A Memetic Genetic Particle Swarm Optimization for Druglike Molecule Discovery

Matías Gabriel Rojas, Ana Carolina Olivera, and Pablo Javier Vidal

***Abstract*—The vast chemical search space, where potential ligands reside, poses a significant challenge in drug discovery. Efficiently identifying ligands with favourable properties is crucial to reducing development costs and time. While neural networks are often employed for this purpose, they can produce chemically invalid molecules. Metaheuristics offer an alternative with promising results but usually face limitations in molecule diversity and quality. This work presents a novel memetic algorithm combining Particle Swarm Optimization and Simulated Annealing, focused on enhancing ligand quality, by ensuring that promising molecules are not overlooked during generation steps. Comparative analysis with eight leading algorithms reveals that our approach generates ligands with improved chemical properties, leading to more effective molecule generation.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/9254

***Index Terms*—De-Novo Drug Discovery, Druglike Molecules Design, Memetic Approach, Metaheuristics.**

I. Introducción

La identificación de moléculas novedosas que posean un conjunto de características específicas es esencial en el diseño y descubrimiento de fármacos [1]. Sin embargo, el espacio de búsqueda químico, que contiene a todas las combinaciones químicas moleculares, es tan amplio que explorarlo resulta lento y costoso en términos monetarios y de uso de recursos [2]. Esto impulsa el interés en técnicas de inteligencia artificial, que recorren eficientemente el espacio de búsqueda de moléculas químicas, especialmente cuando se deben optimizar múltiples objetivos simultáneamente. En particular, las redes neuronales son el centro de atención en el diseño de-novo de moléculas [3]–[5], debido a su capacidad de aprender patrones intrínsecos en los datos y su éxito en la identificación de compuestos prometedores [6].

Una de las herramientas más usadas en el diseño molecular es el Sistema de Introducción Lineal Molecular Simplificada (SMILES, *Simplified Molecular Input Line Entry System*), que permite representar estructuras moleculares en cadenas de caracteres [7]. Desafortunadamente, las redes neuronales han mostrado dificultades para generar cadenas SMILES válidas tanto sintáctica como químicamente [8]. Además, la dependencia de las redes neuronales del conjunto de datos de entrenamiento puede sesgar la generación, limitando la diversidad de las moléculas obtenidas. [9]. En busca de alternativas alternativas confiables que superen estas limitaciones, varios trabajos optaron por las metaheurísticas, que son capaces de abstraerse del conjunto de entrenamiento y lograr soluciones de alta calidad a un costo computacional razonable [10], [11].

Ejemplos del uso de metaheurísticas para el descubrimiento de nuevas moléculas con propiedades de fármacos son [12] y [9], que usan al Algoritmo Genético (GA, *Genetic Algorithm*) como generador de nuevas moléculas, aplicando operadores genéticos específicamente diseñados para recorrer el espacio de búsqueda. En ambos casos, las técnicas lograron identificar nuevas moléculas prometedoras. Sin embargo, estas aproximaciones usan la versión canónica del GA, que no considera, durante el proceso de generación, a las mejores moléculas identificadas, descartando características deseables.

El Algoritmo de Enjambre de Partículas (PSO, *Particle Swarm Optimization*) actualiza cada solución siguiendo a su mejor estado previo y a la mejor solución identificada [13]. Varios trabajos aprovecharon estas características para el diseño de moléculas de-novo. En [14] se usa al PSO para la unión en puntos específicos de fragmentos de moléculas previamente identificados. Winter y otros [15] propusieron un PSO que optimiza el espacio latente de un autoencoder que genera las cadenas SMILES. Las dos variantes mostraron resultados destacados, sin embargo, PSO fue diseñado para la optimización sobre representaciones continuas, por lo que no tiene los mecanismos de diversificación para representaciones de cadena de caracteres. Recientemente, se puso el foco en adaptar estos enfoques para lidiar de manera más directa con la optimización de cadenas SMILES. Por ejemplo, en [16] introducen un PSO con operadores genéticos para el descubrimiento de moléculas mediante la optimización de grafos. Aunque los resultados son prometedores, la aproximación se centra en la optimización de un solo objetivo. Además, las técnicas basadas en grafo tienden a estancarse en óptimos locales. Esto refleja la relevancia de adaptar estos métodos a la optimización de cadenas de caracteres, para aprovechar sus características en la generación de moléculas.

Los Algoritmos Meméticos (MA, *Memetic Algorithms*) [17] son una opción para abordar la falta de diversidad (exploración del espacio de búsqueda) y la perdida de información relevante (explotación del espacio de búsqueda). Se trata de hibridaciones entre una metaheurística (PSO o GA), para la optimización global del espacio de búsqueda, con métodos heurísticos o metaheurísticos que exploten el espacio de soluciones

The associate editor coordinating the review of this manuscript and approving it for publication was Giner Alor-Hernández (*Corresponding author: Matias Gabriel Rojas*).

Matías Gabriel Rojas, A. C. Olivera, P. J. Vidal are with the Instituto Interdisciplinario de Ciencias Básicas, Universidad Nacional de Cuyo, Mendoza, Argentina (e-mails: mrojas@mendoza-conicet.gob.ar, acolivera@conicet.gov.ar, and pjvidal@conicet.gov.ar).

involucrando conocimiento específico del problema [18].

En este trabajo se propone al PSOSA (*Particle Swarm Optimization with Simulated Annealing*), un enfoque memético que híbrida una versión del PSO que usa operadores genéticos, con el algoritmo de Recocido Simulado (SA, *Simulated Annealing*), para el descubrimiento de-novo de moléculas basado en fragmentos. La idea es aportar diversidad al PSO mediante los operadores genéticos, sin perder de vista a las mejores soluciones identificadas. El SA identifica a las distintas disposiciones atómicas de las moléculas descubiertas por el PSO, para mejorar las cualidades químicas de los ligandos y la explosión combinatoria del algoritmo. Como objetivos, se busca que las moléculas tengan características destacadas de fármacos y sean fácilmente sintetizables. Las contribuciones de este trabajo se pueden sintetizar de la siguiente manera.

