)

### 2.6.1.6.1 Método de Punto de Burbuja de Wang y Henke

El primer planteamiento de usar un sistema de ecuaciones MESH para solucionar las etapas de fraccionamiento en cascada fue propuesta en 1966, por Wang y Henke. Se propuso la utilización de un conjunto de variables supuestas como semilla, llamadas variables de goteo o lagrimeo, que deben ser ajustadas por cálculos iterativos. Las variables supuestas por Wang y Henke como semillas fueron temperaturas en todos los platos ( $T_k$ ) y los flujos de vapor de todos los platos ( $V_k$ ), las cuales se corrigen en cada iteración mediante el uso de las ecuaciones de punto de burbuja (BP). Los balances de masa  $\mathbf{M}$  son modificados de forma que sean lineales en las composiciones de líquido desconocidas, para lo cual las salidas líquidas dependientes de la composición deben ser especificadas en función de su flujo molar ( $U_k$ ) y no de su fracción de separación ( $s_k$ ). Con ello quedan planteadas las ecuaciones  $\mathbf{M}$  como ecuación (67),  $\mathbf{E}$  se convierte en ecuación (68) y  $\mathbf{H}$  en ecuación (69).

$$\mathbf{M}_{i,k} \Rightarrow L_{k-1}x_{i,k-1} + V_{k+1}y_{i,k+1} + F_k z_{i,k} - (L_k + U_k)x_{i,k} - (1 + S_k)V_k y_{i,k} = 0 \quad (67)$$

$$\mathbf{E}_{i,k} \Rightarrow y_{i,k} - K_{i,k}x_{i,k} = 0 \quad (68)$$

$$\mathbf{H}_k \Rightarrow L_{k-1}h_{L,k-1} + V_{k+1}h_{V,k+1} + F_k h_{F,k} - (L_k + U_k)h_{L,k} - (1 + S_k)V_k h_{k,j} - Q_k = 0 \quad (69)$$

donde:

$L_k$ : es el flujo molar de líquido en la etapa  $k$ , [kmol/s]

$U_k$ : es el flujo molar de salida de líquido en la etapa  $k$ , [kmol/s]

$V_k$ : es el flujo molar de vapor en la etapa  $k$ , [kmol/s]

$F_k$ : es el flujo molar total que ingresa alimentado en la etapa  $k$ , [kmol/s]

En sustitución de las ecuaciones  $\mathbf{S}$  se suele utilizar un balance de masa global, que es obtenido combinando las ecuaciones (64) y (65) junto a  $\sum_k z_{i,k}$  para todos los componentes entre sumados entre las etapas 1 hasta la  $k$  que da como resultado la ecuación (70):

$$L_k = V_{k+1} + \sum_{j=1}^k (F_j - U_j - S_j V_j) - V_1 \quad (70)$$



La modificación de las ecuaciones **M** de Wang y Henke consisten en la sustitución de (68) en (67) para eliminar dependencia con  $y$ , también de sustituir (70) en (67) para eliminar  $L$  (Seader *et al.*, 2011). De tal manera para cada componente  $i$  y etapa  $k$  queda la ecuación (71) que sintetiza los balances de masa:

$$A_k x_{i,k-1} + B_k x_{i,k} + C_k x_{i,k+1} = D_k \quad (71)$$

donde:

$$A_k = V_k + \sum_{j=1}^{k-1} (F_j - S_j V_j - U_j) - V_1, \quad 2 \leq k \leq N \quad (72)$$

$$B_k = - \left[ V_{k+1} + \sum_{j=1}^k (F_j - S_j V_j - U_j) - V_1 + U_k + (1 + S_k) V_k K_{i,k} \right], \quad 1 \leq k \leq N \quad (73)$$

$$C_k = V_{k+1} K_{i,k+1}, \quad 1 \leq k \leq N-1 \quad (74)$$

$$D_k = -F_k z_{i,k}, \quad 1 \leq k \leq N \quad (75)$$

Que como se indicó en la Fig. 2.6 tiene una única salida de líquido en el fondo,  $W$ , el flujo de residuo, lo que quiere decir  $U_N = W$  y de ser líquido el destilado también  $U_1 = D$ . Además  $V_{N+1} = 0$  a no ser que se trate de una columna alimentada por vapor vivo y  $L_0 = 0$ .

Si las ecuaciones **M** modificadas se agrupan por componente, se pueden escribir como una serie de  $C$  (una por cada componente) sistemas de ecuaciones organizadas en una matriz tridiagonal como se muestra en la ecuación (76).

$$\begin{bmatrix} B_1 & C_1 & 0 & \cdots & & \cdots & 0 \\ A_2 & B_2 & C_2 & 0 & \cdots & \cdots & 0 \\ 0 & A_3 & B_3 & \ddots & \ddots & & \vdots \\ \vdots & 0 & \ddots & \ddots & C_{k-1} & \ddots & \vdots \\ & \vdots & \ddots & A_k & B_k & \ddots & \vdots \\ & & & \ddots & \ddots & C_{N-2} & 0 \\ \vdots & \vdots & & \ddots & A_{N-1} & B_{N-1} & C_{N-1} \\ 0 & 0 & \cdots & \cdots & 0 & A_N & B_N \end{bmatrix}^{(m-1)} \times \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ x_{i,3} \\ \vdots \\ x_{i,k} \\ \vdots \\ x_{i,N-1} \\ x_{i,N} \end{bmatrix}^{(m)} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \vdots \\ D_k \\ \vdots \\ D_{N-1} \\ D_N \end{bmatrix} \quad (76)$$

donde;

$m$ : número de la iteración correspondiente al proceso de tanteo de las variables de goteo

La matriz tridiagonal debe resolverse para hallar las composiciones de líquido del componente  $i$  (vector  $\mathbf{x}_i^{(m)}$ , iteración  $m$ ) utilizando las suposiciones de las variables de lagrimeo  $T_k$  y  $V_k$ , así como también  $\mathbf{x}_i^{(m-1)}$ , de la iteración anterior, de ser necesario como estimado para el cálculo de las constantes de reparto,  $K_i^{(m-1)}$ , como es el caso de EdE cúbicas en el cual los coeficientes de reparto dependen de la composición.

En la Fig. 2.7 se presenta el algoritmo para el método de BP de Wang-Henke. Se debe disponer del número de etapas ( $N$ ), las condiciones térmicas o aislamiento de la columna ( $Q_2-Q_{N-1}$ ), la presión, o equivalentemente pérdida de carga para calcularla, de las etapas teóricas ( $P_1-P_N$ ), los datos completos de todas las alimentaciones y las especificaciones de separación deben ser flujo deseado en el destilado o el residuo ( $D$  o  $B$ ), reflujo ( $R$ ) o relación de reflujo ( $L_{\text{tope}}/D$ ) y además el flujo molar y fase de las salidas laterales.

El algoritmo de BP empieza calculando el valor del flujo de la salida sin especificar,  $D$  o  $B$  y, a continuación, se procede a realizar una estimación inicial de las variables de goteo para comenzar con el tanteo. Por lo regular es suficiente con establecer la suposición de flujos molares de vapor interetapas ( $V_k$ ) constantes, como la suposición intrínseca de Lewis usada en McCabe y Thiele (sección 2.6.1.2), para ello se deben usar los flujos de destilado, alimentación, reflujo y corrientes laterales especificados. Se requiere usualmente suponer una distribución de los compuestos en los productos de destilado y fondo para poder brindar el estimado rápido de temperaturas haciendo coincidir la temperatura en el tope ( $T_1$ ) con la obtenida por el punto de rocío del destilado y la de fondo ( $T_N$ ) con el punto de burbuja en el rehervidor. Al tener los estimados se puede suponer una variación de  $T_k$  lineal con la localización de la cascada de etapas teóricas.

Como en las ecuaciones cúbicas y algunos otros modelos de cálculo de equilibrio de fases las composiciones de líquido son parte fundamental de la estimación del coeficiente de equilibrio  $K_i$ , se propuso en el algoritmo la suposición de variación lineal también de las  $x_{i,k}$  partiendo de los cálculos de equilibrio de los productos con burbuja y rocío que brindan  $\mathbf{x}_{i,D}$ ,  $K_{i,D}$  y  $y_{i,W}$  y  $K_{i,W}$ . Los  $K_{ij}$  también se distribuyen cada etapa linealmente en función a los calculados del tope y fondo.

Con las variables de lagrimeo inicializadas se resuelve el sistema de ecuaciones en forma matricial, ecuación (76), para obtener un conjunto de  $x_{i,k}$  que sea consistente con las variables de lagrimeo  $T_k$ ,  $V_k$  y  $K_{i,k}$  que se tienen. Para resolver el sistema de ecuaciones se utiliza el algoritmo de Thomas, que posee ventajas de consumo de recursos y rapidez (Barboza y Díaz, 2013).

Con el conjunto consistente de variables  $T_k$ ,  $V_k$  y  $x_{i,k}$  se utiliza el cálculo de punto de burbuja

en cada etapa, con el cual también se corrigen para la iteración  $m$  actual las variables  $y_{i,k}^{(m)}$  y  $T_k^{(m)}$ . Partiendo del nuevo conjunto de variables calculados en la iteración actual  $m$  se puede calcular  $V_2$  usando (70). Entonces se calculan las entalpías intensivas molares de todos los platos para líquido y vapor ( $H_V$  y  $H_L$ ). Entonces se tiene todo lo necesario para utilizar (69) para calcular el calor extraído por el condensador. Luego para obtener el calor suministrado al rehervidor se usa la ecuación (77).

$$Q_N = \sum_{k=1}^N (F_k h_{F,k} - U_k h_{L,k} - S_k V_k h_{V,k}) - \sum_{k=1}^{N-1} (Q_k) - V_1 h_{V,1} - L_N h_{L,N} \quad (77)$$

Se calcula entonces un nuevo conjunto de variables de tanteo  $V_j$  mediante la aplicación de un balance de energía modificado, obtenido combinando las ecuaciones (69) y (70) con el propósito de eliminar la dependencia con el flujo de líquido, tal que se obtienen las ecuaciones (78), (79), (80) y (81) (Seader *et al*, 2008).

$$h\alpha_k V_k + h\beta_k V_{k+1} = \gamma_k \quad (78)$$

donde:

$$h\alpha_k = h_{L,k-1} - h_{V,k} \quad (79)$$

$$h\beta_k = h_{V,k+1} - h_{L,k} \quad (80)$$

$$\gamma_k = \left[ \sum_{j=1}^{k-1} [F_j - S_j V_j - U_j] - V_1 \right] (h_{L,k} - h_{L,1}) + F_k (h_{L,k} - h_{F,j}) + S_k V_k (h_{V,k} - h_{L,k}) + Q_k \quad (81)$$

Las entalpías se calculan a las temperaturas  $T_k^{(m)}$ , las calculadas con la condición de punto de burbuja utilizando el nuevo perfil consistente de composiciones de líquido de la torre. La ecuación (78) conforma entonces un sistema de ecuaciones que puede escribirse en forma diagonal como lo indica la ecuación (82).

$$\begin{bmatrix} h\beta_2 & 0 & \dots & & \dots & 0 \\ h\alpha_3 & h\beta_3 & 0 & \dots & \dots & 0 \\ 0 & h\alpha_4 & h\beta_4 & \ddots & & \vdots \\ \vdots & 0 & \ddots & \ddots & & \vdots \\ & \vdots & \ddots & h\alpha_k & h\beta_k & \\ & & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & h\alpha_{N-2} & h\beta_{N-2} & 0 \\ 0 & 0 & \dots & \dots & 0 & h\alpha_{N-1} & h\beta_{N-1} \end{bmatrix} \begin{bmatrix} V_3 \\ V_4 \\ V_5 \\ \vdots \\ V_k \\ \vdots \\ V_{N-1} \\ V_N \end{bmatrix} = \begin{bmatrix} \gamma_2 - h\alpha_2 V_2 \\ \gamma_3 \\ \gamma_4 \\ \vdots \\ \gamma_k \\ \vdots \\ \gamma_{N-2} \\ \gamma_{N-1} \end{bmatrix} \quad (82)$$

La resolución del sistema de ecuaciones se puede realizar por la fórmula genérica para la etapa  $k$ , como se expresa en la ecuación (83).

$$V_k = \frac{\gamma_{k-1} - h\alpha_{k-1}V_{k-1}}{h\beta_{k-1}} \quad (83)$$

Con todo ello se corrige hasta la etapa  $N-1$  los tanteos realizados en la iteración anterior a los flujos de vapor, así como se obtienen las semillas  $V_k$  que alimentan a la iteración subsiguiente, repitiendo cada paso descrito desde la resolución del sistema de ecuaciones en la ecuación (76). El proceso iterativo se termina cuando se cumple con el criterio de convergencia indicado en la ecuación (84).

$$\sum_{k=1}^N \left[ \frac{T_k^{(m)} - T_k^{(m-1)}}{T_k^{(m)}} \right]^2 + \sum_{k=1}^N \left[ \frac{V_k^{(m)} - V_k^{(m-1)}}{T_k^{(m)}} \right]^2 \leq \varepsilon \quad (84)$$

Otro criterio de convergencia mucho más simple fue presentado por Wang y Henke (1966) y se expresa en la ecuación (85).

$$\tau = \sum_{k=1}^N \left[ T_k^{(m)} - T_k^{(m-1)} \right]^2 \leq 0,01N \quad (85)$$

Los valores negativos obtenidos para los flujos de vapor y otras variables de goteo se sustituyen antes de iniciar las subsiguientes etapas por valores positivos cercanos a cero, de forma de evitar que las sucesivas correcciones agraven la discrepancia.

#### 2.6.1.6.2 Método de Newton Raphson

El algoritmo de BP sufre problemas de convergencia para sistemas que contienen mezclas altamente no ideales. Para estos casos se requiere la resolución de las ecuaciones MESH mediante técnicas de corrección simultánea. Uno de ellos fue el propuesto por Naphtali y Sandholm (1971) que agruparon las ecuaciones MESH en el conjunto de  $N(2C+1)$  ecuaciones, siendo las en la sección anterior.

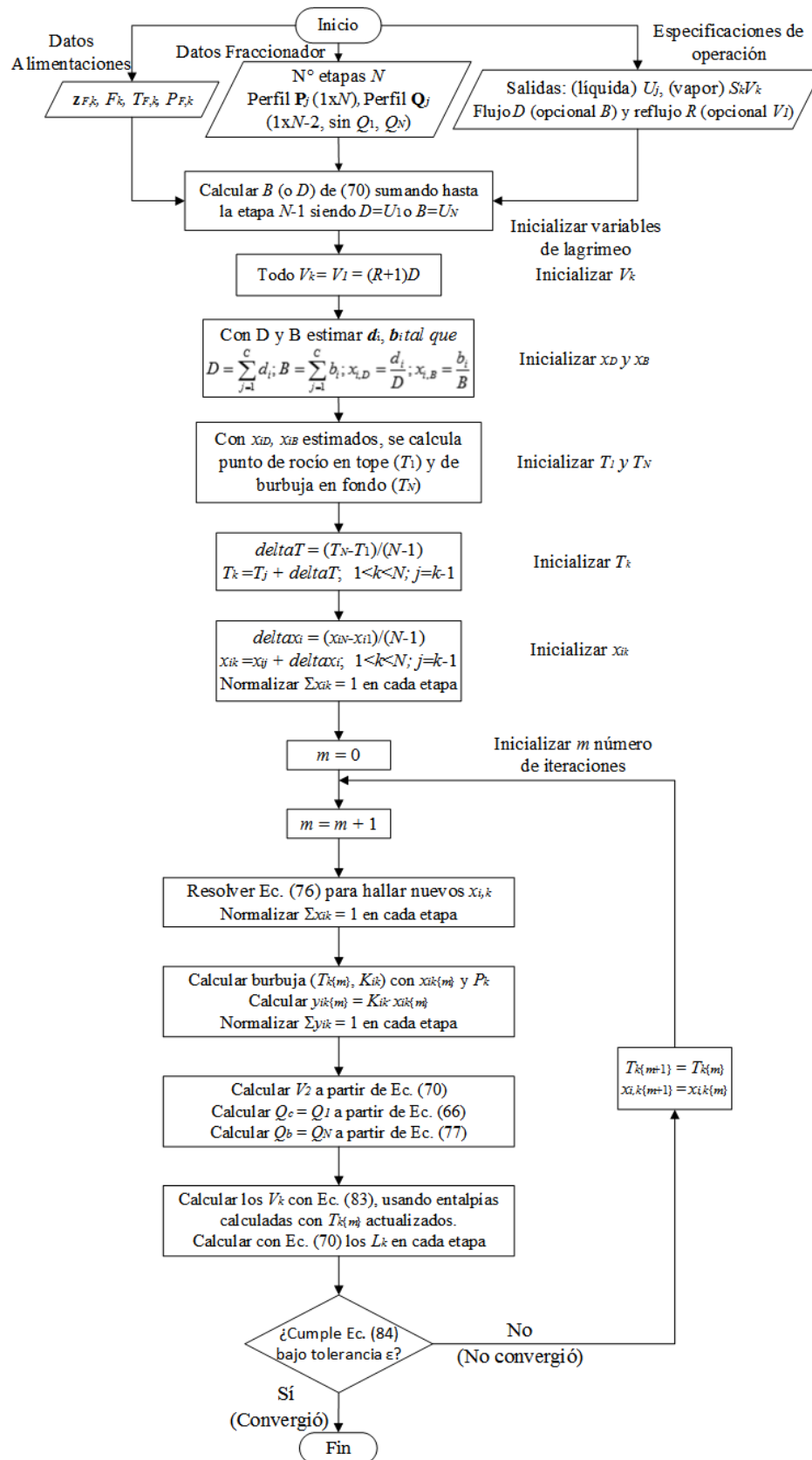


Figura 2.7. Algoritmo de simulación del método de Punto de Burbuja (BP) propuesto por Wang y Henke (adaptado para este estudio basado en Seader *et al.* 2008)

Aprovechando que los flujos interetapas de vapor y líquidos pueden ser escritos en función de sus términos individuales por componentes ( $V_k = \sum_i v_{i,k}$ ,  $L_k = \sum_i l_{i,k}$ ) y las definiciones de las fracciones molares indicadas en las ecuaciones (86) y (87).

$$y_{i,k} = \frac{v_{i,k}}{V_k} \quad (86)$$

$$x_{i,k} = \frac{l_{i,k}}{L_k} \quad (87)$$

Sustituyendo (86) y (87) en las ecuaciones de balance de masa (67) se obtienen las ecuaciones MESH generales (62), (63) (64), (65) y (66) (Seader *et al.*, 2011). Con dicho sistema de ecuaciones planteado en función de las variables  $v_{i,k}$ ,  $l_{i,k}$  y  $T_k$  y agrupando todas ellas por etapas en forma matricial se puede representar el conjunto de todas las variables en el vector  $\mathbf{X}$ , presentado en la ecuación (88).

$$\mathbf{X}_k = [v_{1,k}, v_{2,k}, \dots, v_{i,k}, \dots, v_{C,k}, l_{1,k}, \dots, l_{C,k}, T_k]^T \quad (88)$$

Cada vector  $\mathbf{X}$  está conformado con base en las variables expuestas en cada una de las etapas, por lo que la matriz de variables de estado conteniendo a todas las etapas será la que se expresa en la ecuación (89).

$$\mathbf{X} = [\mathbf{X}_{k=1}, \mathbf{X}_2, \dots, \mathbf{X}_k, \dots, \mathbf{X}_N]^T \quad (89)$$

Las variables de estado varían en un factor determinado en cada iteración  $m$  según lo corregido por las discrepancias de las ecuaciones MESH vistas como funciones objetivo, es decir, el Jacobiano de las funciones de estado brinda el factor de cambio de  $\mathbf{X}$ . Cuando se haya conseguido el punto de operación en estado estable  $\Delta\mathbf{X}$  debe ser cero, como indica la ecuación (90).

$$\Delta\mathbf{X}^{(m)} = - \left[ \left( \frac{\partial \mathbf{F}}{\partial \mathbf{X}} \right)^{-1} \right]^{(m)} \mathbf{F}^{(m)} \quad (90)$$

donde:

$$\mathbf{F} = [\mathbf{F}_{k=1}, \mathbf{F}_2, \dots, \mathbf{F}_k, \dots, \mathbf{F}_N]^T \quad (91)$$

Siendo  $\mathbf{F}_k$  el vector de funciones  $M$ ,  $E$  y  $H$ , siendo las ecuaciones de suma ( $S$ ) garantizadas por la condición de flujos molares por componente, para cada etapa  $k$  arreglados en el orden indicado en la ecuación (92).

$$\mathbf{F}_k = [H_k, M_{i=1,k}, M_{2,k}, \dots, M_{i,k}, \dots, M_{C,k}, E_{1,k}, E_{2,k}, \dots, E_{i,k}, \dots, E_{C,k}]^T \quad (92)$$

Entonces las correcciones simultáneas son calculadas utilizando los factores de cambio calculados en la iteración  $m$  para poder obtener un estimado de las variables de  $\mathbf{X}$  que servirá para utilizar en una iteración subsiguiente, como indica la ecuación (93).

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} + t\Delta\mathbf{X}^{(k)} \quad (93)$$

donde  $t$  es un factor del que depende la convergencia.

Como se especificó, la matriz jacobiana ( $N \times N$ ) que representa el factor  $(\partial \mathbf{F} / \partial \mathbf{X})$  está conformada por una tridiagonal de bloques de matrices de derivadas parciales de todas las funciones MESH con respecto a todas las variables de salida tal y como se indica en la ecuación (94).

$$\frac{\partial \mathbf{F}}{\partial \mathbf{X}} = \begin{bmatrix} \overline{\mathbf{B}}_1 & \overline{\mathbf{C}}_1 & \dots & 0 & \dots & 0 \\ \overline{\mathbf{A}}_2 & \overline{\mathbf{B}}_{k-1} & \overline{\mathbf{C}}_{k-1} & \dots & 0 \\ 0 & \overline{\mathbf{A}}_k & \overline{\mathbf{B}}_{N-1} & \overline{\mathbf{C}}_{N-1} \\ 0 & \dots & 0 & \overline{\mathbf{A}}_N & \overline{\mathbf{B}}_N \end{bmatrix} \quad (94)$$

La matriz es de forma de bloque tridiagonal debido a que las dependencias de las variables de estado de la etapa  $k$  se conforman por variables de estado de las etapas anteriores y posteriores también, es decir,  $F_k$  depende de  $k-1$ ,  $k$  y  $k+1$ . Cada bloque  $\overline{\mathbf{A}}_k$ ,  $\overline{\mathbf{B}}_k$  y  $\overline{\mathbf{C}}_k$  corresponde con submatrices de derivadas parciales de funciones de la etapa  $k$  con respecto a las variables de salida de las etapas  $k-1$ ,  $k$  y  $k+1$ . De tal manera, en las ecuaciones (95), (96) y (97).

En las ecuaciones (95), (96) y (97) se señalizan mediante un marco los componentes de la matriz Jacobiana que corresponden a términos diferentes de cero, mientras que los términos no señalados son todos nulos.

(96)

$$\bar{\mathbf{B}}_k = \frac{\partial \mathbf{F}_k}{\partial \mathbf{X}_k} =$$



$$\bar{\mathbf{C}}_k = \frac{\partial \mathbf{F}_k}{\partial \mathbf{X}_{k+1}} = \begin{array}{c} \mathbf{F}_k \quad \mathbf{X}_{k+1} \\ \left[ \begin{array}{cccc|ccc} \frac{\partial \mathbf{H}_k}{\partial v_{1,k+1}} & \frac{\partial \mathbf{H}_k}{\partial v_{2,k+1}} & \dots & \frac{\partial \mathbf{H}_k}{\partial v_{C,k+1}} & \frac{\partial \mathbf{H}_k}{\partial l_{1,k+1}} & \dots & \frac{\partial \mathbf{H}_k}{\partial l_{C,k+1}} & \frac{\partial \mathbf{H}_k}{\partial T_{k+1}} \\ \frac{\partial \mathbf{M}_{1,k}}{\partial v_{1,k+1}} & & & \frac{\partial \mathbf{M}_{1,k}}{\partial v_{C,k+1}} & \frac{\partial \mathbf{M}_{1,k}}{\partial l_{1,k+1}} & \dots & \frac{\partial \mathbf{M}_{1,k}}{\partial l_{C,k+1}} & \frac{\partial \mathbf{M}_{1,k}}{\partial T_{k+1}} \\ \vdots & \dots & \dots & \vdots & \dots & \dots & & \vdots \\ \frac{\partial \mathbf{M}_{C,k}}{\partial v_{1,k+1}} & & & \frac{\partial \mathbf{M}_{C,k}}{\partial v_{C,k+1}} & \dots & \dots & \frac{\partial \mathbf{M}_{1,k}}{\partial l_{C,k+1}} & \\ \frac{\partial \mathbf{E}_{1,k}}{\partial v_{1,k+1}} & \frac{\partial \mathbf{E}_{1,k}}{\partial v_{2,k+1}} & \dots & & \frac{\partial \mathbf{E}_{1,k}}{\partial l_{1,k+1}} & \dots & & \\ \vdots & \vdots & \dots & \vdots & \vdots & \dots & & \vdots \\ \frac{\partial \mathbf{E}_{C,k}}{\partial v_{1,k+1}} & \frac{\partial \mathbf{E}_{C,k}}{\partial v_{2,k+1}} & \dots & & & \dots & & \frac{\partial \mathbf{E}_{C,k}}{\partial T_{k+1}} \end{array} \right] \end{array} \quad (97)$$

Para el sistema no lineal de la matriz tridiagonal por bloques se usa el algoritmo de Thomas con operaciones matriciales. En la Fig. 2.8 se muestra el algoritmo del método Newton Raphson.

