View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Servicio de Difusión de la Creación Intelectual

Problema de ensamblado de fragmentos de ADN resuelto mediante metaheurísticas y paralelismo

Tesis presentada para cumplir con los
requerimientos del grado de
DOCTOR EN CIENCIAS DE LA COMPUTACIÓN
en la
UNIVERSIDAD NACIONAL DE SAN LUIS
San Luis, Argentina

Autora:
Gabriela F. Minetti
e-mail: minettig@ing.unlpam.edu.ar

Asesores:
Dres. Enrique Alba y Mario Guillermo Leguizamón

7 de Noviembre de 2011

1. Introducción

Esta tesis aborda el problema de ensamblado de fragmentos del genoma[1] de un organismo mediante la utilización de técnicas metaheurísticas. La obtención de un ensamblado completo y de alta calidad de un genoma tiene implicaciones directas en la Biología y la Medicina. Esta tarea es particularmente compleja cuando se trabaja con genomas de gran tamaño, como es el caso de la mayoría de los eucariotas (animales, plantas y hongos). Razón por la cual, es sumamente necesario contar con algoritmos ensambladores que permitan obtener secuencias genómicas de alta calidad en tiempos razonables y, así, proseguir de manera segura y eficiente con las etapas subsiguientes del proyecto de genómica.

1.1. Antecedentes y Motivaciones

La mayoría de los problemas de la vida real muestran un alto grado de vinculación entre los parámetros (epístasis), muchas soluciones localmente óptimas (multimodalidad) y una alta dimensión. Estos problemas complejos están cobrando una mayor notoriedad en la actualidad; esto puede observarse en las áreas de: Comunicaciones, Bioinformática, Planificación, Ambientes Industriales, etc. En éstos y otros campos de investigación a menudo es esencial modelar y resolver tareas de optimización, de aprendizaje o de investigación para aplicaciones que no admiten una fácil formulación. De hecho, son frecuentes los casos donde el problema no es diferenciable, tiene un gran número de restricciones u objetivos, no admite las condiciones de contorno, o no está completamente definido.

[1] Secuencia completa de Ácido Desoxirribonucleico, ADN.

Cuando es necesario tomar decisiones sobre el valor de ciertas condiciones del problema (por ejemplo: costo, peso, ganancias, tiempo, eficiencia, etc.) y tales decisiones afectan el resultado final de su resolución, entonces se enfrenta un problema de optimización. La optimización es una rama de las Matemáticas Aplicadas para encontrar la mejor o una muy buena solución en la resolución de problemas cuantitativos en muchas disciplinas; tales como: Física, Biología, Ingeniería, Bioinformática y Economía. La Bioinformática, específicamente, es un campo interdisciplinardedicado a desarrollar técnicas que permitan: analizar secuencias genéticas, identificar y predecir estructuras moleculares, extraer caracterísitcas de microarreglos de datos, etc. Actividades que, en su mayoría, necesitan ser formuladas como problemas de optimización para poder llevarse a cabo.

El conjunto de técnicas en Bioinformática utilizadas en las distintas áreas de la Biología es extenso y de componentes heterogéneos. Se pueden distinguir dos grandes grupos de técnicas algorítmicas. Uno de ellos está conformado por algoritmos diseñados para un uso bioinformático específico; por ejemplo: BLAST [2, 3] y CLUSTALW-pairwise [40] para alinear un par de secuencias de ADN, FASTA [30, 31], PSI-BLAST [3], SSEARCH [31] y HMMER-HSSP [33] para identificar relaciones entre proteínas, PHRAP [10], TIGR assembler [39], STROLL [4, 5], CAP3 (*Contig Assembly Program*) [11] y Celera Assembler [26] para ensamblar fragmentos de un genoma. En tanto que, el otro subconjunto está formado por un grupo de técnicas modernas de uso generalizado, denominadas metaheurísticas. Éstas se utilizan en casi todas las áreas de la Bioinformática y está representado por numerosas familias algorítmicas, a saber: los algoritmos genéticos (GAs), la optimización basada en colonias de hormigas (ACO), el enfriemiento simulado (SA), PALS y la búsqueda en vecindarios variables (VNS) en la alineación de secuencias [14, 19, 27, 28] y en el ensamblado de fragmentos [15, 17, 18, 20, 29], diversos algoritmos evolutivos se usan en la identificación de relaciones proteínicas [7, 34], la identificación del perfil de la expresión genética [6] y en el análisis de la estructura proteínica [7, 9, 12, 36, 38, 41].