- Se propone un algoritmo memético que combina a una variante genética del PSO con un SA, para el descubrimiento de-novo de moléculas basado en fragmentos.
- Se introduce como operador de búsqueda local al SA, que mejora a las moléculas generadas, identificando a sus distintas conformaciones atómicas.
- La propuesta es evaluada mediante métricas específicas de calidad de generación y test estadísticos.

El manuscrito se organiza de la siguiente manera. La sección II detalla aspectos de implementación del PSOSA. En la sección III se especifican las configuraciones paramétricas y experimentales. Los resultados obtenidos al realizar los experimentos se informan en la sección IV. Por último, en la sección VI, se presentan las conclusiones y trabajos futuros.

II. Propuesta

En este trabajo se propone al PSOSA, una aproximación que hibrida una versión genética del PSO con un SA (que realiza la búsqueda local). El funcionamiento del algoritmo se resume en la Fig. 1 y se describe en el Algoritmo 1.

El proceso comienza inicializando y evaluando la población de soluciones candidatas, tomando moléculas de un conjunto de entrenamiento compuesto por fragmentos de moléculas previamente identificados (Linea 1). Con la población inicializada y evaluada, comienza el proceso iterativo del PSO (Linea 3). Por cada solución de la población, primero se ejecuta la búsqueda local llevada a cabo por el SA (Linea 4).

El SA muta la solución, modificando aleatoriamente las posiciones de los átomos en la molécula. Si la solución mutada tiene un mejor valor de aptitud se mantiene. En caso contrario, se decide si mantenerla o no de acuerdo a un coeficiente de enfriamiento actualizado por el SA. Este proceso se repite hasta satisfacer una condición de parada, retornando la solución mejorada por el SA.

Seguidamente, se aplica el operador de cruza sobre la solución mejorada (Línea 5). Se utiliza el operador propuesto en [9], que es una cruza en un punto, con control de que los paréntesis y ciclos se encuentren balanceados. Se modifica el orden de ejecución del operado para simular el seguimiento de la solución mas apta del PSO canónico. Se obtienen de tres soluciones descendientes que surgen de los siguientes cruzamientos:

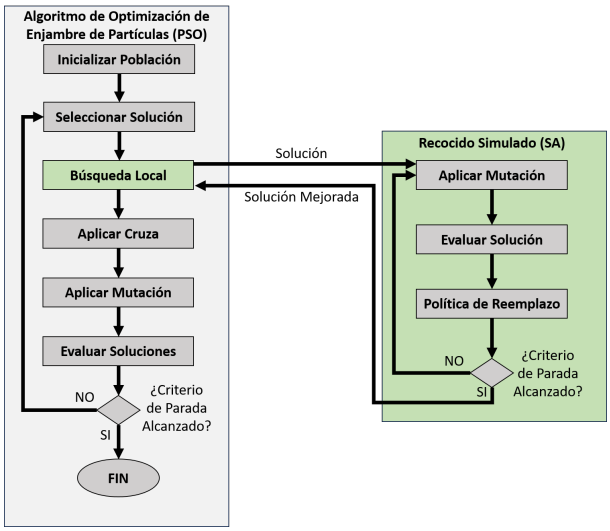


Fig. 1. Diagrama de flujo del PSOSA.

- La cruza entre la solución mejorada (*molMejorada* en el Algoritmo 1) y su mejor estado previo (*mejorMolLocal*).
- La cruza entre la solución mejorada (*molMejorada*) y la mejor solución identificada hasta el momento (*mejorMol*).
- La cruza entre el mejor estado previo de la solución (*mejorMolLocal*) y la mejor solución identificada hasta el momento (*mejorMol*).

El operador de mutación se ejecuta por cada una de las nuevas soluciones (Línea 6). En cada ejecución del operador, se decide, de manera aleatoria con probabilidad uniforme, entre realizar la remoción de un átomo, la inserción de un nuevo átomo o el reemplazo de un átomo, tomando como base la aproximación propuesta en [9]. El objetivo es identificar nuevos puntos no explorados en el espacio de búsqueda.

Luego de ejecutar ambos operadores, las nuevas soluciones son evaluadas por la función aptitud (Línea 7) y son almacenadas en un registro de moléculas descubiertas si su valor de aptitud supera un valor umbral que se actualiza a medida que se ejecuta el algoritmo (Línea 8). El ciclo iterativo del PSOSA se repite hasta satisfacer un criterio de parada, que es realizar un número determinado de evaluaciones de la función aptitud.

Algoritmo 1: Pseudocódigo del PSOSA.

```
   Entrada: Fragmentos de moléculas y tamañoPoblación
   Salida  : Conjunto de moléculas generadas
1  inicializarYEvaluarPoblación(tamañoPoblación);
2  i ⟵ 0;
3  mientras Criterio de parada no alcanzado hacer
4  |  molMejorada ⟵ SA(moléculas[i]);
5  |  descendientes ⟵ Cruza(molMejorada, mejorMol,
   |    mejorMolLocal[i]);
6  |  Mutación(descendientes);
7  |  Evaluar(descendientes);
8  |  ActualizarRegistroMoléculas(descendientes);
9  |  i ⟵ (i+1) mod tamañoPoblación;
10 fin
```

En resumen, el SA permite que se exploren diferentes

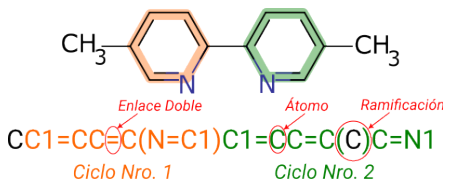


Fig. 2. Molécula representada mediante SMILES.

configuraciones atómicas de las moléculas. Esto contribuye a descubrir nuevas moléculas y a incrementar la explosión combinatoria del algoritmo, ya que los operadores genéticos del PSOSA se aplican sobre una variación de las moléculas previamente identificadas. Por otro lado, los operadores genéticos dotan al PSOSA de una mejor capacidad de exploración y explotación del espacio de búsqueda, a la vez que se evita perder información relevante del proceso iterativo por las cualidades del PSO.

