UN ENFOQUE MULTIOBJETIVO AL ALINEAMIENTO MÚTIPLE DE SECUENCIAS

Cristian Zambrano-Vega[1] Byron Oviedo, Oscar Moncayo
Universidad Técnica Estatal de Quevedo, Ecuador.

ABSTRACT
Multiple Sequence Alignment (MSA) is one of the main topics in the in bioinformatics domain, consists finding an optimal alignment for three or more biological sequences with the number maximum of conserved zones or totally aligned columns. Different scores to assess the quality of the alignments have been proposed, so the problem can be formulated and resolved as a Multi-Objective Optimization Problem (MOP). For this reason, in this paper we present a Multi-Objective approach applied to MSA. We have considered state-of-the-art optimization algorithms aimed at solving different formulations of the MSA: NSGAII, NSGA-III, SPEA2, MOCell, SMS-EMOA, MOEA/D and GWASF-GA. Furthermore we have considered some popular metrics as objectives to be optimized: The weighted Sum-Of-Pairs with a ne gap penalties (wSOP), the Totally Aligned Columns (TC), STRIKE and BaliScore. Finally we have described the main features of our software jMetalMSA, a Multi-Objective optimization software tool applied to MSA problem and illustrated a working example for experimentations purposes.

KEYWORDS: Multiple Sequence Alignment, MultiObjective Optimization Metaheuristics, Bio-informatics.

MSC: 92-08;92D20;68T20

RESUMEN
El Alineamiento Múltiple de Secuencias (MSA por sus siglas en inglés) es uno de los principales tópicos de interés en el campo de la BioInformática, consiste en encontrar un alineamiento óptimo para tres o más secuencias biológicas en el que exista la mayor cantidad de zonas conservadas o columnas de caracteres totalmente alineadas. Diferentes métricas para evaluar la calidad de los alineamientos han sido de nidas en la literatura, lo que hace preciso que el problema MSA sea formulado y resuelto como un Problema de Optimización MultiObjetivo (MOP). Por esta razón, en este artículo presentamos un enfoque de optimización multiobjetivo al problema MSA. Hemos considerado varios algoritmos multiobjetivo recientes aplicados a resolver diferentes formulaciones de MSA: NSGAII, NSGA-III, SPEA2, MOCell, SMS-EMOA, MOEA/D y GWASF-GA. Además, hemos considerado algunas métricas populares como objetivos a optimizar: la suma de pares ponderada con penalizaciones por GAPs a nado (wSOP), columnas totalmente alineadas (TC), STRIKE y BaliScore. Finalmente, describimos las características principales de nuestro software jMetalMSA, una herramienta software de optimización multiobjetivo aplicada al problema de MSA e ilustramos un ejemplo de trabajo para fines de experimentación.

PALABRAS CLAVE: Alineamiento Múltiple de Secuencias, Metaheurísticas de Optimización MultiObjetivo, BioInformática.

1. INTRODUCCIÓN

El alineamiento múltiple de secuencias biológicas, sea ADN, ARN o estructuras primarias proteicas (proteinas), es uno de los principales tópicos de interés dentro del campo de la BioInformática [25]. Su objetivo principal es la de representar y comparar más de dos secuencias de aminoácidos o nucleótidos para resaltar la mayor cantidad de zonas de similitud entre ellas, las cuales podrían indicar relaciones funcionales o evolutivas entre los genes o proteínas consultadas. Su importancia radica en que de la calidad de los alineamientos depende la exactitud y precisión de otros procesos bioinformáticos que se

[1] czambrano@uteq.edu.ec

realizan a partir de tales secuencias alineadas, como son la Inferencia Filogenética y la predicción estructural y funcional de proteínas.

El procedimiento de alineación básica se basa principalmente en la inserción de espacios o huecos (gaps) representados por el carácter "-“ dentro del conjunto de caracteres de las secuencias, para hacer que todas ellas tengan la misma longitud y para lograr la alineación del mayor número de sus columnas. Es importante llevar a cabo la manipulación de las operaciones con los gaps (inserción, eliminación, desplazamiento, agrupamiento, etc.) con el fin de ir generando nuevas alternativas de alineaciones para mejorar la precisión y calidad del alineamiento final, ya que el número de gaps y sus ubicaciones determinan finalmente la calidad del mismo.

Se han propuesto una serie de métricas diferentes para medir la precisión y calidad de los alineamientos, tales como: el porcentaje de columnas totalmente alineadas (TC), el porcentaje de caracteres -No espacios- (NonGapsP), la Suma de pares (Sum-of-Pairs, SOP), la suma ponderada de pares con penalidad de gaps a nes (weighted Sum-of-Pairs, wSOP), Strike [17], Entropy [31], BAliScore [31] o MetAl [4]. Sin embargo, todavía no existe un consenso acerca de qué métrica es la más apropiada o la más precisa para medir la calidad de los alineamientos. Por esta razón, es necesario considerar un enfoque MultiObjetivo para optimizar el problema, que permita obtener de forma simultánea alineamientos optimizados bajo dos o más criteriores de evaluación, a n de que los biólogos puedan disponer, no de una, sino de un conjunto de soluciones que les brinde la posibilidad de escoger una mejor solución disyuntiva.

Es por esto que el objetivo principal de este artículo es brindar un enfoque multiobjetivo al problema del Alineamiento Múltiple de Secuencias. Implementando una herramienta de optimización que incluye varias de las principales metaheurístcias de Optimización multiobjetivo: la técnica mayormente conocida NSGA-II [10], el algoritmo clásico SPEA2 [41], el algoritmo celular MOCell [19] y otros, considerando como funciones objetivo un conjunto de las métricas más comunes y usadas en el problema MSA, como son la Suma Ponderada de Pares con penalidad de gaps afines (wSOP), el porcentaje de columnas totalmente alineadas (TC), una métrica basada en información estructural STRIKE y otras. Y finalmente con esta herramienta realizar un ejemplo del funcionamiento optimizando simultánea- miento tres objetivos de calidad.

El resto del trabajo se organiza de la siguiente manera: una descripción formal y una formulación multiobjetivo del problema se describen en la Sección 2.. La complejidad del problema MSA se detalla en la Sección 3.. En la Sección 4. se detallan las funciones objetivo del Problema. En la Sección 5. se presenta una revisión de los trabajos relacionados a la optimización multiobjetivo aplicada al MSA, la herramienta software jMetalMSA es presentada en la Sección 6.. Finalmente, las conclusiones y líneas de trabajo futuro se comentan en la Sección 7.