La Bioinfórmática se divide, entonces, en distintos campos. Uno de ellos está directamente relacionado con la identificación de secuencias genómicas y proteómicas. En este campo se distinguen 3 áreas bien definidas: alineación de secuencias, identificación de relaciones proteínicas y ensamblado de ADN, siendo la última el objeto de estudio de esta tesis.

El ensamblado de fragmentos de ADN se formula como un problema de optimización combinatoria NP-duro [32]. Por consiguiente: el tamaño del genoma, el número de fragmentos leídos y secuenciados y la presencia de ruido en los datos son factores altamente influyentes en la capacidad de cualquier algoritmo para llevar a cabo esta tarea. También es ampliamente conocido el hecho que las metaheurísticas son técnicas usadas exitosamente en una amplia gama de problemas de optimización combinatoria NP-duros: encaminamiento, telecomunicaciones, secuenciación de tareas, planificación de recursos, corte y empaquetado, diseño ingenieril, entre muchos otros. El éxito de las metaheurísticas en esta clase de problemas se basa fundamentalmente en que no son exhaustivas ni deterministas. Esto reduce considerablemente el esfuerzo computacional empleado; además, permiten producir múltiples resultados para una misma situación. Por otra parte, esta clase de algoritmos pueden prescindir de datos exactos y completos para obtener más y mejores soluciones. Así mismo, las metaheurísticas también han resultado ser eficientes cuando la complejidad del problema es alta y el espacio de soluciones asociado es grande o ambos crecen continuamente. Además son fácilmente paralelizables tanto a nivel algorítmico como de hardware. Estas dos últimas características son muy importantes a la hora de manipular enormes cantidades de información. Tales ventajas y características son difíciles de encontrar o incorporar en los algoritmos diseñados específicamente para resolver un solo tipo de problema.

Todo lo expresado anteriormente parece justificar con creces la elección de las técnicas metaheurísticas para resolver el problema de ensamblado de fragmentos (FAP), pero ¿son capaces de cumplir con las siguientes hipótesis?:

H1. Una metaheurística supera el estado del arte en la resolución de FAP (estado del arte).

H2. Los algoritmos metaheurísticos son capaces de manipular genomas de gran tamaño sin disminuir la calidad de las soluciones halladas (complejidad).

H2.1 Si H2 se confirma, entonces la complejidad temporal no se transforma en un factor prohibitivo (eficiencia).

H3. Las metaheurísticas son robustas a la hora de operar con ruido en los datos (robustez[2]).

Con el propósito de confirmar estas hipótesis, se establecen los siguientes objetivos: aplicar técnicas metaheurísticas al problema de ensamblado de fragmentos de un genoma, analizar los resultados para comprender el comportamiento de estos algoritmos y proponer nuevos métodos para resolver los problemas de manera más eficaz y eficiente. Para llevar a cabo los objetivos planteados se seguieron las fases que establece el método científico [13, 16] según F. Bacon. El método científico se sustenta en dos pilares fundamentales: la reproducibilidad y la falsabilidad, que establece que toda proposición científica tiene que ser susceptible de ser falsada.

En relación a la reproducibilidad, en el documento de la tesis se presentan los detalles necesarios para que cada experimento pueda ser reproducido por cualquier otro investigador interesado en cualquiera de las aplicaciones propuestas en este trabajo. En cuanto a la falsabilidad, en todos los estudios se ofrecen los resultados obtenidos de forma clara, estructurada y sencilla como prueba de las inferencias realizadas. Debido a la naturaleza estocástica de los algoritmos a utilizar, se han realizado un mínimo de 30 experimentos independientes. Además, para asegurar la relevancia estadística de las conclusiones, se aplica un conjunto de análisis estadísticos a los datos en todos los estudios realizados. Los resultados obtenidos han sido publicados en revistas indexadas así como también en congresos nacionales e internacionales (ver apartado 5).

El resto de este artículo se organiza de la siguiente manera. En la próxima sección se describe el problema de ensamblado de fragmentos de ADN (FAP, por sus siglas en Inglés). En la sección 3 se describen y analizan las contribuciones realizadas en esta tesis. En el apartado 4 se presentan las conclusiones y los trabajos futuros.