A. Representación de la Solución

Las soluciones se codifican usando la nomenclatura SMILES [7]. Un ejemplo de esta representación se muestra en la Fig. 2. Los elementos de la solución pueden ser, átomos como el carbono (C), hidrógeno (H), nitrógeno (N) u oxígeno (O), y enlaces entre ellos, que pueden ser simples ($-$), dobles ($=$), triples ($\#$) o aromáticos ($:$).

Las ramificaciones se identifican mediante paréntesis. Los corchetes denotan estados de oxidación y cargas. Los ciclos se especifican introduciendo un número entero junto a los átomos que comienzan y finalizan el ciclo. Los enlaces con isometría cis/trans se representan mediante barras normales ($/$) e invertidas ($\backslash$).

B. Función Aptitud

Dada una solución s, la función objetivo, presentada en Eq. (1), busca maximizar las propiedades de fármacos de las moléculas generadas, al mismo tiempo que se reduce la complejidad de sintetización.

$$(\omega \times QED(s)) - \left((1-\omega) \times \left(1 - \frac{AS(s)}{10}\right)\right). \quad (1)$$

$QED(s)$ es la Estimación Cuantitativa de Atributos de Fármaco (QED, *Quantitative Estimate of Drug-Likeness*). Mide qué tan probable es que la molécula sea una droga válida. Un valor de 0 indica que todas las propiedades son desfavorables. Valores cercanos a 1 sugieren una mayor probabilidad de que la molécula sea un fármaco efectivo.

$AS(s)$ es la Accesibilidad Sintética. Mide la dificultad para sintetizar la droga. Se asigna valores cercanos a 0 a moléculas fáciles de sintetizar y valores cercanos a 10 a las moléculas de sintetización compleja [19].

ω determina la contribución de cada parte de la función aptitud. Se utiliza un valor $\omega = 0.9$ de acuerdo con el rango de valores estudiado en [9]. De esta manera, se obtienen moléculas sintetizables, pero con una complejidad superior a las que se obtendrían si se diera mayor importancia a la sintetizabilidad de los ligandos.

C. Análisis de Complejidad Computacional

Se realiza el análisis de complejidad computacional del PSOSA observando como crece el tiempo de ejecución con respecto al tamaño de la entrada, en el peor caso posible (notación O Grande). Las variables que influyen en la ejecución son: el tamaño de población p, la cantidad de átomos en la cadena SMILES a, cantidad de descendientes d, las iteraciones del operador de búsqueda local l y la cantidad de iteraciones sobre la población g. La complejidad computacional general del PSOSA se presenta en la Eq. (2).

$$O(PSOSA) = \max\left(p, d \times g, d \times a \times g, d \times a \times l \times g\right). \quad (2)$$

El proceso de iniciación y evaluación es $O(p)$. La complejidad computacional del operador de cruza es $O(d \times g)$, teniendo en cuenta que se generan d descendientes. La complejidad del operador de mutación es $O(d \times a \times g)$, ya que en el peor caso, se recorre a toda la molécula. En el caso del operador de búsqueda local, la complejidad es $O(d \times a \times l \times g)$.

III. Diseño Experimental

Se compara el rendimiento del PSOSA contra otras ocho aproximaciones del estado del arte. Tres son distintas arquitecturas de redes neuronales:

- Red Neuronal Recurrente (RNN, *Recurrent Neural Network*) [3]: contiene una capa de embedding y tres capas ocultas con 512 celdas LSTM cada una.
- Autoencoder Variacional (VAE, *Variational Autoencoder*) [4]: La VAE genera un espacio latente para entrenar una red generativa. Mapas auto organizados se emplean como métodos de recompensa para mejorar el espacio latente de la VAE.
- Red Generativa Adversaria (GAN, *Generative Adversarial Network*) [5]: Utiliza una red neuronal recurrente como generador, una red neuronal convolucional como discriminador y una red de aprendizaje por refuerzo para generar las recompenzas que entrenan al generador.

Además, se compara contra la versión del GA presentada en [9] y las versiones adaptadas para cadenas de caracteres del algoritmo de Evolución Diferencial (DE, Differential Evolution) [20], el Algoritmo de Lobo Gris (GWO, Grey Wolf Optimization) [21], el PSO y el SA. La Tabla I, presenta la configuración paramétrica de cada método.

Todos los métodos evaluados tienen como condición de parada evaluar 1.000.000 de moléculas. Las metaheurísticas se ejecutan 30 veces de manera independiente, utilizando conjuntos de entrenamiento distintos en cada ejecución. El VAE, la GAN y la RNN son pre-entrenados una sola vez, debido al costo computacional de entrenarlos.

En las tablas de la sección de resultados, los mejores valores se muestran en negrita y los segundos mejores valores se muestran en cursiva.

Los experimentos se ejecutan en el Cluster TOKO [1], que cuenta con un procesador AMD Opteron/Epyc de 64 núcleos, 128 GB de RAM y sistema operativo Ubuntu 18.04 LTS.

