

Ajuste de datos de equilibrio líquido-vapor a la ecuación NRTL empleando algoritmos de optimización metaheurística

Emilio Alba Robles ¹, Brayán Alejandro González Abundis ², Sergio Yovani Rodríguez Rojo ³, Oscar Daniel Lara Montaña ⁴, Fernando Israel Gómez Castro ⁵

^{1,2,3,4,5} Departamento de Ingeniería Química, División de Ciencias Naturales y Exactas, Campus Guanajuato, Universidad de Guanajuato, Noria Alta S/N Col. Noria Alta, Guanajuato, Guanajuato, 36050, México.
fgomez@ugto.mx⁵

Resumen

En este trabajo se presenta una estrategia para el ajuste de datos experimentales a un modelo clásico para la predicción del equilibrio líquido-vapor. El ajuste se lleva a cabo por medio de la minimización de tres funciones objetivo de mínimos cuadrados, considerando por una parte la desviación con respecto a las composiciones en fase vapor, por otra las desviaciones con respecto a los coeficientes de actividad, y finalmente las desviaciones con respecto a la energía libre de Gibbs. El modelo empleado para el cálculo de los coeficientes de actividad es la ecuación NRTL, el cual se acopla a un entorno de optimización metaheurística codificado en Python. La minimización de las desviaciones se lleva a cabo empleando diversos métodos de optimización, tales como enjambre de partículas, optimizador del lobo gris, entre otros. Se analiza el desempeño de los distintos algoritmos en la solución del problema de ajuste de datos.

Palabras clave: equilibrio líquido-vapor; ecuación NRTL, optimización metaheurística.

Introducción

Mediante el empleo de datos experimentales es posible desarrollar modelos matemáticos que aproximen el comportamiento del fenómeno natural por medio de ecuaciones. En el caso del equilibrio de fases tradicionalmente se emplean modelos termodinámicos, los cuales se dividen en dos categorías principales: las ecuaciones de estado y los modelos de coeficiente de actividad. Para que los modelos termodinámicos predigan de manera adecuada el equilibrio de fases es necesario contar con parámetros de interacción adecuados, específicos para las mezclas estudiadas. Dichos parámetros pueden obtenerse por ajuste de datos experimentales de equilibrio de fases. La información obtenida del ajuste de datos permite conocer los coeficientes de actividad de la mezcla, los cuales a su vez son una medida de la desviación de la idealidad del comportamiento de la mezcla.

Los modelos termodinámicos son importantes ya que permiten predecir el comportamiento de las mezclas. Los modelos de coeficiente de actividad toman en cuenta variables que un modelo ideal no toma en cuenta, como lo son el tamaño de molécula, la polaridad molecular, la carga molecular, etcétera. Existen diferentes modelos, con diferencias notables entre sí, siendo algunos más sencillos que otros, pero todos desarrollados con el fin de considerar las variables que permitan que su ajuste se apegue más a la realidad. Uno de esos modelos está dado por la ecuación NRTL.

El modelo NRTL (Non Random Two Liquids, por sus siglas en inglés), es un modelo derivado del modelo de Wilson. Sin embargo, NRTL puede ser aplicable tanto a mezclas completamente miscibles, como a mezclas parcialmente miscibles. Además, toma en cuenta la energía de la mezcla. Dentro de los parámetros que son manejados por las ecuaciones de este modelo se encuentra la temperatura (T), presión (P), concentración (C), presiones parciales (P_i), y los coeficientes de actividad (γ).

La ecuación NRTL involucra parámetros que dependen del tipo de componentes en la mezcla. La determinación de esos parámetros requiere del ajuste de datos experimentales haciendo uso de herramientas de optimización. Dentro de esta área de las matemáticas, se cuenta con un conjunto de técnicas denominadas metaheurísticas. Estas estrategias son métodos aproximados para la resolución de problemas de optimización combinatoria, y son especialmente útiles para problemas complejos en los cuales los métodos clásicos fallan, por la no linealidad o la discontinuidad de las funciones involucradas. Este tipo de optimización

se ayuda en la inteligencia artificial, teniendo múltiples métodos que se usan para dar solución a los problemas, uno de ellos, es la optimización por enjambre de partículas PSO (*Particle Swarm Optimization*, por sus siglas en inglés). PSO tiene como base la búsqueda de un punto de interés (punto óptimo) usando el comportamiento de múltiples individuos, denominados población, haciendo referencia a los enjambres que se pueden encontrar en la naturaleza, utilizando la interacción social y la mente colectiva en lugar de una mente individual, lo cual ayuda a que la población involucrada se mueva a través de la región de espacio del problema, haciendo iteraciones con cada individuo, los cuales se colocan en diversos puntos aleatorios dentro de la región de búsqueda. Las iteraciones se dan con base a una función objetivo que es evaluada y como resultado se obtiene un valor, el que es comparado con el de los demás individuos del enjambre con la finalidad de compararlo y determinar al individuo que presente el mejor valor ("*best-fitness*"), el cual puede ser un solo individuo o una serie de estos. Con este valor determinado, los individuos del enjambre cambiarán su rumbo con base a la trayectoria que poseen y con base al valor "*best-fitness*", lo que provoca que, en algún punto, todos los individuos se acerquen al mejor valor obtenido por el "*best-fitness*". Así, los individuos se comportan como un enjambre, convergiendo a un punto, el punto óptimo.