2. Problema de ensamblado de fragmentos de ADN (FAP)

Antes de explicar detalladamente el problema de ensamblado de fragmentos es necesario describir el proceso de secuenciación. Uno de los métodos más usados es el *Shotgun Sequencing* de Sanger [35] y consiste en:

1. Generar múltiples copias de la secuencia de ADN original y dividir cada una de ellas en miles de fragmentos aleatorios (ver pasos 1 y 2 de la figura 1).

2. Estos fragmentos son leídos por una máquina de secuenciación de ADN. En esta fase es donde se realiza la secuenciación propiamente dicha, que identifica cada una de las bases nucleótidas presentes en el fragmento (ver pasos 3 y 4 de la figura 1).

3. Un ensamblador une los fragmentos leídos que se superponen, reconstruyendo la secuencia original (ver pasos 5 y 6 de la figura 1).

El ensamblado de fragmentos de ADN es un problema resuelto en las primeras fases del proyecto del genoma y por lo tanto muy importante, ya que los demás pasos dependen de su precisión. El proceso de ensamblado de fragmentos (ver figura 2) consiste en: una primera fase de superposición, una segunda de distribución y una tercera de consenso.

[2]Para estudiar la robustez de un ensamblador se analizan las diferencias entre las soluciones encontradas para las instancias sin y con ruido. Si no se detectan diferencias (estadísticamente significativas), el ensamblador muestra un comportamiento neutro (insensible, indistinto) a pequeñas variaciones en los datos de entrada. Consecuentemente, este ensamblador se considera robusto para resolver instancias ruidosas.

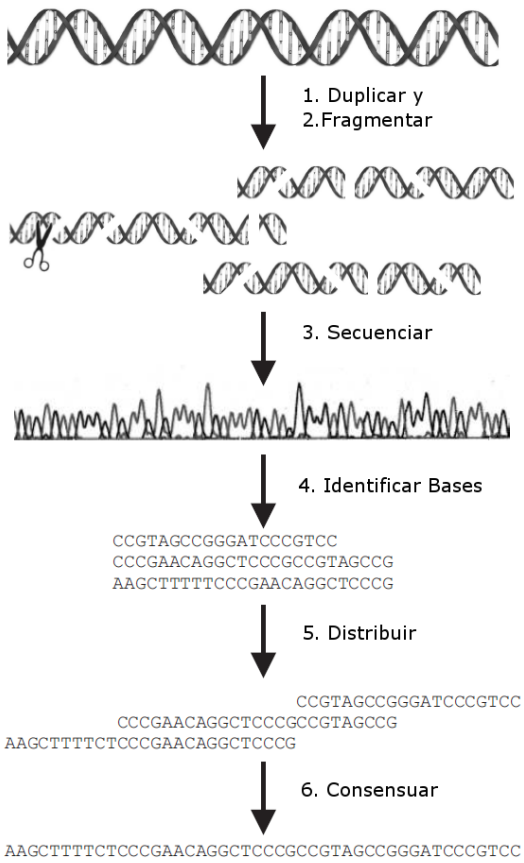

Figura 1: Representación gráfica de secuenciamiento y ensamblado de ADN.

Durante la *fase de superposición* se encuentra y determina el tamaño del solapamiento (o superposición) entre los fragmentos. Esta fase consiste en encontrar la mejor correspondencia o la más larga entre el sufijo de una secuencia y el prefijo de otra. En este paso, se comparan todos los posibles pares de fragmentos para determinar su similitud. Por lo general, un algoritmo de programación dinámica aplicada a la alineación semiglobal se utiliza en este paso. Es muy probable que los fragmentos con un alto grado de solapamiento estén uno seguido de otro en la secuencia destino.

El número de bases que se superponen entre dos fragmentos alineados, se llama *puntaje de solapamiento*. Con el fin de obtener el mencionado puntaje, cada posible orientación de los dos fragmentos es evaluada y luego se eligen: la superposición, la orientación y la alineación que maximice el número de bases coincidentes entre ambos fragmentos. Si no existe coincidencia entre dos fragmentos y ambos aparecen contiguos en la etapa del consenso entonces habrá un vacío en la secuencia final.