[1] toko.uncu.edu.ar

TABLA I
CONFIGURACIÓN PARAMÉTRICA DE LOS ALGORITMOS

Algoritmo	Parámetros
VAE	Tamaño de Batch : 50, Épocas: 10, Tasa de Aprendizaje: 0.0001
GAN	Épocas generador: 240; Épocas discriminador: 50; Entrenamiento con reglas de Lipinsky: 110 épocas.
RNN	Se utiliza modelo pre-entrenado por autores (ver [3]).
DE	Tamaño de población: 100; Operador de Cruza: Cruza de un punto (Proba.: 1.0.); Operador de Mutación: Eliminación, adición o reemplazo de átomo (Proba.: 1.0.)
GA	Tamaño de población: 100; Operador de Cruza: Cruza de un punto (Proba.: 1.0.); Operador de Mutación: Eliminación, adición o reemplazo de átomo (Proba.: 1.0.)
GWO	Tamaño de población: 100; Operador de Cruza: Cruza de un punto (Proba.: 1.0.); Operador de Mutación: Eliminación, adición o reemplazo de átomo (Proba.: 1.0.)
SA	Tamaño de población: 1; Operador de Mutación: Eliminación, adición o reemplazo de átomo (Proba.: 1.0.)
PSO	Tamaño de población: 100; Operador de Cruza: Cruza de un punto (Proba.: 1.0.); Operador de Mutación: Eliminación, adición o reemplazo de átomo (Proba.: 1.0.); Tipo de Vecindario: Global.
PSOSA	Tamaño de población: 100; Operador de Cruza: Cruza de un punto (Proba.: 1.0.); Operador de Mutación: Eliminación, adición o reemplazo de átomo (Proba.: 1.0.); Iteraciones del SA: 10 por cada solución.; Tipo de Vecindario: Global.

Las metaheurísticas fueron implementadas mediante la librería JMetalPy [22]. Las evaluaciones y graficado de las moléculas generadas se realizan mediante la librería RDKIT [2].

Se utiliza la base de datos Papyrus 5.55 [23], que cuenta con 1.270.570 moléculas estandarizadas sin estereoquímica especificada, representadas meadiante SMILES. Para el estandarizado, se usa el pipeline establecido por ChEMBL [24], y se eliminan sales, solventes y fragmentos duplicados.

A. Métricas de Interpretación de los Resultados

Se evalúa la calidad de las moléculas generadas por cada técnica, mediante tres tipos de métricas distintas [19], [25].

Las **métricas de calidad de generación** incluyen a las métricas de *validez*, que es la proporción de moléculas generadas que son químicamente válidas, la *concisión*, cuyo valor es más chico cuando las moléculas generadas son excesivamente más largas que su versión canónica, y la *unicidad* que es la proporción de moléculas sin duplicados.

Las **métricas de dispersión** miden la influencia del conjunto de entrenamiento sobre los métodos de generación. Se evalúan tres métricas, la *novedad*, que es la porción de moléculas generadas que no tienen ocurrencia en el conjunto de entrenamiento, la *variedad*, que mide qué tan diferentes son las moléculas generadas (1.0 implica que las moléculas son distantes), y la *creatividad* que mide la distancia entre las moléculas generadas y las de entrenamiento.

Por último, las **métricas de características químicas** evalúan propiedades químicas farmacológicas de las moléculas. El *LogP* Mide la hidrofilicidad de la molécula generada. Se espera un valor entre 0 y 3 (balance hidrofílico/lipofílico), para maximizar la biodisponibilidad y minimizar posibles efectos tóxicos. La *métrica de Lipinsky* [26] evalúa 5 reglas para asegurar solubilidad y permeabilidad de las moléculas generadas. Asigna un valor de 0.2 por cada regla cumplida, siendo 1 el máximo valor posible.

[2]https://rdkit.org/

IV. RESULTADOS

En esta sección se presentan los resultados obtenidos al ejecutar los experimentos.

A. Comparando Robustez y Efectividad de las Metaheurísticas

La Tabla II compara los valores de función aptitud promedio, máximo, mínimo y la desviación estándar, calculadas considerando a la mejor molécula (mayor aptitud) encontrada por cada *metaheurística* en cada una de las 30 ejecuciones independientes. El objetivo es observar la robustez de las técnicas al utilizar distintos conjuntos de entrenamiento.

TABLA II
PROMEDIO, MÍNIMO, MÁXIMO Y DESVIACIÓN ESTÁNDAR DE LA FUNCIÓN APTITUD OBTENIDA POR CADA METAHEURÍSTICA

	DE	GA	GWO	PSO	SA	PSOSA
Promedio	0.932	0.935	0.910	*0.936*	0.729	**0.937**
Desv. Est.	*0.004*	**0.001**	0.014	**0.001**	0.202	**0.001**
Min.	0.923	*0.933*	0.867	**0.935**	0.099	**0.935**
Max.	0.937	*0.937*	0.934	*0.937*	0.900	**0.938**

El PSOSA obtiene un rendimiento superior al informado por las otras aproximaciones. El segundo mejor es el PSO mostrando un comportamiento similar al PSOSA. Los peores resultados los reporta el SA, con valores de aptitud significativamente inferiores a los informados por las otras técnicas.

Para evaluar la significancia estadística de los resultados, primeramente se aplica el test de Kolmogórov-Smirnov para comprobar si los resultados de cada técnica siguen una distribución normal. En todos los casos, los resultados no siguen una distribución normal y, por lo tanto, se deben usar test no paramétricos para evaluar significancia. El test de Friedman comprueba si existen diferencias significativas entre los algoritmos considerados. El valor P obtenido es $3.09e-25$, indicando diferencias estadísticas significativas.