Otros métodos de optimización metaheurística son el Optimizador del Lobo Gris (GWO, Gray Wolf Optimization) y el Algoritmo Jaya. El método GWO está inspirado en el comportamiento de los lobos grises en la naturaleza, ya que estos animales se organizan en manadas que conviven en comunidad, haciéndolos seres sociales con sus semejantes y que tengan objetivos en común, como lo es la caza o el mantener a la manada unida, en la caza se unen la fuerza y la estrategia para lograr someter presas de mayor tamaño, en la toma de decisiones se tiene una jerarquía muy consistente, se tiene un lobo alfa, que es quien guía a la manada y toma las decisiones, lobos beta que son aquellos que aportan al alfa y podrán sustituirlo en algún futuro, otro escalón de jerarquía más abajo es denominado lobos delta (donde se encuentran los lobos ancianos, que algunos pudieron ser alfas o betas en algún determinado tiempo, exploradores y protectores de la manada), además llega a existir otra jerarquía, los lobos omega, el peldaño inferior que fungen como niños de la manada. El método GWO se basa en los patrones de caza de estos caninos, donde estos rastrean, persiguen y se acercan a la presa para después rodearla y buscar que no tenga escapatoria para finalmente atacar, de esta forma surge el método. El algoritmo de optimización Jaya (palabra sanscrita que significa victoria), trata constantemente de acercarse al éxito, para ello evalúa los valores como mejores o peores, de esta manera si el valor se considera mejor lo toma acercando más a la mejor solución, caso contrario a los valores considerados peores, de los cuales trata de alejarse.

Dentro de los métodos metaheurísticos, se tienen diversos factores para tener en cuenta, uno de ellos es la hipersensibilidad en los parámetros de introducción en los métodos, lo cual hace que, si se cambian los valores introducidos al inicio, se pueden obtener valores completamente diferentes, además de que, al ser métodos basados en optimización combinatoria aleatoria, los valores en cada iteración también serán diferentes por la aleatoriedad de los valores iterativos. Debido a esto, en este trabajo se evaluará la efectividad de los tres métodos descritos en la solución del problema de ajuste de datos experimentales a la ecuación NRTL.

Planteamiento del problema de optimización

El presente trabajo se enfoca en el cálculo de los coeficientes de actividad de diferentes mezclas de sustancias. Se determinarán los parámetros del modelo NRTL que arrojan la menor desviación con respecto a los datos experimentales. Asimismo, se aplican tres métodos de optimización metaheurística, con el fin de determinar las diferencias numéricas entre estas estrategias, evaluando su desempeño en la solución de un problema altamente no lineal como el de ajuste de datos experimentales a un modelo de coeficientes de actividad. El procedimiento para llevar a cabo la resolución de esta incógnita es mediante programación, por lo que se emplea el software libre Python como base de programación, en el cual se introducen las ecuaciones, modelos y parámetros que representan al sistema.

Como se mencionó anteriormente, el modelo termodinámico seleccionado es el NRTL. Las ecuaciones que se emplean se muestran a continuación.

$$\frac{G^E}{x_1 x_2 RT} = - \frac{G_{21} \tau_{21} + G_{12} \tau_{12}}{(x_1 + x_2 G_{21})(x_2 + x_1 G_{12})} \quad (1)$$

$$\ln(\gamma_1) = x_2^2 \left[\tau_{21} \left(\frac{G_{21}}{x_1 + x_2 G_{21}} \right)^2 + \frac{G_{12} \tau_{12}}{(x_2 + x_1 G_{12})^2} \right] \quad (2)$$

$$\ln(\gamma_1) = x_1^2 \left[\tau_{12} \left(\frac{G_{12}}{x_1 + x_2 G_{12}} \right)^2 + \frac{G_{21} \tau_{21}}{(x_2 + x_1 G_{21})^2} \right] \quad (3)$$

Donde:

- $G_{12} = \exp(-\alpha \tau_{12})$, $G_{21} = \exp(-\alpha \tau_{21})$
- $\tau_{12} = \frac{b_{12}}{RT}$
- $\tau_{21} = \frac{b_{21}}{RT}$
- α, b_{12}, b_{21} son parámetros específicos para cada sustancia.

Dentro de este modelo, los parámetros a determinar son α y b_{ij} , los cuales se determinan a partir de datos experimentales.

El problema de optimización que se aborda implica obtener el valor de la función objetivo más pequeño posible. Los valores experimentales permanecen fijos, por lo que las variables de decisión están dadas por los parámetros del modelo. Es decir, es necesario calcular los coeficientes o parámetros que permiten que los valores predichos por el modelo sean lo más aproximados posible a los datos experimentales. La función objetivo indica el error que se tiene entre los valores experimentales y los valores de modelo, por lo que es indispensable que el error sea lo menor posible para que el modelo sea válido.

Como funciones objetivo se toman las reportadas por Tosun (2021), con la finalidad de tener un contraste del desempeño de dichas funciones, dado que, como se menciona con anterioridad, algunas expresiones usan diferentes parámetros. Por tanto, se desea determinar cuáles son las variables que tienen un mayor impacto en la determinación de parámetros adecuados.

Funciones objetivo que son utilizadas:

- Función objetivo 1:

$$F = \sum_{i=1}^N [(y_{1,exp} - y_{1,calc})_i^2 + (y_{2,exp} - y_{2,calc})_i^2]$$

- Función objetivo 2:

$$F = \sum_{i=1}^N [(\gamma_{1,exp} - \gamma_{1,calc})_i^2 + (\gamma_{2,exp} - \gamma_{2,calc})_i^2]$$

- Función objetivo 3:

$$F = \sum_{i=1}^N \frac{(\tilde{G}_{exp}^{ex} - \tilde{G}_{calc}^{ex})_i^2}{RT}$$

La primera función se basa en la fracción mol de los componentes de la mezcla, la segunda emplea los coeficientes de actividad de las mezclas y, finalmente, la tercera función objetivo está basada en la energía libre de Gibbs en exceso.

Es necesario agregar algunas ecuaciones adicionales que se usan dentro del código de programación, ya que son necesarias para los cálculos de los equilibrios de fases, siendo esta ecuación la ley de Raoult,

relacionada con la presión parcial ejercida por cada componente del sistema, teniendo la particularidad que la sumatoria de todas las presiones parciales de cada componente es el valor de la presión total del sistema, además de que con esta ley es posible calcular la concentración de los componentes. Otra ecuación empleada es la Ley de Raoult modificada, esta ecuación describe el equilibrio de fases entre el líquido y el vapor de un sistema de un gas real, considerando las fugacidades de los componentes dentro de la mezcla.