2. DEFINICIÓN DEL PROBLEMA MSA

Esta sección define el dominio del problema del Alineamiento Múltiple de Secuencia en términos formales.

Sea Σ un alfabeto nito, por ejemplo un conjunto nito de caracteres, y $\Sigma \neq 0$, y un conjunto de k secuencias biológicas $S = (s_1, s_2, ..., s_k)$ de longitudes nitas y variables denotadas como l_1 a l_k y compuestas de caracteres $s_i = s_{i1}s_{i2}, ..., s_{il_i}(1 \leq i \leq k)$, S' es una matriz que representa el alineamiento óptimo de S, la cual está definida formalmente por la siguiente ecuación 2..1:

$$S' = (s'_{ij}),\ con\ 1 \leq i \leq k, 1 \leq j \leq l, max(l_i) \leq l \leq \sum_{i=1}^{k} l_i \tag{2.1}$$

Y cumple con:

1. $S'_{ij} \in \Sigma \cup \{-\}$, donde - denota el carácter de espacios o gaps ;
2. cada la $s'_i = s'_{i1}s'_{i2}, ..., s'_{il}(1 \leq i \leq k)$ de S' es exactamente igual a la secuencia

correspondiente s_i si eliminamos todos los gaps;

3. La longitud de todas las k secuencias es exactamente la misma;
4. $S^{'}$ no tiene columnas conformada solo por gaps.

En biología molecular, para las secuencias de ADN, el alfabeto Σ consiste de cuatro nuclétidos repre- sentados por los caracteres {A, T, G, C} y para las secuencias de proteinas, el alfabeto Σ consiste de 20 amino ácidos representados por los caracteres {A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y}. Un ejemplo de alineamiento se muestra a continuación, en el se representan cuatro secuencias con seis columnas alineadas las cuales están marcadas con un asterisco (*).

```
APPSVFAEVPJQKTM-AQPVMKLJ AKRS-V-E-PJFKTMR-IKMK-
                         --
                                  -LISKRA-YPJ-KTM-I---MALP
                                  -SASTIGVEPJCK-M-RA-P--KL
                                   *          ** ***
```

3. COMPLEJIDAD DEL PROBLEMA MSA

El problema MSA es considerado como un problema de Complejidad NP-Completo (NP-Hard), ya que la exploración del espacio de búsqueda se incrementa exponencialmente; según el número de secuencias a alinear k y a su longitud máxima L, definida como $O(k2^kL^k)$ [37]. Para tenerlo un poco más claro, en un grupo de solo 5 secuencias con un máximo de 10 residuos (amino-ácidos o nucleótidos) existen 1038 posibles combinaciones de alineamientos que se pueden generar. Inicialmente el alineamiento de un par de secuencias se realizaba mediante el uso de técnicas de Programación Dinámica [22]. Aunque el uso de estas estrategias garantizan alineamientos matemáticamente óptimos, no pueden ser aplicadas cuando se consideran más de dos secuencias en el proceso, debido a la complejidad antes mencionada. Por estas razones, cada vez más, se considera importante y necesario el uso de metaheurísticas de optimización en la resolución del problema.

4. FUNCIONES OBJETIVOS

Una función objetivo mide la calidad del alineamiento y refleja cuan cerca está dicho del alineamiento óptimo biológico. En esta sección, definimos las funciones objetivo que fueron consideradas en esta investigación, todas están destinadas a ser maximizadas. Para la formulación de estas funciones, hemos considerado al alineamiento a evaluar como S, con un conjunto de k secuencias alineadas representadas como $S = s_1, s_2, ..., s_k$ todas ellas con la misma longitud L.

4.1. Suma de pares (sum-of-pairs SOP)

La suma de pares (SOP) de un alineamiento, presentada en la ecuación 4..1, se calcula sumando todos los puntajes de las comparaciones de pares entre cada residuo en cada columna del alineamiento.

$$SOP(S) = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \sum_{c=1}^{L} ScoringMatrix(s_{ic}, s_{jc})$$

donde $ScoringMatrix$ representa la matriz que determina el costo de sustituir un residuo por otro. Esta matriz incluye también el valor de penalización de un gap que determina el costo de alinear un residuo con un gap. Esta penalización es sólo cuando se alinean un residuo con gap o viceversa, no cuando se alinean dos o más gaps.

4.2. Suma de pares ponderada con afinidad entre gaps (weighted sum of pairs with afine gaps wSOP)

La suma de pares ponderada con afinidad entre gaps (wSOP) se calcula restando el puntaje de suma de pares (comparaciones entre pares de cada uno de los caracteres amino-ácidos o nucleótidos) de cada

una de las columnas del alineamiento menos el puntaje de penalización a los gaps afines de cada una de las secuencias. La wSOP está representada por la ecuación 4.2:

$$wSOP(S) = \sum_{l=1}^{L} SP(l) - \sum_{i=1}^{k} AGP(s_i)$$

donde $SP\,(l)$ representa el puntaje de la suma de pares de la columna l el cuál está denifido como (ecuación 4..3):

$$SP(l) = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} W_{i,j}\ x\ \delta(s_{i,l}, s_{j,l})$$

En la ecuación 4.3, δ representa la matriz de sustitución usada (como pueden ser Pointed Accepted Mutation, - PAM [8] o Block Substitution Matrix, - BLOSUM [15]), la cual proporciona los costos de alineamientos de pares para cada uno de los aminoácidos y el valor de penalidad que se tiene al alinear un carácter con un gap. $W_{i,j}$ representa la ponderación (pesos) entre las sequencias s_i y s_j, definida en la siguiente ecuación 4.4:

$$W_{ij} = 1 - \frac{LD(s'_i, s'_j)}{max(|s'_i|, |s'_j|)} \tag{4.4}$$

donde LD representa la distancia de Levenshtein entre dos secuencias no alineadas (s'_i y s'_j) (el mínimo número de inserciones, eliminaciones o sustituciones de caracteres requeridas para convertir una secuencia en otra).