La forma implementada del algoritmo de Naphtali Sandholm es mostrada en la Fig. 2.8. Este puede utilizar una especificación flexible permitiendo diferentes variables de entrada. Para comenzar el comienzo del algoritmo de resolución se deben inicializar las variables de  $T_k$  y  $V_k$ , para los cuales se hace un cálculo similar al discutido en el apartado de Wang y Henke, se estiman los flujos de vapor interetapas como constantes y el perfil lineal de las temperaturas entre tope y fondo. La convergencia global del proceso depende muy sensiblemente de los estimados iniciales asignados a las variables de la iteración  $v_{i,k}$ ,  $l_{i,k}$ . Una vez inicializadas las variables, la función de discrepancia es evaluada y comparada con el criterio de convergencia presentado en la ecuación (99).

$$\tau_3 = \sum_{k=1}^N \left\{ (H_k)^2 + \sum_{i=1}^C \left[ (M_{i,k})^2 + (E_{i,k})^2 \right] \right\} \leq \varepsilon_3 \quad (98)$$

El error de convergencia  $\varepsilon_3$  se calcula con la ecuación (99).

$$\varepsilon_3 = N(2C+1) \left( \sum_{k=1}^N F_k^2 \right) 10^{-10} \quad (99)$$

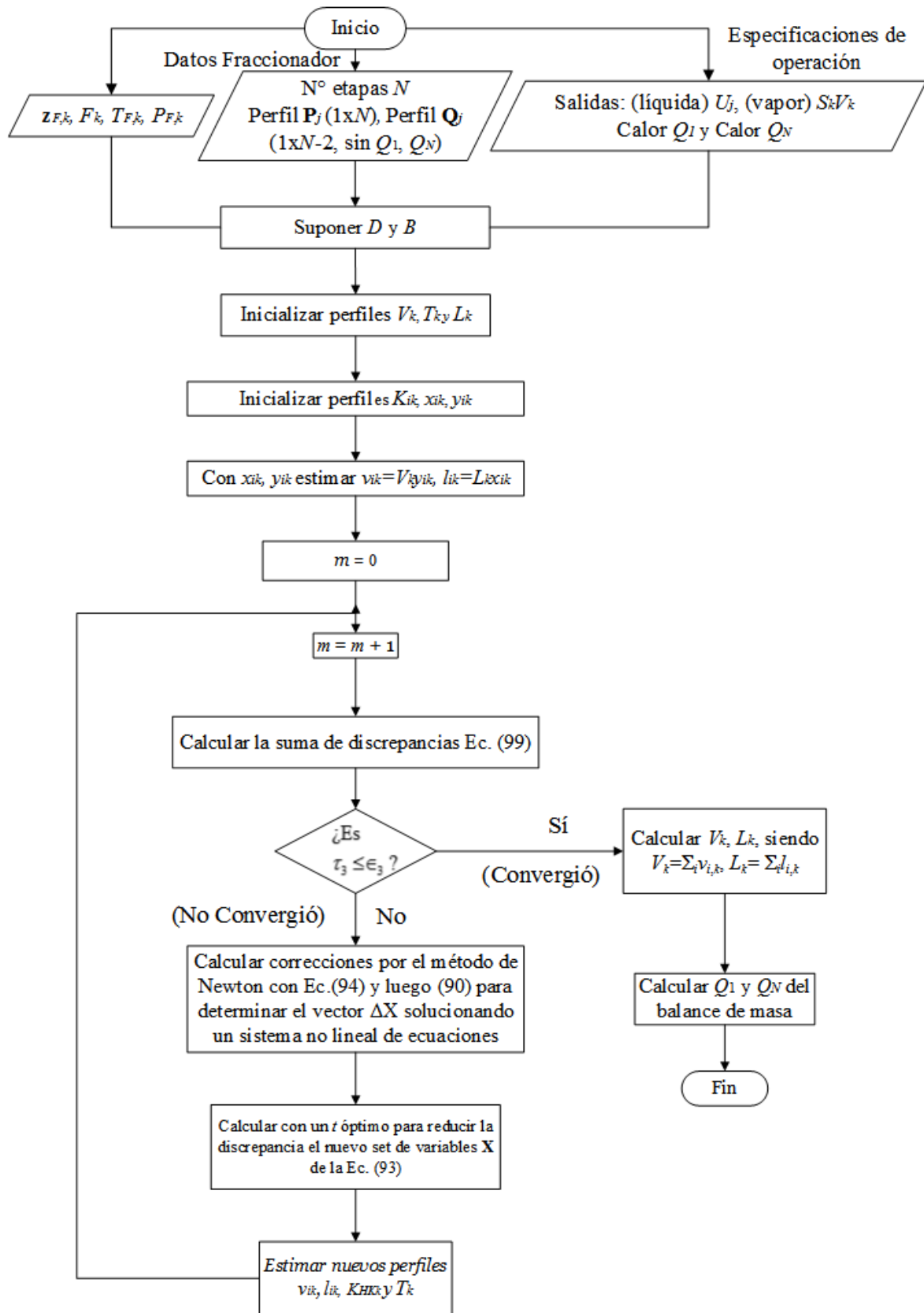


Figura 2.8. Algoritmo de Naphtali Sandholm para la resolución de torres de fraccionamiento por el método de Newton Raphson de Corrección Simultánea (Seader *et. al.*, 2011)

En cada iteración, las correcciones dadas por el método de Newton, ecuación (91), son utilizadas para actualizar las variables de goteo con el uso de la ecuación (90). El parámetro  $t$  en la ecuación (90) sirve para atenuar o acelerar la convergencia, su valor numérico debe ser mayor a cero. Es posible determinar un valor óptimo de  $t$  minimizando la suma de los cuadrados proporcionada por la ecuación (98). Si se llega a producir algún flujo molar negativo, Naphtali y Sandholm recomiendan el uso de la ecuación (100) para reubicar el flujo a un valor positivo cercano a cero:

$$\mathbf{X}^{(m+1)} = \mathbf{X}^{(m)} \exp \left[ \frac{t \Delta \mathbf{X}^{(m)}}{\mathbf{X}^{(m)}} \right] \quad (100)$$

Para cumplir con las condiciones de especificación de la columna, de las que debe proveerse una condición en el destilado y otra en el fondo, se construye una función objetivo por cada condición y se itera hasta reducirla a cero. Tal como fueron planteadas las ecuaciones (62), (63), (64), (65) y (66), las funciones objetivos vienen dadas por la ecuación del balance de energía (H) en el rehervidor y en el condensador ( $H_1$  y  $H_N$ ), para lo cual se debe proveer como condición el calor del condensador en la etapa 1 ( $Q_1 = Q_c$ ) y también el calor del rehervidor de fondo en la etapa  $N$  ( $Q_N = Q_w$ ) y reescribirse de la forma planteada en la ecuaciones (101) y (102).

$$F_1(Q_c, s_1, f_i, h_F) \Rightarrow H_1 + 2Q_1 \Rightarrow h_{L,1}(1 + s_1) \sum_{i=1}^C l_{i,1} + h_{V,1} \sum_{i=1}^C v_{i,1} - h_{V,2} \sum_{i=1}^C v_{i,2} - h_F \sum_{i=1}^C f_{i,1} + Q_c = 0 \quad (101)$$

$$F_N \Rightarrow H_N + 2Q_N \Rightarrow h_{L,N} \sum_{i=1}^C l_{i,N} + h_{V,N}(1 + S_N) \sum_{i=1}^C v_{i,N} - h_{L,N-1} \sum_{i=1}^C l_{i,N-1} - h_F \sum_{i=1}^C f_{i,N} + Q_w = 0 \quad (102)$$

Ambos calores no son condiciones recomendables para la especificación de torres de destilación, debido a que cada uno es muy dependiente del valor del otro y esto conlleva a problemas de convergencia, por lo cual en la Tabla 2.4 se presenta un conjunto de funciones objetivo que pueden utilizarse para el reemplazo de las ecuaciones  $H_1$  y  $H_N$  y únicamente esas dos ecuaciones, de forma que la convergencia dependa de otras distintas variables de especificación, con lo cual los calores sean calculados en base a los resultados ya obtenidos y no entorpezcan la

convergencia (Seader *et al.*, 2011).

Tabla 2.4. Reemplazo para las funciones objetivo para las diferentes condiciones de especificación en el método de Naphtali Sandholm

Especificación	Reemplazo para $F_1$	Reemplazo para $F_N$
Condición de reflujo líquido (L/D) o vapor rehervido (V/W)	$\sum l_{i,1} - (L/D) \sum v_{i,1} = 0$ <p>Si el destilado es vapor</p> $\sum v_{i,2} - (L/D) \left( \sum v_{i,1} + s_1 \sum l_{i,1} \right) - s_1 \sum l_{i,1} - \sum v_{i,1} = 0$ <p>Si el destilado es mezcla o líquido</p>	$\sum v_{i,N} - (V/B) \sum l_{i,N} = 0$
Flujo molar de producto, $D$ o $W$	$\sum v_{i,1} - D = 0$	$\sum l_{i,N} - W = 0$
Flujo molar del componente $i$ en el producto tope o fondo, $d_i$ o $w_i$	$v_{i,1} - d_i = 0$	$l_{i,1} - b_i = 0$

### 2.6.1.6.3 Método de Lewis y Matheson

El método de Lewis y Matheson extiende a sistemas multicomponentes las suposiciones y los balances involucrados en las soluciones según los métodos gráficos binarios y pseudobinarios como el de McCabe y Thiele y el método de Hengstebeck (Towler y Sinnott, 2008).

Entre las suposiciones de Lewis y Matheson se mantiene la de flujos líquidos y vapores interetapas equimolares, además los balances de materia y la relación de equilibrio es resuelta etapa por etapa desde el tope o el fondo de la columna, o inclusive desde ambos el tope y el fondo hasta la intersección de la alimentación (Treybal, 1982).

Para resolver un problema por el método de Lewis y Matheson las siguientes variables deben ser especificadas o determinadas a partir de otras variables.

- Composiciones, flujos y condiciones intensivas de la alimentación ( $z_{i,F}$ ,  $F$ ,  $T_F$ ,  $P_F$ ).
- Distribución entre tope y/o fondo de los compuestos claves.
- Uno de los flujos de salida de destilado extraído en el tope o de producto de residuo de fondo ( $D$  o  $B$ ).
- Relación de reflujo o reflujo líquido real ( $L/D$  o  $L$ ).
- Presión, o perfil de presiones por etapas, de la columna de fraccionamiento.
- Suposiciones en la distribución de los compuestos no claves.

El procedimiento de cálculo contempla el uso de balances de masa para estimar la composición en la etapa  $k$  de algún líquido o vapor, usualmente comenzando por el plato 1 en el que se calcula la composición molar de líquido que está en equilibrio con el vapor que es introducido en el condensador. A partir de ese plato se realiza el cálculo de las composiciones de vapor del plato 2, continuando sucesivamente plato a plato hasta alcanzar que la relación entre los compuestos LK y HK,  $x_{LK,k}/x_{HK,k}$ , la cual ha venido cayendo con cada una de las etapas de equilibrio teóricas calculadas, sea igual o menor a la de la alimentación. Posteriormente se comienza desde el producto de fondo y se realiza el cálculo de los balances de Lewis plato a plato, conjunto al algoritmo de punto de burbuja, hasta alcanzar un plato donde la relación  $x_{LK,k}/x_{HK,k}$  sea similar a la del plato de alimentación determinado en la fase anterior, donde se determina que se alcanzó el punto de acoplado, siendo el plato  $f+1$  el anterior. Entonces se tiene el número de platos total,  $N$ , sumando los obtenidos en cada zona de la cascada. Se puede realizar un ajuste de las composiciones cercanas a la alimentación para que la distribución de los componentes en ambas zonas presente continuidad (Treybal, 1982).

Con ello, el balance de masa en un plato genérico  $k$ , perteneciente a la zona de rectificación, queda como el indicado en la ecuación (103) (Treybal, 1982).

$$y_{i,k+1} = \frac{L}{V} x_{i,k} + \frac{D}{V} z_{i,D} \quad (103)$$

Para la distribución de los no claves se puede despreciar la presencia de los compuestos no claves más pesados que el clave pesado, con lo cual se puede obtener del método una estimación satisfactoria del número de etapas teóricas. Similarmente, para los compuestos no claves más livianos que el clave liviano puede despreciarse su fracción molar en el producto de fondo, en el caso de realizar la distribución en el sentido inverso, desde el fondo hacia la alimentación. Para

usar el cálculo en sentido inverso, se debe hacer uso de la recta de operación de la zona de despojo, que se encuentra presentada en la ecuación (104).

$$x_{i,k'} = \frac{V'}{L'} y_{i,k'+1} + \frac{W}{L'} x_{i,w} \quad (104)$$

El método de Lewis y Matheson posee grandes inestabilidades para su convergencia como método de cálculo computacional (Doherty *et al.*, 2008), por lo cual no se recomienda su uso, salvo si el problema es de cálculo fácil y directo involucrando una mezcla con componentes de la misma naturaleza y sistemas de una torre única que posean una alimentación y ninguna salida lateral (Towler y Sinnott, 2008).

En el caso en el cual se determine que cualquiera de los productos de tope o fondo contiene la totalidad de los compuestos presentes en la mezcla de alimentación, se puede considerar comenzar el algoritmo de cálculo desde dicho producto hasta el otro extremo de la torre, realizando las correcciones a los flujos de vapor y líquido al localizar la etapa de alimentación para considerar el cambio debido a la condición térmica  $q$  visto en la ecuación (29).

## 2.7 Análisis de Grados de Libertad para sistemas de separación en cascada

Un sistema de separación en cascada involucra el grupo de ecuaciones MESH consistentes en restricciones derivadas de los balances de materia, energía y equilibrio entre fases. De esta forma se requiere especificar un conjunto de variables de diseño tal que queden por determinar exactamente el mismo número de variables incógnitas que de ecuaciones disponibles (Seader *et al.*, 2011).

Un método para contar las variables de diseño fue desarrollado por Kwauk (1956), este se presenta en la ecuación 106 y consiste en sustraer el número de ecuaciones ( $N_E$ ) del número total de variables ( $N_V$ ), para obtener el número de variables de diseño ( $N_D$ ) a especificar:

$$N_D = N_V - N_E \quad (105)$$

Las corrientes de alimentación poseen un total de  $C+3$  variables, que están compuestas por una fracción molar o flujo molar por cada componente ( $C$ ), un flujo molar total (+1) y dos variables intensivas que determinan la condición de la fase, sean  $T$  y  $P$  o, equivalentemente, la condición

térmica ( $q$ ) o la fracción vaporizada ( $\beta$ ). Al mismo tiempo existe una restricción tal que la suma de las fracciones molares sea la unidad. Por ende, a cada alimentación van asociadas  $C+2$  variables independientes (Seader *et. al.*, 2011).

## **CAPÍTULO 2**

### **METODOLOGÍA**

En el presente capítulo se presenta el planteamiento orientado a objetos con que se realizó la programación de los diferentes métodos de cálculo corto, aproximado, semiriguroso y riguroso. Se realizó el trabajo con base en el lenguaje de programación MATLAB® de forma que pueda ser utilizado con objetivos pedagógicos para los alumnos de la carrera de Ingeniería Química en la universidad, además de ser una herramienta que ofrece muchas posibilidades implementadas nativamente para la resolución de sistemas de ecuaciones y sistemas de ecuaciones diferenciales como los presentados por los distintos métodos de cálculo en destilación. Además las herramientas cuentan con un mecanismo para determinar los grados de libertad que es mostrado al usuario.

#### **3.1 Programación Orientada a Objetos**

Al realizar una investigación preliminar se encontraron referencias acerca de los beneficios de la programación basada en objetos, su implementación es obligatoria en las extensiones o módulos de cálculos estándares como CAPE-OPEN® y se decidió realizar el proyecto en base a las recientemente implementadas capacidades de construcción de clases en el lenguaje MATLAB® (Navarro, 2003).

La programación orientada a objetos consiste en la creación de estructuras o elementos completamente independientes, cuyo funcionamiento sea estándar y aplicable para dar solución a varios tipos diferentes de problemas, partiendo de la unión de varias, entre dichas estructuras, para poder dar solución a un problema complejo que requiere la aplicación de todas estas técnicas básicas.

Para describir un modelo basado en objetos, primero se deben definir los objetos y sus características y sus conceptos fundamentales. Un objeto es una abstracción bajo una misma delimitación de un grupo de propiedades que describen el estado de lo que éste represente, un grupo de métodos, funcionalidades o eventos que aplican diferentes operaciones, actividades o respuestas para transitar desde el estado inicial hacia un cambio o una salida.

Las propiedades y los métodos se definen dentro del contexto de las denominadas clases. Una clase actúa como un contenedor de características de un objeto. De esta forma el uso de clases permite varias ventajas sobre la programación funcional tradicional. Entre estas se encuentra el



encapsulamiento, la simplificación y el reúso. La ventaja del encapsulamiento es el transformar las operaciones en unidades integradas en las cuales cada etapa de procesamiento almacena, puede manejar, tanto los datos ingresados o disponibles, como las transformaciones a llevar a cabo, con lo cual se aísla de la dependencia de otras piezas de la construcción del programa y puede permitir al programador enfocarse de forma independiente en un componente por vez y no en el sistema completo todo el tiempo, con lo cual se gana en generar aplicaciones de mayor solidez.

La simplificación otorgada por establecer estructuras independientes permite al desarrollador realizar la interconexión de estructuras de distinto origen. El reúso entonces es un aliciente fundamental en la programación orientada a objetos, las rutinas encapsuladas pueden ser empleadas en diferentes aplicaciones posteriores.

### **3.1.1 Descripción de los objetos**

Al describir los objetos se deben establecer tanto sus propiedades y métodos, así como identificarlos y establecer las jerarquías entre ellos que delimitan el rol de su aplicación. Las propiedades son datos asociados con el objeto que clasifican lo que éste “sabe” o cómo el mismo “es” y el estado actual en el cual se encuentra. Los métodos identifican lo que puede hacer y cómo actúa.

En MATLAB® desde 2012 la jerarquía superior de todos los objetos es delimitada por la clase “handle” que es la que facilita al programador la construcción de estructuras con métodos y propiedades y las cohesiona. En versiones anteriores también se podían asociar distintas funciones bajo la misma estructura, aunque cada desarrollador debía tomarse el trabajo adicional de crear las asociaciones que otorgan el encapsulamiento a los objetos. Incluso algunos realizaron la programación de modelos termodinámicos basados en objetos no estándares sin estar basados en la clase handle, como por ejemplo Martín *et al.* (2011) con el propósito de facilitar la enseñanza de cursos de ingeniería.

En este proyecto se realizó uso de los objetos estándares basados en la clase “handle” con la expectativa que perdure la compatibilidad de los algoritmos implementados con las versiones subsiguientes de MATLAB®, así como también debido a la solidez que supuso partir de la estructura ya establecida y poder enfocar el esfuerzo únicamente en la programación de los métodos de cálculo termodinámicos.

### 3.2. Algoritmos de cálculo termodinámicos

Uno de los beneficios obtenidos del enfoque de programación orientado a objetos se encuentra en la implementación de los módulos de cálculo termodinámicos encapsulados, lo que permiten la fácil revisión o adaptación de las ecuaciones existentes, así como también la adición de nuevas ecuaciones sin tener que realizar más que la programación modular de las nuevas ecuaciones, independientemente de los métodos asociados a las interacciones para el cálculo de predicción de equilibrio o de cálculo de propiedades residuales, las cuales fueron realizadas pensando en los casos de ecuaciones cúbicas generalizadas como fue presentado en las ecuaciones (3) a (11).

A continuación se presenta una descripción corta de las clases que componen el modelo termodinámico, tal como se muestran en la Fig. 3.1 y para una mayor descripción se incluye en el Apéndice A el manual de documentación del proyecto.

Para representar a una sustancia pura, se tiene una clase denominada ‘Sustancia’, la cual posee todas las propiedades que caracterizan a una sustancia pura, como el nombre, la fórmula molecular, el peso molecular, los datos críticos de presión y temperatura, así como el factor acéntrico, un conjunto de correlaciones para más de 340 sustancias puras, obtenida en la base de datos del Instituto de Diseño de Propiedades Físicas (*DIPPR*, por sus siglas en inglés). También puede almacenar propiedades de componentes hipotéticos o de otras bases de datos y pueden ser definidas arbitrariamente para los propósitos del usuario. La clase Sustancia también posee un método para poder realizar la búsqueda en la base de datos de todas las propiedades termodinámicas características a partir del nombre del componente deseado.

Representando la combinación de diversas sustancias conjuntamente en una mezcla homogénea, se introduce una clase Mezcla, que almacena las sustancias puras, así como también una composición molar y los parámetros de interacción binaria entre las diferentes sustancias puras, que pueden ser definidos por el usuario para poder realizar la predicción de mezclas no ideales. Un método estima el número de combinaciones binarias entre los compuestos definidos y también los ubica en una matriz cuadrada tal que accediendo al elemento correspondiente se extrae el parámetro entre los compuestos deseados.

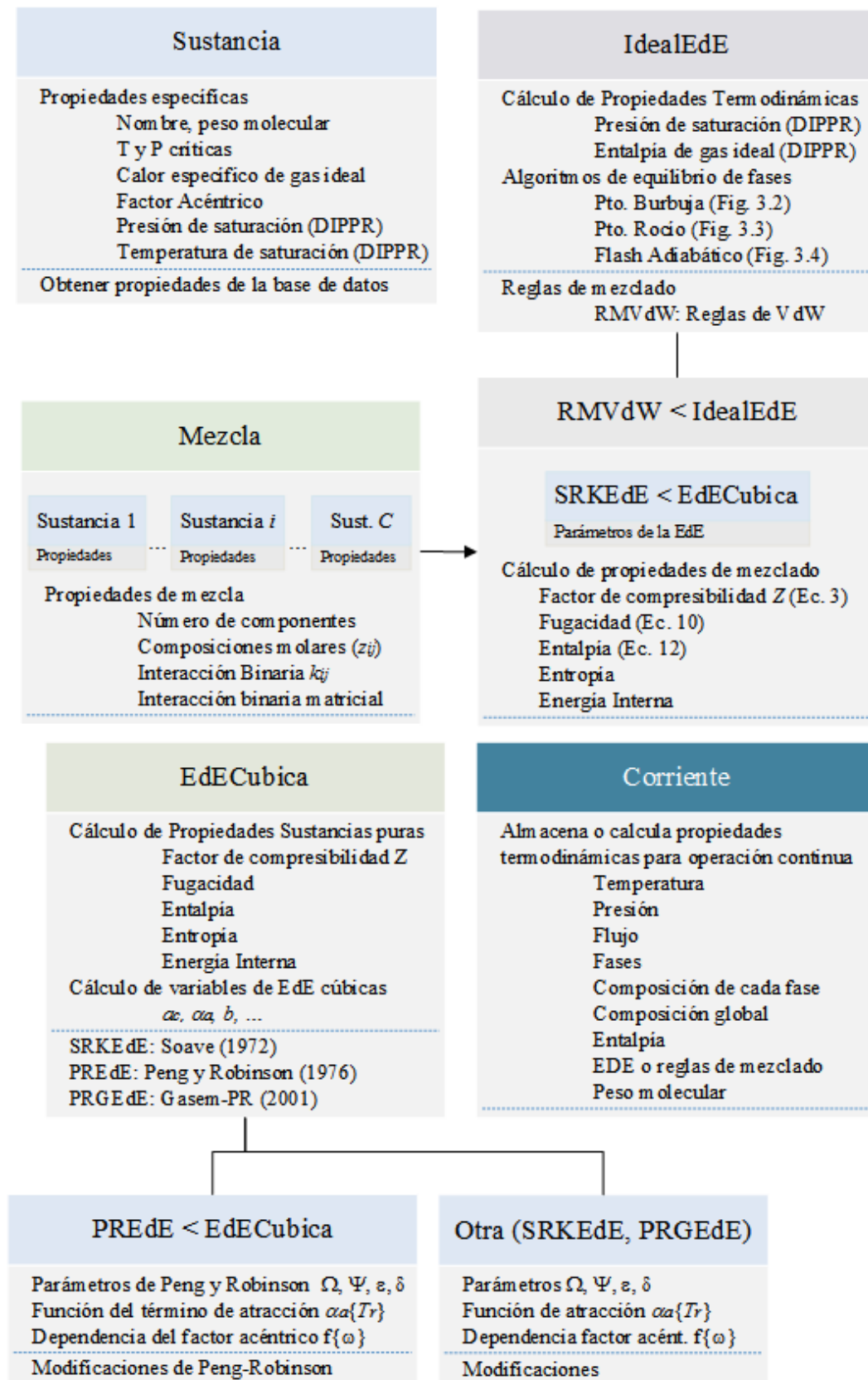


Figura 3.1 Jerarquía de los objetos de cálculo de propiedades termodinámicas

Para la estimación de propiedades de sustancias puras se introdujo el uso de ecuaciones de estado cúbicas en una clase llamada *EdECubica*, encargada de la resolución de propiedades de sustancias puras. Esta clase es abstracta ya que no posee por si misma ninguna ecuación asociada, su función es englobar los métodos de cálculo de propiedades de sustancias puras y como tal de ella heredan los métodos las ecuaciones cúbicas, en dado caso tanto Soave-Redlich-Kwong (*SRKEdE*), como Peng-Robinson (*PREdE*) y Gasem (*PRGEdE*). La función de *EdECubica* es facilitar al usuario la incorporación de nuevas ecuaciones cúbicas, únicamente debe proveer los parámetros de una función cúbica generalizada, como en la ecuación (3) y un método de cálculo del parámetro de atracción correspondiente. El reemplazo del módulo *EdECubica* por algún otro permitiría expandir el alcance para diferentes tipos de ecuaciones de estado de sustancias puras.

Los métodos de cálculo multicomponentes también fueron hechos con el objetivo de lograr el encapsulamiento de los algoritmos de cálculo. Para ello se utilizó una clase superior llamada ‘*IdealEdE*’ que, análogamente a *EdECubica*, almacena los métodos de cálculo multicomponente, como los de equilibrio de fases: el cálculo de la temperatura y composición del punto de burbuja, usando el algoritmo presentado en la Fig. 3.2, también el cálculo de la temperatura y composición del punto de rocío, con el algoritmo de la Fig. 3.3 y también la resolución de un flash isotérmico, Fig. 3.4.

La clase *IdealEdE* engloba a todas las ecuaciones de estado multicomponentes y, al mismo tiempo, también es en ella misma una ecuación de estado multicomponente para equilibrio de fases, que obtiene predicciones de gas ideal y solución ideal por ley de Raoult con datos basados en correlaciones de temperatura y presión de saturación obtenidas de base de datos *DIPPR*. Para ello utiliza un conjunto de métodos para el algoritmo de punto de burbuja (*BubbleT*), para el del punto de rocío (*DewT*), así como otros para entalpía residual y fugacidad.

Otro nivel de abstracción fue el de las reglas de mezclado con la cual se extiende el modelo de las ecuaciones cúbicas para su validez en sistemas de múltiples sustancias. Una variante de reglas de mezclado propuesta por Van der Waals fue programada en una clase denominada ‘*RMVdW*’, el cual agrupa los métodos de cálculo de los parámetros de atracción y de covolumen para mezclas estimadas por ecuaciones cúbicas. El reemplazo de este módulo por otro conjunto de reglas de mezclado, como las de Wong y Sandler, permitiría al programa mejorar las predicciones para otros casos, reutilizando el código de todas las ecuaciones de estado para sustancias puras.

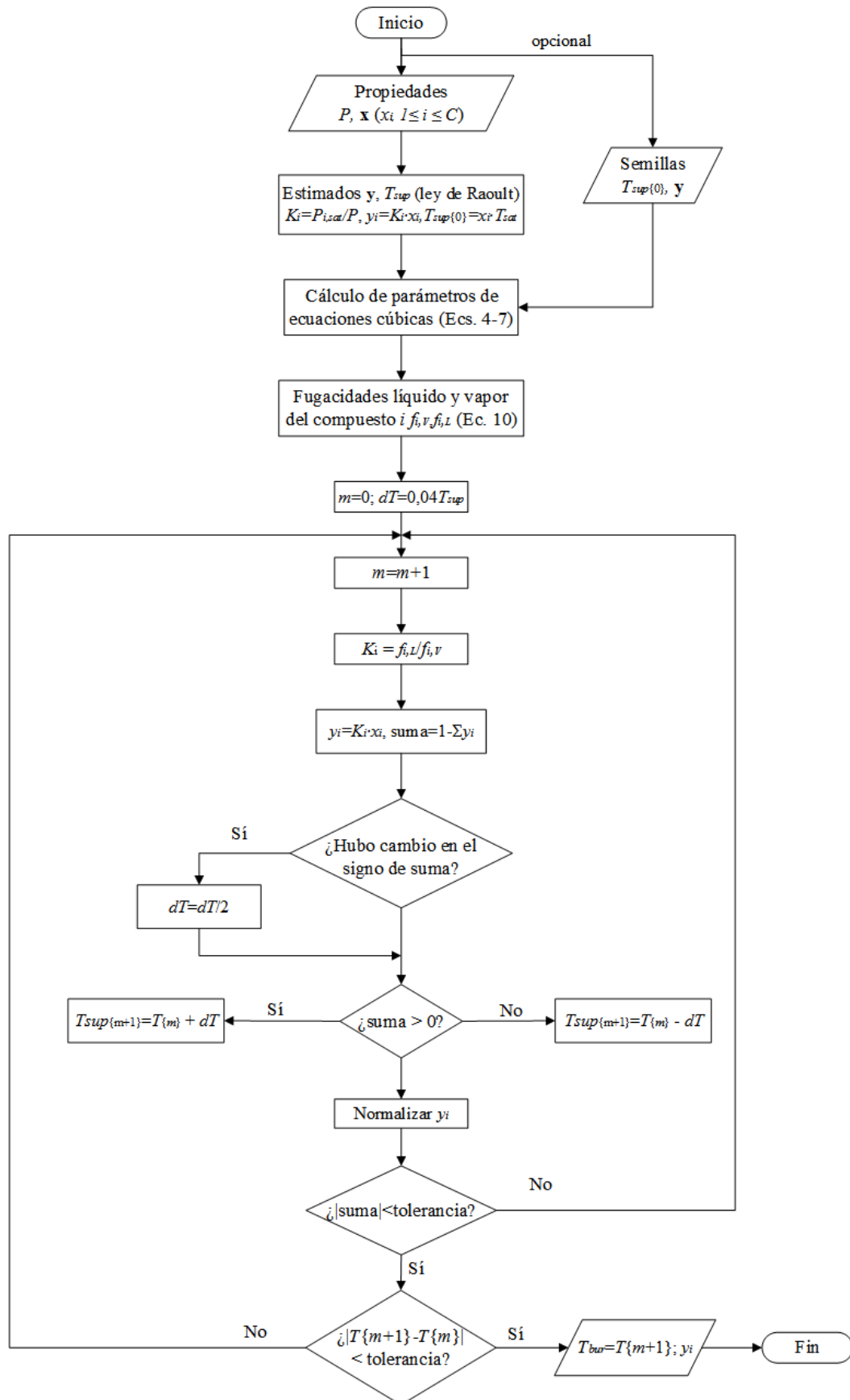


Figura 3.2 Algoritmo para hallar la temperatura de burbuja de una mezcla líquida.