Ley de Raoult $P_i = P_i^0 x_i$

Presión total a partir de presiones parciales $P_{Total} = \sum P_i$

Ley de Raoult modificada $y_i P = \gamma_i x_i P_i^{sat}$

Desarrollo del entorno de optimización

Para satisfacer los objetivos planteados, se ha empleado el lenguaje de programación Python dado que es un lenguaje de código abierto, sencillo, multiparadigma y muy flexible, además de que es posible realizar de una manera automatizada los algoritmos de optimización metaheurísticas (PSO, GWO, Jaya).

Con este lenguaje se elaboró una herramienta de cálculo que recibe el nombre de “Ajuste parámetros NRTL”. Esta herramienta permite trabajar con datos propios experimentales dados en formato csv, siendo necesario que tengan los encabezados específicos que se especifican en el manual de uso.

El programa consta de cuatro módulos principales:

- **Main:** es donde está programada la interfaz gráfica. Se elaboró empleando el patrón modelo – vista para tener una mejor organización y contiene cuatro clases. La clase Vista es donde se programa lo relacionado a la interfaz gráfica, como son los botones o los textos y dónde se posicionan, así como algunas funciones que modifican la apariencia de ésta; la clase Modelo provee métodos necesarios para trabajar con los datos dados, leer archivos de la computadora y llamar a otros módulos para poder acceder a sus distintas funcionalidades y hacer los cálculos necesarios; la clase MainWindow es una clase maestra, es la responsable de la creación de la interfaz y en ésta se conectan las clases entre sí; la clase MplCanvas es la responsable de crear gráficos compatibles con la interfaz y finalmente, la clase EscrituraExcel tiene métodos que van a permitir registrar los resultados en algún archivo .xlsx que el usuario especifique.
- **Evaluaciones:** este módulo se emplea para la creación de objetos para cada mezcla binaria. Están definidas las funciones objetivo como parte de esta clase. Como parte del módulo también se definieron otras funciones maestras que reciben una instancia del objeto mezcla_bin y se encargan de evaluar los métodos de optimización para ajustar los parámetros a los datos experimentales dados. Existen otras funciones que le dan formato a los datos para que puedan ser empleados por otras funciones.
- **Optimizadores:** es el módulo que contiene todos los algoritmos de optimización en forma de funciones. Para los casos de interés, se emplearon las funciones “PSO”, “gwo” y “jaya”. Todas estas funciones reciben los valores límites de operación (lb y ub), la dimensión (dim), el tamaño de la población o el número de individuos (PopSize), el número de iteraciones (iters) y la función objetivo, que es la que se quiere minimizar. Este módulo contiene más métodos además de los mencionados previamente.
- **Termo:** este último módulo contiene funciones termodinámicas, como para calcular los coeficientes de actividad según el modelo NRTL, la ley de Raoult modificada o para calcular la presión de saturación. Contiene también algunas otras para darle formato a los datos.

Los cuatro módulos se relacionan entre sí de manera ciclada, de manera que se reproduce el algoritmo hasta converger con el número de iteraciones solicitadas dentro del ejecutable, para una manera más visual, véase la Figura 1.

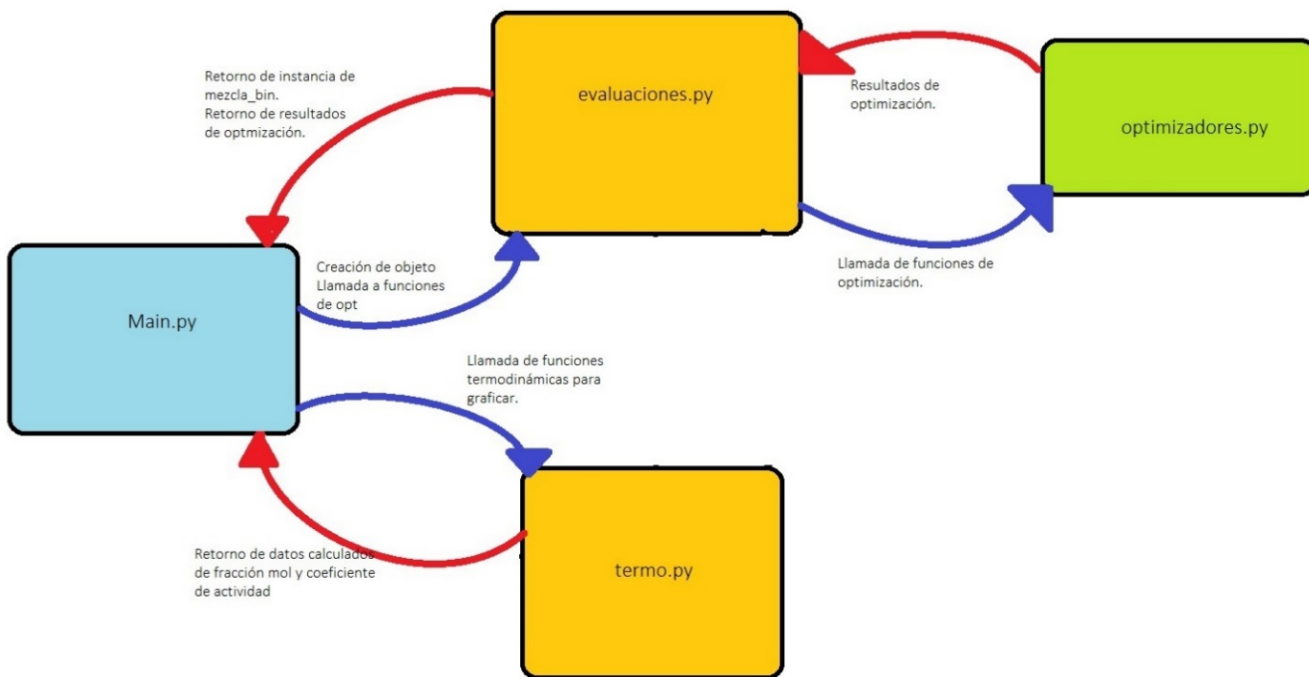


Figura 1. Relación de los módulos contenido en el entorno de programación.