Finalmente, en la Ecuación 4..2, $AGP(s'_i)$ representa la penalización por gaps afines de la secuencia s_i, la cual está definida en la siguiente ecuación 4..5:

$$AGP(s_i) = (g_{open}\ x\ \#gaps) + (g_{extend}\ x\ \#spaces) \tag{4.5}$$

en la que g_{open} es el peso por empezar con un gap y g_{extend} es el peso por extender el gap con uno o mas espacios.

4.3.. Bali-score (SP y TC)

Baliscore es un software proporcionado por el benchmark BAliBASE v3.0 [34, 33], que incluye dos funciones de calidad la Suma de Pares (SP) y Columna Total (TC), las cuales estiman la precisión de calidad de los alineamientos de entrada en comparación a alineamientos de referencias generados por el benchmark. Por una parte, la métrica SP se calcula como la razón de la suma de las puntuaciones p para todos los pares de residuos en cada columna del alineamiento de entrada por la suma de las puntuaciones en el alineamiento de referencia; p = 1 si el par de residuos comparados se ajusta de forma idéntica al alineamiento de referencia, de lo contrario p = 0. Así, el puntaje SP aumenta con el número de secuencias alineadas correctamente. Por otra parte, la métrica TC se calcula considerando la relación de la suma de las puntuaciones c por el número de columnas en el alineamiento de entrada, siendo c = 1 si todos los residuos en la columna se alinean de forma idéntica al alineamiento de referencia, de lo contrario c = 0.

4.4.. Single structure induced evaluation (STRIKE)

STRIKE [17] representa una nueva métrica para evaluar la calidad de los alineamientos basada en información estructural, de al menos, una de las secuencias del alineamiento. La información estructural de las secuencias de proteínas es comúnmente obtenida desde el sitio web del Protein Data Bank (PDB) [2].

Esta métrica de evaluación permite identificar de mejor manera la exactitud en los alineamientos mejor que otras puntuaciones clásicas como BLOSUM62 [15] and PAM250 [8]. Además, supera claramente a las otras métricas clásicas cuando las secuencias son evolutivamente más distantes [17]. STRIKE también muestra un fuerte efecto de correlación no-paramétrico con los valores BAliscore (subsección 4.3.). Es decir, en una comparativa entre dos diferentes alineamientos, tanto

BAliscore como STRIKE, generalmente identifican al mismo alineamiento como el mejor (alrededor del 79 % de los casos) [17]].

4.5.. El porcentaje de columnas totalmente alineadas

El porcentaje de columnas totalmente alineadas (TC) se re ere al número de columnas que están compuestas totalmente del mismo carácter en cada una de sus las (amino ácidos o nucleótidos). Esta función objetivo necesita ser maximizada para asegurar la mayor cantidad de regiones conservadas dentro del alineamiento. TC puede ser definida como (Ecuación 4..6):

$$TC(S) = 100 \sum_{l=1}^{L} \frac{ColumnaAlineada(S_l)}{L}$$

donde S_l representa la l-ésima columna del alineamiento S, tal que $S_l = s_{il}\ \forall i = 1, .., k$, y la función $ColumnaAlineada(S_l)$ está definida como (Ecuación 4..7):

$$ColumnaAlineada(S_l) = \begin{cases} 1 & Si\ s_{il} = s_{1l}\ \forall i = 2, ..., k \\ 0 & caso\ contrario \end{cases}$$

4.5.1.. Porcentaje de no-gaps

El porcentaje de no-gaps mide el número de residuos con respecto al número de gaps dentro del alineamiento, está de nido en la Ecuación4..8:

$$NonGaps(S) = 100 \sum_{i=1}^{k} \sum_{j=1}^{L} \frac{EsNonGap(s_{ij})}{k * L}$$

donde s_{ij} representa el símbolo en la j-ésima posición de la i-ésima secuencia en el alineamiento S. La función $EsNonGap$ para un determinado residuo del alineamiento está de nido en la siguiente Ecuación 4.9:

$$EsNonGap(residuo) = \begin{cases} 1 & si\ residuo = \text{“}-\text{”}(gap) \\ 0 & caso\ contrario \end{cases}$$

5. ESTADO DEL ARTE

Recientemente ha habido un creciente interés en la formulación multiobjetivo de los problemas de optimización que surgen en el campo de la Bioinformática. Handl et al. muestra en [14] los beneficios de la Optimización Multiobjetivo aplicada específicamente en el campo de la Bioinformática en comparación con los enfoques monoobjetivo. A continuación se detallan algunos trabajos:
En los últimos tiempos, se han publicado varias propuestas multiobjetivo para resolver el problema del MSA usando técnicas metaheurísticas. La primera aproximación multiobjetivo fue presentada por Seeluangsawat et al. quienes publicaron MOMSA (Multiple Objective Multiple Sequence Alignment) un algoritmo evolutivo que optimiza dos objetivos de calidad implementados en una sola función, con el n de mejorar las soluciones obtenidas desde el software Clustal X [32] en [30]. La población inicial del algoritmo se genera a partir de los resultados generados por el software Clustal X extendiendo el tamaño de los alineamientos un 10 % mas. Considera dos objetivos a optimizar, la Suma de Pares

y La Penalidad de Gaps, empleando como matriz de distancia Blosum45. MOMSA implementa un operador de cruce de dos puntos y tres operadores de mutación (Movimientos de columnas, Cambio de posición de Gaps e Intercambio aleatorio entre residuo y grupos de gaps). Este algoritmo propuesto fue probado con nueve conjuntos de datos del benchmark BAliBASE 2.0 [35].

Ortuño et al. presentaron en [24] MO-SAStrE (Multiobjective Optimizer for Sequence Alignments based on Structural Evaluations), cuyo algoritmo está basado en la clásica metaheurística NSGA- II y trata de optimizar tres objetivos de calidad, uno basado en información estructural STRIKE, el porcentaje de no-Gaps y el porcentaje de columnas totalmente conservadas. En MO-SAStrE, la población inicial se genera mediante la estrategia basada en alineamientos precomputados generados por ocho enfoques representativos del estado del arte, tales como Muscle, ClustalW, Ma t, T-Coffee, Kalign, RetAlign, ProbCons y FSA. Emplea el operador de cruce de un solo punto y como operador de mutación desplazamiento aleatorio de una región de gaps dentro de una secuencia. El rendimiento del algoritmo fue evaluado resolviendo los 218 problemas del benchmark BAliBASE (v3.0) [33]. Los resultados multiobjetivo de MOSAStrE fueron evaluados usando el indicador de calidad multiobjetivo Hypervolumen [42].