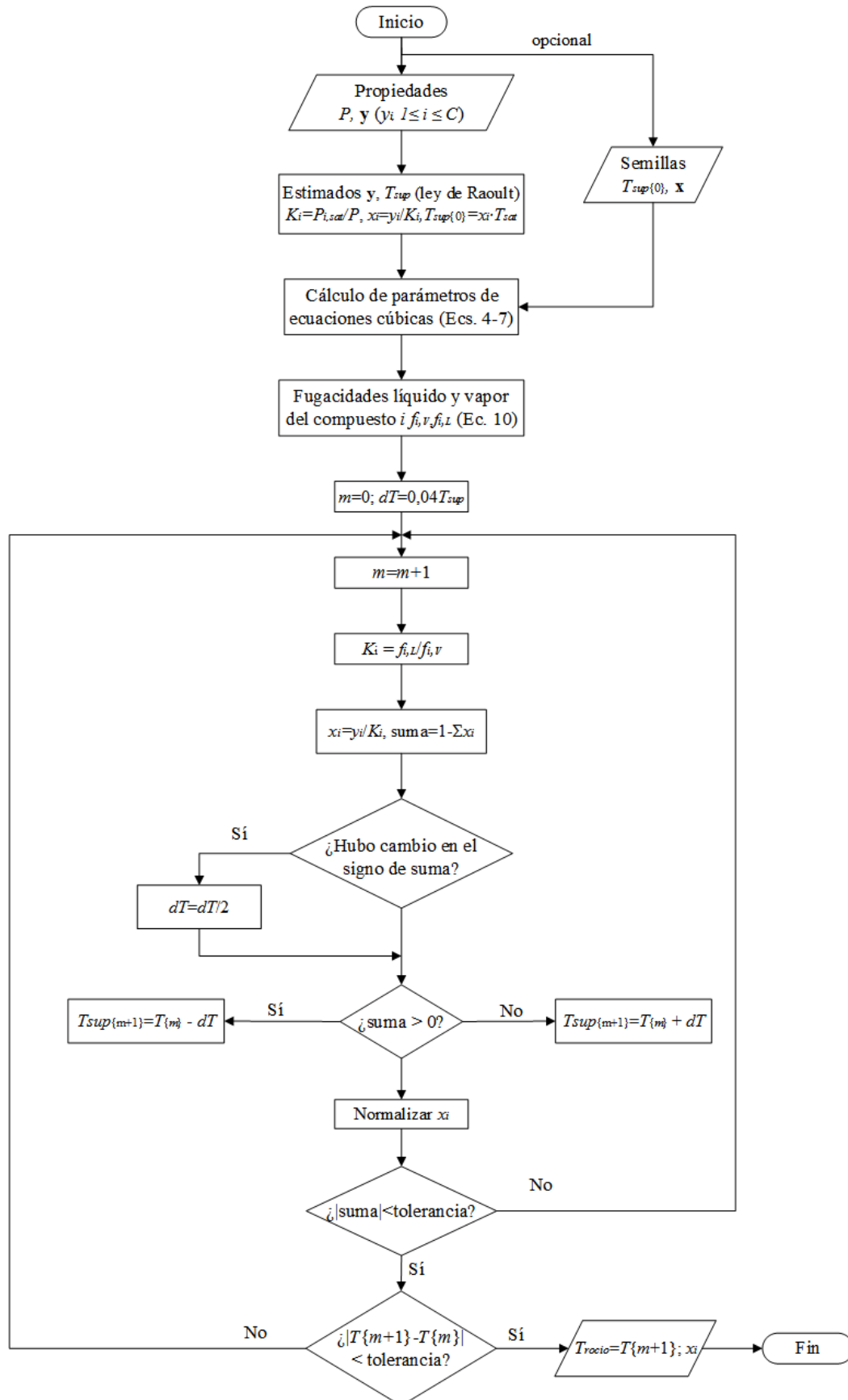


Figura 3.3 Algoritmo para hallar la temperatura de rocío de una mezcla vapor

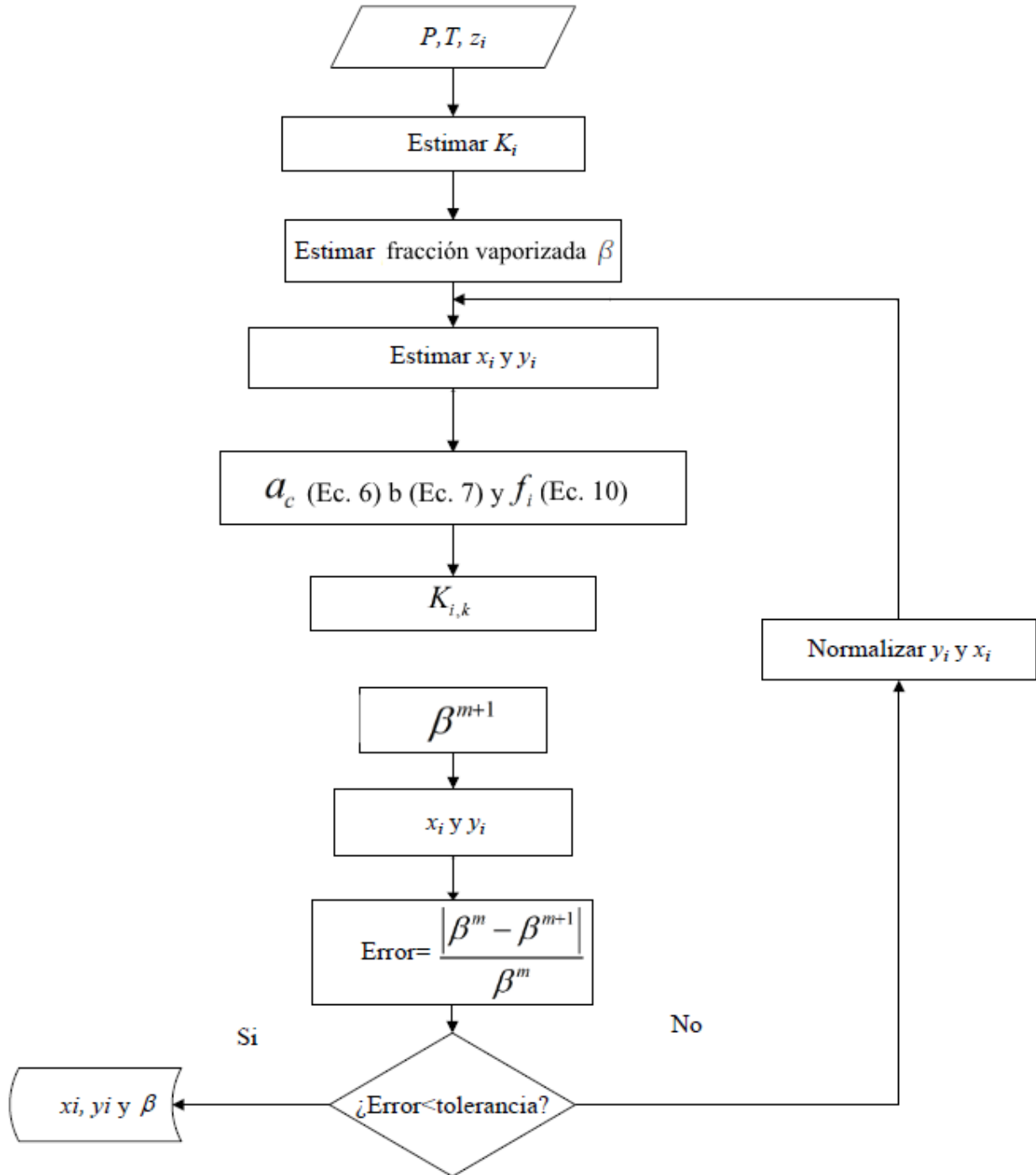


Figura 3.4 Algoritmo para hallar la fracción vaporizada de una mezcla

Los coeficientes de las correlaciones obtenidas de la base de datos *DIPPR* se encuentran tabulados en un conjunto de archivos de base de datos, de extensión .DAT y con la modificación de estos ya se amodifican las mismas para el caso de corrección de cualesquiera errores o de

cambios en los parámetros en ellas registradas.

Para los casos de operación continua se agrupan todos los elementos anteriores, las Sustancias, las Mezclas, las EDE y los métodos de cálculo y de predicción de fases en una clase llamada Corriente, la cual es responsable de obtener una mezcla y las propiedades de estado, como presión, temperatura, fracción vaporizada y flujo molar o másico, para calcular el resto de las propiedades como densidad, caudal, peso molecular, las fases y las composiciones de cada una de ellas, la entalpía y la condición térmica  $q$ . La clase corriente posee un identificador de texto que le permitiría distinguirla de otras en un entorno gráfico de trabajo, como los presentes en simuladores, lo que la programación orientada a objetos promueve.

### 3.3 Métodos de cálculo de torres de destilación

Los métodos de cálculo de columnas de destilación fueron programados con base en el enfoque de programación orientada a objetos y en particular su diseño fue basado en eventos, de forma que se pudiera desarrollar una interfaz gráfica de usuario (GUI, por sus siglas en inglés) alrededor de ellos para facilitar su uso para propósitos de enseñanza de métodos de destilación en los cursos del plan de estudio de Ingeniería Química.

El diseño basado en eventos permite la utilización de fases, o eventos, ante los cuales un objeto responde generando un grupo de resultados y al finalizar regresa a un estado de pausa, en el que espera recibir una nueva instrucción a partir de la cual generar otro grupo de resultados.

Los diagramas de los objetos para el cálculo de torres de destilación se muestran en la Fig. 3.5. En esta se muestran las diferentes propiedades que almacenan o calculan y los métodos con los cuales calculan sus resultados. Todas las clases están basadas en las extensiones multicomponentes de los métodos de cálculo, a partir de las cuales el caso binario es caso particular derivado.

Se requiere para todas las clases de torres de destilación el conjunto completo de C++ variables definidas en cada una de las alimentaciones tal que las mismas no son apropiadas para la resolución de requerimientos aguas arriba. A todas las clases de torres de destilación se les asoció una propiedad que almacena una clase Corriente para el producto de destilado y una para el producto de residuo, de tal forma que en un futuro pudiera implementarse un conjunto continuo cuya resolución dependa de los valores de cada equipo previo, como se realiza en simuladores comerciales.



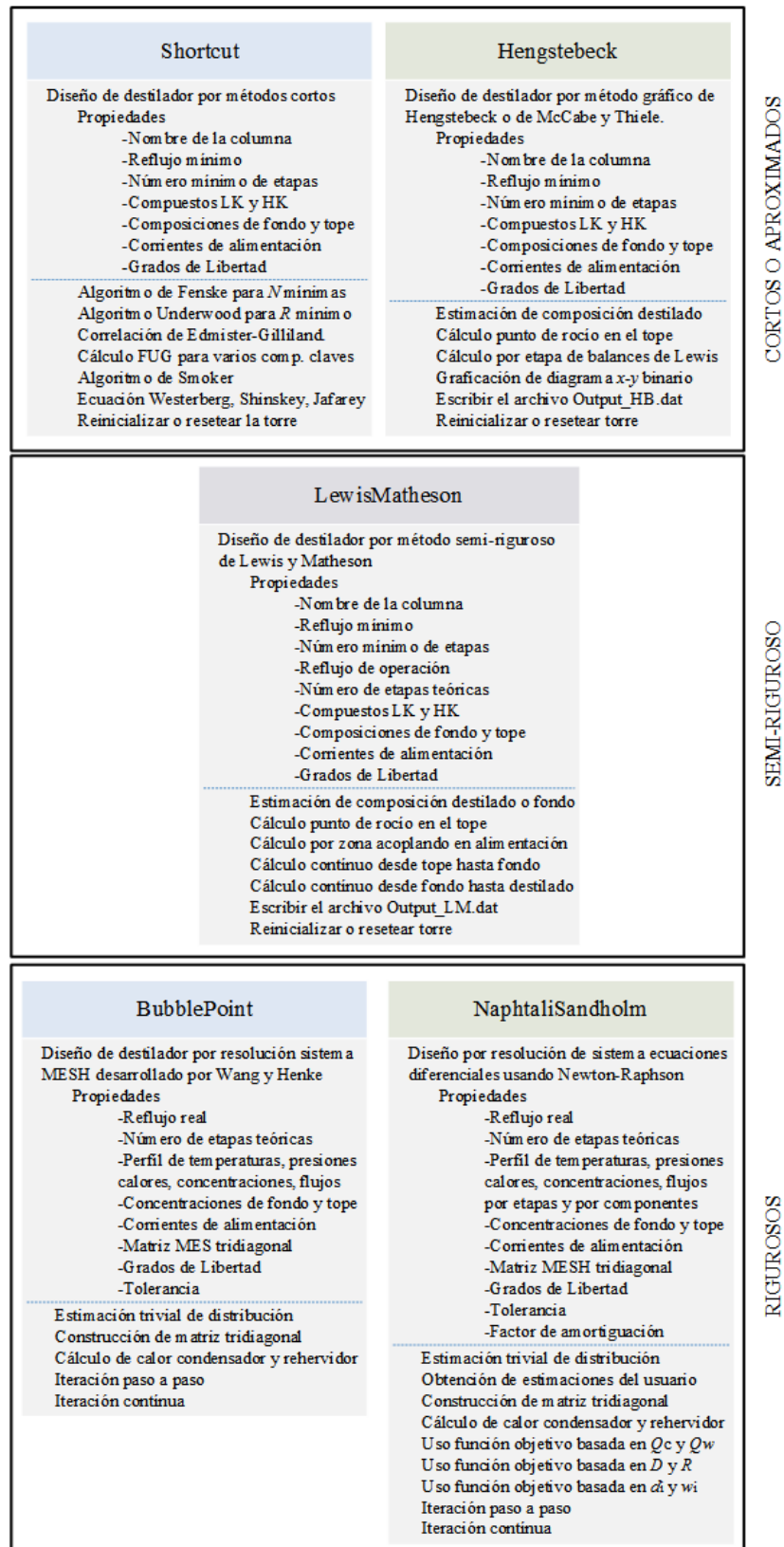


Figura 3.5 Diagrama de los objetos de cálculo de métodos de destilación

Para el cálculo de métodos cortos y aproximados de destilación se realizó la clase *Shortcut*. En ella se obtienen cálculos rápidos por el método FUG para destilación multicomponente y se aproximan a un caso pseudobinario las ecuaciones de Smoker, Shinskey, Jafarey y Westerberg. Las especificaciones que acepta la torre son presiones en el plato de tope y de fondo, recuperación fraccional del componente clave liviano (LK) en el tope o del componente clave pesado (HK) en el fondo, así como también composición deseada del componente LK en el tope o composición deseada del componente HK en el fondo o, por último una combinación de estas variables de diseño. Se puede otorgar como especificación una recuperación conjunta de un grupo de componentes de volatilidades consecutivas y el método realiza el tanteo necesario para obtener con precisión dicha recuperación a lo largo de múltiples componentes. La clase va calculando los grados de libertad y automáticamente procede a la resolución de la torre en cuanto se tienen suficientes datos iniciales.

Para la resolución corta o aproximada de un sistema en cascada por método gráfico de McCabe y Thiele, tanto binario como su extensión a multicomponente por analogía a una torre pseudobinaria, se utilizó la clase Hengstebeck. Las entradas admisibles son coeficientes de recuperación del producto clave liviano en el tope y del producto clave pesado en el fondo.

El método de Lewis y Matheson se calcula con la clase *LewisMatheson*. En esta se cuentan los grados de libertad, se admiten recuperaciones fraccionales o composiciones como variables de entrada y se puede indicar que se realice el cálculo desde el destilado hasta el fondo, desde el tope hasta el fondo o con unión en la alimentación desde los estimados de los productos, en cuyo caso realiza la corrección de la composición indicada en Treybal (1982). Los resultados incluyen un perfil de relaciones entre el líquido y el vapor para cada una de las etapas, con lo cual se puede estimar ambos perfiles de flujo de vapor y líquido si se supone que uno de los perfiles es constante a lo largo de la torre.

Para los métodos rigurosos basados en la resolución del sistema de ecuaciones MESH planteado en forma de matriz tridiagonal en lugar de utilizar el algoritmo de Thomas, se usaron las capacidades intrínsecas de Matlab para resolución de sistemas de ecuaciones matriciales.

El método de punto de burbuja basado en el algoritmo de Wang y Henke (1966), únicamente admite como variables de diseño la relación de reflujo en el tope de la columna ( $R$ ) y uno de los flujos de producto ( $D$  o  $W$ ). Se utilizó como criterio de tolerancia la ecuación (84), siendo el máximo error relativo en el cálculo de los flujos de vapor y el perfil de temperaturas  $1 \times 10^{-8}$ . Con

ello se observó que las temperaturas no variaron en sucesivas correcciones en más de una décima para ninguno de los casos probados.

Para el método de Newton Raphson se programaron tres casos de variables de diseño, basados en las funciones objetivo de la Tabla 2.4. En un primer caso, se tiene el calor de especificación tanto para el producto de destilado como el de residuo. También se pueden especificar análogamente al método de punto de burbuja los valores de la relación entre el flujo de líquido al primer plato y el destilado total ( $R$ ) y uno de los flujos de producto. Como criterio de tolerancia se utilizó la ecuación 100, aunque con una tolerancia elevada en  $10^4$  ya que se observó que el número de iteraciones para la convergencia se incrementaba rápidamente y la mayor precisión no hacía variar ni los flujos de vapor ni las temperaturas más allá de la sexta cifra significativa. El método acepta como estimados iniciales algunas variables de estado, por ejemplo: perfiles de temperatura, flujos por componente de salida, flujos molares de líquido o vapor.

### 3.4 Interfaz gráfica de usuario

La interfaz gráfica de usuario, presentada en la Fig. 3.6, presenta los siguientes elementos:

1. Menú desplegable de selección del método termodinámico a utilizar.
2. Menú desplegable para la selección del método termodinámico.
3. Herramienta de selección de sustancias puras componentes de la mezcla de alimentación.
4. Botones para añadir o remover sustancias puras de la alimentación.
5. Menú para la visualización y edición de las propiedades críticas y el factor acéntrico para las sustancias puras.
6. Edición de opciones avanzadas de la mezcla, en particular la adición de parámetros de interacción binaria.
7. Menú para la definición de las variables intensivas de la corriente de alimentación.
8. Tabla para la edición de los flujos molares individuales para cada componente y el flujo total.
9. Ubicación de la etapa para la corriente de alimentación (si aplica al método) y opción para la normalización de los flujos molares.
10. Número total de etapas teóricas del arreglo en cascada.
11. Tabla para la definición del perfil de presiones y de calor en la torre.
12. Selección del set de variables de diseño.

13. Tabla de condiciones de las corrientes de alimentación agregadas.
14. Obtención de variables de especificación para el diseño de la torre.
15. Resultados de flujos molares, perfiles, error y número de iteraciones, si aplica.
16. Distribución molar por componente de productos en las salidas.
17. Diagrama esquematizado de la torre de separación.
18. Opciones de resolución de estimados, cálculos finales o paso por paso.

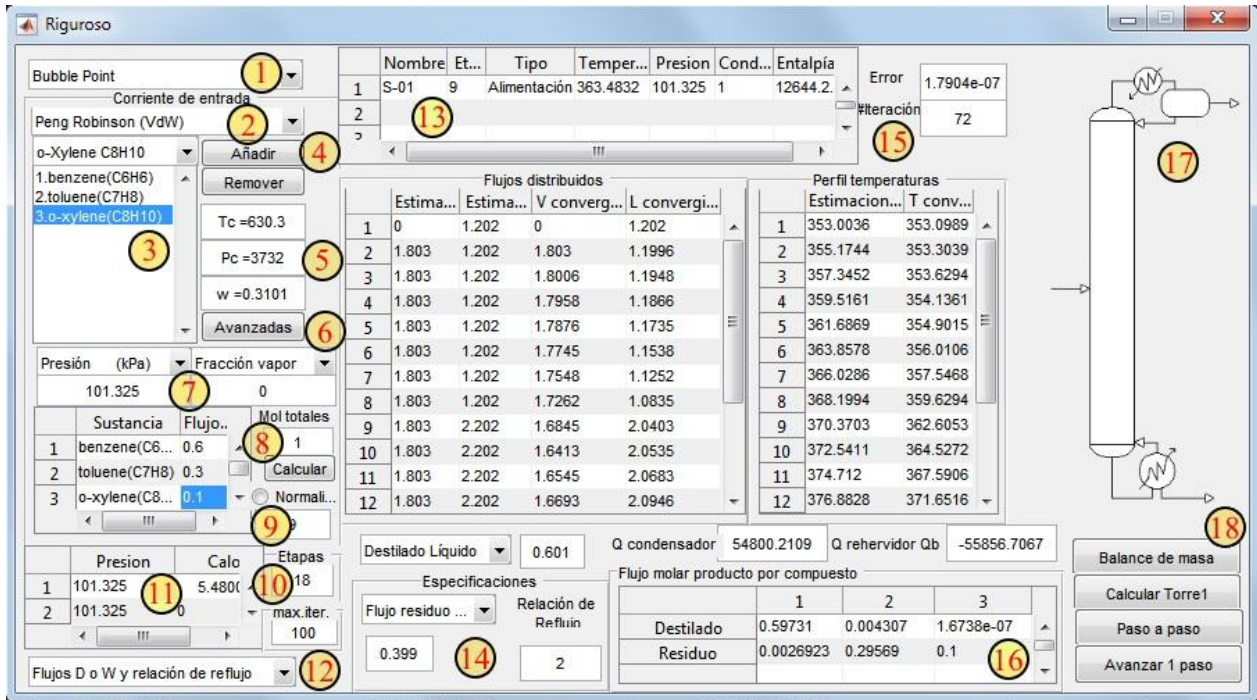


Figura 3.6 Interfaz gráfica de usuario de DESTILMATLABUSB

## **CAPÍTULO 3**

### **RESULTADOS Y DISCUSIÓN**

La simulación de métodos de cálculo de destilación para separación de mezclas multicomponentes puede ser vista como un proceso que involucra varias etapas, desde establecer las características de la separación deseada: estimar cuáles productos y en qué pureza son requeridos, luego fijar un conjunto de variables mínimas con las cuales se cumpla con los grados de libertad y sea posible especificar el problema, pasando por la obtención de estimaciones de otros conjuntos de variables que permitan la utilización de métodos más avanzados de cálculo o que permitan realizar correcciones a las estimaciones anteriores, hasta el uso de los métodos rigurosos de cálculo etapa por etapa para la resolución simultánea de las ecuaciones de balances de masa y energía.

#### **4.1 Planificación y desarrollo de los métodos de cálculo**

Para este proyecto se disponía como recursos en Matlab® a los métodos de cálculo desarrollados por Asuaje (2009) y Majzoub (2011) enfocados en resolución de métodos de destilación binaria por los algoritmos gráficos de McCabe y Thiele y de Ponchon-Savarit, así como el trabajo de Porras (2012) de desarrollo de un programa de cálculo multicomponente por métodos cortos de Fenske, Underwood y Gilliland.

Al evaluar los recursos mencionados, se consiguió que buena parte de aquellos algoritmos han comenzado a fallar por diferencias en nuevas versiones de Matlab® que convirtieron algunas funciones en incompatibles con las versiones anteriores. Además, cada uno de los programas hace uso de un sistema termodinámico implementado en su propia manera, e incluso algunos de los cuales están restringidos al uso de uno único entre la variedad de métodos termodinámicos, por lo cual se optó por construir desde los cimientos el conjunto de algoritmos que servirían para alimentar a los métodos de cálculo más avanzados.

Al buscar en la literatura entre las diversas formas de implementación de métodos termodinámicos, una de las más completas recopilaciones de cálculo de distintas ecuaciones de estado es la de Martín *et al.* (2011). En ella se implementa un conjunto de métodos de cálculo de equilibrio de fases y de propiedades de mezclado para un conjunto numeroso de ecuaciones de

estado, haciendo uso de un enfoque de programación orientado a objetos. Aunque se inspiró gran parte de los algoritmos de cálculo en su trabajo, no se hizo uso directo de esta plataforma de código abierto por varias razones.

En primer lugar se consiguió que una buena parte de los problemas de incompatibilidades introducidas por las nuevas versiones de Matlab® también afectaban a sus algoritmos. En segundo lugar una parte de las incompatibilidades provenían del hecho de haber establecido un enfoque orientado a objetos previo a la implementación oficial de este soporte en el entorno de programación y debido a que gran parte de los cambios entre las versiones recientes se han enfocado en expandir las capacidades en este respecto, la mayor probabilidad de supervivencia en cuanto a compatibilidad futura se obtendría de utilizar las capacidades intrínsecas implementadas, como la definición de clases mediante la redacción “classdef”. Como tercera y última consideración se hizo evidente que el mecanismo de definición de ecuaciones cúbicas como objetos propios no era tan satisfactorio como el uso de los desarrollos genéricos de ecuaciones cúbicas (ecuaciones 5-10) y podía mejorarse para la comodidad del usuario si se implementaban bajo un conjunto de métodos de cálculo genéricos heredadas a todas las ecuaciones derivadas, de forma que únicamente con implementar un conjunto de nuevos parámetros genéricos se tuviese una ecuación cúbica relacionada derivada.

Los algoritmos de código abierto de Martín (2011) no incluyeron una ecuación cúbica derivada por Gasem (2001) y que cambia drásticamente la forma del parámetro de atracción, su inclusión a este proyecto conjunto a los métodos clásicos de Soave (1972) y de Peng-Robinson (1976) sirvió para probar que los desarrollos genéricos de la forma cúbica y el cálculo de propiedades residuales presentados por Figueira (2005 y 2010) funcionaban para derivados, incluso teniendo estas formas distintas para las ecuaciones de los parámetros internos.

Luego se decidió que sería necesario proveer a los métodos rigurosos un conjunto de estimados basados en métodos cortos, así se estableció primeramente como objetivo el uso de métodos de simulación corta o aproximada. Para los cuales se utilizaron los de FUG y el de Hengstebeck.

La programación de los métodos cortos se enfocó en el objetivo de proporcionar flexibilidad en la obtención de estimados para diferentes conjuntos de variables de especificación, de forma que pudiera utilizarse como base para la resolución con métodos rigurosos de columnas complejas multicomponentes.

Para ello se estableció un sistema interno a las torres cortas para poder ir descontando los grados de libertad, para las cuales al momento de obtener el conjunto suficiente de datos procediera automáticamente a realizar su solución por el mecanismo correspondiente.

Sin embargo, los métodos cortos clásicos únicamente son únicamente útiles para la resolución de columnas simples (Navarro, 2003). Un método derivado de los métodos clásicos fue desarrollado por Youseff *et al.* (1989), aunque una de sus debilidades es que necesita sobre-especificar los grados de libertad (Navarro, 2003) por lo cual no es particularmente útil para propósitos pedagógicos.

Adicionalmente a ello, al implementar los métodos cortos se evidenció que la velocidad de corrida en el lenguaje interpretado del entorno Matlab<sup>®</sup> introduce fuertes retrasos al ser utilizado para programación orientada a objetos y un conjunto de primeros intentos produjo tiempos de corrida para torres simples y de pocas etapas muy elevados, así que se optó por desvincular el cálculo previo de métodos simples de la resolución de métodos rigurosos.

En lugar de generar estimados mediante las columnas simples, los métodos rigurosos realizan una distribución ideal de los componentes etapa por etapa, en la cual todo el contenido introducido de los componentes livianos es recuperado en el tope hasta cumplir con el flujo requerido de destilado, o cumplir el balance de masa en el caso de recibir el flujo de fondo como requerimiento. Los componentes y flujos restantes son distribuidos idealmente al fondo. Posteriormente a partir de la distribución supuesta se genera un perfil de temperatura a partir de resolver la temperatura del punto de burbuja en el rehervidor y la de rocío o burbuja en el condensador y se realiza la distribución linealmente de las composiciones, además de flujos molares constantes entre el grupo de etapas intermedias.

Para demostrar el funcionamiento de los algoritmos de cálculo fueron utilizadas todas las clases de columnas de destilación para resolver el siguiente ejemplo tomado de Robinson y Gilliland (1950).

## 4.2 Ejemplo 1

En la Fig. 4.1 se muestran los datos del problema y a continuación se explican los datos obtenidos por medio de los distintos métodos de cálculo aproximado y riguroso, así como también los resultados del programa de simulación comercial PRO/II para comparación. En el apéndice B

se muestra una guía ilustrada de cómo introducir paso a paso los datos para resolver el ejemplo y obtener los resultados utilizando la interfaz gráfica y en el Apéndice C se muestra el archivo de entrada de PRO/II.

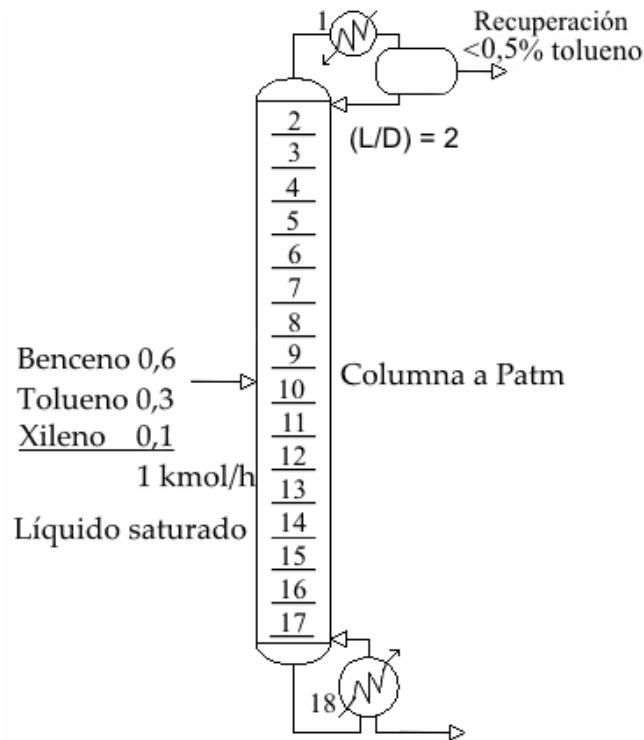


Figura 4.1. Datos del ejemplo 1

En la Tabla 4.1 se presentan los resultados de la distribución de los componentes para los productos de residuo y de tope. Es de notar que los algoritmos de Hengstebeck y de Lewis-Matheson de DESTILMATLABUSB utilizan como distribución hacia los productos los resultados del algoritmo FUG, por lo cual el resultado de estos fue igual al de *Shortcut*.