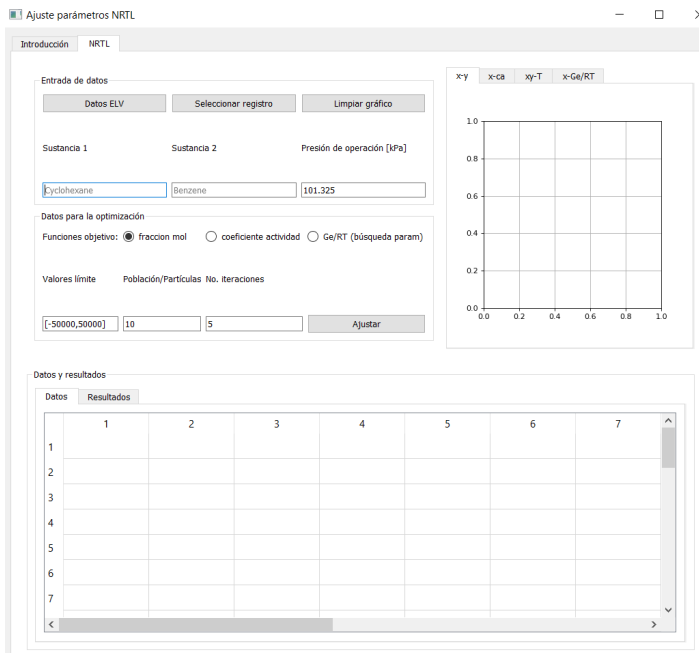


Figura 2. Entorno de programación, interfase final para usuarios.

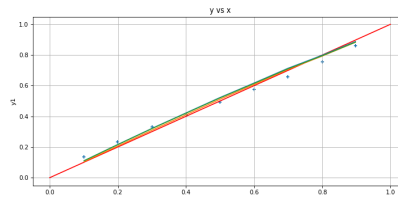


Figura 3. Gráficos realizados por el software, mezcla ciclohexano-benceno.

```

=====
Método: pso
a: 0.6000
b: [30236.5764535 49120.8335 ]
error: 0.1154 tiempo: 1.2179
-----
Método: jaya
a: 0.6000
b: [50000. -196.02770746]
error: 0.1102 tiempo: 1.2179
-----
Método: gwo
a: 0.6000
b: [50000. 50000.]
error: 0.1129 tiempo: 1.2179
=====
  
```

Figura 4. Resultados de análisis por el software, mezcla ciclohexano-benceno.

Para realizar los análisis correspondientes es necesario que el usuario cuente con los siguientes elementos: los coeficientes de Antoine para la mezcla que es analizada, la presión a la que se encuentra el sistema de

interés. Con estos requisitos adicionales, el usuario puede hacer uso de la interfaz gráfica, en esta se introducirán los valores límite de la región de búsqueda, además de los valores del tamaño de la población deseada o número de partículas de iteraciones que se desean realizar.

Con los requisitos introducidos en el software, este mostrará un gráfico donde se representa el comportamiento ideal de la mezcla, además de mostrar el comportamiento de los modelos matemáticos de las funciones objetivo, lo que ayuda a visualizar los resultados de manera gráfica. El mismo software, presenta los resultados de manera numérica en la pestaña de resultados, donde además de mostrar los parámetros buscados, se muestra el margen de error, con base a la función de mínimos cuadrados y el tiempo que le toma de cómputo que fue necesario para la iteración deseada.

Todas las iteraciones que se realizan pueden ser almacenados los resultados de estas en archivos .xlsx (archivos Excel) donde es posible hacer comparaciones del comportamiento de los sistemas con modificaciones a los parámetros para obtener resultados acordes al sistema, estos resultados están sujetos al error asociado al método, dado que al ser métodos numéricos se tienen errores asociados a los métodos.

Casos de estudio

Para el análisis de la desviación de la idealidad de mezclas, se examinaron datos experimentales de equilibrio liquido-vapor de cinco diferentes mezclas medidas a una presión 101.325 kPa, siendo las siguientes: ácido acético - N-Metil-2-pirrolidona (nmp), ácido fluorhídrico - agua, alcohol alílico - etilenglicol, ciclohexano - benceno y benceno -dimetilacetamida (dmac).

De estas mezclas, las de ácido acético – nmp, alcohol alílico - etilenglicol y benceno – dmac comparten la característica de ser mezclas sin azeótropos, por lo que su análisis en bastante similar, por ello se utilizara la mezcla alcohol alílico - etilenglicol como representante de este tipo de mezclas. Los datos experimentales de equilibrio liquido – vapor de esta mezcla se muestran en la Tabla 1.

Tabla 1. Datos experimentales para la mezcla alcohol alílico – etilenglicol (Dong et al., 2019).

T(K)	x1	y1	γ_1	γ_2
372.93	0.8721	0.995	1.0292	1.8612
376.27	0.7718	0.9905	1.033	1.665
379.17	0.6584	0.9863	1.0947	1.3829
383.96	0.5308	0.9786	1.1534	1.2383
387.16	0.4454	0.9733	1.236	1.1186
392.16	0.3599	0.9638	1.2998	1.0367
396.45	0.311	0.9532	1.3102	1.0218
401.16	0.266	0.9387	1.3181	1.0174
406.6	0.2216	0.9186	1.3322	1.0059
411.13	0.1901	0.8978	1.345	1.0029
417.52	0.153	0.8622	1.3621	0.9965
426.76	0.1103	0.7927	1.388	0.9962
433.5	0.0852	0.7261	1.4101	0.9969
437.49	0.0722	0.6795	1.4258	0.9965
442.15	0.0588	0.618	1.441	0.9944
448.76	0.0418	0.5124	1.4669	0.9963

455.8	0.026	0.3755	1.5055	0.9979
459.55	0.0184	0.2906	1.534	0.999
467.25	0.0046	0.0872	1.6018	1.0016
469.4	0.0012	0.0235	1.5939	1.0017

Por otro lado, las mezclas restantes presentan azeótropos, esto implica que en cierta composición la mezcla se comporta como si estuviera formada por un solo componente por lo que al hervir su fase vapor presentara la misma composición que fase líquida. La mezcla ciclohexano - benceno contiene diversos azeótropos a lo largo de las condiciones de operación. Los datos experimentales de equilibrio líquido – vapor de esta mezcla se presentan en la Tabla 2.