En la Tabla III se muestra el resultado del test de a pares de Wilcoxon. Un triángulo hacia la izquierda ($\triangleleft$) indica que el PSOSA es superior al algoritmo de la columna con una diferencia estadística significativa. Si el algoritmo de la columna es mejor que el PSOSA, se utiliza un triángulo ($\triangle$). Si no se observa una diferencia estadística significativa, se utiliza un guion ($-$). Además, se informa el valor P retornado por el test y se analiza la significancia estadística de los resultados mediante la corrección de Bonferroni, que ajusta el nivel de significancia inicial 0.05 considerando la cantidad de comparaciones realizadas, resultando en $\alpha = 0.01$.

TABLA III
TEST DE WILCOXON ENTRE PARES DE METAHEURÍSTICAS

	DE	GA	GWO	PSO	SA
PSOSA	$\triangleleft$	$\triangleleft$	$\triangleleft$	$-$	$\triangleleft$
Valor P	$1.57e-08$	$3.64e-05$	$7.18e-11$	$3.93e-02$	$9.57e-11$
¿Significativo?	SI	SI	SI	NO	SI

Se puede afirmar que el PSOSA es significativamente superior a la mayoría de las metaheuristicas consideradas. Con respecto al PSO, no se cuenta con evidencias suficientes para

afirmar que la diferencia es significativa, aunque al ser un valor P menor a 0.05, se puede inferir que existen diferencias entre los conjuntos comparados.

En la Tabla IV se presentan los tiempos de ejecución promedio, mínimos, máximos y desviación estándar, requeridos por cada técnica, para generar 1.000.000 de moléculas.

TABLA IV
TIEMPOS DE EJECUCIÓN PROMEDIO, MÍNIMO, MÁXIMO Y DESVIACIÓN ESTÁNDAR, MEDIDOS EN MINUTOS

	DE	GA	GWO	SA	PSO	PSOSA
Promedio.	**24.721**	329.724	268.699	*49.755*	75.629	90.151
Desv. Est.	10.978	135.564	532.122	6.947	*4.437*	**3.538**
Min.	**15.961**	195.109	*25.459*	30.593	68.493	82.895
Max.	*77.830*	703.814	1972.818	**65.787**	82.318	97.865

El DE es el algoritmo más rápido, mostrando una gran diferencia con respecto al resto de las metaheurísticas. El SA es el segundo algoritmo más rápido y el PSOSA se ubica en cuarta posición, lo que tiene sentido, ya que es una combinación de dos métodos de optimización distintos. Aun así, el incremento en tiempo de procesamiento del PSOSA no es significativo con respecto al PSO. El GA es el algoritmo más lento.

B. Comparación Contra Técnicas del Estado del Arte

Los algoritmos evaluados difieren en la cantidad de ligandos generados sin repeticiones ni redundancias, lo que deriva en comparaciones entre conjuntos de distintas dimensiones. Para garantizar una comparación equitativa, esta sección se enfoca en las 1.000 mejores moléculas generadas por cada método, evaluando la calidad de las mejores moléculas que cada aproximación puede producir en una ejecución determinada. La Tabla V muestra los valores promedio, mínimos, máximos y desviación estándar de la función aptitud, alcanzados por cada técnica. En el caso de las metaheurísticas las 1000 mejores moléculas son extraídas de la ejecución en la que se encontró al ligando con mejor valor de aptitud.

TABLA V
VALORES DE FUNCIÓN APTITUD DE LAS 1000 MEJORES MOLÉCULAS DESCUBIERTAS POR CADA MÉTODO

	GAN	RNN	VAE	DE	GA	GWO	SA	PSO	PSOSA
Promedio	0.728	*0.930*	0.915	0.909	0.905	0.894	0.907	0.917	**0.935**
Desv. Est.	0.031	*0.002*	0.007	0.010	0.015	0.013	0.002	0.006	**0.001**
Min.	0.696	*0.927*	0.905	0.896	0.880	0.875	0.905	0.909	**0.934**
Max.	0.859	*0.937*	0.934	0.936	0.934	0.935	0.911	0.934	**0.938**

Se observa que el PSOSA es el método que alcanza los mejores valores de aptitud, informando un valor promedio de 0.935 y logrando una diferencia significativa en los valores máximos y mínimos alcanzados. Esto implica que las 1000 moléculas descubiertas por el PSOSA presentan un mejor balance entre QED y accesibilidad sintética. La segunda mejor técnica es la RNN, con un valor promedio de 0.930.

Al aplicar el test de Kolmogórov-Smirnov, se obtiene que los resultados no siguen a la distribución normal, por lo que se usan los test no paramétricos para la evaluación de significancia estadística. El test de Friedman retorna un valor P de 0.0, lo que es una fuerte evidencia de que existen diferencias significativas en el comportamiento de los algoritmos. En la Tabla VI, se muestran los resultados del test de Wilcoxon y se informa si las diferencias son significativas o no.

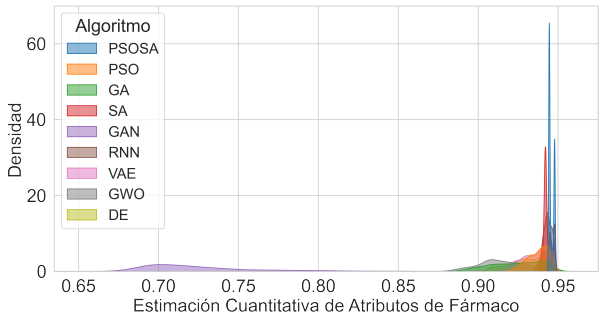


Fig. 3. Diagramas de densidad del QED por cada método.