Para lograr la convergencia del método de Newton Raphson se tuvo que utilizar aparte del conjunto de variables de diseño un estimado inicial de la distribución de los componentes al producto de fondo, que fue dada por el resultado de la torre de métodos cortos, dado que suponiendo una distribución trivial el procedimiento de cálculo encuentra una matriz mal escalada o mal condicionada que imposibilita la convergencia. Ante este escenario, por lo regular realizando una disminución del factor de amortiguamiento de forma que los cambios en las variables de estado no



sean muy pronunciados y se favorezca la estabilidad del algoritmo. En el caso actual, se utilizó un factor de amortiguamiento (*damping*) de 0,65.

Tabla 4.1 Comparación de la distribución de los componentes para el ejemplo 1 entre los diferentes métodos del ejemplo 1

Corriente	Compuesto	Fracción molar			Inside Out PRO/II
		<i>Shortcut</i>	BubblePoint	Newton Raphson	
Destilado	Benceno	0,9970	0,9928	0,9934	0,993
	Tolueno	0,0030	0,0072	0,0066	0,007
	o-xileno	0,0000	0,0000	0,0000	0,000
	Total	0,5988	0,6010	0,6010	0,601
Fondo	Benceno	0,0011	0,0067	0,0074	0,008
	Tolueno	0,1343	0,7422	0,7420	0,742
	o-xileno	0,8646	0,2511	0,2506	0,250
	Total	0,4012	0,3990	0,3990	0,399

Se puede observar que las distribuciones entre los productos del método de Punto de Burbuja y de Newton Raphson son cercanas entre sí, al mismo tiempo que el método FUG obtiene un buen estimado inicial, lo bastante cercano para facilitar la convergencia de Newton Raphson al ser utilizado como estimado inicial. Además, la distribución calculada por el simulador comercial PRO/II es hasta la precisión reportada, la misma que el del resultado de los métodos rigurosos de DESTILMATLABUSB, por lo cual se puede considerar que los resultados de los módulos de cálculos son correctos en cuanto a la distribución de los componentes en cada una de las corrientes del ejemplo.

El número de etapas mínimo calculado por el método de Fenske fue de 12 y el reflujo mínimo estimado fue de 1,09 moles de líquido por cada mol de flujo retirado en el destilado. Con la aplicación de la ecuación de Molokanov (1972) para el ajuste de Gilliland, el método *Shortcut* realiza el cálculo de 20 etapas teóricas. Estos resultados pueden ser comparados con el módulo de resolución por el método de Hengstebeck, el cual arrojó como resultado 12 etapas mínimas, como se muestra en la Fig. 4.2, además de una condición mínima de reflujo de 1,0249.

Mediante el método de Hengstebeck se realizó el trazado a un reflujo de operación igual a 2, con lo que se estimó un número de 18 etapas, como se representa en el diagrama de McCabe y

Thiele obtenido en la Fig. 4.3. Se puede así determinar que el número de etapas teóricas sea cercano y es esperable que al tratarse de un método corto resulte una sobreestimación que sea corregida por métodos más precisos, como pudiera ser el método semiriguroso de Lewis y Matheson, que representa para Treybal (1982) el último estimado requerido para el número de etapas.

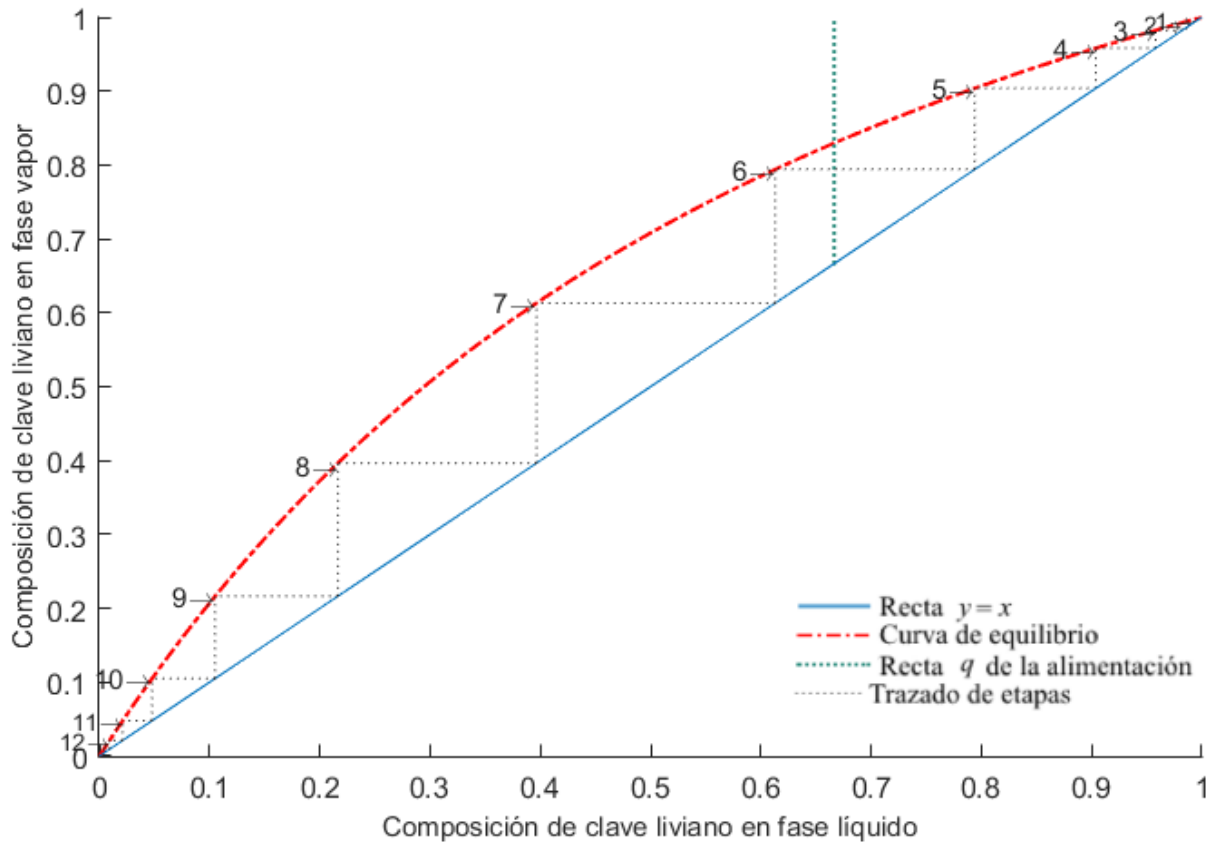


Figura 4.2 Trazado del número mínimo de etapas para el ejemplo por el método de Hengstebeck para aproximación pseudobinaria a McCabe-Thiele

Debido a que la resolución de un método riguroso viene condicionada directamente a la selección de antemano del número de etapas teóricas como especificación, se debe obtener con métodos previos un estimado cercano. Con el método de Lewis Matheson se corrigió el estimado inicial de 20 etapas teóricas que dio el método de Gilliland, obteniéndose como resultado 18 etapas incluyendo al condensador y el rehervidor, en concordancia con los resultados de Robinson y Gilliland (1950) que reportan que la separación puede lograrse con 16 platos. El output de texto generado por el cálculo de distribución del algoritmo de Lewis-Matheson se presenta en el Apéndice D, en el cual se muestran etapa a etapa las iteraciones. Según este método se alcanza el

plato de alimentación en la etapa 13 desde el tope, sin embargo el criterio del método para la selección se encuentra basado en la similitud en composición entre el plato y la alimentación, similarmente al método gráfico de McCabe y Thiele, en lugar del criterio de optimización térmica, por lo que es de esperarse que los calores obtenidos por Lewis-Matheson de  $Q_c = 5,494 \cdot 10^4$  kJ/h y de  $Q_b = 7,697 \cdot 10^4$  kJ/h sean afinados por los métodos rigurosos

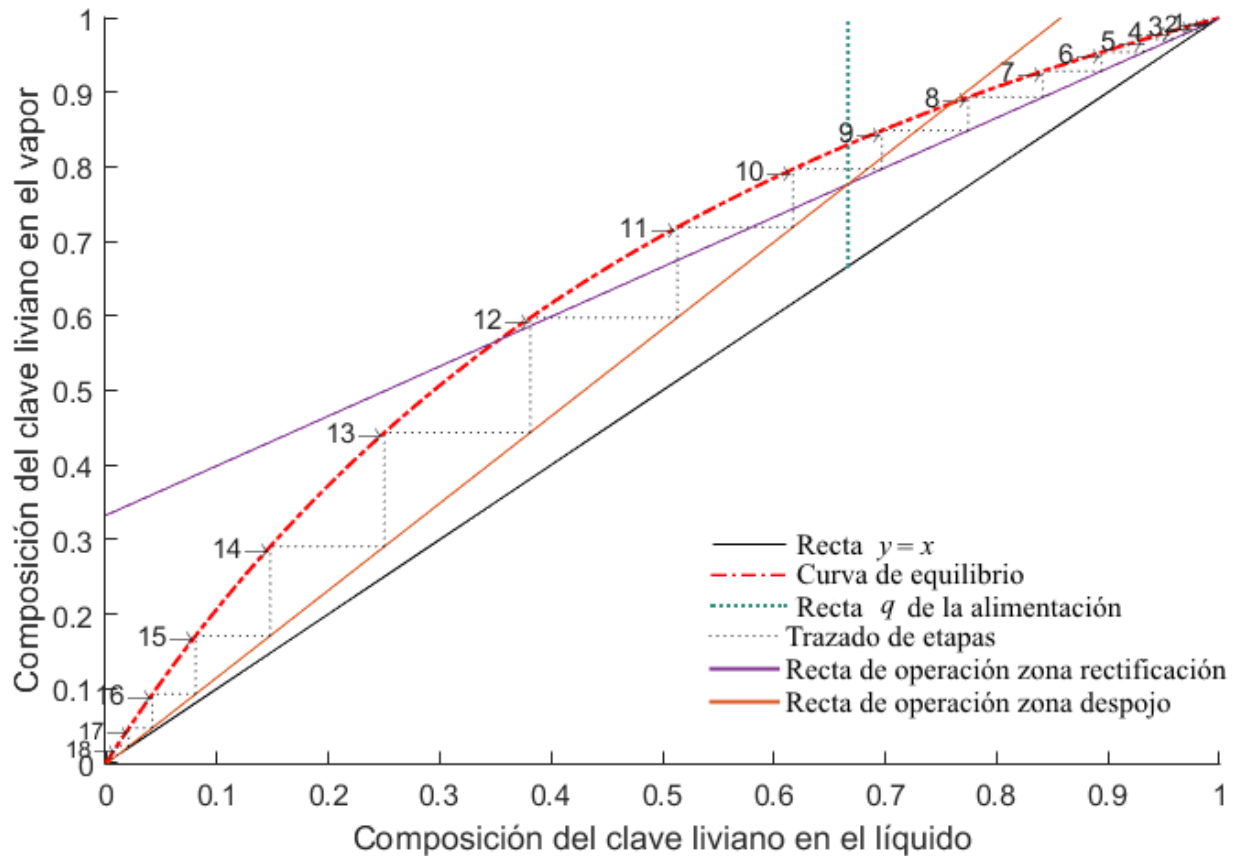


Figura 4.3 Resultado del método de Hengstebeck a reflujo real ( $L/D = 2$ )

A partir de la corrección obtenida que indica un número suficiente de 18 etapas teóricas, se puede proceder a utilizar los métodos rigurosos. Para el método de punto de burbuja se requiere como dato la relación de reflujo de 2 usada en Lewis-Matheson y también un flujo total de destilado o de fondo. Debido a que la relación de reflujo puede variar entre el mínimo (calculado en 1,08) e infinito, se supone que hay parecido con la condición de Underwood de reflujo mínimo (Shiras, 1950), para la cual fue calculada un flujo de destilado de 0,601, que se introdujo como dato en los métodos rigurosos de BubblePoint y NaphtaliSandholm y en el de *Inside-Out* en el simulador PRO/II®.

En la Fig. 4.4 se muestra el resultado de los perfiles de flujos molares de vapor en correspondencia con cada etapa.

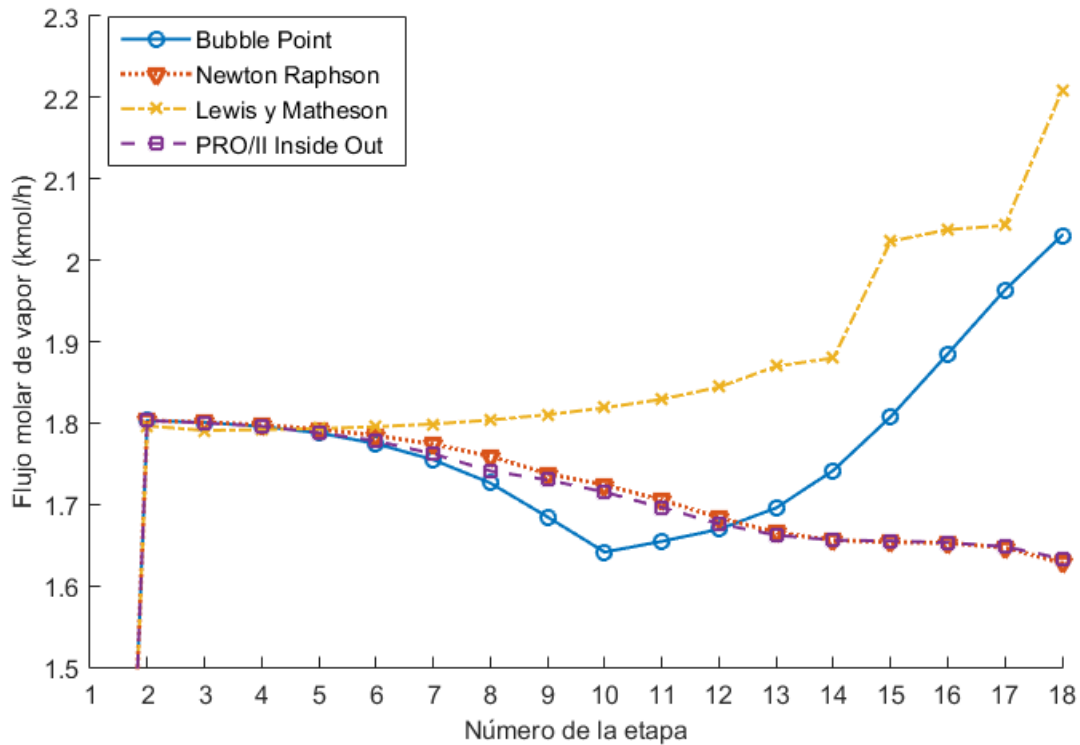


Figura 4.4 Flujos molares de vapor para los algoritmos de LM, BP, NR e *Inside-Out* de PRO/II®

En el caso del método de *Bubble Point* y de Lewis-Matheson se observa una tendencia similar a obtener perfiles elevados de vapor para las etapas inferiores, lo que viene de la naturaleza de dichos métodos a depender exclusivamente de la convergencia de los balances de masa, para los cuales es independiente la consistencia de los balances de energía, que corrige tal que en etapas en donde no se localiza alguna alimentación todo el calor cedido por la fase vapor ascendente de la etapa siguiente debe ser retenido por las fases líquidas y el vapor que sale a la etapa superior. Al ser la fase de vapor una de mucho mayor almacenamiento energético que la de líquido, en el momento de plantear las ecuaciones H con base en los resultados convergidos por *Bubble Point*, varias de las etapas de la zona de despojo de creciente flujo de vapor no son cero, por lo cual debe disminuir el flujo de vapor en esa zona al plantear el sistema completo de ecuaciones MESH y converger en conjunto con el balance de energía, que es por lo que se observa la disminución del

flujo de vapor en los métodos de Newton Raphson y de *Inside-Out* en PRO/II que lo toman en consideración.

Además, por la similitud importante al hacer la comparación entre *Inside-Out* con el algoritmo programado para Newton Raphson, se puede determinar que el método fue bien implementado y el criterio para la tolerancia que, en este trabajo, se elevó en cuatro órdenes de magnitud sobre la recomendación de Henley y Seader, es suficiente para los propósitos pedagógicos que persigue este proyecto.

En la Fig. 4.5 se presenta la comparación entre los perfiles de líquido plato por plato de los distintos algoritmos de cálculo.

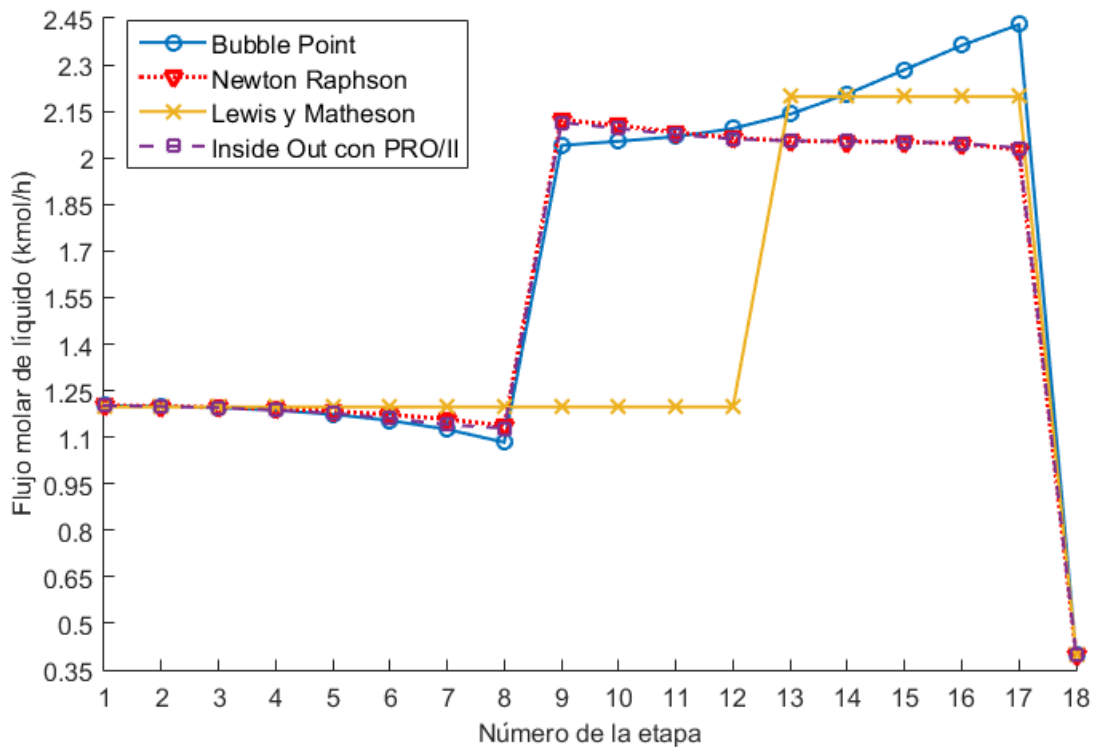


Figura 4.5 Flujos molares de líquido para los algoritmos de LM, BP, NR e *Inside-Out* de PRO/II®

En dicho gráfico la diferencia más resaltante se corresponde con la diferencia en el plato de alimentación que propone el criterio para su ubicación de Lewis-Matheson, que lo ubica en la etapa 13 que es aquella donde los balances de Lewis alcanzan el punto en que la relación de composición entre los claves es similar a la composición. Este método semirriguroso frecuentemente alcanza

puntos de cambio infinitesimal en la relación de composiciones entre los claves, por lo que cuando se tiende a estos estima un número de platos superior.

Entre los métodos de Newton Raphson y de *Inside-Out* la diferencia entre los flujos de líquido es despreciable, validando los resultados obtenidos.

En la Fig. 4.6 se observa el perfil de temperaturas por etapas para los resultados en los métodos de cálculo y en la Fig. 4.7 el perfil de composiciones por cada componente de la mezcla en cada una de las etapas teóricas.

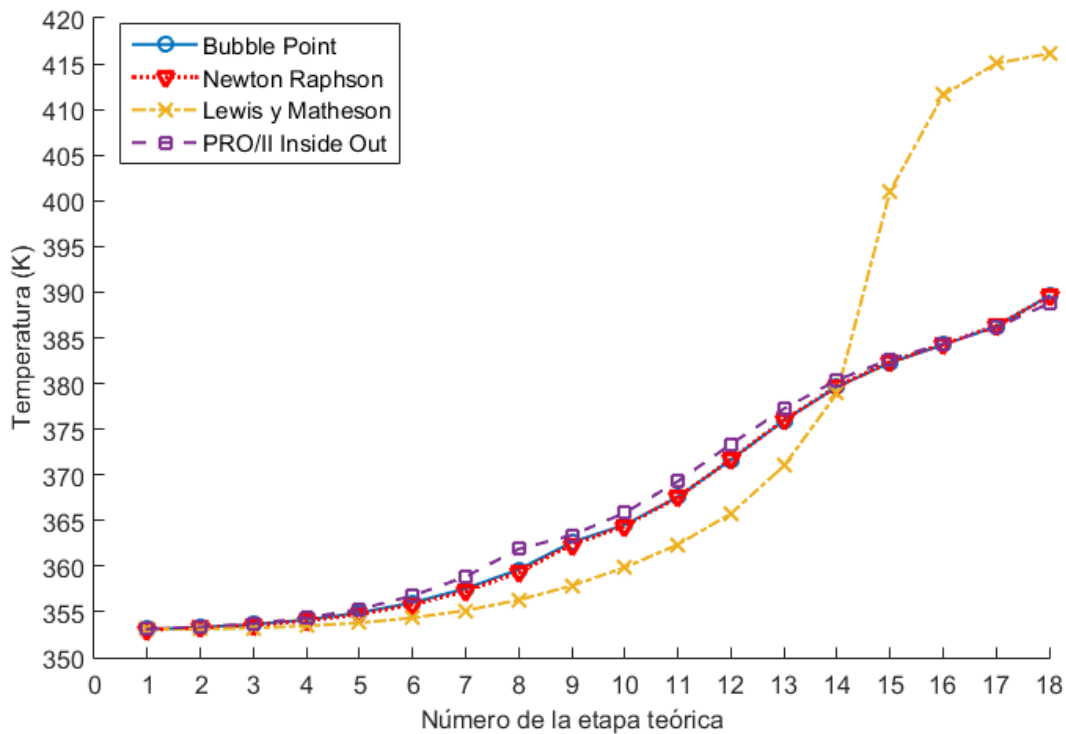


Figura 4.6 Perfil de temperaturas por etapa teórica para los algoritmos de LM, BP, NR y de *Inside-Out* de Pro/II®

El perfil de temperaturas corrobora la mayor debilidad del método de Lewis-Matheson, al comenzar con una suposición de nulo flujo de componentes no claves más pesados que el HK en el producto de tope, la composición de este es cero hasta la etapa de alimentación y alcanzar la etapa de alimentación por el criterio de relación se dificulta por la cercanía en volatilidades de los compuestos.

En los métodos rigurosos la composición del compuesto más pesado, el xileno, se mantiene aproximadamente constante durante toda la zona de despojo, en cambio para el tolueno y el benceno se evidencia un rápido incremento y disminución de su fracción molar con cada etapa teórica, respectivamente, tanto para el método de punto de burbuja como el de Newton Raphson.

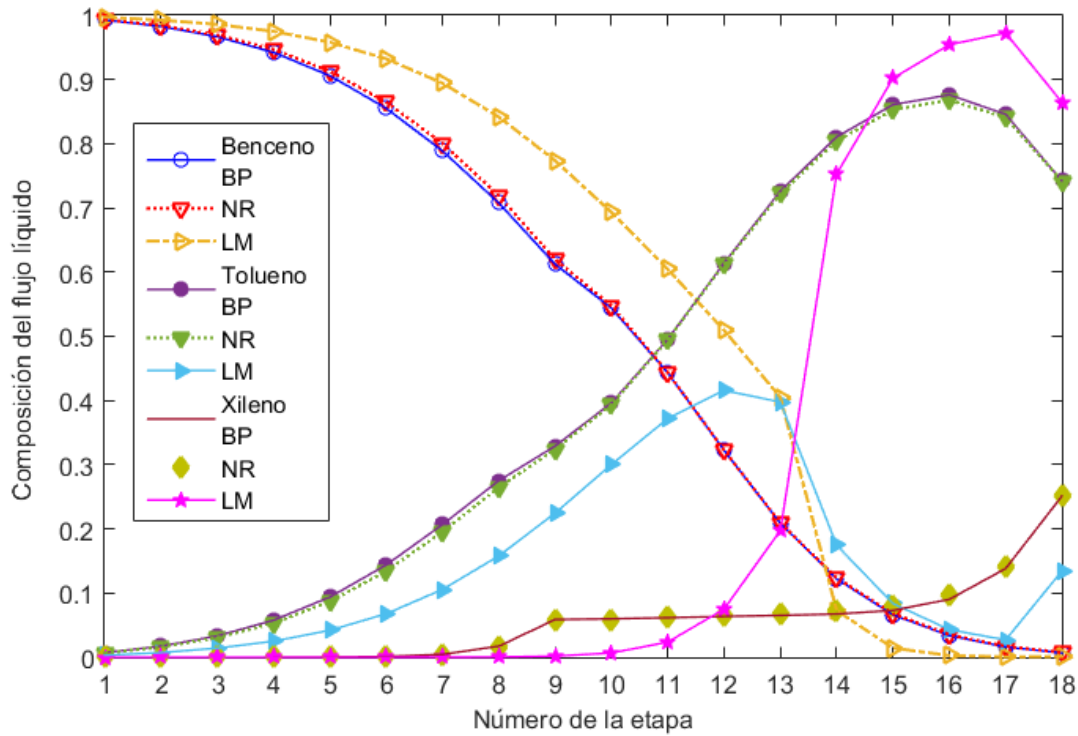


Figura 4.7 Perfil de composiciones para cada uno de los componentes por etapa teórica calculado con los algoritmos de LM, BP y NR

Se esperaba observar una discrepancia entre los perfiles de líquido y de vapor que reportaron los resultados de Navarro (2003), con diferencias de hasta un 40%, sin embargo los métodos de DESTILMATLABUSB son acordes a los del simulador comercial. Esto pudiera deberse a que en los datos de ingreso al modelo termodinámico del simulador para realizar la comparación se cambió el método por defecto de cálculo de la densidad de líquido, que suele ser configurada automáticamente a API, por el método de Peng Robinson para realizarlo en las mismas condiciones del algoritmo siendo probado, haciendo a ambos modelos termodinámicos exactamente iguales.

En la Tabla 4.2 se presentan los resultados de los cálculos de los calores del rehervidor y del condensador para los métodos rigurosos, así como también el resultado con el simulador.

Tabla 4.2 Calor del condensador y del rehervidor de la columna obtenido por los métodos rigurosos de *Bubble Point*, Newton Raphson y usando el simulador comercial PRO/II

Método	BubblePoint	Newton Raphson	<i>Inside-Out</i>
Calor del condensador $Q_c$ ( $\times 10^4$ kJ/h)	5,48	5,48	5,48
Calor del rehervidor $Q_b$ ( $\times 10^4$ kJ/h)	5,59	6,15	5,58

Se observa nuevamente cercana correspondencia entre los métodos, como se podía esperar dado que se están calculando calores a un grupo de variables de proceso similares como son los flujos molares, perfiles de temperatura y composiciones que se vieron entre si en las figuras anteriores.

El módulo de *Bubble Point* requirió un total de 72 iteraciones para su convergencia, mientras que el módulo de Newton Raphson, al cual se le dio inicialmente un estimado de la distribución cercano, requirió únicamente 37 iteraciones habiendo reducido el factor de amortiguamiento a 0,65, por lo cual pudiera ser menor de utilizar mejores estimados iniciales.

## 4.2 Ejemplo 2

Con el propósito de demostrar las capacidades de cálculo avanzadas de los algoritmos de cálculo rigurosos *BubblePoint* y Naphtali-Sandholm programados, se comparan los resultados obtenidos por ellos para el caso de una columna con cinco componentes y un par de alimentaciones, como se muestra en la Fig. 4.8 y en la Tabla. 4.3

En este ejemplo se desea utilizar una misma columna para la separación de crudos adicionales, por lo cual se considera utilizar una primera corriente de alimentación ‘S-01’ y también



separar los compuestos de una segunda corriente de alimentación ‘S-02’ en un equipo en cascada existente de 9 etapas teóricas, incluyendo el rehervidor y el condensador.

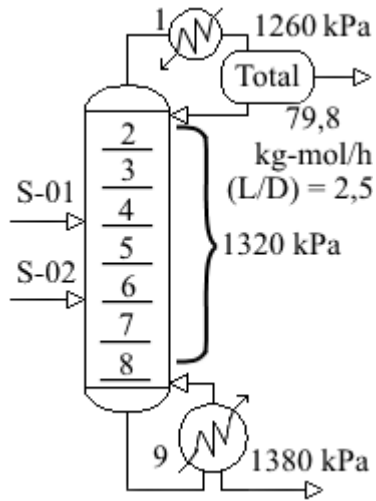


Figura 4.8. Esquema y parámetros de diseño del ejemplo 2

Tabla 4.3. Condiciones de las corrientes de alimentación a separar.