Soto y Becerra propusieron en [31] un algoritmo evolutivo multiobjetivo, también inspirado en NSGA-II, para optimizar alineamientos múltiples de secuencias previamente alineadas. Para evaluar la calidad de los individuos utilizan dos métricas de calidad MetAl [5] y Entropy, esta última mide la variabilidad de un MSA de niendo las frecuencias de la ocurrencia de cada letra en cada columna, Siendo estas dos los objetivos a optimizar dentro del algoritmo. Los resultados fueron validados por cuatro métricas de calidad estándar: La suma de pares (SOP), Número de Columnas Totalmente Alineadas (TC), MetAl e Hypervolume [42]. SP y TC fueron computadas usando el script de Baliscore [33]. Similar a MO-SAStrE, esta propuesta construye la población inicial usando los alineamientos producidas por otros técnicas MSA de última generación, más algunas modi caciones de operadores genéticos. Como operadores de variación aplicaron cruzamiento de dos puntos y mutación de inserción aleatoria y desplazamiento. El método propuesto fue validado resolviendo los 218 instancias del benchmark BAliBASE (3.0).

Kaya et al. presentaron MSAGMOGA [16], basado también en NSGA-II, considera tres objetivos conflictivos a optimizar: minimización de la penalidad de gaps afín y de Similitud y la maximización de la Compatibilidad. MSAGMOGA aplica operadores de cruce de un solo punto y dos puntos y tres operadores de mutación (cambio aleatorio de gaps, desplazamiento hacia derecha e izquierda de bloques degaps). Los resultados se obtuvieron del benchmark BAliBASE 2.0 [35].

da Silva et al. presentaron Parallel Niche Pareto AlineaGA (PNPAlineaGA) una versión multiobjetivo de su algoritmo Parallel AlineaGA en [7]. Usa dos objetivos a optimizar: La Suma de pares (SOP) con la matriz de distancia PAM350 y el número de columnas totalmente alineadas en el alineamiento. PNPAlineaGA implementa tres operadores de cruce y seis versiones de operadores de mutación. Los resultados fueron validados sobre 8 datasets del benchmark BAliBASE 2.0 [35].

Abbasi et al. publicaron en su trabajo [1] varios técnicas de Búsqueda Local aplicadas al Alineamiento Múltiple de Secuencias Multiobjetivo, sus objetivos a optimizar fueron maximizar la Suma de Pares y minimizar el número de gaps. A pesar que ilustran buenos resultados obtenidos gracias a su técnica propuesta, Pareto Local Search, indican que deben mejorarla, ya que en algunos casos se estanca en óptimos locales. Para ello, sugieren perturbar el conjunto de alineamientos en un archivo de soluciones y reiniciar la búsqueda local cada cierto número de iteraciones. El rendimiento de los algoritmos de búsqueda local fue probados resolviendo 38 instancias del benchmark BAliBASE 3.0. Zhu et al. presentaron en [39] una propuesta basada en el algoritmo evolutivo multiobjetivo basado en la descomposición (MOEA/D) aplicado a resolver el problema del MSA, llamado MOMSA. Zhu et al. resaltan dos nuevas aportaciones dentro de su algoritmo: la generación de la población inicial y un nuevo operador de mutación. La población inicial es generada mediante una nueva técnica basada en inserción de gaps similar al funcionamiento del algoritmo en SAGA [23]. A partir de alineamientos previamente obtenidos de alguna técnica como ClustalW, las secuencias son divididas al azar en dos grupos, se insertan un número gaps en posiciones aleatorias, y luego estos grupos de secuencias son unificados para conformar el alineamiento final. El rendimiento de esta técnica se comparó con varios métodos de alineamientos basados en algoritmos evolutivos, y también con técnicas basadas en métodos progresivos, usaron el conjunto de datos del BAliBASE 2.0 y BAliBASE 3.0 para evaluar su rendimiento.

Recientemente, Rubio-Largo et al. propusieron dos nuevas técnicas para resolver el problema MSA basadas, el algoritmo Hybrid Multiobjective Arti cial Bee Colony (HMOABC) [29] y el algoritmo hybrid multiobjective memetic metaheuristic (H4MSA) [28] Ambos basados en algoritmos bioinspirados, HMOABC inspirado en el comportamiento natural de las colonias de abejas y, H4MSA inspirado en la metaheurística memética Shu ed Frog-Leaping Algorithm (SFLA) [13]. Con el objetivo de preservar la calidad y consistencia de sus alineamientos, ambos consideran dos funciones objetivo: La Suma Ponderada de Pares con penalidad de gaps afines(WSP) y el número de columnas totalmente conservadas (TC), respectivamente. La metodología híbrida de ambos algoritmos, está basada en el uso de la técnica progresiva KAlign [18], unos de los software más rápidos y precisos del estado del arte. En HMOABC, esta técnica adicional se utiliza en la fase de exploración del algoritmo ABC (Artificial Bee Colony) y, en H4MSA ésta se emplea como un procedimiento de búsqueda local que busca alinear mejormente pequeñas porciones de los alineamientos. Ambos algoritmos, generan aleatoriamente la población inicial. El desempeño de H4MSA fue probado en tres benchmarks de referencia: BAliBASE [33], Protein REFerence Alignment Benchmark (PREFAB) [12] y el Sequence Alignment Benchmark (SABmark) [36] y, el desempeño de HMOABC fue probado únicamente utilizando el benchmark BAliBASE (v3.0).
Y por último, Ranjani Rani et al. propusieron en [27] dos algoritmos: Hybrid Genetic Algorithm with Arti cial Bee Colony Algorithm (GA-ABC) y Bacterial Foraging Optimization Algorithm (MO-BFO), pero su trabajo se centra principalmente en el rendimiento del algoritmo MO-BFO ya que obtiene un mejor rendimiento y porque identifica mayormente bloques conservados dentro de los alineamientos. Ranjani Rani et al. incorporaron en su trabajo cuatro objetivos a optimizar: la maximización de la Similitud, el porcentaje de no-gaps y Bloques Conservados, y la Minimización de la penalidad por gaps. Los algoritmos propuestos fueron evaluados resolviendo el benchmark BAliBASE v3.0 y comparados con otros métodos MSA clásicos muy usados como: ClustalW y Clustalω, KAlign, MUSCLE, MAFFT y con varios algoritmos genéticos .