TABLA VI
RESULTADOS DEL TEST DE WILCOXON PARA EL TOP 1000

	GAN	RNN	VAE	DE
PSOSA	◁	◁	◁	◁
Valor P	$0.00e+00$	$1.36e-296$	$0.00e+00$	$0.00e+00$
¿Significativo?	SI	SI	SI	SI
	GA	**GWO**	**SA**	**PSO**
PSOSA	◁	◁	◁	◁
Valor P	$0.00e+00$	$0.00e+00$	$0.00e+00$	$0.00e+00$
¿Significativo?	SI	SI	SI	SI

De acuerdo con el test, y considerando la corrección de Bonferroni que ajusta el nivel de significancia inicial 0.05 considerando la cantidad de comparaciones realizadas, lo que resulta en un nuevo nivel de significancia $\alpha = 0.00625$, se cuenta con evidencia suficiente para afirmar que las diferencias mostradas por el PSOSA son significativas estadísticamente.

En la Fig. 3 se muestra el diagrama de densidad de los valores de QED alcanzados por cada técnica. La mayoría de las técnicas muestran valores de QED entre 0.9 y 0.95. El PSOSA, destaca porque todas las soluciones poseen valores de QED próximos a 0.95, lo que evidencia robustez en la generación de moléculas con marcadas características de fármacos.

En la Fig. 4 se presenta el gráfico de densidad para los valores de AS logrados por cada técnica evaluada. Las moléculas del PSOSA se agrupan en torno al rango de valores entre 1.5 y 2. Las otras técnicas muestran una mayor dispersión, lo que explica la disminución en los valores de aptitud alcanzados.

Los resultados muestran que el PSOSA es capaz de mejorar los valores de aptitud y que eso se debe a que logra ser robusto en los valores alcanzados y a que logra un balance destacado entre las dos partes de la función aptitud.

C. Análisis de Calidad de Generación

En la Tabla VII, se analiza la calidad de generación de cada técnica mediante las métricas presentadas en la sección III-A. Para estas evaluaciones se considera a la totalidad de moléculas producidas por cada técnica en una generación determinada, con el objetivo de identificar fortalezas y debilidades.

Los valores de validez muestran que todas las metaheurísticas siempre generan moléculas sintáctica y químicamente

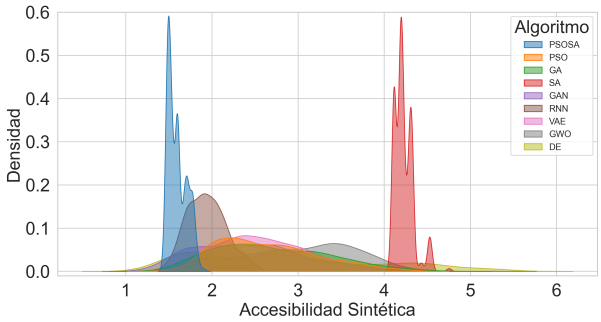


Fig. 4. Diagramas de densidad del AS por cada método.

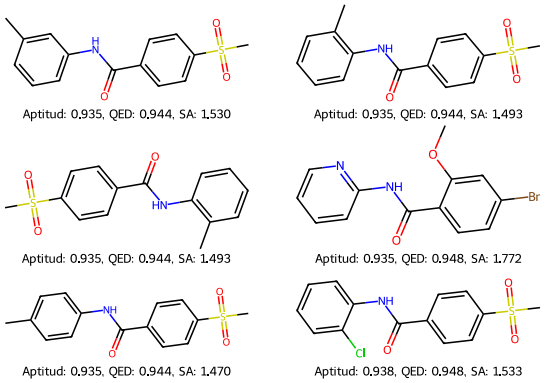


Fig. 5. Seis moléculas aleatorias identificadas por el PSOSA.

válidas, debido a los mecanismos de control de validez en los procesos de generación. Esto supone una ventaja con respecto a las técnicas basadas en redes neuronales, que informan un considerable porcentaje de soluciones no válidas. Los valores de concisión reflejan que, en general, las moléculas generadas no son considerablemente más largas que sus versiones canónicas, lo que significa que se reduce la presencia de átomos insignificantes en las moléculas obtenidas. En cuanto a la unicidad, el VAE fue el mejor algoritmo, pero de acuerdo con la validez, la mayor parte son moléculas sintácticamente inválidas. La RNN fue el segundo mejor y el resto de los algoritmos oscila valores cercanos a 0.5. El peor fue el SA, con un valor de 0.013.

TABLA VII
VALORES PROMEDIO DE LAS MÉTRICAS DE CALIDAD DE GENERACIÓN, DE DISPERSIÓN Y DE CARACTERÍSTICAS QUÍMICAS

Métricas	**GAN**	**RNN**	**VAE**	**DE**	**GA**	**GWO**	**SA**	**PSO**	**PSOSA**
Métricas de Calidad de Generación									
Validez	*0.975*	0.945	0.111	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
Concisión	**1.000**	*0.998*	0.992	0.944	0.909	0.988	0.920	0.871	0.885
Unicidad	0.420	*0.957*	**1.000**	0.392	0.574	0.478	0.013	0.680	0.539
Métricas de Dispersión									
Novedad	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
Variedad	**0.975**	0.916	*0.920*	0.816	0.902	0.826	0.286	0.682	0.906
Creat.	**0.964**	0.917	*0.920*	0.924	0.916	0.918	0.933	0.908	0.918
Métricas de Características Químicas									
LogP	1.490	3.503	3.238	3.200	3.541	2.813	2.844	2.621	2.711
Lipinsky	**1.000**	0.921	0.975	0.984	0.891	0.922	**1.000**	**1.000**	**1.000**

Con respecto a las métricas de dispersión, todos los algoritmos identificaron moléculas novedosas, diferentes a las existentes en el conjunto de entrenamiento. Los valores de variedad muestran que el PSOSA mejora a la generación del PSO incrementando la distancia entre las moléculas generadas. Esto refleja la contribución del SA en aportar diversificación durante el proceso de generación. A su vez, el SA obtiene los peores valores de variedad, lo que tiene sentido ya que es un algoritmo de búsqueda local. Por parte de la creatividad(*Creat.* en la tabla), todos los algoritmos generan moléculas distantes de las existentes en el conjunto de entrenamiento.