En la *fase de distribución* se encuentra el orden de los fragmentos basado en el puntaje de solapamiento computado en la fase anterior. Además de la complejidad que implica resolver el problema de optimización combinatoria de ordenamiento, este paso resulta el más complicado ya que es difícil conocer el solapamiento real dados los siguientes inconvenientes:

1. *Orientación desconocida.* Después de cortar a la secuencia original en muchos fragmentos, se pierde la orientación. No se sabe qué cadena debe ser seleccionada. Si un fragmento no tiene ningún tipo de solapamiento con otro, todavía es posible que su complemento sí presente esa coincidencia.

2. *Errores en la identificación de bases.* Hay tres tipos de errores en la identificación de bases: inserción, sustitución y eliminación. Se producen debido a errores experimentales durante la

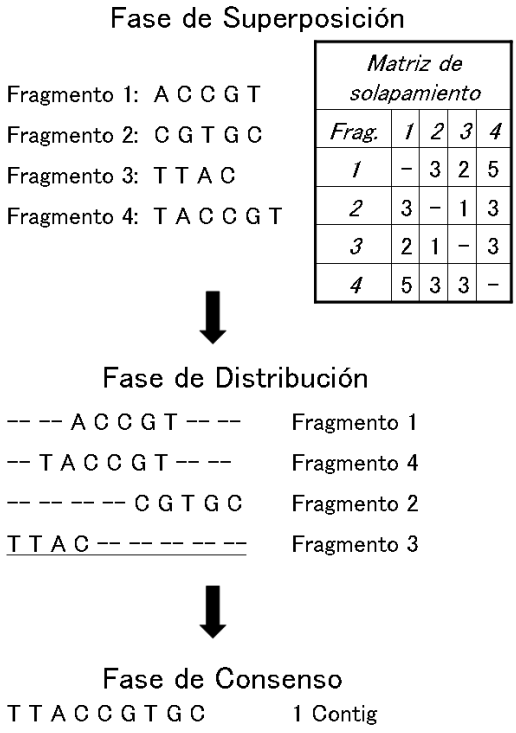

Figura 2: Fases en el ensamblado de fragmentos.

electroforesis (el método utilizado en los laboratorios para leer las secuencias de ADN). Los errores afectan la detección de solapamientos entre fragmentos. Por lo tanto, la determinación del consenso requiere de múltiples alineaciones en las regiones de alta.

3. *Cobertura incompleta.* Se produce cuando el algoritmo no es capaz de ensamblar un determinado conjunto de fragmentos en un solo *contig*. Un *contig* es una secuencia en la que la solapamiento entre los fragmentos adyacentes es mayor o igual a un umbral predefinido (parámetro de corte denominado $cutoff$).

4. *Regiones repetidas o Repeats.* Son secuencias que aparecen dos o más veces en la secuencia de ADN. Las regiones repetidas han causado problemas en muchos proyectos de secuenciación de genomas, y se torna dificultosa para los programas actuales de ensamblado poder manejarlos perfectamente.

5. *Quimeras y contaminación.* Las quimeras surgen cuando dos fragmentos que no son adyacentes o superpuestos en la molécula se unen en un solo fragmento. La contaminación se produce debido a la depuración incompleta del fragmento desde el vector de ADN.

En tanto que, en la *fase de consenso* se deduce la secuencia de ADN usando el ordenamiento de fragmentos obtenidos anteriormente. La técnica más comúnmente utilizada en esta fase es aplicar la regla de la mayoría en la construcción del consenso. Para medir la calidad de un consenso, podemos ver la distribución de la cobertura. La cobertura en una posición de base se define como el número de fragmentos en esa posición. Se trata de una medida de la redundancia de los datos del fragmento; que denota el número promedio de fragmentos que se espera aparezca un nucleótido en la secuencia de ADN. Se calcula como el número de bases leídas en los fragmentos sobre la longitud de la cadena

de ADN [37]:

$$Cobertura = \frac{\sum_{i=1}^{n} longitud\ \ del\ \ fragmento\ \ i}{longitud\ \ de\ \ la\ \ secuencia} \quad (1)$$

donde n es la número de fragmentos. Cuanto mayor sea la cobertura, menor es el número de espacios en blanco y mejor es el resultado.