Corriente	Componentes	Fracción Molar	Condiciones
S-01	Etano	0,5342	P = 1320 kPa
	Propano	0,3163	Punto de
	Butano	0,0612	burbuja
	Pentano	0,0701	Flujo
	Hexano	0,0183	154,3 kg-mol/h
S-02	Etano	0,2236	P = 1320kPa
	Propano	0,5194	Punto de
	Butano	0,1119	Burbuja
	Pentano	0,1230	Flujo
	Hexano	0,0221	64,02 kg-mol/h

A partir de los datos de la Fig. 4.8 se realizó el modelaje con los métodos de cálculo rigurosos programados. Para el caso del método *BubblePoint* se pudo contar con la convergencia en 15 iteraciones a partir de una suposición inicial de distribución ideal, con la cual el etano fue supuesto el único componente presente en el destilado. A partir de la distribución ideal se pudo suponer perfiles de temperatura, flujos molares de vapor y composiciones lineales con las cuales se empezó la primera iteración.

El algoritmo de Newton Raphson es mucho más sensible a la convergencia, y su dependencia de la estimación inicial es importante. Debido a que la suposición de distribución ideal no alcanzó para obtener resultados dentro del margen de tolerancia, se disminuyó el factor de amortiguamiento a 0,45 y se suministraron como estimados los perfiles de líquido, vapor, temperatura y composiciones obtenidos con el método de *BubblePoint*. El algoritmo puede mejorarse y automatizarse acoplándolo a un módulo de generación de estimados iniciales de flujos molares interetapas por métodos cortos, como pudiera ser el propuesto por Youseff *et al.* (1989).

El *script* con el cual se generan los resultados del ejemplo 2, que se muestran en la Tabla 4.4, se muestra en el Apéndice E, el cual se encuentra comentado para el mejor entendimiento de cada paso.

Tabla 4.4. Resultados de la simulación con los métodos rigurosos de *BubblePoint*, Newton Raphson y los obtenidos con el uso de *Inside-Out* en un simulador comercial

Corriente	Compuesto	Fracción molar		
		<i>BubblePoint</i>	Newton Raphson	<i>Inside Out</i> PRO/II
Destilado	Etano	0,990	0,989	0,989
	Propano	0,010	0,011	0,011
	Butano	0,000	0,000	0,000
	Pentano	0,000	0,000	0,000
	Hexano	0,000	0,000	0,000
	Flujo total (kg-mol/h)	79,8	79,8	79,8
Fondo	Etano	0,128	0,129	0,128
	Propano	0,587	0,586	0,586
	Butano	0,120	0,120	0,120
	Pentano	0,135	0,135	0,135
	Hexano	0,031	0,031	0,031
	Flujo total (kg-mol/h)	138,5	138,5	138,5

En la predicción de las composiciones de los productos, ambos métodos muestran concordancia de un error máximo de 0,1% en la composición de etano y de 0% en la del resto de los componentes, así como el error con respecto al simulador comercial que es de la misma magnitud. El simulador comercial se introduce como una referencia útil, cuyo objetivo es tener

contra quién comparar la convergencia de los resultados, a pesar que la validez de los modelos, que ya ha sido estudiada, debe realizarse comparando contra una columna real de operación.

En la Fig. 4.9 se muestra la comparación de los perfiles de temperatura entre los modelos de punto de burbuja, el método de Newton Raphson de Naphtali-Sandholm y el método de *Inside-Out* de PRO/II. Se puede observar que la diferencia entre el método de *Bubble-Point* y el de Newton Raphson está entre el 0,1% y un máximo de 1,5%, mientras que al mismo tiempo ambos resultados están en el rango de tolerancia reportado por el simulador, con hasta una diferencia de 0,8% entre los métodos programados y el simulador.

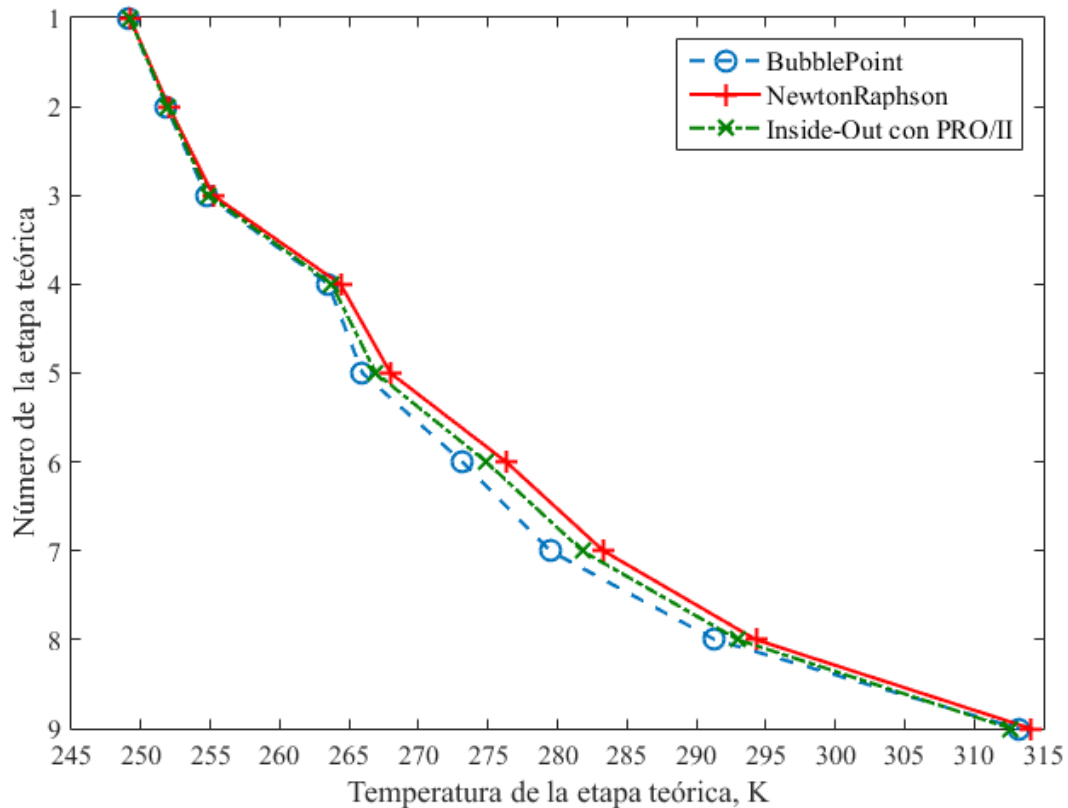


Figura 4.9 Perfil de temperaturas en cada etapa teórica para los métodos de *BubblePoint*, Newton-Raphson de Naphtali-Sandholm y el de *Inside-Out*

En la Fig. 4.10 se observa la diferencia entre los perfiles de flujo de vapor entre los modelos rigurosos de *BP* y *NR*, que se encuentran entre un mínimo de 0,1% y un máximo de 33%. Aquí resalta la vinculación existente entre los métodos de Newton-Raphson y el método de *Inside-Out*, que consiste en que la mayoría de las ocasiones el método riguroso de corrección simultánea NR conforma el algoritmo interno de convergencia, con lo que no es extraño ver que el error máximo

en que difiera el método programado con el simulador comercial sea en 2%, debido a que ambos realizaron la simulación del conjunto completo de ecuaciones MESH. La mayor parte de la diferencia se puede atribuir a la suposición para la corrida del programa DESTILMATLABUSB que los parámetros de interacción binaria son ideales, en cambio PRO/II selecciona, por defecto, valores de una base de datos.

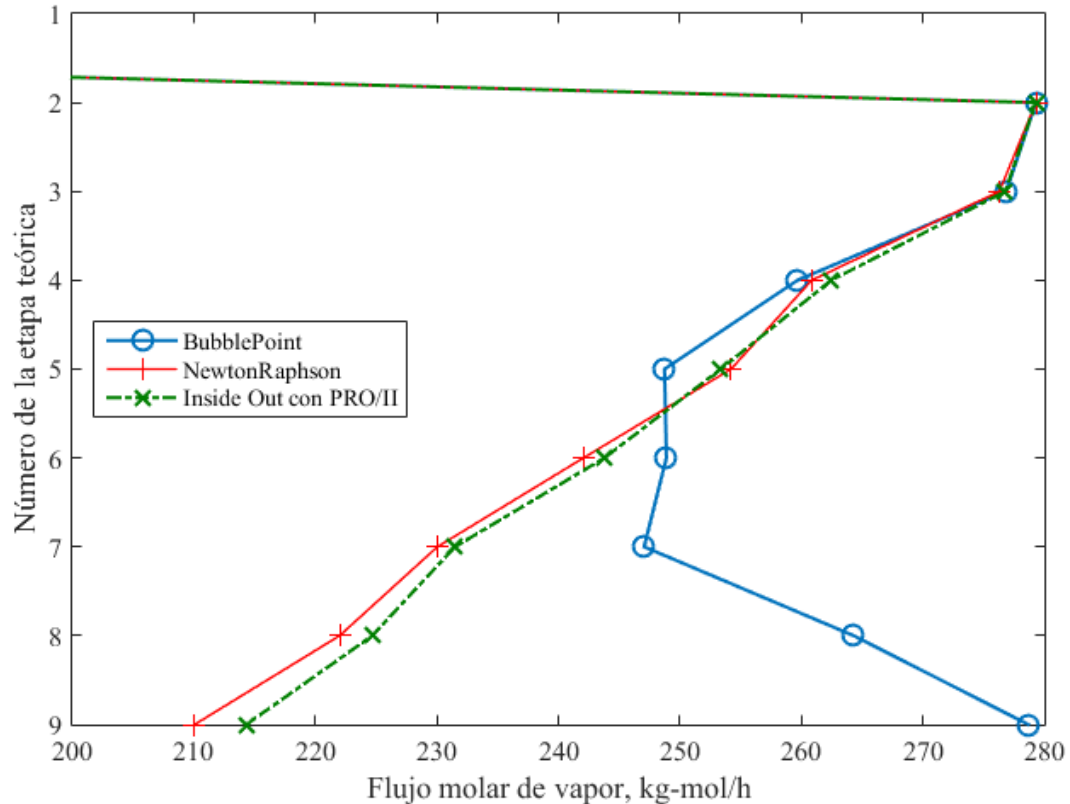


Figura 4.10. Perfil de flujo molar de vapor que abandona cada etapa teórica, según el resultado arrojado por los algoritmos de BP, NR y de *InsideOut*

Se observa similarmente diferencias de hasta un 22% entre los resultados predichos por el algoritmo de *BP* y el de *NR* para los perfiles de líquido interetapas, que se muestran en la Fig. 4.11. En este caso la diferencia máxima entre el simulador comercial y el método riguroso *NR* programado en DESTILMATLABUSB es inferior al 1%, lo que ofrece una buena perspectiva acerca de la comparación que se desea, para corroborar que los cálculos programados estén acordes a la exactitud suministrada por otros métodos que también simulan el conjunto de ecuaciones de energía acopladas con las de materia, en conjunto.

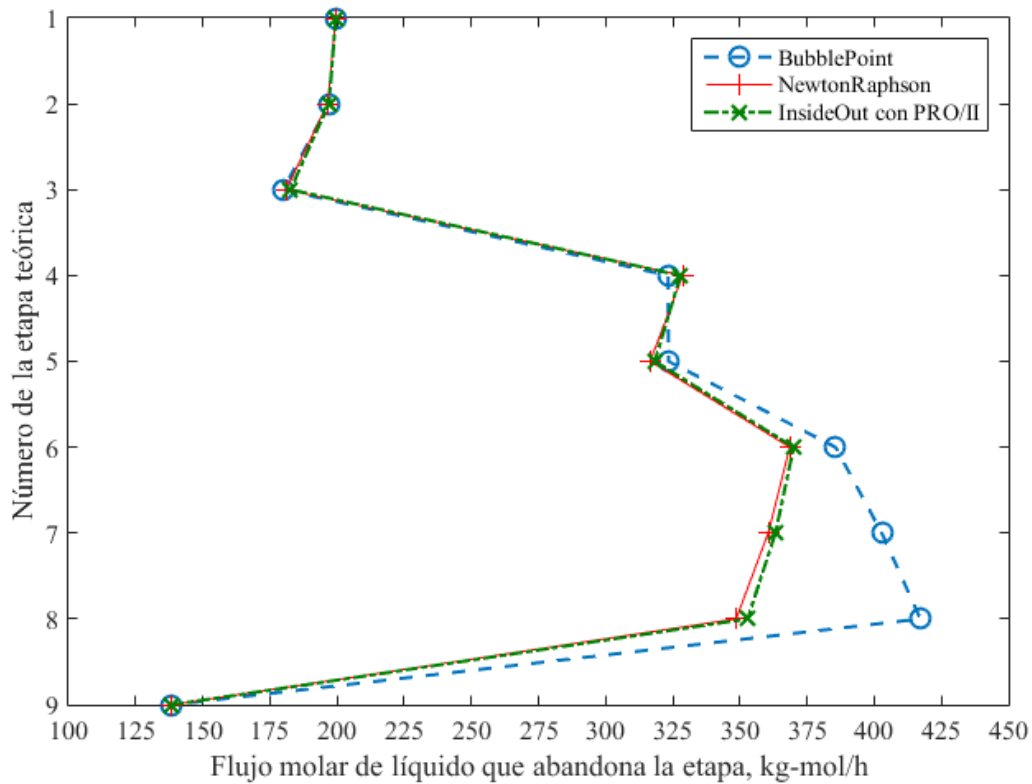


Figura 4.11. Flujo molar de líquido para cada etapa teórica, obtenida mediante los métodos de BP, NR y de IO

La tendencia del método de *BubblePoint* es a realizar la mayoría de los cambios a los perfiles de flujo de vapor, de temperatura y de líquido en las primeras iteraciones, y posteriormente disminuir la tasa de cambios hasta cumplir con el criterio de convergencia escogido basado en el error relativo entre un par de iteraciones consecutivas, así como se muestra en las Figs. 4.12, a 4.14.

Aunque comparando la convergencia de los perfiles de flujos de vapor, líquido y temperatura de cada plato con el algoritmo de Newton Raphson, disponible en las Figs. 4.15 a 4.17, respectivamente, se observa que, incluso utilizando el bajo factor de amortiguamiento de 0,45, el método obtiene una convergencia más rápida que durante las primeras iteraciones que el método de Punto de Burbuja.

El tiempo de corrida del algoritmo de punto de burbuja fue de 30 segundos, mientras que el de Newton Raphson tardó 25 segundos en un procesador de arquitectura AMD Phenom II®.

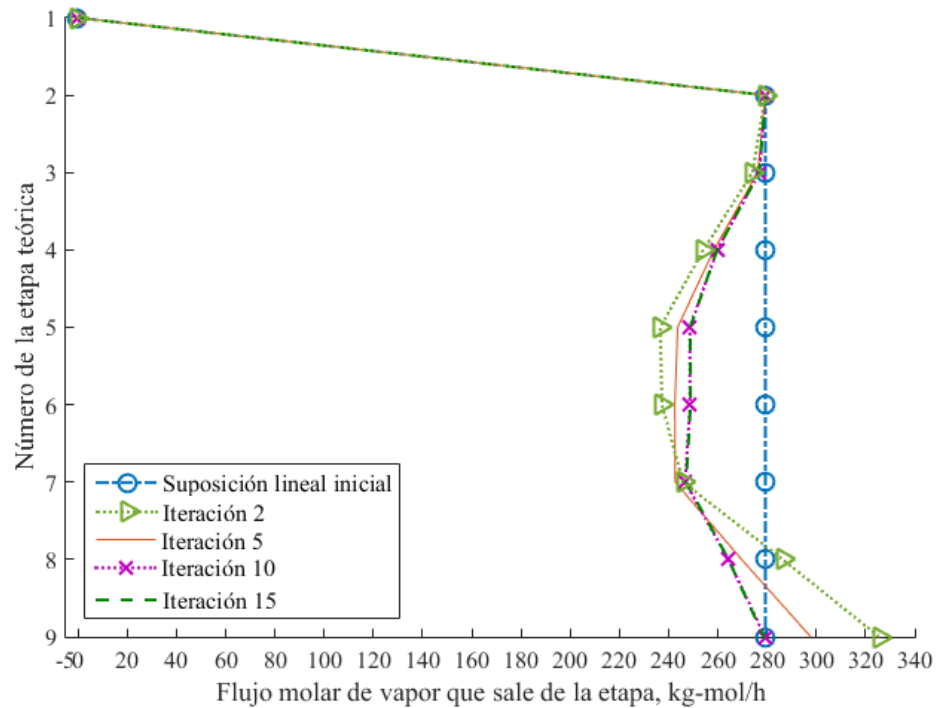


Figura 4.12. Convergencia de los perfiles de flujo molar de vapor según el método de punto de burbuja

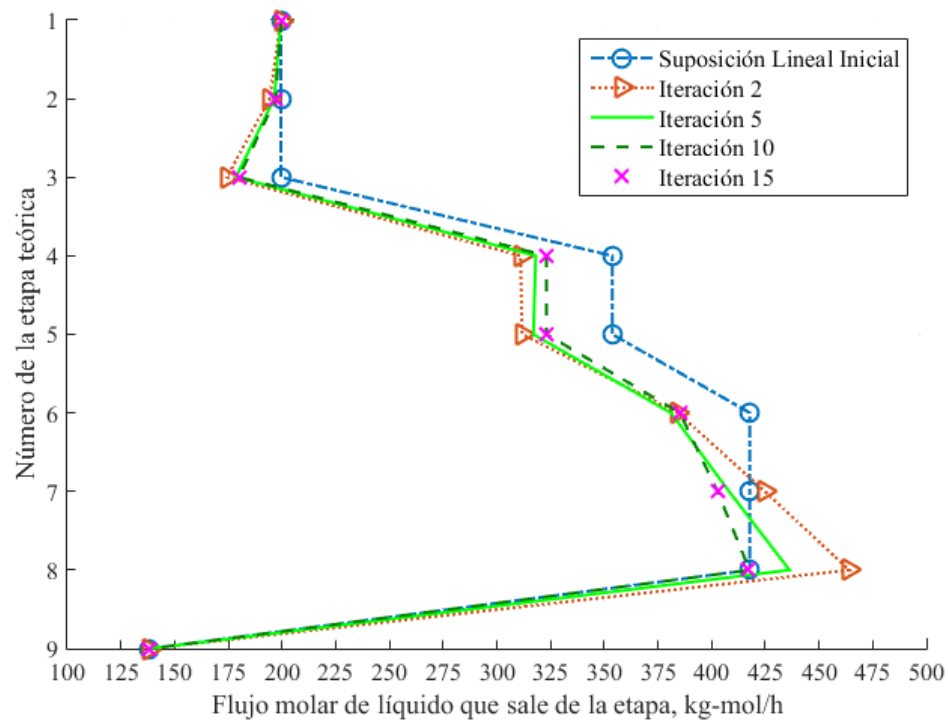


Figura.4.13. Convergencia de los perfiles de flujo molar de líquido utilizando el método de punto de burbuja

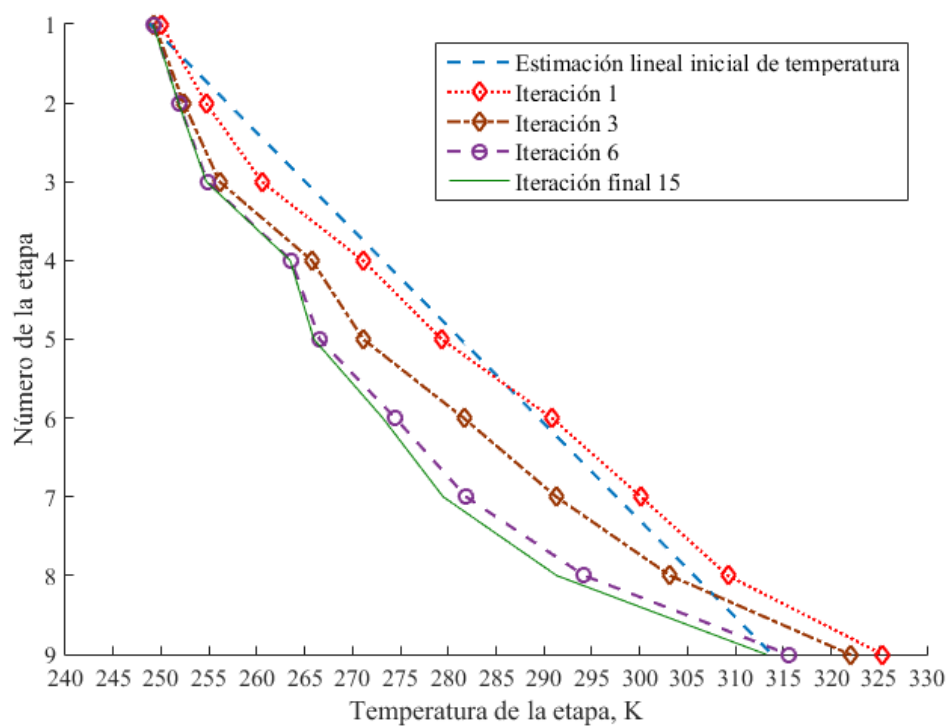


Figura 4.14. Convergencia de los perfiles de temperatura en cada plato utilizando el método *Bubble Point*

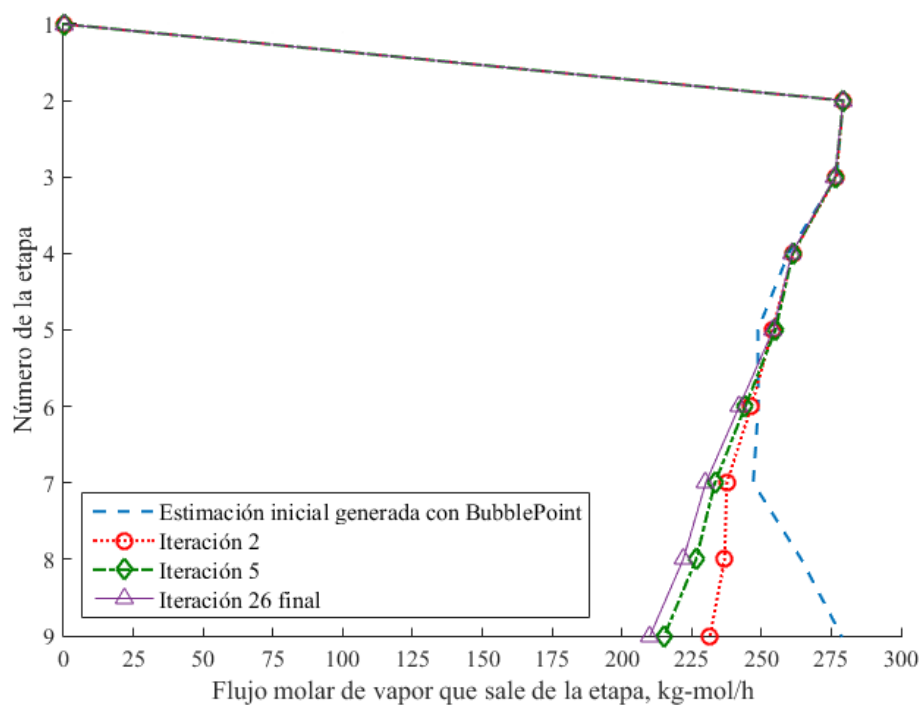


Figura 4.15. Convergencia de los perfiles de flujo de vapor en cada etapa mediante el método de Newton Raphson

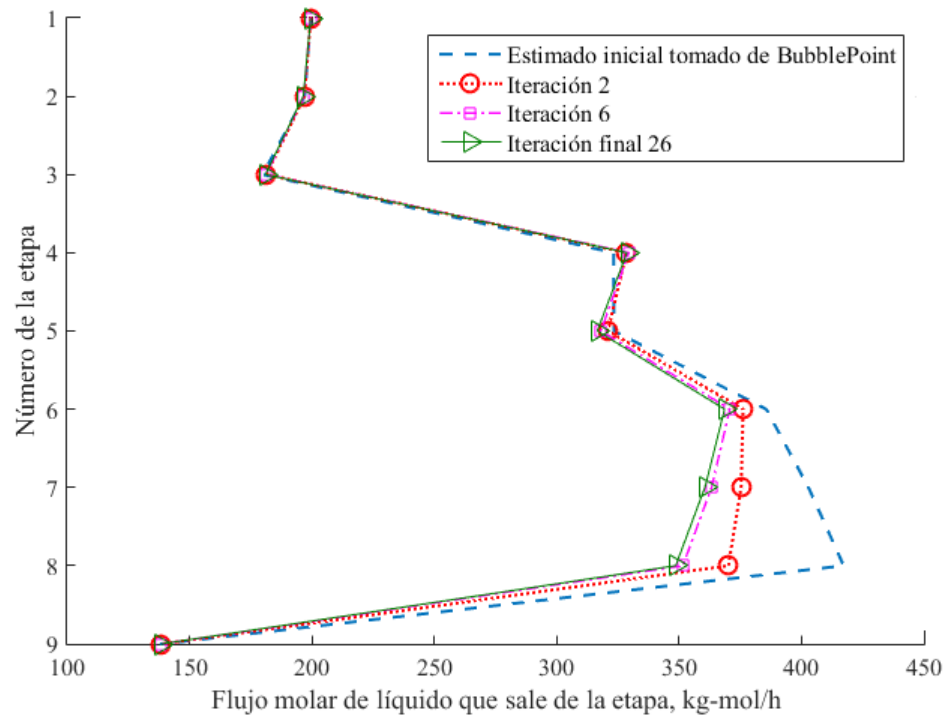


Figura 4.166. Convergencia de los perfiles de flujo de líquido en cada etapa mediante el método de Newton Raphson

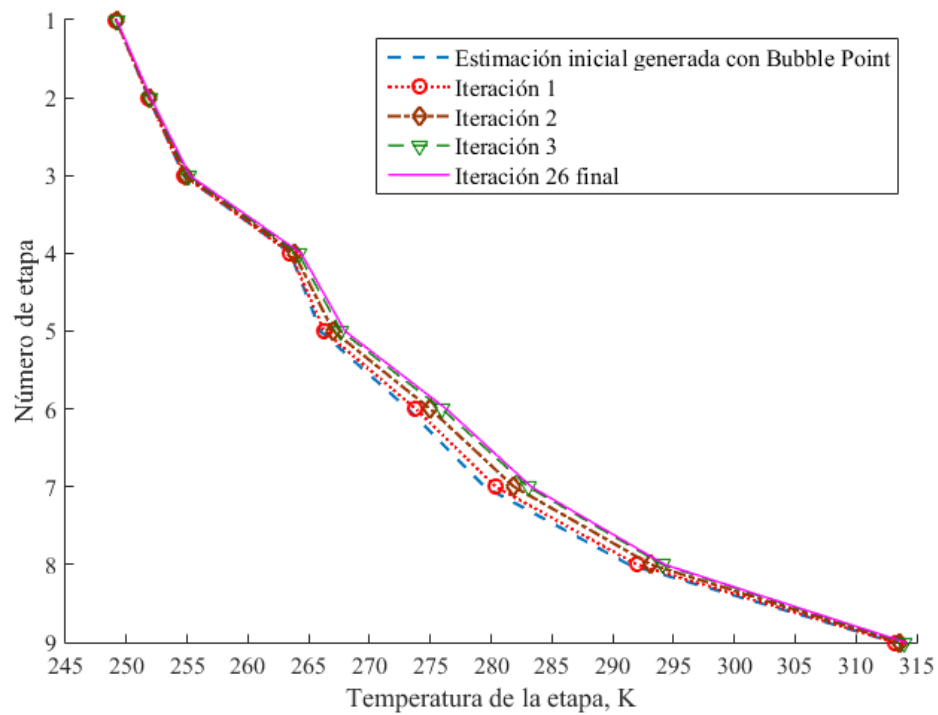


Figura 4.177. Convergencia de los perfiles de temperatura en cada plato utilizando el método Newton Raphson



## CONCLUSIONES Y RECOMENDACIONES

Este trabajo corresponde a un avance en el alcance que tiene MATLAB®, así como cualquier otro lenguaje de programación orientado a objetos que pudiera dictarse en el futuro en el programa de estudios de Ingeniería Química, para el desarrollo de aplicaciones en el área de la termodinámica de código abierto que permitan una mejor enseñanza sobre el funcionamiento en que están construidas las aplicaciones que el ingeniero de procesos utiliza, como lo son los simuladores.

Como resultado de este trabajo se llevó a cabo la programación de cinco módulos de cálculo independientes que simulan unidades de destilación multicomponentes, con los métodos cortos *FUG* y de Hengstebeck, el método semi-riguroso de Lewis y Matheson y los métodos rigurosos de *Bubble Point* y de Newton Raphson, así como una robusta base de datos *DIPPR* de sustancias puras y un módulo termodinámico modular para ecuaciones cúbicas basado en programación orientada a objetos, de código reusable y fácilmente expandible para incorporar un gran número de otras ecuaciones cúbicas aparte de las tres realizadas: Peng y Robinson, Soave y de Gasem.

Los resultados numéricos generados por los módulos mostraron estar en concordancia con los reportados por Robinson y Gilliland (1950) y por el programa de simulación comercial PRO/II.