Con respecto al valor de LogP, todos los algoritmos identifican moléculas que tienen un balance deseable entre lipofilicidad e hidrofilicidad. Por último, el SA, el PSO y el PSOSA son las únicas técnicas que generan, en todos los casos, ligandos que cumplen con todas las reglas de Lipinsky, implicando una alta biodisponibilidad oral de las moléculas generadas.

La Fig. 5 muestra 6 moléculas tomadas aleatoriamente de las 1000 mejores moléculas identificadas por el PSOSA. Las moléculas tienen mínimas diferencias, que son capaces de incrementar los valores de función aptitud de manera considerable. Esto resalta la importancia del SA para identificar las conformaciones atómicas más destacadas.

V. DISCUSIÓN

Los resultados del PSOSA resaltan como puntos de fortaleza la robustez del algoritmo y la capacidad de lograr valores destacados en ambas partes de la función aptitud agregativa, lo que posibilita la obtención de moléculas de alta calidad. La diferencia con respecto a las otras metaheurísticas es que el PSOSA mantiene una memoria de las mejores moléculas identificadas, lo que evita la pérdida de conocimiento. Además la búsqueda local del SA ayuda a explotar las características de las moléculas destacadas, mediante la identificación de conformaciones moleculares con mejores propiedades.

Una posible limitación del PSOSA es la desventaja del diseño de fármacos basados en unión de fragmentos, ya que al utilizar un conjunto predefinido de fragmentos para inicializar la población, restringe la porción de espacio de búsqueda que se explora. Los operadores de mutación y la búsqueda local ayudan a incrementar la diversidad, pero aún se tienen limitaciones químicas pertenecientes a los fragmentos iniciales.

VI. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se introduce al PSOSA, un algoritmo memético que combina a una versión genética del Algoritmo de Enjambre de Partículas (PSO) y el algoritmo de Recocido Simulado (SA), para el problema del descubrimiento de-novo de moléculas basado en fragmentos.

Se compara al PSOSA contra 5 metaheurísticas y 3 redes neuronales del estado del arte. Los resultados mostraron que el PSOSA es capaz de mejorar los rendimientos exhibidos por las otras técnicas, mostrándose robusto en términos de función aptitud, a expensas de tiempos de ejecución aceptables. Al analizar la calidad de las generaciones, el PSOSA informó que las moléculas descubiertas poseen cualidades aceptables e

incluso mejores que los algoritmos con los que fue comparado, tanto desde el punto de vista de la diversidad, como así también desde las características químicas exhibidas.

Como trabajo futuro, se pretende extender la aplicación del PSOSA a la generación de drogas con respecto a una proteína objetivo. En este sentido, también se busca mejorar la función aptitud para ampliar la cantidad de objetivos a ser optimizados.

AGRADECIMIENTOS

Los fondos para realizar esta investigación provienen de la Universidad Nacional de Cuyo, Secretaría de Investigación, Internacionales y Posgrado (Proyecto 06/B052-T1) y del Consejo Nacional de Investigaciones Científicas y Técnicas (PIBAA 2022-2023 1070). Los autores agradecen al Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina.

REFERENCES

[1] C. M. Dobson, "Chemical space and biology," *Nature*, vol. 432, no. 7019, pp. 824–828, Dec. 2004. doi: 10.1038/nature03192

[2] T. Pereira, M. Abbasi *et al.*, "Optimizing blood–brain barrier permeation through deep reinforcement learning for de novo drug design," *Bioinformatics*, vol. 37, no. Supplement_1, pp. i84–i92, Jul. 2021. doi: 10.1093/bioinformatics/btab301

[3] X. Liu, K. Ye *et al.*, "DrugEx v2: de novo design of drug molecules by Pareto-based multi-objective reinforcement learning in polypharmacology," *J. Cheminf.*, vol. 13, no. 1, p. 85, Dec. 2021. doi: 10.1186/s13321-021-00561-9

[4] A. Zhavoronkov, Y. A. Ivanenkov *et al.*, "Deep learning enables rapid identification of potent DDR1 kinase inhibitors," *Nat. Biotechnol.*, vol. 37, no. 9, pp. 1038–1040, Sep. 2019. doi: 10.1038/s41587-019-0224-x

[5] B. Sanchez-Lengeling, C. Outeiral *et al.*, "Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC)," Aug. 2017.

[6] F. Zhu, X. X. Li *et al.*, "Clinical Success of Drug Targets Prospectively Predicted by *In Silico* Study," *Trends in Pharmacological Sciences*, vol. 39, no. 3, pp. 229–231, Mar. 2018. doi: 10.1016/j.tips.2017.12.002

[7] D. Weininger, "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules," *J. Chem. Inf. Comput.*, vol. 28, no. 1, pp. 31–36, Feb. 1988. doi: 10.1021/ci00057a005

[8] L. Schoenmaker, O. J. M. Béquignon *et al.*, "UnCorrupt SMILES: a novel approach to de novo design," *J. Cheminf.*, vol. 15, p. 22, Feb. 2023. doi: 10.1186/s13321-023-00696-x

[9] Y. Kwon and J. Lee, "MolFinder: an evolutionary algorithm for the global optimization of molecular properties and the extensive exploration of chemical space using SMILES," *J. Cheminf.*, vol. 13, no. 1, p. 24, Mar. 2021. doi: 10.1186/s13321-021-00501-7