Resumiendo, el conjunto de fragmentos de ADN en una secuencia de consenso que corresponde a la secuencia padre constituye el "problema de ensamblado de fragmentos" [37]. Es un problema de permutación NP-duro [32], por lo tanto, no existe (asumiendo P $\neq$ NP) un algoritmo exacto que resuelva este problema en tiempo polinómico. Desde el punto de vista de la optimización combinatoria, la construcción de un consenso es similar a la de un recorrido en un problema del viajante de comercio (*Travelling Salesman Problem*, TSP). Esto es porque cada fragmento tiene una ubicación específica en la formación de una secuencia en la etapa de consenso. Aunque los puntos terminales de un recorrido de TSP sean irrelevantes ya que su solución es un recorrido circular de ciudades, en el caso de FAP estos puntos son importantes ya que ellas representan los extremos opuestos de la secuencia original de ADN. En TSP el ordenamiento de las ciudades es la solución final al problema. En cambio para FAP, el ordenamiento de fragmentos es sólo un resultado intermedio que será utilizado en la fase de consenso.

3. Contribuciones de esta tesis

Previo a las investigaciones desarrolladas en este trabajo, se ha realizado una amplia exploración del estado del arte en algoritmos ensambladores, específicamente en aquellos basados en metaheurísticas. Tras este estudio inicial, se desprende que los GAs son una de las metaheurísticas más utilizadas para resolver este problema. También las metaheurísticas basadas en trayectoria, tales como SA, PALS y VNS, forman parte de la literatura asociada a FAP. A partir de esto, surgen las diversas contribuciones de esta tesis, a saber: algoritmos heurísticos para la *generación de semillas*, creación de *instancias de mayor complejidad*, diseño, desarrollo y evaluación de *nuevos ensambladores metaheurísticos, basados en trayectoria y en población, centralizados y distribuidos*. Estos aportes son descriptos en los siguientes apartados.

3.1. Generación de semillas.

Se ha desarrollado una nueva estrategia voraz para generar soluciones iniciales, que incorpora información heurística sobre el problema en soluciones representadas por una permutación. Se comprobó que introducir *semillas* en los primeros pasos del método permite, desde el inicio, guiar a la búsqueda hacia regiones prometedoras y, así, encontrar soluciones de calidad visiblemente superior a la obtenida cuando el inicio es aleatorio. Dado que en promedio el porcentaje de esta mejora es del 19 % y el costo de su aplicación es insignificante se aconseja siempre utilizar semillas generadas mediante dicha técnica voraz como estrategia de inicio [22, 24, 25].

3.2. Generación de instancias de mayor complejidad.

Con el objetivo de realizar un estudio del comportamiento de los ensambladores propuestos que permita concluir sobre el problema en general, resulta necesario analizar un número superior de instancias y también de mayor complejidad que las proporcionadas por la literatura. Por lo tanto, se ha implementado DNAgen un generador de instancias con el cual se obtuvieron un nuevo conjunto de instancias de alta complejidad [25]. Con la utilización de DNAgen se crea un nuevo grupo de instancias, denominadas *acin*. Estas instancias se caracterizan por la alta complejidad dada por el

tamaño de las mismas, es decir que están conformadas por entre 307 y 1049 fragmentos con una longitud promedio de 1000 bases cada uno.

Pero este estudio es realmente significativo cuando se abordan problemas reales originados por el ruido en los datos. En este trabajo se han identificado tres fuentes de ruido (el proceso de secuenciación, la fase de solapamiento y el cálculo de fitness), en función a esto se han generado nuevos grupos de instancias acorde a dichas fuentes [21, 23, 8].

La incorporación de todas estas nuevas instancias (sin y con ruido) a la experimentación ha permitido concluir de manera general sobre el problema con respecto a la precisión, eficiencia y robustez de los algoritmos propuestos en esta tesis.

3.3. Diseño, desarrollo y evaluación de nuevos ensambladores metaheurísticos basados en trayectoria.

ISA y SAX son dos nuevos algoritmos basados en SA [25, 21], el primero utiliza un procedimiento de inversión para generar vecinos y el segundo se hibrida con el operador de cruce genético *Order Crossover*. FVNS y CVNS [1] son dos ensambladores basados en VNS, el primero maximiza el puntaje de solapamiento y el segundo minimiza el número de contigs. Todos ellos son ensambladores centralizados pero, la complejidad de las grandes instancias con ruido conlleva a la utilización de métodos distribuidos y paralelos, así surge PH-PALS. Este es un nuevo ensamblador distribuido y paralelo basado en PALS que se hibrida con ISA [23, 8]. El comportamiento de cada uno de estos algoritmos es contrastado con PALS, una metaheurística especialmente diseñada para resolver FAP.