Una de las futuras implementaciones de proyectos pudiera incorporar una interfaz gráfica de usuario similar a las de entornos de trabajo de simulación con equipos de procesos independientes, de los cuales los cinco módulos de cálculo presentados aquí sean el referente a las unidades de separación por destilación, lo que pudiera ser expandido a bombas, intercambiadores, flujos de procesos, entre otros para orientar hacia la consecución de un entorno de simulación integrado con el cual poder brindar un servicio que sirva como un potencial método de autofinanciamiento.

En lo referido a los métodos de simulación, pudiese ser incorporado un método de cálculos aproximados basados en los algoritmos de Youseff *et al.* (1989) para la simulación de sistemas de torres complejos con salidas laterales, debido a que los modelos rigurosos son sensibles al uso de estimados iniciales consistentes y este tipo de torres complejas requieren la generación de un buen conjunto de suposiciones para las variables de goteo para que no arroje sistemas de ecuaciones mal condicionados que entorpezcan la capacidad de convergencia del método.

En lo referido a la extensión de las capacidades de los algoritmos pudiera implementarse un módulo de conversión desde cualquier otro sistema de unidades de medida para el sistema que aceptan los módulos, que es el sistema internacional utilizado en este trabajo.

Puede realizarse una optimización de la programación de los métodos de las clases para poder lograr tiempos de convergencia más cortos que serán sin lugar a dudas necesarios si se quiere integrar en un entorno de simulación completo, así como también la optimización del código ayudará en disminuir los tiempos de convergencia de los algoritmos rigurosos.

La modularidad de las clases de modelos termodinámicos permite la expansión del proyecto para albergar diferentes métodos de definición de las sustancias puras, en particular se podría expandir el alcance de este proyecto para incluir métodos de caracterización de crudos por ensayos estándares, así como la caracterización de aceites pesados y otras mezclas o sustancias que no están incluidas en la base de datos *DIPPR*.

Por último, pero no menos importante, se recomienda la migración en la medida de lo posible de los algoritmos a otros lenguajes de programación, por ejemplo, pudiera realizarse en C++ si el objetivo es optimizar los tiempos de convergencia, pero también existe la opción de un lenguaje de alto nivel de gran valor pedagógico y que también integra rutinas de resolución de sistemas de ecuaciones LAPACK y BLAS, que son en los que basa su popularidad MATLAB®, dicha opción es Python. Además existe una mayor cantidad de recursos en Python con los que un estudiante puede aprender o con los cuales disponer para aplicaciones específicas.

## REFERENCIAS

- Asuaje, J. (2009). *Diseño e Implementación de un Software de Simulación para la Evaluación de Columnas de Destilación*. Caracas: Universidad Simón Bolívar.
- Barboza, A. y Díaz, J. (2013). *Modulo Tutorial para el Estudio de una Torre de Destilación Fraccionada a Escala Piloto: Modelado y Simulación*. Cartagena: Universidad de Cartagena. Facultad de Ingeniería.
- Bausa, J., Watzdorf, R. y Marquardt, W. (1998). *Shortcut Methods for Nonideal Multicomponent Distillation*. 1. Simple Columns. *AIChE*, 2181-2198.
- Boston, J. (1980). *Inside-Out Algorithms for Multicomponent Separation Process Calculations*. *Computer Applications to Chemical Engineering*, 124, 135.
- Boston, J. y Britt, H. (1978). A Radically Different Formulation and Solution of the Single-Stage Flash Problem. *Comput. Chem. Engng.*, 2, 577.
- Brüggemann, S. y Marquardt, W. (2004). *Shortcut Methods for Nonideal Multicomponent Distillation*. 3. Extractive Distillation Columns. *AIChE*, 1129-1149.
- Burman, Å. y Ström, K. (2011). Excess Enthalpy and Vapor-Liquid Equilibria with the MHV2 and Soave Mixing Rules. *Chem. Eng. Comm.*, 1435-1452.
- Carlberg, N. y Westerberg, A. (1988). *Underwood's Method for Side Strippers and Enrichers*. Pittsburgh: Carnegie Mellon University.
- Del Nogal, F. (1995). *Métodos Cortos y Semi-rigurosos en Destilación Multicomponente*. Caracas: Universidad Simón Bolívar.
- Doherty, M., Fidkowski, Z., Malone, M. y Taylor, R. (2008). Distillation. En D. Green, & R. Perry, *Perry's Chemical Engineers' Handbook* (págs. 13-1:13-116). New York: McGraw-Hill Companies.
- Figueira, F. (2005). *Desarrollo de Ecuaciones de Estado del tipo van der Waals para fluidos puros polares y no polares*. Caracas. Universidad Simón Bolívar.
- Figueira, F. (2010). *Desempeño de la Función de Cohesión SOF en la Predicción de Propiedades Termodinámicas del Sistema Amoniaco + Agua*. Caracas: Universidad Simón Bolívar.

- Green, D. y Perry, R. (2008). *Perry's Chemical Engineering Handbook*. McGraw-Hill.
- Hengstebeck, R. (1961). *Distillation: Principles and Design Procedures*. Reinhold Publishing.
- Ivakpour, J. y Kasiri, N. (2008). Improve Speed and Convergence of Distillation Column Simulation. *Hydrocarbon Processing*, 75-80.
- Jafarey, A., Douglas, J. M. y McAvoy, T. J. (1979). Short-cut Techniques for Distillation Column Design and Control. *Ind. & Eng. Chemistry Process Des. & Dev.*, 18, 197-202.
- Jenny, P. J. (1939). Graphical Solution of Problems in Multicomponent Fractionation. *Transactions of the American Institute of Chemical Engineers*, 35(635), 636.
- King, J. (1980). *Procesos de Separación*. Barcelona: Editorial Reverté.
- Kister, H. (1992). *Distillation Design*. McGraw-Hill.
- Kwauk, M. (1956). A System for Counting Variables in Separation Processes. *Hydrocarbon Research, Inc.*, 240-248.
- Leonard, S. (2005). Object Oriented Programming in Control System Software Engineering. Estados Unidos de América. Recuperado el 09 de 08 de 2015, de <https://www.matrikonopc.com/downloads/173/whitepapers/index.aspx>
- Li, C.-S., Zhang, X.-p. y Zhang, S.-j. (2006). Simulation of Multi-component Multi-stage Separation Process. An Improved Algorithm and Application. *The Chinese Journal of Process Engineering*, 247-254.
- Majzoub, M. (2011). *Desarrollo de un Programa con Interfaz Gráfica Guiada para Resolución de Problemas de Destilación Continúa por Métodos Gráficos*. Universidad Simón Bolívar. Coordinación de Ingeniería Química.
- Martín, Á., Bermejo, M., Mato, F. y Cocero, M. (2011). Teaching Advanced equations of state in applied thermodynamics courses using open source programs. *Education for Chemical Engineers*, e114-e121.
- Mathias, P. (2009). Visualizing the McCabe-Thiele Diagram. *American Institute of Chemical Engineers*, 36.
- Molokanov y., Korablina, T., Mazurina, N. y Nikiforov, G. (1972). An Approximate Method of Calculating the Basic Parameters of a Multicomponent Fractionation. *Ind. Eng. Chem.*, 209.

- Monroy-Loperena, R. (2003). Simulation of Multicomponent Multistage Vapor-Liquid Separations. An Improved Algorithm Using the Wang-Henke Tridiagonal Matrix Method. *Ind. Eng. Chem. Res.*, 175-182.
- Naphtali, L. y Sandholm, D. (1971). Multicomponent Separation Calculations by Linearization. *AIChE*, 17, 148-153.
- Navarro, M. (2003). *Módulo de Simulación de una Unidad de Destilación Multicomponente de Hidrocarburos Compatible CAPE-OPEN*. Caracas: Universidad Simón Bolívar.
- Parekh, V. y Mathias, P. (1998). Efficient Flash Calculations for Chemical Process Design. Extension of the Boston-Britt "Inside-Out" Flash Algorithm to Extreme Conditions and New Flash Types. *Computers & Chemical Engineering*, 1371-1380.
- Peng, D. y Robinson, D. (1976). A new two-constant equation of state. *Ind. Eng. Chem. Fundamentals*, 15, 59-64.
- Porras, S. (2012). *Programa Computacional para la Resolución de Problemas y Dimensionamiento de Columnas de Destilación y Absorción Multicomponente Aplicando Métodos Cortos*. Caracas. Universidad Simón Bolívar. Coordinación de Ingeniería Química.
- Prausnitz, J., Poling, B. y O'Connell, J. (2001). *The Properties of Gases and Liquids* (Quinta edición ed.). New York: McGraw-Hill.
- Redlich, O. y Kwong, J. (1949). On the thermodynamics of solutions V: An equation of state. Fugacities of gaseous solutions. *Chem. Rev.*, 44, 233.
- Reyes, J. A. (1998). *Diseño de Columnas de Rectificación y Extracción Multicomponente: Cálculo del Reflujo Mínimo*. Alicante: Biblioteca Virtual Miguel de Cervantes. Universidad de Alicante. Obtenido de <http://www.cervantesvirtual.com/obra/disenio-por-ordenador-de-columnas-de-rectificacion-y-extraccion-multicomponente-calculo-del-reflujo-minimo/ff38fc4c-82b1-11df-acc7-002185ce6064.pdf>
- Robinson, C. S. y Gilliland, E. R. (1950). *Elements of Fractional Distillation*. Toronto: McGraw-Hill Book Company.
- Russel, R. (1983). A Flexible and Reliable Method Solves Single-Tower and Crude-Distillation Column Problems. *Chemical Engineering*, 53.

- Seader, J., Henley, E. y Roper, D. (2011). *Separation Process Principles* (Tercera ed.). Estados Unidos de América: John Wiley & Sons, Inc.
- Shinskey, F. (1984). *Distillation Control*. New York: McGraw-Hill.
- Shiras, R. (1950). Calculation of Minimum Reflux in Distillation Columns. *Industrial and Engineering Chemistry*, 871-876.
- Simandl, J. y Svrcek, W. (1991). Extension of the Simultaneous-solution and *Inside-Outside* Algorithms to Distillation with Chemical Reactions. *Computers & Chemical Engineering*, 15, 337-348.
- Skogestad, S. (1997). Dynamics and Control of Distillation Columns. A Tutorial Introduction. *Distillation and Absorption*. 75, pág. Part. A. Maastricht: Trans. IChemE.
- Skogestad, S. y Morari, M. (1987). *Shortcut Models for Distillation Columns*. I: Steady State Behaviour. En S. Skogestad, *Studies in Robust Control of Distillation Columns*. Pasadena: California Institute of Technology.
- Smith, J., Van Ness, H., & Abbott, M. (2005). *Introduction to Chemical Engineering Thermodynamics*. McGraw-Hill.
- Smoker, E. H. (1938). Analytic Determination of Plates in Fractionating Columns. *Transactions of the American Institute of Chemical Engineers*, 34, 165-172.
- Soave, G. (1972). Equilibrium constants from a modified Redlich-Kwong equation of state. *Chem. Eng. Sci.*, 1197-1203.
- Tlacuatl, G. (2004). *Método Semi-riguroso de Simulación para Columnas de Destilación Multicomponente Conteniendo Mezclas Altamente No Ideales*. México, D.F.: Instituto Politécnico Nacional.
- Towler, G. y Sinnott, R. (2008). *Chemical Engineering Design: Principles, Practice and Economics of Plant and Process Design*. San Diego: Elsevier, Inc.
- Treybal, R. (1982). *Operaciones de Transferencia de Masa* (Segunda ed.). México: McGraw-Hill.
- Wang, J. C. y Henke, G. E. (1966). Tridiagonal Matrix for Distillation. *Hydrocarbon Processing*, 156-170.

- Watzdorf, R., Bausa, J., & Marquardt, W. (2004). *Shortcut* Methods for Nonideal Multicomponent Distillation. 2. Complex Columns. *AIChE*, 1615-1628.
- Youseff, S., Domenech, S. y Piboleau, L. (1989). Calcul d'une colonne de rectification à multiples alimentations et soutirages . *The Chemical Engineering Journal*, 153-165.

## APÉNDICE A

### CÁLCULO DE EJEMPLO DE LA INTERFAZ GRÁFICA

#### Funciones Varias:

“*Riguroso.m*”: Función de control de la interfaz gráfica para los métodos rigurosos y multicomponentes.

“*mccabethiele.m*”: Función de control para el método gráfico de McCabe y Thiele.

“*tridiagThomas.m*”:  $[\Delta X, Fik] = \text{tridiagThomas}(Bik, Aik, Cik, Dk)$ .

Función que recibe vectores o matrices bloque *Bik*, *Aik*, *Cik*, *Dk* con los cuales construye la matriz tridiagonal *Fik* del sistema de ecuaciones matricial  $Fik \cdot \Delta X = Dk$ . Los vectores o matrices *Bik*, *Aik*, *Cik* deben ser fila, horizontales e internamente se distribuyen a los elementos correspondientes en la matriz tridiagonal. El sistema es resuelto con  $\Delta X = Fik \backslash Dk$

“*indices.mat*”, “*indices1.mat*”: Bases de datos que relacionan el string de texto o caracteres de nombre de un compuesto con el número de índice en el cual este está ubicado en las bases de datos, ordenados alfabéticamente. El primero recibe un string y retorna un índice entre 1 y 345, el segundo recibe un índice entre 1 y 345 y retorna un string de nombre. Se utiliza el primero para ubicar la línea de los datos de los archivos contenedores de los compuestos, se utiliza el segundo para recuperar el string con el cual definir una clase Sustancia y mediante su método de definición interno defina todas las propiedades.

“*get\_indices.m*”, “*get\_indices1.m*”. En caso de modificación o expansión de las bases de datos de componentes se fabrican unas nuevas bases de datos “*indices.mat*” e “*indices1.mat*” al correr los algoritmos *get\_indices()* y *get\_indices1()*, respectivamente.

“*interpolLagrange2.m*”:  $\text{obj} = \text{interpolLagrange2}(x, Ex, Ey, A, B)$

Función objetivo para realizar mediante un *fsolve()* interpolación de Lagrange cuadrática que halle el punto horizontal *x* donde corta una línea recta que atraviesa los vectores de coordenadas *A*, *B* [1x2] sobre una curva construida construyendo pares de coordenadas [*x y*] tomando valores de los vectores *Ex*, *Ey* de mínimo 3 elementos y arbitrariamente largos. Utilizado en la construcción del diagrama de Hengstebeck para el trazado gráfico de las etapas en el diagrama *x-y*.

“*underwood.m*”. [*rmin*, *dflow*, *incog*] = *underwood*(*composic*, *flow*, *sup\_recup\_d*, *sup\_recup\_b*, *KD*, *light\_key*, *heavy\_key*, *beta*).



Función que realiza la distribución a reflujo mínimo de los productos de destilado y fondo así como calcula el valor del reflujo mínimo ( $r_{min}$ ), del flujo de destilado ( $d_{flow}$ ) y la distribución de compuestos no claves ( $incog$ ) de longitud arbitraria. Recibe las composiciones, flujos, una suposición de recuperación en destilado y residuo de los componentes clave liviano y pesado, un vector de coeficientes de distribución líquido vapor y una fracción vaporizada de la alimentación.

*“obj\_mccabe.m”*.  $[salida] = obj\_mccabe(y, zf, q, P, mezcla, light, heavy, RM, kij)$ .

Función objetivo para obtener mediante un  $fsolve()$  la fracción molar de clave liviano en la mezcla vapor, según el equilibrio de fases el cual predice la curva y según el modelo termodinámico provisto *“RM”*, para obtener la ordenada y en dónde una recta  $q$  de alimentación de fracción molar de liviano  $zf$ , condición térmica  $q$  y a presión  $P$  corta el equilibrio.

*“heats\_vaporization.dat”*. Base de datos *DIPPR* de correlación para la predicción de los calores intensivos molares de cambio de fase líquido a vapor para más de 340 sustancias puras. Requerido en el algoritmo de IdealEdE en el cual se estima el calor de líquido mediante la substracción del calor de vaporización al calor calculado por la suposición de gas ideal.

*“get\_latent\_heat.m”*. Constructor del function handle que correlaciona los calores de vaporización para distintas sustancias puras mediante una función de la temperatura. Si se utilizan múltiples bases de datos, en el archivo *“heats\_vaporization.dat”* se puede cambiar para el componente el identificador de la correlación, que por defecto es *“A”* y agregar nuevos constructores *“B”* u otros.

*“cpgasideal.dat”*. Base de datos *DIPPR* de correlación para la predicción de los calores específicos molares en gas ideal para más de 340 sustancias puras para el cálculo del calor sensible. Requerido en el algoritmo de IdealEdE en el cual se estima el calor por la suposición de gas ideal y en el cálculo de la entalpía real para substraerle el calor residual del modelo utilizado.

*“get\_ig,cp.m”*. Constructor del function handle que correlaciona los calores específicos de más de 340 sustancias de la base de datos *DIPPR* en función de la temperatura. Si se utilizan múltiples bases de datos, en el archivo *“cpgasideal.dat”* se puede cambiar para el componente el identificador de la correlación, que por defecto es *“A”* y agregar nuevos constructores *“B”* u otros.

*“Under\_theta.m”*.  $Under\_theta(theta, alfa, Flow, Z, beta)$ .

Es una función objetivo que sirve de argumento al  $fsolve()$  para hallar raíces  $theta$  que satisfagan el sistema de ecuaciones de Underwood.

*“get\_properties.m”*. Obtiene las propiedades de sustancias puras de la base de datos *DIPPR*, de la sustancia cuyo string le sea proveído. Extrae los datos del archivo *“properties.dat”*

*“get\_vapour\_pressure.m”*. Constructor del function handle que correlaciona la presión de saturación de más de 340 sustancias de la base de datos *DIPPR* en función de la temperatura. Si se utilizan múltiples bases de datos, en el archivo *“vapourpressure.dat”* se puede cambiar para el componente el identificador de la correlación, que por defecto es “A” y agregar nuevos constructores “B” u otros. Es utilizado por la clase *IdealEdE* para el cálculo de la fugacidad de solución ideal.

*“properties.dat”*. Base de datos de propiedades de sustancias puras para la base de datos *DIPPR*, ordenada alfabéticamente.

*“RachfordRice.m”*. *RachfordRice(beta, z, K)*.

Función objetivo para obtención de la fracción vaporizada por el método de Rachford Rice para su uso en un *FSOLVE()* en la resolución de un flash isotérmico.

*“fenske.m”*. *[nmin, recup\_d, recup\_b] = fenske(composic, flow, sup\_recup\_d, sup\_recup\_b, light\_key, heavy\_key, volatildest, volatilfondo)*

Función para la resolución del algoritmo de fenske para la estimación del mínimo número de etapas de una torre de destilación en función de las características de alimentación: composición, flujo, suposición trivial de recuperación de componentes clave liviano y pesado y volatilidades relativas de destilado y fondo.

*“antoinedata.dat”*. Archivo reservado para la posible implementación de una base de datos de constantes para la correlación de Antoine para el cálculo de presión de vapor de sustancias puras.

*“Under\_Riguroso”*. *Under\_Riguroso(soluc, theta, alfa, Recup\_D, Z, Flow, claves)*.

Función objetivo para la resolución con *fsolve()* del sistema de ecuaciones de una columna de destilación simple en condición de reflujo mínimo por el algoritmo de Underwood.

Clase **Sustancia** < handle

Objetos en esta clase representan sustancias puras, que son componentes en una mezcla. Funciona para almacenar las propiedades necesarias para la aplicación de modelos termodinámicos.

Propiedades:

id: Nombre o etiqueta de la sustancia, string de caracteres.

molmass: Peso molecular de la sustancia, [kg/kmol]. Clase 'double'

tcrit: Temperatura crítica de la sustancia, [K]. Clase 'double'.

pcrit: Presión crítica de la sustancia, [kPa]. Clase 'double'

w<sub>acent</sub>: Factor acéntrico de la sustancia. Clase 'double'

psat: Clase 'cell' 1x3. Almacena un function handle en el primer elemento, la mínima temperatura en K para el que es válido el handle en el segundo elemento y la temperatura máxima de validez en K para el tercer elemento.

tsat: Clase 'cell' 1x3. Almacena un function handle para obtener la temperatura de saturación en K en función de la presión en kPa. También almacena el mínimo y máximo valor de presión para la validez de la correlación.

href: Entalpía de referencia a 273,15K y 101,325kPa, para el cálculo de entalpía con las ecuaciones de estado. Clase 'double'. Por defecto es 0.

cp<sub>gi</sub>: Clase 'cell' 1x3. Almacena un function handle para correlacionar el cp de gas ideal [kJ/(kmol·K)] en función de la temperatura de la base de datos *DIPPR*. También almacena la mínima y máxima temperatura de aplicación

Hvap: Clase 'cell' 1x3. Almacena un function handle para correlacionar la entalpía de vaporización [kJ/kmol] en función de la temperatura obtenido de la base de datos *DIPPR*. También almacena la mínima y máxima temperatura de aplicación.

formul: String de caracteres que almacena la fórmula molecular de la sustancia.

indice: Entero que almacena la línea de ubicación en los archivos de la base de datos de la sustancia.

ParamEdE: Variable reservada para el almacenamiento de parámetros específicos de la sustancia en determinadas ecuaciones de estado.

Métodos:

Sustancia(identificador, varargin). Constructor de la clase.

Clase **Mezcla** < handle

Objetos de esta clase representan una mezcla de dos o más componentes. La clase almacena elementos de clase Sustancia que representan a cada sustancia pura, también almacena la composición de la mezcla y los coeficientes de interacción binaria entre elementos.

Propiedades:

comp: Clase Sustancia 1xC. Almacena clases Sustancia, una por cada componente de la mezcla.

num\_sust: Almacena el número de sustancias,  $C$

conc: Vector 1xC que almacena las fracciones molares de la mezcla.

kij: Vector que almacena los parámetros de interacción binaria entre los componentes.

kbinario: Vector de las dimensiones de kij que almacena los string de índices entre los parámetros de interacción binaria.

kmatrix: Matriz cuadrada, cada dimensión es del tamaño de la longitud de kij, que almacena los parámetros de interacción binaria en forma matricial.

Métodos:

Mezcla(componentes, concent, interaccion\_bin). Constructor de la clase.

matrixk(). Construye a partir del vector kij la matriz kmatrix.

definek(kij). Permite definir a una mezcla ya creada los parámetros de interacción binaria, introduciendo un vector kij.

Clase: **EdECubica** < handle

Esta clase implementa el cálculo de las propiedades termodinámicas de sustancias puras. La clase provee métodos para calcular todas esas propiedades, que son todas heredadas por las clases EdE derivadas de ellas

Propiedades:

familia = 'Cubicas'.

Métodos.

presion(T, v, comp, parametros, alfa\_func). Calcula la presión a partir de la temperatura y el volumen específico para una clase sustancia en 'comp', utilizando parámetros de la ecuación de estado cúbica generalizada, ecuaciones (3), (4) y (5) y el function handle en 'alfa\_func' para el cálculo del parámetro de atracción de la ecuación de estado

cmpr(T, P, comp): Calcula el factor de compresibilidad Z a condiciones T y P.

isofugT(P, comp, tol, maxite): Realiza el cálculo por isofugacidades de la temperatura de saturación de la sustancia comp, con la tolerancia tol y en maxite iteraciones

isofugP(T, comp, tol, maxite): Realiza el cálculo por isofugacidades de la presión de saturación de la sustancia comp, con la tolerancia tol y en maxite iteraciones

a\_handle(T, comp): Realiza el cálculo del parámetro crítico de atracción  $a_c$ , para cualquier ecuación de estado.

entalpia(T, P, comp, fase): Cálculo de la entalpía residual a condiciones de temperatura T y presión P, para la sustancia comp, en la fase que puede ser 'liq' para líquido o 'vap' para vapor.

fugF(T, P, comp, fase): Cálculo de la fugacidad a condiciones de temperatura T y presión P, para la sustancia comp, en la fase que puede ser 'liq' para líquido o 'vap' para vapor.

intnrg(T, P, comp, fase): Cálculo de la energía interna a condiciones de temperatura T y presión P, para la sustancia comp en la fase que puede ser 'liq' para líquido o 'vap' para vapor.

entropia(T, P, comp, fase): Cálculo de la entropía de la sustancia comp a condiciones de temperatura T y presión P, para fases 'liq' o 'vap'.

b\_fun(comp). Cálculo del covolumen para cualquier ecuación cúbica en función de los parámetros críticos de la sustancia pura comp.

Lista de archivos:

EdECubica.m, a\_handle.m, entalpia.m, fugF.m, intnrg.m, entropia.m, b\_fun.m

Clases: **SRKEdE, PRedE, PRGEEdE < EdECubica**

Implementación específica de las ecuaciones cúbicas para compuesto puro derivadas.

Propiedades:

id: String identificación del nombre de la ecuación cúbica, ej. 'Soave-Redlich-Kwong'

propied: Vector fila de almacenamiento de las constantes asociadas a la ecuación cúbica,  $\delta$ ,  $\varepsilon$ ,  $\psi_a$ ,  $\Omega$ .

alfa\_inp: Function handle opcional para reemplazar la ecuación específica para el cálculo del parámetro de atracción  $\alpha$

m\_inp: Function handle opcional para reemplazar la ecuación específica en el parámetro de atracción asociada al factor acéntrico

derivada\_alfa: Function handle opciones para el cálculo de la derivada de una ecuación alfa\_inp opcional, para poder calcular propiedades residuales.

Métodos:

a\_fun(T, comp): Cálculo de la función del parámetro crítico de atracción  $ac$ .

alfa\_fun(T, comp, m\_inp). Cálculo de la función del parámetro de atracción a temperatura T para una sustancia comp.

dalfa\_fun(T, comp): Cálculo de la derivada de la función  $\alpha$  para la ecuación de estado específica.

m\_fun(omega): Función dependiente del factor acéntrico.

Lista de archivos:

SRKEdE.m, alfa\_fun.m, dalfa\_fun.m, m\_fun.m.

Clase: **IdealEdE** < handle

Esta clase implementa el cálculo de las propiedades termodinámicas y el equilibrio de fases para mezclas, considerando comportamiento ideal (Ley de gases ideales y ley de Raoult). La clase provee los métodos para calcular esas propiedades a todas las clases inferiores que heredan de ella

Métodos:

fug(T, P, mezcla, fase). Determina la fugacidad de una mezcla en determinada fase ('liq' o

‘vap’) a condiciones de P y T.

entalpia(T, P, mezcla, fase). Determina la entalpía de una mezcla en determinada fase (‘liq’ o ‘vap’) a condiciones de P y T.

flash(T, P, mezcla, sup\_x, sup\_y, sup\_beta, sup\_T, tol, max\_iter). Realiza el algoritmo de flash isotérmico a una mezcla de condiciones T y P en un número máximo de iteraciones max\_iter (por defecto 100) hasta alcanzar cumplir con una tolerancia de tol en dos composiciones sucesivas de vapor y líquido (por defecto 1e-6).

fugF(T, P, mezcla, fase, elemento). Obtiene un vector con la fugacidad parcial en una mezcla fluida para todas las sustancias componentes a condiciones de T, P y determinada fase (‘liq’ o ‘vap’). Si se quiere uno individual, se introduce en ‘elemento’ el componente deseado.

BubbleT(P, mezcla, sup\_y, tol, max\_iter, sup\_T): Obtiene la temperatura de la mezcla a presión P en su punto de burbuja, así como la composición de la primera burbuja de vapor y el coeficiente de reparto  $K_{ij}$  por componente. La tolerancia (por defecto 5e-8) es en términos relativos en función tanto de temperatura como de fracciones molares (max\_iter por defecto es 500).

DewT(T, mezcla, sup\_x, tol, max\_iter, sup\_T): Obtiene la temperatura de la mezcla a presión P en su punto de rocío, así como la composición de la primera gota de líquido y el coeficiente de reparto  $K_{ij}$  por componente. La tolerancia (por defecto 5e-8) es en términos relativos en función tanto de temperatura como de fracciones molares (max\_iter por defecto es 500).

BubbleP(T, mezcla, sup\_y, tol, max\_iter, sup\_T): Obtiene la presión de la mezcla a temperatura T en su punto de burbuja, así como la composición de la primera burbuja de vapor y el coeficiente de reparto  $K_{ij}$  por componente. La tolerancia (por defecto 5e-8) es en términos relativos en función tanto de temperatura como de fracciones molares (max\_iter por defecto es 500).

DewP(T, mezcla, sup\_x, tol, max\_iter, sup\_T): Obtiene la presión de la mezcla a temperatura T en su punto de rocío, así como la composición de la primera gota de líquido y el coeficiente de reparto  $K_{ij}$  por componente. La tolerancia (por defecto 5e-8) es en términos relativos en función tanto de temperatura como de fracciones molares (max\_iter por defecto es 500).

Lista de archivos:

IdealEdE.m, fug.m, entalpia.m, flash.m, fugF.m, BubbleT.m, DewT.m, BubbleP.m,

DewP.m.

### Clase **RMVdW** < **IdealEdE**

Esta clase implementa el cálculo de propiedades termodinámicas y de estimación de equilibrio de fases a mezclas utilizando las reglas de mezclado de Van der Waals con las ecuaciones de estado cúbicas. La clase hereda los métodos genéricos de ecuaciones cúbicas para el cálculo de propiedades a todas las formas de ecuaciones cúbicas derivadas

Propiedades:

familia = 'Cubicas'

EdE: Clase EdECubica. Almacena un objeto ecuación de estado cúbica para sustancias puras, que el a la cual se le aplicarán reglas de mezclado.