Tabla 2. Datos experimentales para la mezcla ciclohexano - benceno (Dong et al., 2019).

T(K)	x1	y1	γ_1	γ_2
356.7	0.1008	0.1352	1.2285	0.863
356.28	0.1989	0.2343	1.093	0.8686
355.89	0.301	0.33	1.0291	0.8815
355.4	0.4012	0.41	0.9734	0.9195
355.1	0.4993	0.4915	0.9463	0.9568
354.57	0.5998	0.5744	0.9354	1.0182
354.27	0.6989	0.6591	0.9294	1.0936
353.56	0.8001	0.7547	0.9497	1.2114
353.26	0.8976	0.861	0.9747	1.353

La mezcla ácido fluorhídrico – agua presenta un solo azeótropo. Los datos experimentales de equilibrio líquido – vapor de esta mezcla se muestran en la Tabla 3.

Tabla 3. Datos experimentales para la mezcla ácido fluorhídrico – agua (Dong et al., 2019).

T(K)	x1	y1	γ_1	γ_2
373.92	0.0495	0.0078	0.0093	0.9841
375.32	0.0919	0.0183	0.0151	0.9505
379.93	0.1894	0.064	0.0371	0.8315
380.93	0.228	0.1057	0.0493	0.7745
383.23	0.2794	0.1781	0.0696	0.6957
384.73	0.3382	0.3053	0.0978	0.6023
385.03	0.344	0.3207	0.101	0.5932
385.33	0.3517	0.3401	0.1054	0.581
385.23	0.3578	0.3571	0.1086	0.5711
385.13	0.3583	0.3582	0.1088	0.5702

385.03	0.3673	0.3859	0.1138	0.5554
384.63	0.3967	0.4748	0.1311	0.5078
381.43	0.444	0.633	0.1599	0.4298
374.42	0.5028	0.8104	0.1983	0.334
371.62	0.5219	0.862	0.2115	0.304
363.82	0.5604	0.9218	0.2374	0.2444
359.22	0.5817	0.9701	0.2525	0.2133
351.62	0.6165	0.9889	0.2787	0.1665
347.12	0.6382	0.9856	0.2967	0.1403
334.72	0.6984	0.9867	0.3534	0.0794
318.03	0.7976	0.9922	0.4874	0.0199
306.43	0.8793	0.9945	0.66	0.0025

Para cada una de las diferentes funciones objetivo se trabajó mediante los tres métodos de optimización metaheurística, con el fin de poder comparar los métodos con base en la cantidad de iteraciones y del número de individuos que son necesarios para obtener un resultado acorde, dado por el margen de error del método utilizado. Todas las mezclas fueron probadas con el programa de tal forma que se realizaron una serie de combinaciones de hiperparámetros de los algoritmos de optimización para cada mezcla, variando el tamaño de la población desde 5 hasta 25 individuos, así como el número de iteraciones realizadas desde 5 hasta 50. Esto con el fin de observar con cual combinación de variables los métodos metaheurísticos convergían en un error menor y con un tiempo de cómputo adecuado.

Resultados

En esta sección se presentan los resultados obtenidos para las mezclas estudiadas. Se realizó un análisis de sensibilidad a cada mezcla para determinar el número adecuado de iteraciones y tamaño de la población. Los resultados obtenidos para las distintas mezclas se presentan en las Tablas 4 a 9.

Tabla 4. Resultados obtenidos con la primera función objetivo para la mezcla ciclohexano-benceno.

Población	Iteración	GWO		Jaya		PSO
		Error	Tiempo	Error	Tiempo	Error
30	5	0.113	0.018	0.118	0.022	0.196
30	10	0.113	0.037	0.113	0.039	0.115
30	15	0.113	0.054	0.112	0.053	0.336
30	20	0.113	0.073	0.113	0.073	0.129
30	25	0.113	0.090	0.110	0.086	0.172
30	30	0.109	0.108	0.111	0.107	0.196
30	35	0.113	0.127	0.111	0.123	0.316
30	40	0.110	0.149	0.113	0.137	0.119
30	45	0.113	0.163	0.113	0.159	0.119
30	50	0.110	0.180	0.111	0.164	0.123

Tabla 5. Resultados obtenidos con la segunda función objetivo para la mezcla ciclohexano-benceno.

Población	Iteración	GWO		Jaya		PSO	
		Error	Tiempo	Error	Tiempo	Error	Tiempo
30	5	0.309	0.043	0.309	0.054	0.278	0.046
30	10	0.267	0.093	0.180	0.096	2.725	0.091
30	15	0.181	0.150	0.308	0.144	0.312	0.138
30	20	0.171	0.193	0.308	0.188	0.310	0.183
30	25	0.308	0.260	0.178	0.237	0.309	0.224
30	30	0.308	0.290	0.178	0.279	0.302	0.291
30	35	0.308	0.365	0.211	0.363	0.309	0.329
30	40	0.175	0.422	0.299	0.359	0.377	0.376
30	45	0.308	0.420	0.175	0.402	0.312	0.447
30	50	0.170	0.495	0.172	0.443	0.308	0.471

Tabla 6. Resultados obtenidos con la segunda función objetivo para la mezcla alcohol alilico - etilenglicol.