A partir de la aplicación de todos ellos a FAP se recomienda: aplicar ISA y SAX para resolver las instancias sin ruido, ya que son los ensambladores que resuelven con mayor precisión y eficiencia estas instancias al encontrar siempre la solución óptima en menos de 60 segundos. También se sugiere el uso de PH-PALS para resolver instancias ruidosas (cualquiera sea la fuente de ruido), dado que encuentra soluciones que reducen en más del 50 % el número de contigs con respecto a las halladas por el resto de los ensambladores propuestos. Además, logra una reducción importante del costo temporal al utilizar dos niveles de paralelismo.

Por otra parte, se detecta que PALS requiere menos tiempo de ejecución que ISA y SAX. Pero, dado que la calidad de los resultados obtenidos por éste es inferior a la lograda por los dos ensambladores basados en SA, se infiere que PALS converge rápidamente a óptimos locales en las instancias de mayor complejidad (sin y con ruido).

Con respecto a las hipótesis planteadas al inicio de esta tesis, ISA, PALS, SAX y PH-PALS confirman las hipótesis **H1** (estado del arte) y **H2** (complejidad y eficiencia), mientras que sólo PALS y PH-PALS confirman la **H3** (robustez).

3.4. Diseño, desarrollo y evaluación de nuevos ensambladores metaheurísticos basados en población.

Se han propuesto nuevos ensambladores basados en GAs que utilizan distintas estrategias de inicio (aleatoria, 2-opt y voraz), además de aplicar diferentes operadores de cruce (*Partial Mapped Crossover*, *Order Crossover*, *Cycle Crossover* y *Edge Recombination*). De esta forma surgen los siguientes algoritmos: $GA2o_{50}$, $GA2o_{100}$, GAG_{50} y GAG_{100} [22]. También se propone GA+VNS, que nace de hibridar a GAG_{50} con FVNS como un tercer operador genético [24]. De la aplicación de todos ellos para resolver FAP surge que: el ensamblador, basado en población, que logra un mejor compromiso entre calidad y costo computacional es GAG_{50} cuando utiliza OX. Pero, cuando se compara el comportamiento de GAG_{50} con el de los ensambladores basados en trayectoria, se detecta que estos últimos superan a la calidad obtenida por GAG_{50} en más del 7 % en las instancias de mediana y alta complejidad (sin y con ruido). En cuanto al costo temporal, sólo ISA y PH-PALS se diferencian de GAG_{50} requiriendo tiempos de ejecución significativamente menores al empleado por este GA. Finalmente, GAG_{50} confirma las 3 hipótesis planteadas.

4. Conclusiones y trabajos futuros

En esta tesis doctoral se ha propuesto la aplicación de técnicas metaheurísticas para solucionar el problema de ensamblado de fragmentos de ADN. Particularmente, el trabajo se enfoca en resolver el problema combinatorio NP-duro que surge al llevarse a cabo la fase de distribución. El objetivo es entonces encontrar el ordenamiento de fragmentos que permita obtener la secuencia original de ADN.

Dado los resultados obtenidos en esta tesis, se recomienda resolver FAP usando ensambladores metaheurísticos que utilicen estrategias de inicio heurísticas, como es la técnica voraz propuesta en esta tesis. De esta forma, el ensamblador mejora la calidad promedio de los resultados finales en un 19 % siendo insignificante el costo computacional resultante de aplicar esta estrategia.

Por otra parte, se sugiere la utilización de ensambladores metaheurísticos basados en trayectoria, dado que en general son más precisos y eficientes que los basados en población. En particular, para las instancias sin ruido se recomienda el uso de ISA y SAX por obtener siempre la distribución óptima de fragmentos en menos de 60 segundos. En tanto que para las instancias ruidosas, el ensamblador que brinda mayor precisión y eficiencia en los resultados es PH-PALS. Es decir, este ensamblador logra soluciones con al menos un 50 % más de calidad que el resto de los algoritmos. Además, se trata de un ensamblador robusto.