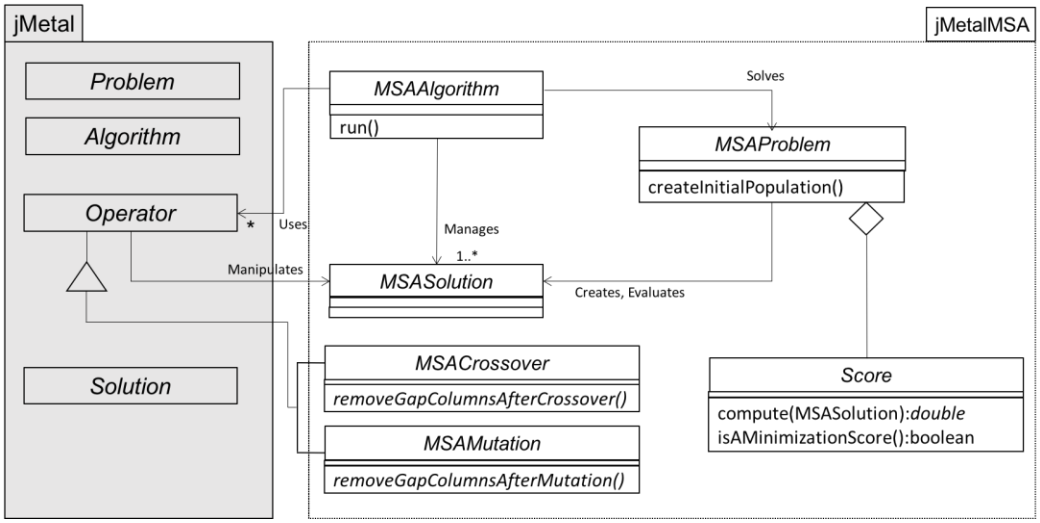

Figura 1: Arquitectura de jMetalMSA (algunas relaciones de herencia no han sido incluidas para simplificar el diagrama).

6.. HERRAMIENTA SOFTWARE: JMETALMSA

Con el objetivo de ofrecer a la comunidad científica de la biología computacional una plataforma libre que incluya algoritmos de optimización de última generación dirigidos a resolver diferentes formulaciones del MSA, presentamos jMetalMSA, una herramienta software de código abierto para el alineamiento múltiple de secuencias con metaheurísticas de optimización multiobjetivo. El código

fuente del proyecto se encuentra disponible públicamente en GitHub[1].
A continuación, describimos la arquitectura de software y sus principales características, incluyendo dos casos de uso práctico de la herramienta.

6.1.. Arquitectura de jMetalMSA

jMetalMSA está basado en el framework de optimización multiobjetivo jMetal [11][20], del cual toma la mayoría de las clases centrales. La arquitectura orientada a objetos de jMetalMSA se muestra en la Figura 1, en la que podemos observar que está compuesta de cuatro clases principales (interfaces Java).
Tres de ellos (MSAProblem, MSAAlgorithm, y MSASolution) heredan de sus equivalentes en jMetal (algunas relaciones de herencia se han omitido en el diagrama con el n de simplificarlo y facilitar su comprensión), y además hay una clase Score para representar las métricas de calidad MSA .
Muchas propuestas para resolver el problema del MSA con metaheurísticas incluyen un método de inicialización basado en la toma de un conjunto de alineamientos pre-calculadas obtenidas por otras metodologías MSA no metaheurísticas (tales como Clustal-W, MAFFT, MUSCLE, etc.), por lo que la clase MSAProblem incluye el método createInitialPopulationMethod[2] () la cual está destinada para incorporar este tipo de estrategias.

6.2.. Algoritmos incluidos en jMetalMSA

Como jMetalMSA se basa en jMetal, la mayoría de los algoritmos incluidos en el último se puede utilizar en el primero. La codificación de los alineamientos (ver sección 6.3.) está basada en la representación de los grupos de gaps, por lo que los algoritmos de optimización continua, tales co- mo la optimización de enjambre de partículas y la evolución diferencial no pueden ser utilizados en sus versiones clásicas. Los algoritmos multiobjetivos disponibles en jMetalMSA son: NSGA-II [10], NSGA-III [9], SMS-EMOA [3], SPEA2 [40], MOEA/D [38], MOCell [21], y GWASF-GA [26]. Estos algoritmos constituyen el conjunto de algoritmos evolutivos multiobjetivos representativos del estado del arte: están los referencias clásicas (NSGA-II, SPEA2), celular (MOCell), basada en la descompo-sición (MOEA/D), basada en indicadores (SMS-EMOA) y basado las preferencias (GWASF-GA).

6.3.. Codificación de las soluciones (MSA)

Con el objetivo de reducir el alto costo de memoria y el tiempo de ejecución que requieren las codi - caciones clásicas de los alineamientos, cadenas de caracteres o matrices numéricas como en el caso de MO-SAStrE [24], hemos implementado una codificación, rápida y de bajo costo de memoria, basada en los grupos de gaps dentro de las secuencias, similar a la propuesta por [28]. Esta representación MSA almacena únicamente las posiciones (inicio y n) de los grupos de gaps dentro de las secuencias alienadas. En la Figura 2 se ilustra un ejemplo.

Figura 2: Ejemplo de un alineamiento (izquierda) y como es codificada en jMetalMSA (derecha).

Por lo que, dada una secuencia S, esta es codificada a $S^{'}$ de la siguiente manera:
$S^{'}$: $[(Igg_1, Fgg_1), (Igg_2, Fgg_2), ..., (Igg_n, Fgg_n)]$
donde n es el número de grupos de gaps de la secuencia S e Igg_x y Fgg_x representan la posición inicial y la posición final del grupo de gaps x dentro de la secuencia S, respectivamente. Esta codificación reduce el tiempo de ejecución de los operadores genéticos de cruce y mutación, ya que únicamente se ejecutan operaciones numéricas sobre los grupos gaps y no se tienen que realizar operaciones sobre

[2] [1] Web del Proyecto jMetalMSA en GitHub: http://github.com/jmetal/jmetalmsa

grandes secuencias de caracteres.