Población	Iteración	GWO		Jaya		PSO	
		Error	Tiempo	Error	Tiempo	Error	Tiempo
30	5	0.116	0.038	0.155	0.044	0.141	0.040
30	10	0.048	0.084	0.108	0.080	0.820	0.086
30	15	0.116	0.116	0.064	0.144	1.034	0.135
30	20	0.116	0.169	0.056	0.166	0.835	0.157
30	25	0.075	0.203	0.059	0.200	0.480	0.204
30	30	0.040	0.251	0.044	0.249	0.340	0.240
30	35	0.116	0.302	0.042	0.272	0.080	0.283
30	40	0.075	0.369	0.063	0.321	0.447	0.313
30	45	0.116	0.377	0.051	0.342	0.727	0.382
30	50	0.116	0.404	0.095	0.391	0.669	0.397

Tabla 7. Resultados obtenidos con la segunda función objetivo para la mezcla alcohol alilico – etilenglicol.

Población	Iteración	GWO		Jaya		PSO	
		Error	Tiempo	Error	Tiempo	Error	Tiempo
30	5	0.116	0.038	0.155	0.044	0.141	0.040
30	10	0.048	0.084	0.108	0.080	0.820	0.086
30	15	0.116	0.116	0.064	0.144	1.034	0.135
30	20	0.116	0.169	0.056	0.166	0.835	0.157
30	25	0.075	0.203	0.059	0.200	0.480	0.204
30	30	0.040	0.251	0.044	0.249	0.340	0.240
30	35	0.116	0.302	0.042	0.272	0.080	0.283
30	40	0.075	0.369	0.063	0.321	0.447	0.313
30	45	0.116	0.377	0.051	0.342	0.727	0.382
30	50	0.116	0.404	0.095	0.391	0.669	0.397

Tabla 8. Resultados obtenidos con la segunda función objetivo para la mezcla ácido fluorhídrico -agua.

Población	Iteración	GWO		Jaya		PSO	
		Error	Tiempo	Error	Tiempo	Error	Tiempo
25	5	0.919	0.145	1.268	0.167	7.941	0.154
25	10	0.790	0.296	0.728	0.309	11.472	0.291
25	15	0.096	0.478	1.405	0.513	12.471	0.435
25	20	0.345	0.562	1.310	0.569	1.538	0.596
25	25	0.094	0.754	1.535	0.731	1.270	0.717
25	30	0.094	0.847	0.215	0.888	2.262	0.876
25	35	0.345	1.006	0.243	0.990	0.597	1.000
25	40	0.345	1.196	0.430	1.136	3.949	1.170
25	45	0.094	1.290	0.101	1.244	1.299	1.283
25	50	0.094	1.445	0.445	1.398	4.087	1.446

Tabla 9. Resultados obtenidos con la segunda función objetivo para la mezcla ácido fluorhídrico -agua.

Población	Iteración	GWO		Jaya		PSO	
		Error	Tiempo	Error	Tiempo	Error	Tiempo
25	5	1.542	0.367	0.506	0.442	0.639	0.364
25	10	0.134	0.755	0.053	0.811	0.914	0.829
25	15	0.189	1.136	0.048	1.219	0.445	1.107
25	20	0.184	1.529	0.082	1.523	1.371	1.507
25	25	0.184	2.013	0.015	1.957	0.268	2.102
25	30	0.184	2.337	0.007	2.357	1.066	2.300
25	35	0.002	3.135	0.024	3.518	1.904	2.900
25	40	0.184	3.156	0.030	3.183	0.501	3.292
25	45	0.184	3.401	0.004	3.386	0.304	3.850
25	50	0.184	3.768	0.012	3.752	0.695	3.732

Como se puede apreciar en las Tablas 4-9, los métodos Jaya y GWO presentan una tendencia de convergencia desde valores bajos de población e iteraciones, no obstante, el método PSO presenta una estabilidad de convergencia con una población de 30 individuos/partículas y 40 iteraciones. Como los métodos Jaya y GWO, presentan la estabilidad con valores inferiores y se mantiene la convergencia, se emplearán estos parámetros como punto de comparación entre los diferentes métodos de optimización y el impacto en las funciones objetivos que se plantean (Tablas 10 a 12). Cabe mencionar las iteraciones para el alcohol alílico con la tercera función objetivo (energía libre de Gibbs), mostraban una serie de errores dentro del algoritmo, por lo cual, no se presentan.

Tabla 10. Resultados obtenidos con el método PSO para las diferentes mezclas, con una población de 30 individuos y 40 iteraciones.

MODELO PSO (función Objetivo 1, Fracciones mol)				
Mezcla	Error	α	β_{12}	β_{21}
	0.11905532	0.199998009	7611.165528	-4945.14587
Ciclohexano-benceno				

	0.44660793	0.579657453	25478.81007	46506.98578
Alcohol alílico-etilenglicol				
	0.47732469	0.251489299	-5801.090791	-9831.259582
HF-H2O				
MODELO PSO (función Objetivo 2, Coeficientes de actividad)				
Mezcla	Error	α	β_{12}	β_{21}
	0.37660288	0.40849674	44221.2748	32431.1628
Ciclohexano-benceno				
	1.30791262	0.236210584	16863.01628	-4699.700736
Alcohol alílico-etilenglicol				
	0.30164076	0.28434876	-6730.61947	-10204.5362
HF-H2O				
MODELO PSO (función Objetivo 3, Energía libre de Gibbs)				
Mezcla	Error	α	β_{12}	β_{21}
	271472	0.6	-50000	-26450.6684
Ciclohexano-benceno				
	Ver nota 1	Ver nota 1	Ver nota 1	Ver nota 1
Alcohol alílico-etilenglicol				
	285286303	0.53321056	-37686.8153	-43985.104
HF-H2O				

Tabla 11. Resultados obtenidos con el método Jaya para las diferentes mezclas, con una población de 30 individuos y 40 iteraciones.