[10] R. Devi, S. S. Sathya, and M. Coumar, "Evolutionary algorithms for de novo drug design – A survey," *Applied Soft Computing*, vol. 27, pp. 543–552, Feb. 2015. doi: 10.1016/j.asoc.2014.09.042

[11] S. Luukkonen, H. W. van den Maagdenberg *et al.*, "Artificial intelligence in multi-objective drug design," *Curr. Opin. Struct. Biol.*, vol. 79, p. 102537, Apr. 2023. doi: 10.1016/j.sbi.2023.102537

[12] N. Jain, A. Hornback, and M. D. Wang, "EMxDesign: A Genetic Algorithm for High Affinity Drug Design," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ser. GECCO '24 Companion. New York, NY, USA: ACM, Aug. 2024. doi: 10.1145/3638530.3654423. ISBN 9798400704956 pp. 439–442.

[13] T. M. Shami, A. A. El-Saleh *et al.*, "Particle Swarm Optimization: A Comprehensive Survey," *IEEE Access*, vol. 10, pp. 10 031–10 061, 2022. doi: 10.1109/ACCESS.2022.3142859

[14] M. Hartenfeller, E. Proschak *et al.*, "Concept of Combinatorial De Novo Design of Drug-like Molecules by Particle Swarm Optimization," *Chem. Biol. Drug. Des.*, vol. 72, no. 1, pp. 16–26, 2008. doi: 10.1111/j.1747-0285.2008.00672.x

[15] R. Winter, F. Montanari *et al.*, "Efficient multi-objective molecular optimization in a continuous latent space," *Chem. Sci.*, vol. 10, no. 34, pp. 8016–8024, Aug. 2019. doi: 10.1039/C9SC01928F

[16] H.-P. Liu, F. K. H. Phoa, and S. Dutta, "Molecule discovery and optimization via evolutionary swarm intelligence," *Scientific Reports*, vol. 14, no. 1, p. 24510, Oct. 2024. doi: 10.1038/s41598-024-75515-w

[17] C. Cotta, L. Mathieson, and P. Moscato, *Memetic Algorithms*. Cham: Springer International Publishing, 2017, pp. 1–32. ISBN 978-3-319-07153-4

[18] P. Moscato and C. Cotta, "An Accelerated Introduction to Memetic Algorithms," in *Handbook of Metaheuristics*, M. Gendreau and J.-Y. Potvin, Eds. Cham: Springer International Publishing, 2019, vol. 272, pp. 275–309. ISBN 978-3-319-91085-7 978-3-319-91086-4

[19] V. Bagal, R. Aggarwal *et al.*, "MolGPT: Molecular Generation Using a Transformer-Decoder Model," *J. Chem. Inf. Model*, vol. 62, no. 9, pp. 2064–2076, May 2022. doi: 10.1021/acs.jcim.1c00600

[20] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, 12 1997. doi: 10.1023/A:1008202821328

[21] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 3 2014. doi: 10.1016/j.advengsoft.2013.12.007

[22] A. Benítez-Hidalgo, A. J. Nebro *et al.*, "jMetalPy: A Python framework for multi-objective optimization with metaheuristics," *Swarm Evol. Comput.*, vol. 51, p. 100598, Dec. 2019. doi: 10.1016/j.swevo.2019.100598

[23] O. J. M. Béquignon, B. J. Bongers *et al.*, "Papyrus: a large-scale curated dataset aimed at bioactivity predictions," *J. Cheminf.*, vol. 15, no. 1, p. 3, Jan. 2023. doi: 10.1186/s13321-022-00672-x

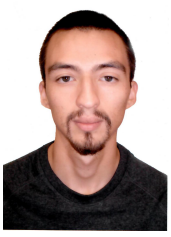
[24] A. P. Bento, A. Hersey *et al.*, "An open source chemical structure curation pipeline using RDKit," *J. Cheminf.*, vol. 12, no. 1, p. 51, Sep. 2020. doi: 10.1186/s13321-020-00456-1

[25] S. Choudhuri, M. Yendluri *et al.*, "Recent Advancements in Computational Drug Design Algorithms through Machine Learning and Optimization," *Kinases and Phosphatases*, vol. 1, no. 2, pp. 117–140, Jun. 2023. doi: 10.3390/kinasesphosphatases1020008

[26] C. A. Lipinski, F. Lombardo *et al.*, "Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings," *Adv. Drug Deliv. Rev.*, vol. 23, no. 1, pp. 3–25, Jan. 1997. doi: 10.1016/S0169-409X(96)00423-1

Matías Gabriel Rojas is a doctoral fellow at National Council of Scientific and Technological Researches from the Minister of Science and Technology of the Argentina Republic. He is an informatics engineer who graduated in 2019. His research interests focus on using artificial intelligence in the bioinformatics field, centring on optimisation algorithms.

Ana Carolina Olivera is an Inpedendent Researcher at National Council of Scientifics and Technological Researches from the MINCyT, Argentine. Dr. in Computer Science from Universidad Nacional del Sur. She is an Associate Professor at the Facultad de Ingeniería from Universidad Nacional de Cuyo. Her research focuses on metaheuristics and optimization in complex problems. She has published several book chapters, articles in indexed journals and proceedings of refereed international conferences.



Pablo Javier Vidal is an Adjunct Professor at the Universidad Nacional de Cuyo, and at the Universidad Nacional de la Patagonia Austral, Argentine. Dr. in Software Engineering and Artificial Intelligence, from Universidad de Málaga, Spain. He is an Adjunct Researcher at National Council of Scientifics and Technological Researches from the Ministerio de Ciencia y Tecnología de la Nación, Argentine. His main research topics are: parallel and distributed computing, bioinformatics and metaheuristics.