Métodos:

RMVdW(EdECubica). Constructor de la clase.

cmpr(T, P, mezcla). Calcula el factor de compresibilidad Z de una mezcla a condiciones T y P.

b\_factor(mezcla). Calcula el covolumen de una mezcla de sustancias. Ecuación (7)

a\_factor(mezcla, T). Calcula el factor de atracción crítico de una mezcla a temperatura T.

a\_ipuro(mezcla, T). Calcula un vector de tamaño igual al número de componentes con los factores de atracción (*ac*) de cada compuesto como sustancia pura.

delta\_ipuro(mezcla, T). Calcula un vector de tamaño igual al número de componentes con la derivada del factor de atracción de cada compuesto como sustancia pura, tomada a temperatura T.

aij\_fun(mezcla, ai\_puro). Organiza y calcula en forma matricial el conjunto de factores de atracción como mezcla binaria (mediante promedio geométrico) de los componentes entre si.

volumesp(T, P, mezcla, fase): Volúmen específico de una mezcla a condiciones T y P y en determinada fase ('liq' o 'vap').

entropia(T, P, mezcla, fase): Cálculo de la entropía de una mezcla a condiciones T y P y en



determinada fase ('liq' o 'vap')

entalpia(T, P, mezcla, fase): Cálculo de la entropía de una mezcla a condiciones T y P y en determinada fase ('liq' o 'vap').

intnrg(T, P, mezcla, fase): Cálculo de la energía interna de una mezcla a condiciones T y P y en determinada fase ('liq' o 'vap').

fugF(T, P, mezcla, fase): Cálculo de la fugacidad de una mezcla en determinada fase ('liq' o 'vap') a condiciones de presión P y temperatura T.

Lista de archivos:

RMVdW.m, volumesp.m, entalpia.m, entropía.m, fugF.m, intnrg.m,

Clase **Plato** < handle

Objeto que almacena un grupo consistente de propiedades de una etapa teórica de una columna de destilación.

Propiedades:

nro: Número de la etapa teórica

z\_i: Composición en fracciones molares conjunta de todas las fases en la mezcla del plato.

x\_i: Fracciones molares por componente de la fracción de líquido en la etapa teórica.

y\_i: Fracciones molares por componente de la fracción de vapor en la etapa teórica.

v\_i: Flujos molares de vapor por componente que abandona la etapa teórica.

l\_i: Flujos molares de líquido por componente que abandona la etapa teórica.

T: Temperatura de la etapa teórica.

P: Presión de la etapa teórica

V: Flujo total de vapor que abandona la etapa teórica

L: Flujo total de líquido que abandona la etapa teórica

K: Coeficiente de reparto líquido vapor por componente en la etapa teórica.

num\_sust: Número total de componentes en la mezcla del plato teórico

aliment: Clase Corriente. Corriente que es ingresada a la etapa teórica.

salidaV: Magnitud de flujo molar total en fase vapor que es retirado como salida de la etapa teórica.

salidaL: Magnitud de flujo molar total en fase líquido que es retirado como salida de la etapa teórica.

pumparoundOut: Clase 'cell'. Salida de la etapa teórica que se bombea a otra etapa teórica, almacena el índice de la etapa teórica objetivo y la magnitud del flujo molar de líquido.

pumparoundIn: Clase 'cell'. Entrada en la etapa teórica que se bombea desde otra distinta etapa teórica. Almacena el índice de la etapa teórica origen y la magnitud del flujo molar de líquido

Métodos:

Plato(nro, T, P, v\_i, l\_i, x\_i y\_i, V, L, K, alim, saleV, saleL, pumpout, pumpin). Constructor de la clase.

setvi(v\_i). Establece el flujo de vapor por componente (v\_i) y reescribe el conjunto de variables y\_i, V y z\_i tal que sea consistente.

setli(l\_i). Establece el flujo de líquido por componente (l\_i) y reescribe el conjunto de variables x\_i, L y z\_i tal que sea consistente.

setxi(y\_i). Establece la composición de vapor por componente (y\_i) y reescribe el conjunto de variables v\_i tal que sea consistente.

setyi(x\_i). Establece la composición de líquido por componente (x\_i) y reescribe el conjunto de variables l\_i tal que sea consistente.

setV(V). Establece el flujo molar de vapor por componente (V) y reescribe el flujo de vapor por componente (v\_i) tal que sea consistente.

setL(L). Establece el flujo molar de líquido por componente (L) y reescribe el flujo de líquido por componente (l\_i) tal que sea consistente.

mixtured(). Escribe z\_i a partir de las variables de líquido y de vapor.

Clase *Shortcut* < handle

Objeto que almacena las propiedades y provee los métodos de cálculo requeridos para la distribución de una torre de destilación mediante el algoritmo de FUG, Smoker, Westerberg, Shinskey, Jafarey.

flujo: Flujo molar total de cada una de las alimentaciones a la columna de destilación.  
[kmol/h]

Dconc: Concentración de cada componente en la fracción extraída en el tope como destilado

Bconc: Concentración de cada componente en la fracción líquida extraída en el fondo

nteo: Número de etapas teóricas estimadas.

nmin: Número de etapas mínimas a reflujo total.

rmin: Reflujo de tope ideal a condición de infinitas etapas

concFeed: Matriz que almacena las concentraciones de cada alimentación para todos los componentes, almacena 1 fila por cada alimentación.

num\_sust: Número entero que representa el número total de sustancias.

Qc: Calor transferido en el condensador, [kJ/(kmol·h)]

Qb: Calor transferido en el rehervidor, [kJ/(kmol·h)]

cdest: Clase Corriente. Reservado para la implementación de una corriente de salida de destilado.

cbot: Clase Corriente. Reservado para la implementación de una corriente de salida de fondo.

cfeed: Clase Corriente. Almacena el objeto con las propiedades de la corriente de ingreso a la torre de destilación.

lk: Número entero que representa la ubicación del componente clave liviano

hk: Número entero que representa la ubicación del componente clave pesado

reflujo: Magnitud de la relación de reflujo en operación entre líquido recirculado a la primera etapa y el destilado extraído en el tope. Se estima con el método de Gilliland.

volat: Volatilidad relativa, promedio geométrico, para cada componente, aplicable a la torre de destilación en toda su extensión.

volat\_top: Volatilidad relativa para cada componente en el tope de la torre de destilación

volat\_bot: Volatilidad relativa para cada componente en el fondo de la columna.

Clase **Hengstebeck** < handle

Objeto que almacena las propiedades y provee los métodos de cálculo requeridos para la distribución de una torre de destilación mediante el método gráfico pseudobinario de Hengstebeck basado en los diagramas de McCabe-Thiele

Propiedades:

Fi: Flujo molar por componente total suma del flujo molar por componente para cada una de las alimentaciones a la columna de destilación, [kmol/h]

comp: Clase Sustancia. Almacena un vector de sustancias, una por cada componente

dsubi: Flujo molar de cada componente en la fracción extraída en el tope como destilado, [kmol/h].

bsubi: Flujo molar de cada componente en la fracción extraída en el fondo como destilado, [kmol/h].

nteo: Número de etapas teóricas estimadas.

nmin: Número de etapas mínimas a reflujo total.

rmin: Reflujo de tope ideal a condición de infinitas etapas

conceed: Matriz que almacena las concentraciones de cada alimentación para todos los componentes, almacena 1 fila por cada alimentación.

num\_sust: Número entero que representa el número total de sustancias.

Qc: Calor transferido en el condensador, [kJ/(kmol·h)]

Qb: Calor transferido en el rehervidor, [kJ/(kmol·h)]

cdest: Clase Corriente. Reservado para la implementación de una corriente de salida de destilado.

cfond: Clase Corriente. Reservado para la implementación de una corriente de salida de fondo.

alimentaciones. Clase Corriente. Almacena el objeto con las propiedades de la corriente de ingreso a la torre de destilación.

vivo: Variable lógica. Por defecto falso. Verdadero si en la columna se utiliza vapor vivo.

rectarectmin: Cálculo de la recta de la zona de rectificación para la condición de reflujo mínimo.

rectastripmin: Cálculo de la recta de zona de despojo para la condición de reflujo '*boil-up*' mínimo.

rectaoperacion: Matriz que almacena las

lk: Número entero que representa la ubicación del componente clave liviano

hk: Número entero que representa la ubicación del componente clave pesado

reflujo: Magnitud de la relación de reflujo en operación entre líquido recirculado a la primera etapa y el destilado extraído en el tope. Se estima con el método de Gilliland.

volat: Volatilidad relativa, promedio geométrico, para cada componente, aplicable a la torre de destilación en toda su extensión.

volat\_top: Volatilidad relativa para cada componente en el tope de la torre de destilación

volat\_bot: Volatilidad relativa para cada componente en el fondo de la columna.

Clase ***LewisMatheson*** < handle

Objeto que almacena las propiedades y provee los métodos de cálculo requeridos para la distribución de una torre de destilación mediante el algoritmo de Lewis y Matheson

Propiedades:

id. Identificador o nombre de la unidad de destilación.

nteo: Número de etapas teóricas estimadas.

nmin: Número de etapas mínimas a reflujo total.

rmin: Reflujo de tope ideal a condición de infinitas etapas

reflujo: Magnitud de la relación de reflujo en operación entre líquido recirculado a la primera etapa y el destilado extraído en el tope.

reflujofondo: Magnitud de la relación de reflujo en operación entre vapor rehervido y extracto líquido de fondo

concFeed: Matriz que almacena las concentraciones de cada alimentación para todos los componentes, almacena 1 fila por cada alimentación.

num\_sust: Número entero que representa el número total de sustancias.

Qc: Calor transferido en el condensador, [kJ/(kmol·h)]

Qb: Calor transferido en el rehervidor, [kJ/(kmol·h)]

lk: Número entero que representa la ubicación del componente clave liviano

hk: Número entero que representa la ubicación del componente clave pesado

ratioLV: Representa la relación etapa por etapa entre flujo de líquido y de vapor en las etapas de la zona de rectificación, entre el destilado de tope y la alimentación

ratioLsVs: Representa la relación etapa por etapa entre flujo de líquido y de vapor en las etapas de la zona de despojo, entre la alimentación y el residuo de fondo.

MEdE: Clase IdealEdE, reglas de mezclado (ej. RMVdW) o ecuación de estado multicomponente (hija de IdealEdE) con la cual se resolverá el equilibrio de fases en la columna de destilación.

PlatosTot: Número de platos totales utilizando un coeficiente de eficiencia por plato.

concVapPlato: Matriz que representa la concentración de cada componente en la fase vapor en cada etapa teórica.

concLiqPlato: Matriz que representa la concentración de cada componente en la fase líquida en cada etapa teórica.

flujo: Flujo molar total de cada una de las alimentaciones a la columna de destilación. [kmol/h]

feedtray: Etapa de entrada de la alimentación estimado en base al método Lewis y Matheson

Dconc: Concentración de cada componente en la fracción extraída en el tope como destilado

Bconc: Concentración de cada componente en la fracción líquida extraída en el fondo

cdest: Clase Corriente. Reservado para la implementación de una corriente de salida de destilado.

cbot: Clase Corriente. Reservado para la implementación de una corriente de salida de fondo.

cfeed: Clase Corriente. Almacena el objeto con las propiedades de la corriente de ingreso a la torre de destilación.

beta\_feed: Fracción vaporizada de la alimentación a la torre en cascada.

q: Condición térmica de la alimentación a la columna.

heat\_loss: Estimación de calor retirado por etapa.

starttop: Variable lógica. Verdadero por defecto. En el caso de ser verdadero, la columna comenzará a estimarse desde el destilado hacia las etapas sucesivas.

startbottom: Variable lógica. Falso por defecto. En el caso de ser verdadero, la columna comenzará a estimarse desde el residuo de fondo hasta el tope de la columna.

TPlatoPlato: Vector que almacena los estimados de temperatura en cada una de las etapas teóricas, [K]

## APÉNDICE B

### EJEMPLO DE USO DE LA INTERFAZ GRÁFICA

Para el ejemplo 1 en el capítulo 4 se utilizó la interfaz gráfica de la siguiente forma para la simulación de los resultados. En primer lugar se ejecutó el archivo ‘Riguroso.m’ que llama a la interfaz gráfica de usuario. En la pantalla aparece la Fig. A.1. La interfaz posee un mecanismo intuitivo de señalización de los requerimientos, utilizando color rojo cuando aún los grados de libertad no son cero y se requiere la asistencia del usuario para añadir los datos experimentales.

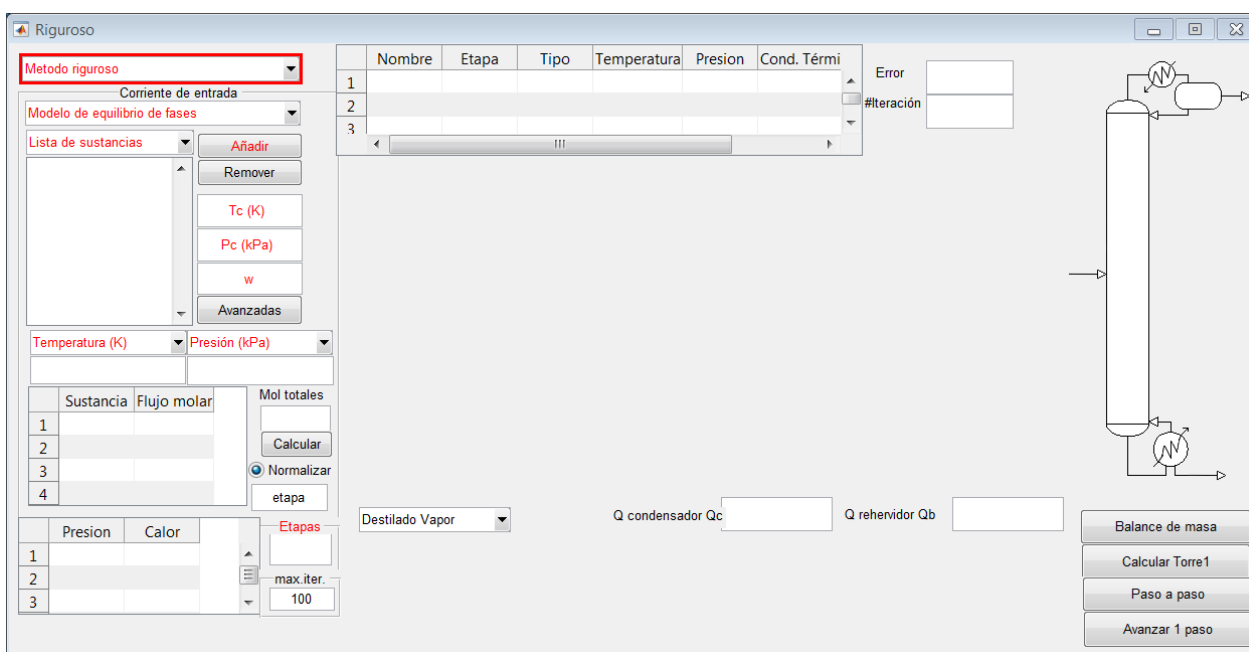


Fig. A.1 Primer paso: selección del método

Lo primero a realizar es seleccionar el método de cálculo a utilizar mediante el menú ‘Método riguroso’. En este ejemplo se utilizó el método de *Bubble Point*. Se despliega la pestaña resaltada en la Fig. A.1 y se selecciona, quedando el menú en color negro como es mostrado en la foto del paso siguiente en la Fig. A.2.

A continuación se requiere escoger un método termodinámico para el cálculo de la corriente principal y de la torre. La modularidad de DESTILMATLABUSB permitirá a corrientes de alimentación secundarias ser resueltas con modelos termodinámicos distintos, de resultar más conveniente, pero la torre obtendrá su modelo termodinámico de la primera corriente alimentada.



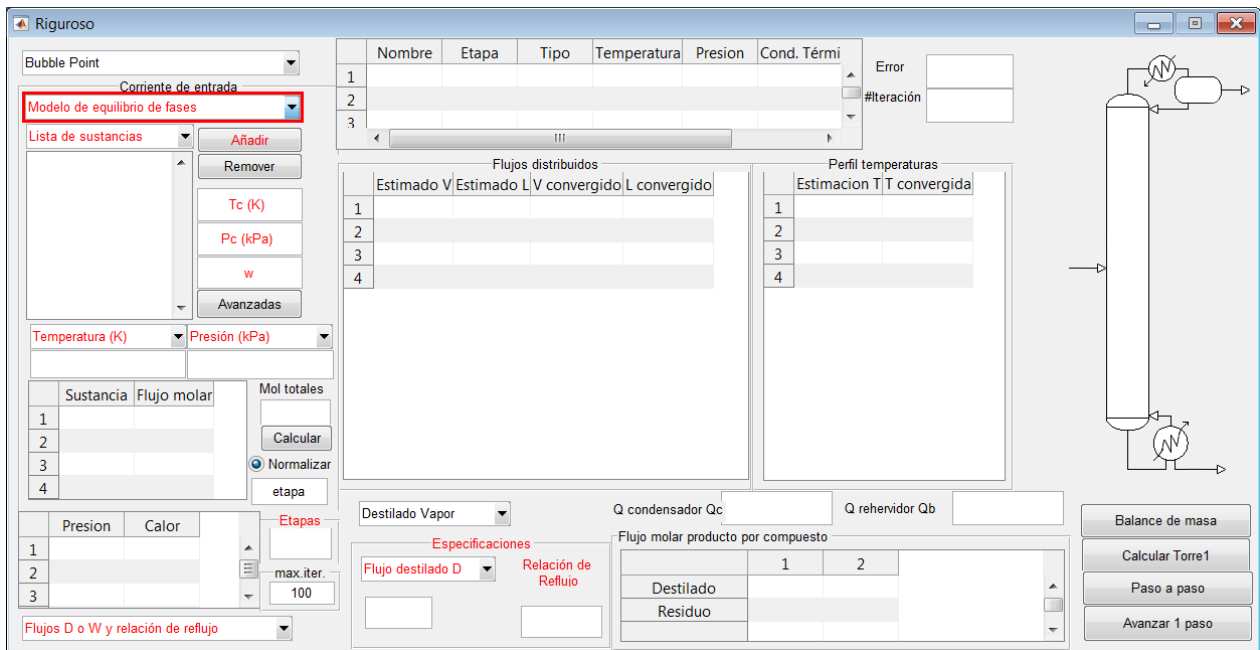


Fig. A. 2. Segundo paso: Selección del modelo termodinámico para la corriente de alimentación principal y para la resolución de la columna en cascada

A todas las corrientes de alimentación deben agregárseles todas las sustancias que van a ser necesarias en todas las alimentaciones de la torre, incluso si algunas no están presentes en todas. Es conveniente que en el paso de proveer la composición a los componentes de fracción despreciable se les asigne un flujo molar o una fracción molar despreciable distinta de cero, por ejemplo,  $1e-11$ .

Por tanto el siguiente paso es escoger las sustancias. Ello se logra utilizando el menú resaltado en la Fig. A.3 y seleccionando de la base de datos la sustancia deseada en orden de volatilidad (internamente el programa los pondrá en orden de ser necesario, pero si estuviesen desordenados por el usuario ocasionaría que los resultados sean presentados en un orden disímil al suministrado por el usuario y ocasionaría confusión, así que es mejor introducirlos en orden). En el caso del ejemplo, la sustancia más volátil es el benceno, por lo cual se le selecciona en la lista como en la Fig. A.4 y se presiona el botón añadir resaltado como se muestra en la Fig. A.5.

Para la comodidad de búsqueda de componentes en el listado de sustancias de la base de datos *DIPPR*, se puede usar el teclado y buscar presionando la letra inicial del componente para ir alternando entre los componentes con dicha inicial hasta llegar al deseado y presionar enter estando el componente resaltado. Alternativamente también puede utilizarse el *mouse* y la barra de desplazamiento, cuando se le consigue, hacer click con el botón izquierdo del ratón.

**Riguroso**

Bubble Point

Corriente de entrada: Peng Robinson (VdW)

Lista de sustancias (highlighted in red)

Añadir

Remover

Tc (K)

Pc (kPa)

w

Avanzadas

Temperatura (K)

Presión (kPa)

Sustancia

Flujo molar

Mol totales

Calcular

Normalizar

etapa

Presión

Calor

Etapas

max.iter. 100

Flujos D o W y relación de reflujo

Nombre

Etapas

Tipo

Temperatura

Presión

Cond. Térm.

Error

#iteración

Flujos distribuidos

Estimado V

Estimado L

V convergido

L convergido

Perfil temperaturas

Estimación T

T convergida

Destilado Vapor

Q condensador Qc

Q rehevador Qb

Flujo molar producto por compuesto

Destilado

Residuo

Balance de masa

Calcular Torre1

Paso a paso

Avanzar 1 paso

Fig. A.3 Paso 3: Selección de la lista de componentes requeridos en la columna

**Riguroso**

Bubble Point

Corriente de entrada: Peng Robinson (VdW)

Lista de sustancias (highlighted in red)

Añadir

Definido por el usuario

Acetaldehyde C2H4O

Acetamide C2H5NO

Acetic acid C2H4O2

Acetic anhydride C4H6O3

Acetone C3H6O

Acetonitrile C2H3N

Acetylene C2H2

Acrolein C3H4O

Acrylic acid C3H4O2

Acrylonitrile C3H3N

Aire

Ammonia H3N

Anisole C7H8O

Argon Ar

Benzamide C7H7NO

**Benzene (C6H6)** (highlighted in blue)

Benzenethiol C6H6S

Benzoic acid C7H6O2

max.iter. 100

Flujos D o W y relación de reflujo

Nombre

Etapas

Tipo

Temperatura

Presión

Cond. Térm.

Error

#iteración

Flujos distribuidos

Estimado V

Estimado L

V convergido

L convergido

Perfil temperaturas

Estimación T

T convergida

Destilado Vapor

Q condensador Qc

Q rehevador Qb

Flujo molar producto por compuesto

Destilado

Residuo

Balance de masa

Calcular Torre1

Paso a paso

Avanzar 1 paso

Fig. A. 4 Paso 4: Selección del componente Benceno presente en el ejemplo del Capítulo 4

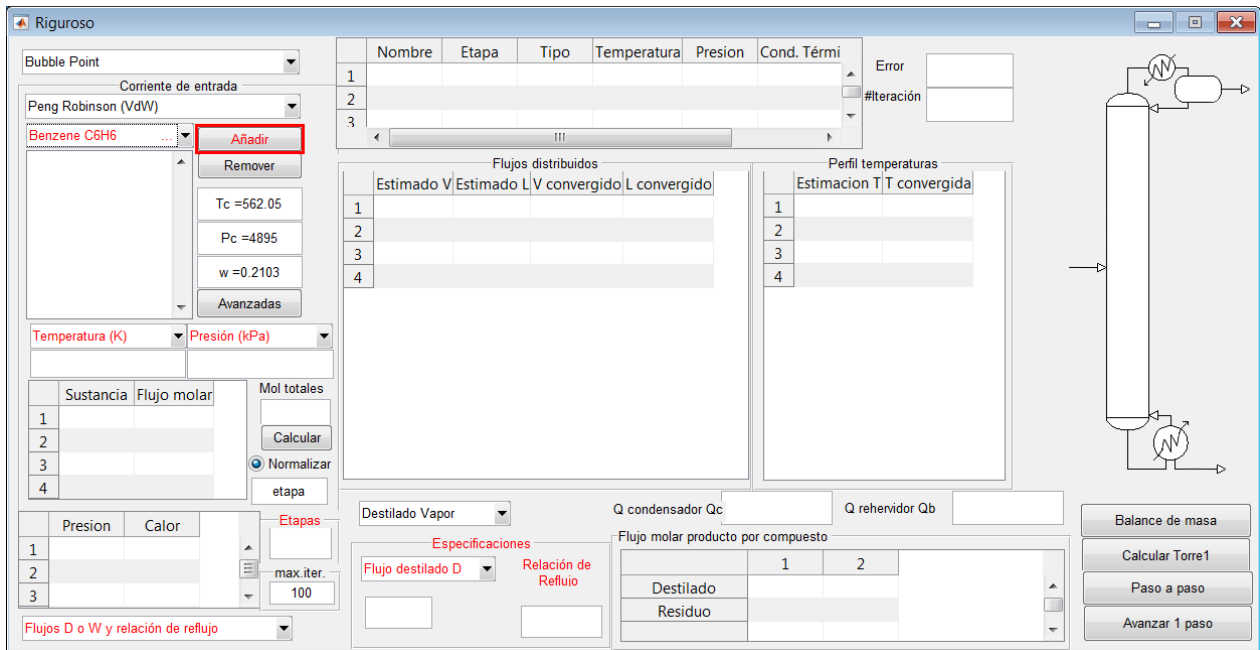


Fig. A.5 Paso 5. Activar el botón "Añadir" para ingresar el Benceno a la lista de sustancias de la alimentación

Como se ve en la Fig. A.6, uno de los componentes en la lista permite el ingreso de un compuesto hipotético de características diseñadas por el usuario para sus fines. Al añadirlo en la lista de compuestos, será necesario proveer los datos críticos de la sustancia, en las casillas resaltadas en la Fig. A.7, que se marcan en rojo al ser necesarias. También puede modificarse cualquiera de las propiedades críticas para una sustancia incluida en la base de datos, el usuario debería borrar la casilla de datos e introducir un valor propio, si así lo desea.

Para el ejemplo en el capítulo 4 no se requieren compuestos diseñados por el usuario, debido a que se utilizó el o-xileno como representante del grupo de todos los isómeros xilenos (pudo haberse definido uno intermedio, pero el o-xileno se encuentra en el rango medio de volatilidad entre los tres). Por lo tanto, el compuesto definido por el usuario debe ser eliminado de la lista, para lo cual se utiliza el botón de remover resaltado en la Fig. A.8 y este ya no será visible ni en la lista de compuestos ni en la tabla de composiciones molares.

Para equiparar el resultado de DESTILMATLABUSB con el algoritmo de PRO/II se extrajo los valores de parámetros de interacción binaria allí usados. El  $k_{12}$  es  $2,27e-3$ , así que debe ser introducido al presionar el botón de "Avanzadas", ante lo que aparece la tabla de parámetros

de interacción binaria de la Fig. A.9, donde se etiquetan numéricamente los parámetros que pueden ser introducidos, como indica la Figura. Al finalizar, presionar el botón “Atrás”.

The screenshot shows the 'Riguroso' software interface. On the left, a list of substances is displayed, with 'Definido por el usuario' (Defined by the user) highlighted. Below this list, input fields for critical properties are visible:  $T_c$  (K),  $P_c$  (kPa), and  $w$ . The main area contains a table for stage data with columns: Nombre, Etapa, Tipo, Temperatura, Presion, Cond. Térm, Error, and #Iteración. The table has 3 rows. Below the table, there are sections for 'Flujos distribuidos' (Estimated V, Estimated L, V converged, L converged) and 'Perfil temperaturas' (Estimation T, T converged). At the bottom, there are buttons for 'Calcular', 'Normalizar', and 'etapa', along with a 'max iter.' field set to 100. On the right, there is a schematic diagram of a distillation column and buttons for 'Balance de masa', 'Calcular Torre1', 'Paso a paso', and 'Avanzar 1 paso'.

Fig. A.6. Paso 6. Introducir una sustancia hipotética definida por el usuario

The screenshot shows the 'Riguroso' software interface. On the left, a list of substances is displayed, with 'UserDef.1()' highlighted. Below this list, input fields for critical properties are visible:  $T_c$  (borrar y en),  $P_c$  (borrar y en), and  $w$  (borrar y en). The main area contains a table for stage data with columns: Nombre, Etapa, Tipo, Temperatura, Presion, Cond. Térm, Error, and #Iteración. The table has 3 rows. Below the table, there are sections for 'Flujos distribuidos' (Estimated V, Estimated L, V converged, L converged) and 'Perfil temperaturas' (Estimation T, T converged). At the bottom, there are buttons for 'Calcular', 'Normalizar', and 'etapa', along with a 'max iter.' field set to 100. On the right, there is a schematic diagram of a distillation column and buttons for 'Balance de masa', 'Calcular Torre1', 'Paso a paso', and 'Avanzar 1 paso'.