MODELO Jaya (función Objetivo 1, Fracciones mol)				
Mezcla	Error	α	β_{12}	β_{21}
	0.29931823	0.2	-4778.93151	7822.27628
Ciclohexano-benceno				
	0.15911597	0.45559685	3410.88213	-623.526233
Alcohol alílico-etilenglicol				

	0.00802655	0.46874564	-4298.82524	-7538.04256
HF-H2O				
MODELO Jaya (función Objetivo 2, Coeficientes de actividad)				
Mezcla	Error	α	β_{12}	β_{21}
	0.112852688	0.6	50000	50000
Ciclohexano-benceno				
	0.06333566	0.526916564	5120.475454	-1648.652807
Alcohol alílico-etilenglicol				
	0.231132962	0.267953125	-2363.188395	-10581.90143
HF-H2O				
MODELO Jaya (función Objetivo 3, Energía libre de Gibbs)				
Mezcla	Error	α	β_{12}	β_{21}
	271472	0.6	-50000	-27370.3853
Ciclohexano-benceno				
	Ver nota 1	Ver nota 1	Ver nota 1	Ver nota 1
Alcohol alílico-etilenglicol				
	284663719	0.6	-50000	-50000
HF-H2O				

Tabla 12. Resultados obtenidos con el método GWO para las diferentes mezclas, con una población de 30 individuos y 40 iteraciones.

MODELO GWO (función Objetivo 1, Fracciones mol)				
Mezcla	Error	α	β_{12}	β_{21}
	0.10990162	0.2	85.7881788	-278.773271
Ciclohexano-benceno				
	0.07515246	0.6	23533.5255	699.446837
Alcohol alílico-etilenglicol				
	0.09386073	0.2	-2520.0182	-11727.9717
HF-H2O				
MODELO GWO (función Objetivo 2, Coeficientes de actividad)				
Mezcla	Error	α	β_{12}	β_{21}

	0.17537434	0.6	50000	632.884837
Ciclohexano-benceno				
	0.15902877	0.36075495	4217.07644	-1206.08276
Alcohol alílico-etilenglicol				
	0.00064466	0.53045209	-4491.46535	-7002.97369
HF-H2O				
MODELO GWO (función Objetivo 3, Energía libre de Gibbs)				
Mezcla	Error	α	β_{12}	β_{21}
	271472	0.6	-50000	-26450.6684
Ciclohexano-benceno				
	Ver nota 1	Ver nota 1	Ver nota 1	Ver nota 1
Alcohol alílico-etilenglicol				
	284663719	0.6	-50000	-50000
HF-H2O				

En las Figuras 5 a 8 se muestra el comportamiento que tiene cada algoritmo de optimización. Los algoritmos Jaya y GWO tienden a presentar un comportamiento que oscila entre valores por encima y por debajo del punto óptimo, lo que hace que conforme se avanza en iteraciones, esta oscilación converja a un valor. Por su parte, el algoritmo PSO muestra una tendencia más estable.

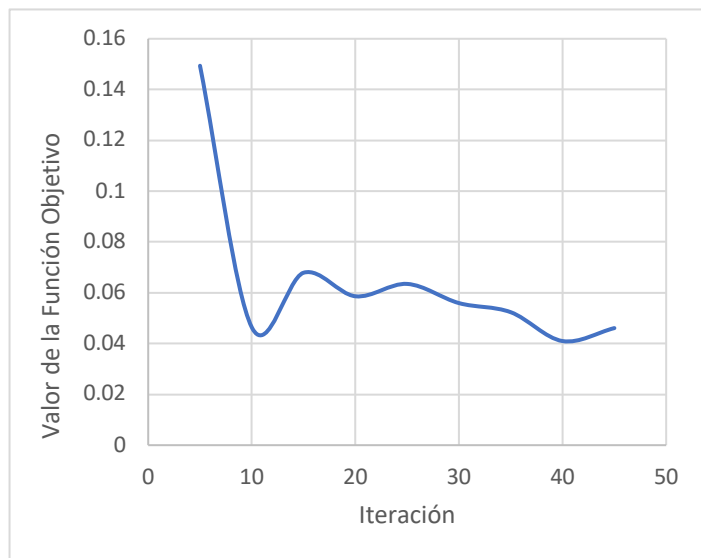


Figura 5. Comportamiento del método Jaya en la mezcla alcohol alílico-etilenglicol con la función objetivo 1 (fracción mol).

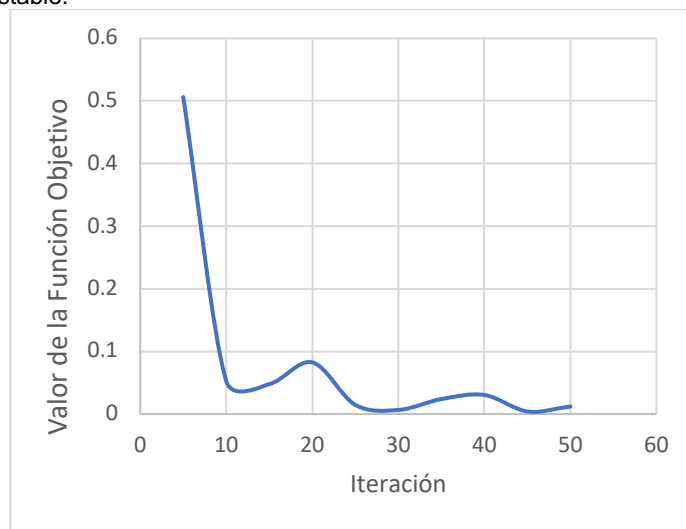


Figura 6. Comportamiento del método GWO en la mezcla HF-H2O con la función objetivo 2 (coeficientes de actividad)