6.4.. Operadores evolutivos

El operador de cruce es el operador de Cruce de un sólo punto que ha sido adaptado a los alineamien- tos [6]. La lista de operadores de mutación incluida en jMetalMSA es:

- Desplazamiento de grupo de gaps: selecciona una secuencia aleatoriamente del MSA; Y del grupo de huecos que contiene, se escoge uno al azar y se lo desplaza a otra posición aleatoria dentro de la misma secuencia (véase Figura 3).

- Separación de grupos No-gaps: se selecciona aleatoriamente un grupo de residuos (no-gaps) y se lo divide en dos grupos insertando un gap en una posición aleatoria entre ellos. (véase Figura 4).

- Inserción de un gap: Inserta un gap en una posición aleatoria para cada secuencia del alinea- miento (véase Figura 5).

- Fusión de dos grupos de gaps adyacentes: Selecciona un grupo aleatorio de gaps y éste se fusiona con su grupo de gaps más cercano (véase Figura 6).

- Mutación múltiple: Es una combinación del resto de operadores en uno solo.

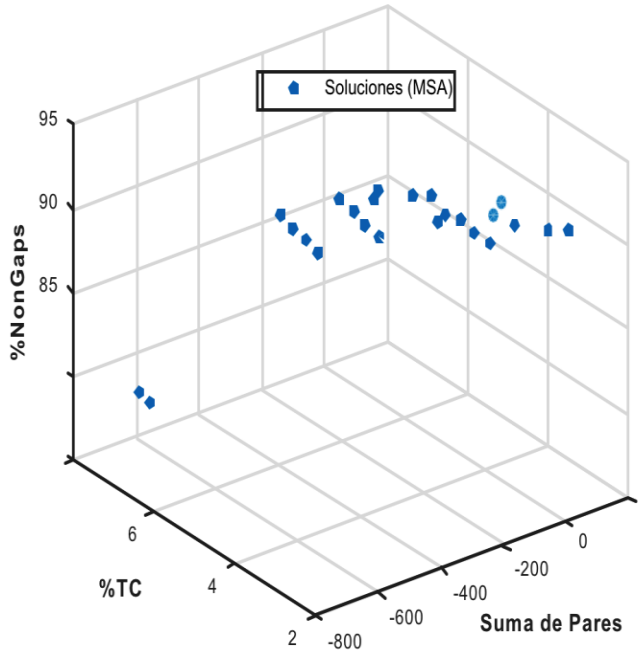

Figura 7: Aproximaciones del frente de Pareto obtenida por el algoritmo MOCell resolviendo la instancia BB11001 del BAliBASE 3.0 con una formulación de tres objetivos Suma de Pares, TC y Porcentaje de Non-gaps

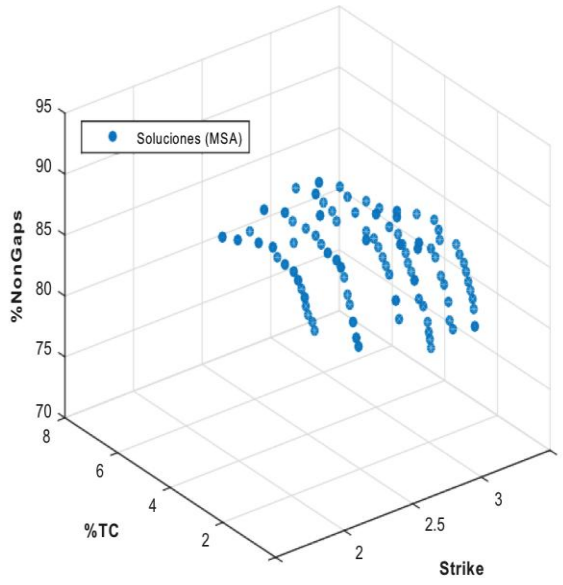

Figura 8: Aproximaciones del frente de Pareto obtenida por el algoritmo NSGAII resolviendo la instancia BB11001 del BAliBASE 3.0 con una formulación de tres objetivos son: STRIKE, TC y Porcentaje de Non-gaps

```
GKGDPKKPR-GK--MSSYAFFVQTSREEHKKKHPDASVNFSEFSKKCSERWKTMSAKEKGKFEDMAKADKARYEREMKTY--I----PPK---- GE
MQDRVKRP------MNAFIVWSRDQRRKMALENPRMR-N-SEISKQLGYQWKMLTEAEKWPFFQEAQKLQAMHREKYPNY--KYRP-RRKAKMLPK
MKKLKKHPDFPKKPLTPYFRFFMEKRAKYAKLHPEMS-N-LDLTKILSKKYKELPEKKKMKYIQDFQREKQEFERNLARF--REDH-PDLIQNAKK
MH--IKKP LNAFMLYMKEMRANVVAESTLKE-S-AAINQILGRRWHALSREEQAKYYELARKERQLHMQLYPGWSARDNYGKKKKRKREK
```

Figura 9: MSA con el mejor Suma de Pares

```
GKGDPKKPR-GKMSSYA-F--FVQTSREEHKKKHPDASVNFSEFSKKCSERWKTMSAKEKGKFEDMAKADKA ----------------- RYEREMKTYIP--PKGE
MQDRVKRP------MNA-FIVWSRDQRRKMALENPRMR-N-SEISKQLGYQWKMLT--EAEKWPFFQEAQKLQAMH---REKYPNYKYRPRRKAKML-------PK--
MKKLKKHPDFPKKPLTPYFR-FFMEKRAKYAKLHPEMS-N-LDLTKILSKKYKELP--EKKKMKYIQDFQREKQ----------EFERNLARFREDHPDLIQNAKK--   MH--
IKKP------LNA-FMLYMKEMRANVVAESTLKE-S-AAINQILGRRWHALSREEQAKYYELARKERQLHMQLYPGWSARDNYGKKKRKRE----------K--
```