Fig. A.7. Paso 7. Introducir las propiedades críticas de una sustancia definida por el usuario

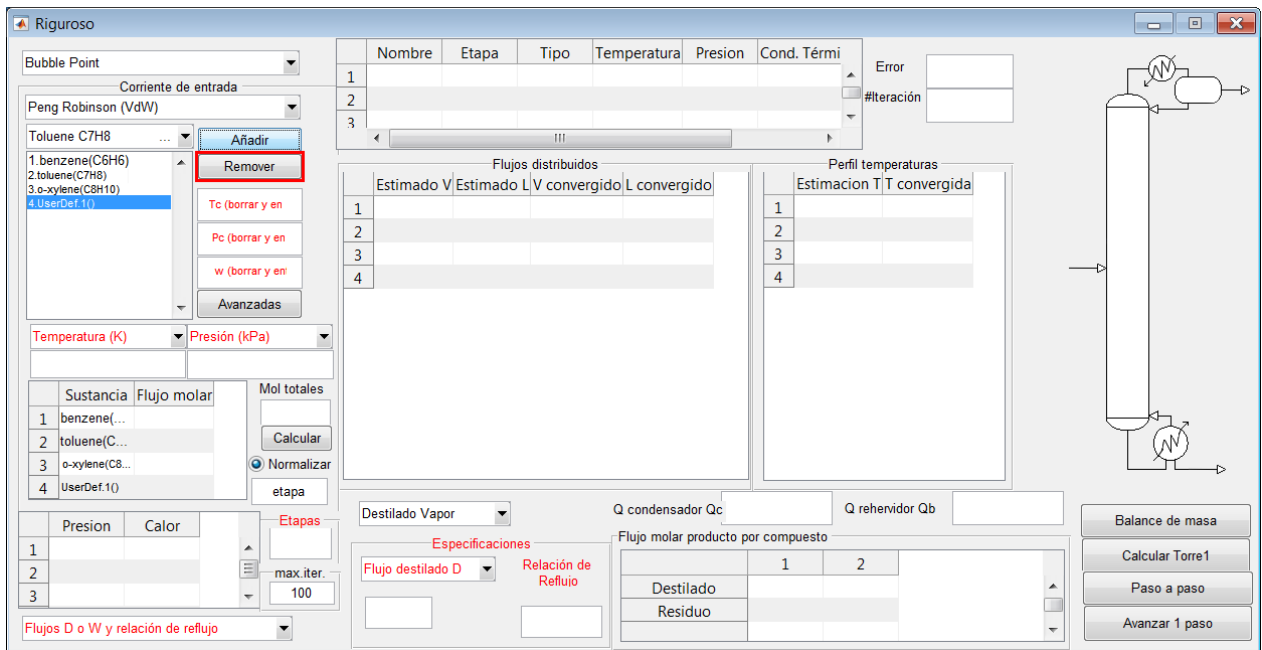


Fig. A.8 Paso 8. Remover una sustancia de la lista de sustancias

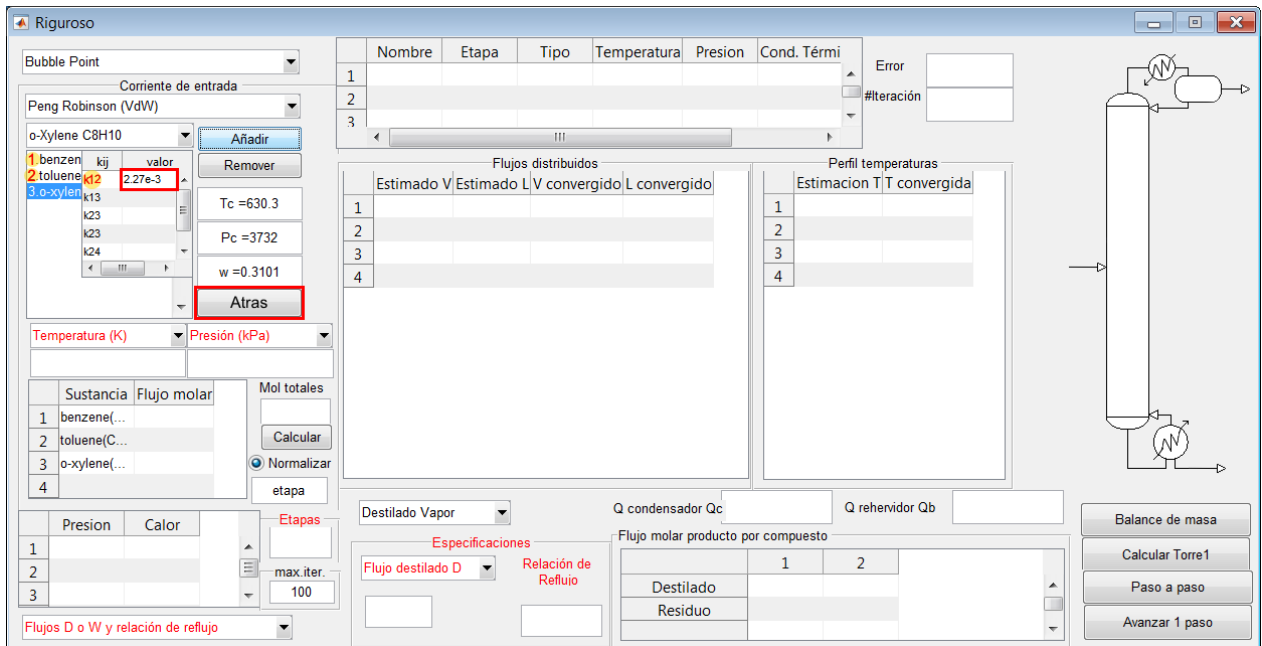


Fig. A.9 Paso 9. Introducir los parámetros de interacción binaria requeridos por los pares de sustancias definidos

Una vez que la mezcla de sustancias haya sido definida con todas las sustancias deseadas y los parámetros que fuesen requeridos, lo siguiente marcado en rojo es dar dos propiedades intensivas que definan a la corriente, que pudieran ser presión y temperatura o fracciones vaporizadas o condición térmica. Se introducen en las casillas marcadas en la Fig. A.10.

Fig. A.10. Paso 10: Introducir las variables intensivas que definan la fase de la corriente entre las variables resaltadas en la casilla correspondiente

Después es necesario proveer los flujos molares de cada componente en la tabla ubicada en la parte inferior, o alternatively se puede elegir introducir fracciones molares e indicar los moles totales y preferir que el algoritmo no vaya a normalizar los flujos.

Fig. A.11 Paso 11: Introducir flujos o fracciones molares en la tabla seleccionada y flujo total en la casilla "Mol totales" y/o normalizar

Fig. A.12 Paso 12. Introducir la etapa en la cual se ubica la corriente de alimentación y calcular la corriente

En cuanto alguna corriente haya sido introducida, la tabla de alimentaciones en la parte superior, señalizada en la Fig. A.13 mostrará las condiciones calculadas de la corriente. La misma se encuentra almacenada como variable global en el *workspace* de nombre “corriente1” y puede ser accedida citando a la mencionada variable global en el *command history* para obtención de datos adicionales.

Como es mostrado en la Fig. A.14, una vez que se tiene al menos una corriente para separar se podrá introducir el primero de los datos específicos de la torre, que es el número de etapas teóricas totales a utilizar. Para ello se hace uso de la casilla “Etapas” como muestra la figura. Adicionalmente ello producirá el relleno de las condiciones de presión y de calor en la torre en la tabla a la izquierda, que el usuario puede modificar para simular torres que no presenten perfil constante de presión o que tengan algún requerimiento energético.

Entonces se podrá terminar de especificar la torre haciendo uso del panel de especificaciones señalado en la Fig. A.15, en donde se debe introducir la condición dato de la relación de reflujo (2) y la condición dato del flujo molar de destilado, de 0,601, aparte de especificar la fase en que se desea el destilado, entre vapor y líquido, o una mezcla de fases.

**Riguroso**

Bubble Point

Corriente de entrada

Peng Robinson (VdW)

o-Xylene C<sub>8</sub>H<sub>10</sub>

1.benzene(C<sub>6</sub>H<sub>6</sub>)

2.toluene(C<sub>7</sub>H<sub>8</sub>)

3.o-xylene(C<sub>8</sub>H<sub>10</sub>)

Añadir

Remove

Tc =630.3

Pc =3732

w =0.3101

Avanzadas

Presión (kPa)

101.325

Fracción vapor

0

Sustancia

Flujo molar

1 benzene(...) 0.6

2 toluene(C...) 0.3

3 o-xylene(...) 0.1

4

Mol totales

1

Calcular

Normalizar

8

Presión

Calor

1 101.325

2 101.325 0

3 101.325 0

4

Etapas

max.iter.

100

Flujos D o W y relación de reflujo

Nombre	Etap	Tipo	Temperatura	Presión	Cond. ...	Entalpía ...
S-01	8	Alimentación	363.6643	101.325	1	12673.8066
2						
3						
4						

Error

#Iteración

Flujos distribuidos

	Estimado V	Estimado L	V convergido	L convergido
1				
2				
3				
4				

Perfil temperaturas

	Estimación T	T convergida
1		
2		
3		
4		

Destilado Vapor

Q condensador Qc

Q rehevador Qb

Especificaciones

Flujo destilado D

Relación de Reflujo

Flujo molar producto por compuesto

	1	2
Destilado		
Residuo		

Balance de masa

Calcular Torre1

Paso a paso

Avanzar 1 paso

Fig. A.13 Paso 13: Revisar en el panel superior que haya sido añadida la corriente deseada exitosamente

**Riguroso**

Bubble Point

Corriente de entrada

Peng Robinson (VdW)

o-Xylene C<sub>8</sub>H<sub>10</sub>

1.benzene(C<sub>6</sub>H<sub>6</sub>)

2.toluene(C<sub>7</sub>H<sub>8</sub>)

3.o-xylene(C<sub>8</sub>H<sub>10</sub>)

Añadir

Remove

Tc =630.3

Pc =3732

w =0.3101

Avanzadas

Presión (kPa)

101.325

Fracción vapor

0

Sustancia

Flujo molar

1 benzene(...) 0.6

2 toluene(C...) 0.3

3 o-xylene(...) 0.1

4

Mol totales

1

Calcular

Normalizar

8

Presión

Calor

1 101.325

2 101.325 0

3 101.325 0

4

Etapas

max.iter.

100

Flujos D o W y relación de reflujo

Nombre	Etap	Tipo	Temperatura	Presión	Cond. ...	Entalpía ...
S-01	8	Alimentación	363.6643	101.325	1	12673.8066
2						
3						
4						

Error

#Iteración

Flujos distribuidos

	Estimado V	Estimado L	V convergido	L convergido
1				
2				
3				
4				

Perfil temperaturas

	Estimación T	T convergida
1		
2		
3		
4		

Destilado Vapor

Q condensador Qc

Q rehevador Qb

Especificaciones

Flujo destilado D

Relación de Reflujo

Flujo molar producto por compuesto

	1	2
Destilado		
Residuo		

Balance de masa

Calcular Torre1

Paso a paso

Avanzar 1 paso

Fig. A.14 Paso 14. Introducir el número de etapas teóricas de dato al método riguroso de la torre de destilación



**Riguroso**

Bubble Point

Corriente de entrada

Peng Robinson (VdW)

o-Xylene C8H10

1. benzene(C6H6)

2. toluene(C7H8)

3. o-xylene(C8H10)

Añadir

Remove

Tc = 630.3

Pc = 3732

w = 0.3101

Avanzadas

Presión (kPa)

Fracción vapor

101.325

0

Sustancia

Flujo molar

1 benzene(... 0.6

2 toluene(C... 0.3

3 o-xylene(... 0.1

4

Mol totales

1

Calcular

Normalizar

8

Presión

Calor

1 101.325

2 101.325 0

3 101.325 0

4

Etapas

18

max.iter.

100

Flujos D o W y relación de reflujo

Nombre	Etapas	Tipo	Temperatura	Presión	Cond. ...	Entalpia ...
1 S-01	8	Alimentación	363.6643	101.325	1	12673.8066
2						
3						
4						

Error

#Iteración

Flujos distribuidos

	Estimado V	Estimado L	V convergido	L convergido
1				
2				
3				
4				

Perfil temperaturas

	Estimación T	T convergida
1		
2		
3		
4		

Destilado Líquido

Especificaciones

Flujo destilado D

Relación de Reflujo

60.1

Q condensador Qc

Q rehedidor Qb

Flujo molar producto por compuesto

	1	2
Destilado		
Residuo		

Balance de masa

Calcular Torre1

Paso a paso

Avanzar 1 paso

Fig. A.15 Paso 15. Introducir las especificaciones de la torre de destilación

Una vez que todos los datos requeridos están en color negro, como la Fig. A.16, se está listo para hacer uso de la modalidad de resolución más conveniente entre los ofrecidos por los botones resaltados en la figura.

**Riguroso**

Bubble Point

Corriente de entrada

Peng Robinson (VdW)

o-Xylene C8H10

1. benzene(C6H6)

2. toluene(C7H8)

3. o-xylene(C8H10)

Añadir

Remove

Tc = 630.3

Pc = 3732

w = 0.3101

Avanzadas

Presión (kPa)

Fracción vapor

101.325

0

Sustancia

Flujo molar

1 benzene(... 0.6

2 toluene(C... 0.3

3 o-xylene(... 0.1

4

Mol totales

1

Calcular

Normalizar

8

Presión

Calor

1 101.325

2 101.325 0

3 101.325 0

4

Etapas

18

max.iter.

100

Flujos D o W y relación de reflujo

Nombre	Etapas	Tipo	Temperatura	Presión	Cond. ...	Entalpia ...
1 S-01	8	Alimentación	363.6643	101.325	1	12673.8066
2						
3						
4						

Error

#Iteración

Flujos distribuidos

	Estimado V	Estimado L	V convergido	L convergido
1				
2				
3				
4				

Perfil temperaturas

	Estimación T	T convergida
1		
2		
3		
4		

Destilado Líquido

Especificaciones

Flujo destilado D

Relación de Reflujo

60.1

2

Q condensador Qc

Q rehedidor Qb

Flujo molar producto por compuesto

	1	2
Destilado		
Residuo		

Balance de masa

Calcular Torre1

Paso a paso

Avanzar 1 paso

Fig. A.16. Paso 16. Estimar el balance de masa o calcular los resultados del algoritmo de *Bubble Point*

**APÉNDICE C**  
**ARCHIVO EXPORTADO DE DATOS DEL SIMULADOR COMERCIAL PRO/II**

\$ Generated by PRO/II Keyword Generation System <version 9.1>

\$ Generated on: Sat Sep 05 20:53:30 2015

TITLE

TOLERANCE STREAM =0.01,-0.555556,0.01,0.01, TEMPERATURE=-  
0.0555556, &

PRESSURE=0.005, DUTY=0.001, MISCELLANEOUS=0.003,  
FLASH=3E-6

DIMENSION SI, TEMP=K, PRES=KPA, WT=KG, TIME=HR, LENGTH=M, &

FLLENGTH=MM, LIQVOL=M3, VAPVOL=M3, LDENSITY=KG/M3, &

VDENSITY=KG/M3, XDENSITY=DENS, SPVOL=M3/KG-MOL, &

SPVVOL=M3/KG-MOL, ENERGY=KJ, WORK=KJ/HR, DUTY=KJ/HR, &

CONDUCT=W/MK, HTCDEF=KJ/H, FOUL=MK/KW, VISCOSITY=PAS, &

KVIS=CST, SURFACE=N/M, STDTEMP=273.15, STDPRES=101.325, &

STDVAP (M3/KG-MOL)=22.414, PBASIS (KPA)=101.325

SEQUENCE SIMSCI

CALCULATION TRIALS=20, RECYCLE=ALL, TVPBASIS=310.928, &

RVPBASIS=APIN, COMPCHECK=CALC, MAXOPS=1000000, CDATA=FIX,  
&

FLASH=DEFAULT, DVARIABLE=ON

COMPONENT DATA

LIBID 1,BENZENE/2,TOLUENE/3,OXYLENE, BANK=SIMSCI,PROCESS

THERMODYNAMIC DATA

METHOD SYSTEM=PR, DENSITY(L)=PR, SET=PR01, DEFAULT

## STREAM DATA

PROPERTY STREAM=S1, PRESSURE=101.33, PHASE=L, RATE (M)=1, &  
COMPOSITION (M)=1,0.6/2,0.3/3,0.1, SET=PR01

## UNIT OPERATIONS

COLUMN UID=T1

PARAMETER TRAY=18, IO

FEED S1,9, SEPARATE

PRODUCT OVHD (M)=S2,0.601, BTMS (M)=S3, SUPERSEDE=ON

CONDENSER TYPE=BUBB

DUTY 1,1,,CONDENSER

DUTY 2,18,,REBOILER

PSPEC PTOP=101.33

PRINT PROPTABLE=PART

ESTIMATE MODEL=CONVENTIONAL, RRATIO=2

SPEC ID=COL1SPEC1, STREAM=S2, RATE (KGM/H),TOTAL,WET, &  
VALUE=0.601

SPEC ID=COL1SPEC2, RRATIO, PHASE=L, VALUE=2

VARY DNAME=CONDENSER,REBOILER

REBOILER TYPE=KETTLE

END

## APÉNDICE D

### ARCHIVO DE SALIDA GENERADO POR ALGORITMO DE LEWIS MATHESON

El número mínimo de etapas es  $N_{min}=12.052143$

y el reflujo mínimo es  $R_{min}=0.256865$

Con un reflujo de: 2.000000

Para el plato de tope, los flujos molares de liquido y vapor son:  $L=1.197608$   $V=1.796411$

Con ello, se calcula las relaciones  $V/L = 0.666667$  y  $D/V = 0.333333$

En la alimentacion  $x_F(LKey)/x_F(HKey)=6.000000$

El plato 1 subiendo del calderín tiene una composición:  
 $T=353.101393$  K

X	Y
0.992700	0.994129
0.007300	0.005871
0.000000	0.000000

El plato 2 subiendo del calderín tiene una composición:  
 $T=353.240417$  K

X	Y
0.985828	0.989548
0.014172	0.010452
0.000000	0.000000

El plato 3 subiendo del calderín tiene una composición:  
 $T=353.462179$  K

X	Y
---	---

0.974925 0.982279

0.025074 0.017720

0.000001 0.000001

El plato 4 subiendo del calderín tiene una composición:  
T=353.811399 K

X Y

0.957901 0.970930

0.042095 0.029067

0.000004 0.000003

El plato 5 subiendo del calderín tiene una composición:  
T=354.350410 K

X Y

0.931976 0.953647

0.068004 0.046340

0.000020 0.000013

El plato 6 subiendo del calderín tiene una composición:  
T=355.157443 K

X Y

0.893965 0.928306

0.105943 0.071633

0.000092 0.000061

El plato 7 subiendo del calderín tiene una composición:  
T=356.314071 K

X Y

0.841204 0.893132

0.158390 0.106597

0.000406 0.000271

El plato 8 subiendo del calderín tiene una composición:  
T=357.880279 K

X	Y
0.773135	0.847752
0.225164	0.151113
0.001702	0.001134

El plato 9 subiendo del calderín tiene una composición:  
T=359.881327 K

X	Y
0.692488	0.793988
0.300872	0.201585
0.006640	0.004427

El plato 10 subiendo del calderín tiene una composición:  
T=362.384230 K

X	Y
0.603837	0.734887
0.372471	0.249318
0.023692	0.015795

El plato 11 subiendo del calderín tiene una composición:  
T=365.764643 K

X	Y
0.508834	0.671552
0.416221	0.278485
0.074945	0.049963

El plato 12 subiendo del calderín tiene una composición:  
 $T=371.002070 \text{ K}$

X	Y
0.404576	0.602046
0.397945	0.266301
0.197480	0.131653

Pasando el plato de alimentación con la fracción de liviano a  
 pesado de 0.000500

El plato 1 bajando desde la alimentación tiene una  
 composición:  $T=378.919481 \text{ K}$

X	Y
0.297270	0.602046
0.303311	0.266301
0.399419	0.131653

El plato 2 bajando desde la alimentación tiene una  
 composición:  $T=396.988836 \text{ K}$

X	Y
0.079424	0.244365
0.271705	0.383629
0.648871	0.372005

El plato 3 bajando desde la alimentación tiene una  
 composición:  $T=406.065651 \text{ K}$

X	Y
0.017813	0.066290
0.203480	0.357794
0.778707	0.575917

El plato 4 bajando desde la alimentación tiene una composición:  $T=409.625186$  K

X	Y
0.003986	0.015926
0.158115	0.302024
0.837899	0.682050

El plato 5 bajando desde la alimentación tiene una composición:  $T=411.028797$  K

X	Y
0.001126	0.004623
0.134309	0.264941
0.864566	0.730436

El plato 5 es el fondo

Para el plato de tope, los flujos molares de liquido y vapor son:  $L=1.197608$   $V=1.796411$

Con ello, se calcula las relaciones  $V/L = 0.666667$  y  $D/V = 0.333333$

En la alimentacion  $x_F(LKey)/x_F(HKey)=6.000000$

El plato 1 subiendo del calderín tiene una composición:  $T=353.100771$  K

X	Y
0.992731	0.994154
0.007269	0.005846
0.000000	0.000000

El plato 2 subiendo del calderín tiene una composición:  $T=353.239214$  K



X	Y
0.985887	0.989592
0.014112	0.010408
0.000000	0.000000

El plato 3 subiendo del calderín tiene una composición:  
T=353.460063 K

X	Y
0.975029	0.982352
0.024971	0.017647
0.000001	0.000001

El plato 4 subiendo del calderín tiene una composición:  
T=353.807890 K

X	Y
0.958071	0.971047
0.041925	0.028950
0.000004	0.000003

El plato 5 subiendo del calderín tiene una composición:  
T=354.344862 K

X	Y
0.932241	0.953827
0.067739	0.046159
0.000020	0.000013

El plato 6 subiendo del calderín tiene una composición:  
T=355.149104 K

X	Y
0.894354	0.928569

0.105555 0.071370

0.000091 0.000061

El plato 7 subiendo del calderín tiene una composición:  
T=356.302297 K

X Y

0.841733 0.893489

0.157863 0.106242

0.000405 0.000270

El plato 8 subiendo del calderín tiene una composición:  
T=357.864930 K

X Y

0.773789 0.848193

0.224513 0.150675

0.001697 0.001131

El plato 9 subiendo del calderín tiene una composición:  
T=359.863176 K

X Y

0.693212 0.794475

0.300161 0.201107

0.006627 0.004418

El plato 10 subiendo del calderín tiene una composición:  
T=362.365011 K

X Y

0.604538 0.735359

0.371799 0.248866

0.023663 0.015776

El plato 11 subiendo del calderín tiene una composición:  
 $T=365.746731\text{ K}$

X	Y
0.509414	0.671943
0.415685	0.278123
0.074901	0.049934

El plato 12 subiendo del calderín tiene una composición:  
 $T=370.988366\text{ K}$

X	Y
0.404959	0.602306
0.397584	0.266056
0.197457	0.131638

Pasando el plato de alimentación con la fracción de liviano a pesado de 0.000500

El plato 1 bajando desde la alimentación tiene una composición:  $T=378.912619\text{ K}$

X	Y
0.297450	0.602306
0.303089	0.266056
0.399461	0.131638

El plato 2 bajando desde la alimentación tiene una composición:  $T=401.036654\text{ K}$

X	Y
0.072652	0.243353
0.174495	0.272276
0.752853	0.484370

El plato 3 bajando desde la alimentación tiene una  
composición:  $T=411.583841\text{ K}$

X	Y
0.014364	0.059594
0.083645	0.167158
0.901991	0.773248

El plato 4 bajando desde la alimentación tiene una  
composición:  $T=415.042855\text{ K}$

X	Y
0.002694	0.011947
0.043009	0.092894
0.954297	0.895158

El plato 5 bajando desde la alimentación tiene una  
composición:  $T=416.125236\text{ K}$

X	Y
0.000532	0.002407
0.026974	0.059677
0.972494	0.937916

El plato 5 es el fondo

Corrigiendo las composiciones para la alimentación Plato 12

x	y
---	---

## APÉNDICE E

### HISTORIAL DE CÁLCULO DEL EJEMPLO 2

A continuación se reproducen los pasos para obtener los resultados demostrados para el ejemplo 2, un *script* de cálculo, que por ser una torre de difícil convergencia no puede reproducirse mediante las limitadas capacidades de la interfaz gráfica.

```
%Defino un conjunto de sustancias C2H6, C3H8, nC4H10, nC5H10, nC6H12

sust = [Sustancia('ethane'), Sustancia('Propane'), Sustancia('Butane'), Sustancia('Pentane'),
Sustancia('Hexane')];

% la concentracion de la alimentación 1 es:

conc1 = [0.5342, 0.3163, 0.0612, 0.0701, 0.0183];

%la concentracion de la alimentacion 2 es:

conc2 = [0.2236, 0.5194, 0.1119, 0.1230, 0.0221];

%kij ideales, también se hubiera podido usar los de Peng y Robinson arrojados por PRO/II

kij = 0;

mezcla1 = Mezcla(sust, conc1, kij);

mezcla2 = Mezcla(sust, conc2, kij);

%Se utiliza la ecuacion de estado de Peng y Robinson

EdE = PREdE();
```

%Se usan las reglas de mezclado de Van der Waals

MEdE = RMVdW(EdE);

% Teniendo todo se puede calcular la primera corriente, a 1320 kPa y en su

% punto de burbuja y un flujo molar de 154.3 kg-mol/h, de la forma:

P = 1320;

vargin1 = 'P'; %P significa el argumento introducido es una presión en kPa

x = 0;

vargin2 = 'x'; %x significa utilizar fracción vaporizada de argumento

flujo1 = 154.3;

varginflujo = 'm'; %m significa utilizar un flujo en base molar

corriente1 = Corriente(mezcla1, P, vargin1, x, vargin2, flujo1, varginflujo, MEdE, 'S-01');

%El flujo de la corriente 2 es de 64.02 kg-mol

flujo2 = 64.02;

corriente2 = Corriente(mezcla2, P, vargin1, x, vargin2, flujo2, varginflujo, MEdE, 'S-02');

alim = {4, corriente1, 6, corriente2};

% El número de etapas teóricas es 9, incluyendo rehervidor y el condensador

```
etapas = 9;
```

```
%La columna es prácticamente isobárica a 1320 kPa salvo por el condensador
```

```
%a menor presión (1260 kPa) y el rehervidor a la mayor presión (1380 kPa)
```

```
presion_por_plato = ones(1,9).*P;
```

```
presion_por_plato(1) = 1260;
```

```
presion_por_plato(end) = 1380;
```

```
%Se desea condensador total, es decir, una salida de destilado líquido en
```

```
% el tope, siendo esta de 79.8 kg-mol/hr. De ser el destilado vapor, no es
```

```
% necesario definir
```

```
salidatope = {1,1,79.8};
```

```
%Un par de corrientes placeholders en las cuales se definirían los
```

```
%productos (aún no implementado)
```

```
dest = Corriente.empty(0,1);
```

```
fondo = Corriente.empty(0,1);
```

```
% El calor intercambiado por plato es despreciable, la columna es adiabática
```

```
calor = 0;
```

%El reflujo (L/D) es de 2

reflujo = 2.5;

% La otra condición es dada por el balance de masa global, con la cual al

% entrar 218.32 kg-mol alimentados y salir en el tope 79.8 kg-mol, en el

% fondo van a salir 138.52 kg-mol y se le suministra 'B' para indicar el

% total de fondo

flujofondo = 138.52;

condflujo = 'B';

%Se le restringirá que el máximo a realizar son 200 iteraciones

iteraciones = 200;

%Ahora sí teniendo todos los datos requeridos se procede a definir la torre

torreBP = BubblePoint(etapas, alim, dest, fondo, salidatope, presion\_por\_plato, calor,  
reflujo, flujofondo, condflujo, iteraciones);

%Se realiza el balance de materia

torreBP.balanmasa();



%Inicio del método Bubble Point de Wang y Henke

torreBP.wanghenke();

%Para la torre de Newton Raphson se define un amortiguamiento bajo de 0.45

damping = 0.45;

torreNS = NaphtaliSandholm(etapas, alim, dest, fondo, salidatope, presion\_por\_plato, calor, iteraciones, flujofondo, condflujo, [], [], damping, reflujo, 'R', flujofondo, condflujo);

% Se realiza una distribución ideal de los compuestos mediante un balance de masa

torreNS.balanmasa();

%Puesto que no se posee generador de estimados iniciales, se procede a utilizar como %estimados de Newton Raphson los valores finales de la convergencia de *BubblePoint*

torreNS.perfil\_t = torreBP.perfil\_t; %Estimados de temperatura

torreNS.perfil\_v = torreBP.perfil\_v; %Estimados de flujo molar de vapor por plato

torreNS.perfil\_l = torreBP.perfil\_l; %Estimados de flujo molar de líquido por plato

torreNS.perfil\_li = torreBP.perfil\_li; %Estimados de flujo molar líquido por componente por plato

torreNS.perfil\_vi = torreBP.perfil\_vi; %Estimados de flujo molar vapor por componente por plato

torreNS.perfil\_k = torreBP.perfil\_k; %Estimados de coeficiente de reparto por componente

por plato

```
torreNS.xsubj = torreBP.xsubj; %Estimados de fracción molar de fase líquida por
componente por plato
```

```
torreNS.ysubj = torreBP.ysubj; %Estimados de fracción molar de fase vapor por
componente por plato
```

```
%Debido a que el flujo de vapor destilado debe restringirse a cero, para tener un estimado
%consistente se debe restringir a 0 el destilado vapor
```

```
torreNS.dflujo = 0;
```

```
torreNS.perfil_v(1) = 0;
```

```
%El estimado de flujo de residuo líquido por la última etapa es el mismo que con el método
%de BubblePoint
```

```
torreNS.bflujo = torreBP.bflujo;
```

```
%Finalmente teniendo un set de estimados iniciales que es consistente y que debería
%generarse acoplando un algoritmo de Youseff para la estimación de perfiles por plato y por
%componente mediante un cálculo corto
```

```
torreNS.newtonraphson2(); %El 2 hace referencia a la especificación de reflujo y flujo
desilado
```