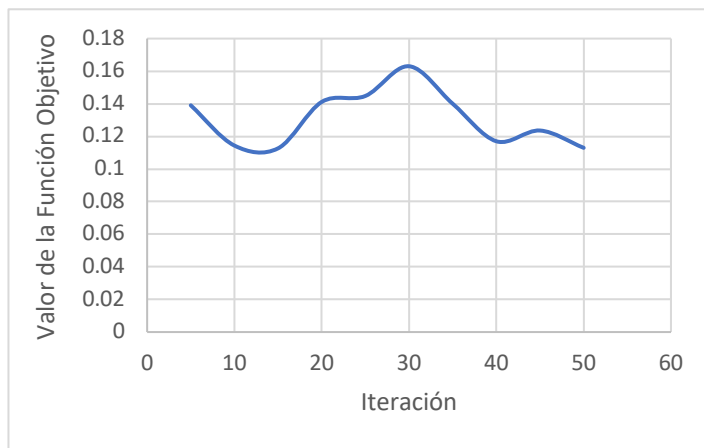


Figura 7. Comportamiento del método PSO en la mezcla ciclohexano-benceno con la función objetivo 1 (fracción mol).

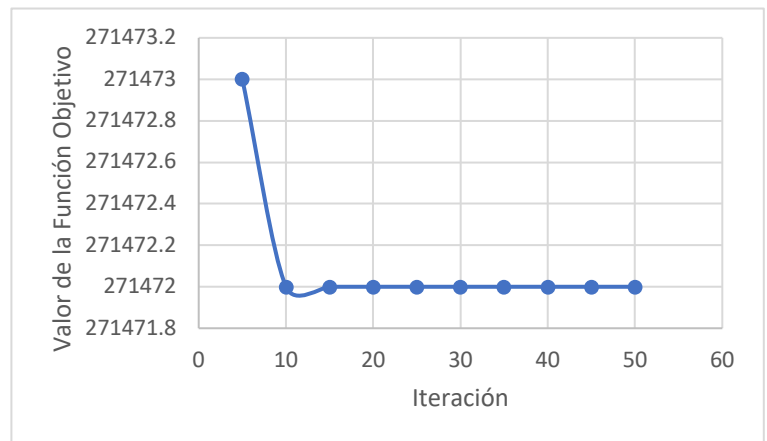


Figura 8. Comportamiento del método PSO en la mezcla ciclohexano-benceno con la función objetivo 3 (Energía libre de Gibbs)

Conclusiones

Se desarrolló un código en lenguaje de programación Python capaz de analizar mezclas binarias de sustancias de diversas naturalezas, así como realizar los ajustes de datos a mezclas con la presencia de azeótropos en el punto final de ebullición o con múltiples azeótropos a lo largo del rango de operación del sistema, además de poder realizar el análisis en sistemas que no presenta azeótropos. Se ha observado que al usar la función objetivo que involucra la energía libre de Gibbs, se tiene una tendencia a incrementar el error en valores grandes, por lo que se podría descartarse esta función para la optimización en los problemas de equilibrio de fases.

De los puntos visibles dentro del método PSO es que llega a converger a valores que son descartados por el margen de error, esto se debe a que el método utiliza al mejor valor "best-fitness" de un miembro de la población, provocando que, si un individuo se localiza en un punto local, este podrá guiar a los demás individuos hacia el punto local, estancando el método en dicho punto. Aunque esto es solucionable agregando mayor cantidad de individuos, por lo que, si uno de ellos se localiza en un punto local, es posible que otro individuo se localice en algún punto global, haciendo que las partículas se vean favorecidas hacia este punto del conjunto de valores posibles. Por su parte, el método Jaya tiene la particularidad que es un método que tiene a presentar una tendencia de oscilar entre valores positivos y negativos hasta converger a un punto, siendo este el punto máximo global.

Cada método tiene características que lo hacen comportarse de manera única, en general, el método PSO tiene a hacer más preciso con un número mayor de partículas, casi siendo esto un factor determinante de la confiabilidad del método, ya que con un número de 50 partículas y 5 iteraciones para la mezcla ciclohexano-benceno es muy similar a la de 50 partículas y 1000 iteraciones, obteniendo valores de función objetivo de 0.1392 y 0.1125, respectivamente, para cada valor, lo que hace que el método funcione mejor para números grandes de individuos de población. Por el lado de la optimización Jaya se observa un comportamiento oscilatorio, hasta que converge a un valor estable.

Referencias

- Yeo, Y. K. (2018). *Chemical Engineering Computation with MATLAB®*. Amsterdam University Press.
- Benitez, R. A., & Rojas, J. (2020). Implementation of the meta-heuristic algorithm Gray Wolf Optimization for the optimization of standard objective functions. *Serie Científica de la Universidad de las Ciencias Informáticas*, 13(8), 174–194. <https://publicaciones.uci.cu/index.php/serie/article/download/716/591/>
- Dong, Y., Huang, S., Guo, Y., & Lei, Z. (2019). COSMO-UNIFAC model for ionic liquids. *AIChE Journal*, 66(1). <https://doi.org/10.1002/aic.16787>
- Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence*, 1(1), 33–57. <https://doi.org/10.1007/s11721-007-0002-0>
- Smith, J. M., van Ness, H. C., & Abbott, M. M. (1997). *Introducción a la termodinámica en Ingeniería Química / SÉPTIMA EDICIÓN (5.a ed.)*. MCGRAW HILL EDUCATION.
- Orejel, J. F., & González, R. (2008). Solución Matricial de Modelos para Cálculo de Equilibrio Líquido-Vapor. *Información Tecnológica*, 19(3), 51–58. <https://bit.ly/3ltobla>
- Tosun, I. (2021). *The Thermodynamics of Phase and Reaction Equilibria (2a ed.)*. Elsevier.
- Vasquez, V., & Whiting, W. (2000). Regression of binary interaction parameters for thermodynamic models using an inside-variance estimation method (IVEM). *Fluid Phase Equilibria*, 170(2), 235–253. [https://doi.org/10.1016/s0378-3812\(00\)00325-3](https://doi.org/10.1016/s0378-3812(00)00325-3)