Figura 10: MSA con el mejor TC

```
GKGDPKKP---RGKMSSYAFFVQTSREEHKKKHPDAS--VNFSEFSKKCSER--WKTMSA-KEKGKFEDMAKADKARYEREMKTYIPPKGE
MQDRVKRP---MNAFIVWSRDQRRKMALENPRMRNSEISKQLGYQWKMLTE---AEKWPFFQEAQKLQAMHREKYPNYKYRPRRKAKMLPK
MKKLKKHPDFPKKPLTPYFRFFMEKRAKYAKLHPEMSNLDLTKILSKKYKELPEKKKMKYIQDFQREKQEFERNLARFREDHPDLIQNAKK
MH--IKKP---LNAFMLYMKEMRANVVAESTLKESAAINQILGRRWHALSREEQAKYYELARKERQLHMQLYPGWSARDNYGKKKKRKREK
```

Figura 11: MSA con el mejor %Non-Gaps

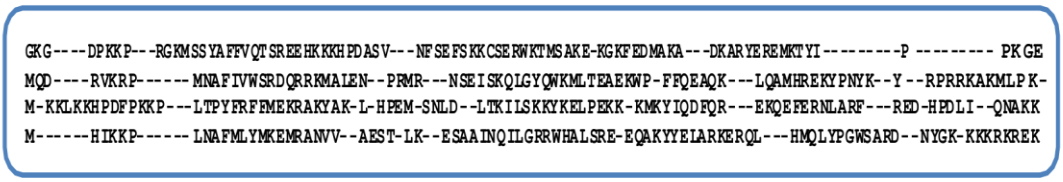

Figura 12: MSA con el mejor STRIKE

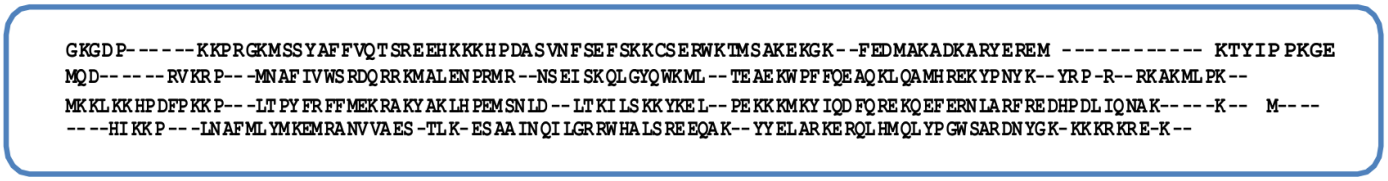

Figura 13: MSA con el mejor TC

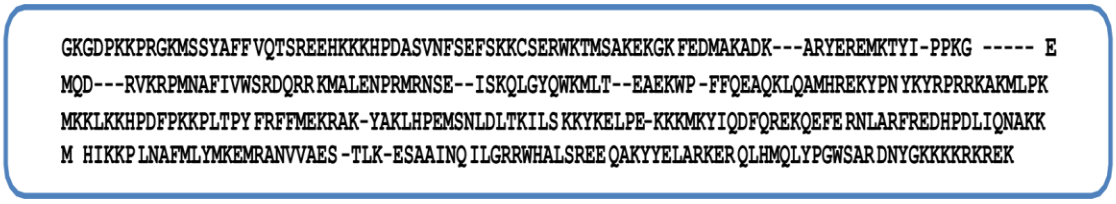

Figura 14: MSA con el mejor %Non-Ga

RECEIVED: DECEMBER, 2019.
REVISED: MARCH, 2020.

REFERENCIAS

[1]ABBASI, M., PAQUETE, L., and PEREIRA, P. B. (2015): Local search for multiobjective multiple sequence alignment. In **Bioinformatics and Biomedical Engineering**, volume 9044 of Lecture Notes in Computer Science, 175-182. Springer International Publishing, Berlin.

[2]BERMAN, H., WESTBROOK, J., FENG, Z., GILLILAND, G., BHAT, T., WEISSIG, H., SHINDYALOV, I., and BOURNE, P. (2000): The protein data bank. **Nucleic Acids Research**, 28, 235-242.

[3]BEUME, N., NAUJOKS, B., and EMMERICH, M. (2007): Sms-emoa: Multiobjective selection based on dominated hypervolume. **European Journal of Operational Research**, 18, 1653 1669.

[4]BLACKBURNE, B. and WHELAN, S. (2012a): Measuring the distance between multiple sequence alignments. **Bioinformatics**, 28, 495-502.

[5]DA SILVA, F., PÉREZ, J. S., PULIDO, J. G., and RODRÍGUEZ, M. V. (2010): Alineaga a genetic algorithm with local search optimization for multiple sequence alignment. **Applied Intelligence**, 32, 164-172.

[6]DA SILVA, F. J. M., PÉREZ, J. M. S., PULIDO, J. A. G., and RODRÍGUEZ, M. A. V. (2011): Parallel Niche Pareto AlineaGA an evolutionary multiobjective approach on multiple sequence alignment. **Journal of Integrative Bioinformatics**, 8,174.

[7]DAYHO , M., SCHWARTZ, R., and B.C. ORCUTT, B. (1978): A model of evolutionary change in proteins. **Atlas of Protein Sequences and Structure**, 5,345-352.

[8] DEB, K. and JAIN, H. (2014): An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints **IEEE Transactions on Evolutionary Computation,** 18, 577-601.

[9] DEB, K., PRATAP, A., AGARWAL, S., and MEYARIVAN, T. (2002): A fast and elitist multiobjective genetic algorithm: NSGA-II. **IEEE Transactions on Evolutionary Computation**, 6,182 197.

[10] DURILLO, J. J. and NEBRO, A. J. (2011): jmetal: A java framework for multi-objective optimization. **Advances in Engineering Software**, 42, 760-771.

[11] EDGAR, R. C. (2004): Muscle: multiple sequence alignment with high accuracy and high throughput. **Nucleic Acids Research**, 32, 1792-1817.

[12] EUSU , M., LANSEY, K., and PASHA, F. (2006): Shu ed frog-leaping algorithm: a memetic meta- heuristic for discrete optimization. **Engineering Optimization**, 38,129-154.

[13] HANDL, J., KELL, D. B., and KNOWLES, J. (2007): Multiobjective optimization in bioinformatics and computational biology. **IEEE/ACM Transactions On Computational Biology And BioinforMatics / IEEE, ACM**, 4, 279-292.

[14] HENIKO , S. and HENIKO , J. (1992): Amino acid substitution matrices from protein blocks **Proceedings of the National Academy of Sciences**, 89, 10915-10919.

[15] KAYA, M., SARHAN, A., and ABDULLAH, R. (2014): Multiple sequence alignment with a ne gap by using multi-objective genetic algorithm. **Computer Methods and Programs in Biomedicine**, 114, 38-49.

[16] KEMENA, C., TALY, J., KLEINJUNG, J., and NOTREDAME, C. (2011): Strike: evaluation of protein msas using a single 3d structure. **Bioinformatic**s, 27, 3385-3391.

[17] LASSMANN, T. and SONNHAMMER, E. L. (2005): Kalign an accurate and fast multiple sequence alignment algorithm BMC. **Bioinformatics**, 6, 1-9.

[18] NEBRO, A., DURILLO, J., LUNA, F., DORRONSORO, B., and ALBA, E. (2007): Design issues in a multiob- jective cellular genetic algorithm. In Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., and Murata, T., editors, **Evolutionary Multi-Criterion Optimization. 4th International Conference, EMO 2007**, volume 4403 of Lecture Notes in Computer Science, 126-140. Springer, N. York.

[19] NEBRO, A., DURILLO, J. J., and VERGNe, M. (2015): Redesigning the jmetal multi-objective optimization framework. In **Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation, GECCO Companion '15**, 1093-1100, New York, NY, USA. ACM.

[20] NEBRO, A. J., DURILLO, J. J., LUNA, F., DORRONSORO, B., and ALBA, E. (2009): Mocell: A cellular ge- netic algorithm for multiobjective optimization. **International Journal of Intelligent Systems**, 24, 723-725.

[21] NEEDLEMAN, S. and WUNSCH, C. (1970): A general method applicable to the search for similarities in the amino acid sequence of two proteins. **Journal of Molecular Biology**, 48, 443-453.

[22] NOTREDAME, C. and HIGGINS, D. G. (1996): Saga: Sequence alignment by genetic algorithm. **Nucleic Acids Research**, 24,1515-1524.

[23] ORTUÑO, F., VALENZUELA, O., ROJAS, F., POMARES, H., FLORIDO, J., URQUIZA, J., and ROJAS, I. (2013): Optimizing multiple sequence alignments using a genetic algorithm based on three objectives: struc- tural information, non-gaps percentage and totally conserved columns. **Bioinformatics**, 2112-2121.

[24] PEI, J. (2008): Multiple protein sequence alignment. **Current Opinion in Structural Biology,** 18, 382-386.

[25] RANI, R. R. and RAMYACHITRA, D. (2016): Multiple sequence alignment using multi-objective based bacterial foraging optimization algorithm. **Biosystems**, 150,177-189.

[26] RUBIO-LARGO, A., VEGA-RODRIGUEZ, M., and GONZALEZ-ALVAREZ, D. (2015): A hybrid multiobjective memetic metaheuristic for multiple sequence alignment Evolutionary Computation. **IEEE Transactions** , 99, 1-16.

[27] RUBIO-LARGO, A., VEGA-RODRÍGUEZ, M., and GONZÁLEZ-ALVAREZ, D. (2016): Hybrid multiobjective artificial bee colony for multiple sequence alignment. **Applied Soft Computing**, 41,157-168.

[28] SABORIDO, A. R. and LUQUE, M. (2016): **Global wasf-ga: An evolutionary algorithm in multi- objective optimization to approximate the whole pareto optimal front**. Evolutionary Computation. In Press.

[29] SEELUANGSAWAT, P. and CHONGSTITVATANA, P. (2005): A multiple objective evolutionary algorithm for multiple sequence alignment. In **Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, GECCO '05**, 477-478, New York,

[30] SOTO, W. and BECERRA, D. (2014): A multi-objective evolutionary algorithm for improving multiple sequence alignments In Campos, S., editor, **Advances in Bioinformatics and Computational Biology,** 8826 of Lecture Notes in Computer Science, 73-82. Springer International Publishing, Berlin.

[31] THOMPSON, J., D.G.HIGGINS, and GIBSON, T. (1994): Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-speci c gap penalties and weight matrix choice. **Nucleic Acids Research**, 22, 4673-4680.

[32] THOMPSON, J., KOEHL, P., and POCH, O. (2005): Balibase 3.0: latest developments of the multiple sequence alignment benchmark. **Proteins**, 61, 127-136.

[33] THOMPSON, J. D., PLEWNIAK, F., and POCH, O. (1999a): Balibase: a benchmark alignment database for the evaluation of multiple alignment programs. **Bioinformatics**, 15, 87-88.

[34] THOMPSON, J. D., PLEWNIAK, F., and POCH, O. (1999b): A comprehensive comparison of multiple sequence alignment programs. **Nucleic Acids Research**, 27, 2682-2690.

[35] VAN WALLE, I., LASTERS, I., and WYNS, L. (2005): Sabmark?a benchmark for sequence alignment that covers the entire known fold space. **Bioinformatics**, 21, 1267-1268.

[36] WATERMAN, M., SMITH, T., and BEYER, W. (1976): Some biological sequence metrics . **Advances in Mathematics**, 20, 367-387.

[37] ZHANG, Q. and LI, H. (2007): MOEA/D: A multiobjective evolutionary algorithm based on decomposition. **IEEE Trans. Evolutionary Computation**, 11, 712-717

[38] ZHU, H., HE, Z., and JIA, Y. (2016): A novel approach to multiple sequence alignment using multiobjective evolutionary algorithm based on decomposition. **IEEE Journal of Biomedical and Health Informatics**, 20, 717-727.

[39] ZITZLER, E., LAUMANNS, M., and THIELE, L. (2001): SPEA2: Improving the strength pareto evolutionary algorithm. **Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology** (ETH), Zurich.

[40] ZITZLER, E., LAUMANNS, M., and THIELE, L. (2002): SPEA2: Improving the strength pareto evolutionary algorithm. In Giannakoglou, K., Tsahalis, D., Periaux, J., Papailou, P., and Fogarty, T., editors, **EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems**, 95-100, Athens.

[41] ZITZLER, E. and THIELE, L. (1999): Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. **IEEE Transactions on Evolutionary Computation**, 3, 257-271.