Diversas son las líneas de trabajo que surgen de las investigaciones desarrolladas en esta tesis doctoral. Una de ellas es diseñar una medida de calidad que, por un lado, considere el puntaje de solapamineto pero, por otro, contemple también el número de contigs. Otra es incorporar información heurística durante la búsqueda para tomar mejores decisiones sobre las regiones a explorar y explotar, y así se evitar óptimos locales y reducir el tiempo de ejecución. En cuanto a la manipulación de instancias ruidosas, en primer lugar se propone incrementar el *speedup* de PH-PALS; también es de interés diseñar una versión de PH-PALS heterogénea, que permita una exploración simultánea y controlada del espacio de búsqueda. Otra propuesta relacionada con la mejora en la calidad de las soluciones es el intercambio de información útil entre los individuos en una isla.

5. Publicaciones que sustentan la tesis doctoral

En este apartado se presenta el conjunto de trabajos publicados como resultado de las investigaciones desarrolladas a lo largo de esta tesis doctoral. Estas publicaciones avalan el interés, la validez y las contribuciones de dicha tesis en la literatura, dado que estos trabajos se han publicado en foros de prestigio y, por lo tanto, se han sometido a procesos de revisión por reconocidos investigadores especializados. A continuación se muestran las referencias de todas las publicaciones.

Revistas indexadas

- Gabriela Minetti, Enrique Alba y Gabriel Luque. Seeding strategies and recombination operators for solving the DNA fragment assembly problem. *Information Processing Letters*, Volume 108, Número 3, pág. 97-100, Octubre 2008.
- Gabriela Minetti, Guillermo Leguizamón y Enrique Alba. Assembling DNA Sequences Containing Noisy Information With Metaheuristic Algorithms. En evaluación *Journal of Information Sciences*, Elsevier, 2011.
- Gabriela Minetti, Guillermo Leguizamón y Enrique Alba. A new Parallel and Hybrid Metaheuristic for Solving Noisy DNA Strands. En evaluación *Journal of Information Sciences*, Elsevier, 2011.

Congresos Internacionales

- Gabriela Minetti y Enrique Alba. Metaheuristic Assemblers of DNA strands: Noiseless and Noisy Cases. *2010 IEEE Congress on Evolutionary Computation*. Barcelona, España. Julio 2010.
- Gabriela Minetti, Gabriel Luque, Guillermo Leguizamón y Enrique Alba. A new Hybrid SA for Solving the DNA Fragment Assembly Problem. *XXVIII Internacional Conference of the Chilean Computing Science Society (SCCC)*, pág. 109-116, Noviembre de 2009.
- Gabriela Minetti, Gabriel Luque y Enrique Alba. Variable Neighborhood Search as Genetic Algorithm Operator for DNA Fragment Assembling Problem. *Eighth International Conference on Hybrid Intelligent Systems*, IEEE Computer Society, pág. 714-719, 2008.

Congresos Nacionales

- Gabriela Minetti, Enrique Alba y Gabriel Luque. Variable Neighborhood Search for Solving the DNA Fragment Assembly Problem. *XIII Congreso Argentino de Ciencias de la Computación (CACIC 2007)*, pág. 1359-1371, Octubre de 2007.

Referencias

[1] E. Alba, G. Luque, and G. Minetti. Variable neighborhood search for solving the DNA fragment assembly problem. In *Anales del XIII Congreso Argentino de Ciencias de la Computación (CACIC)*, pages 1359 – 1370, Corrientes y Resistencia, Argentina, October 2007.

[2] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, (1990):403–410, 1990.

[3] S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, (25):3398–3402, 1997.

[4] T. Chen and S.S. Skiena. A case study in genome-level fragment assembly. *The Eighth Symposium on Combinatorial Pattern Matching*, pages 206–223, 1997.

[5] T. Chen and S.S. Skiena. A case study in genome-level fragment assembly. *Bioinformatics*, 16, 2000.

[6] K. Deb and A.R. Reddy. Reliable classification of two-class cancer data using evolutionary algorithms. *BioSystems*, 72:111–129, 2003.

[7] G. Fogel and D. Corne. *Evolutionary Computation in Bioinformatics*. Morgan Kaufmann, San Francisco, 2002.

[8] G. Minetti and G. Leguizamón and E. Alba. A new Parallel and Hybrid Metaheuristic for Solving Noisy DNA Strands. *Journal of Information Sciences, Elsevier (en evaluación)*, 2011.

[9] D. Gehlhaar, G. Verkhivker, P. Rejto, C. Sherman, D. B. Fogel, L. J. Fogel, , and S. Freer. Molecular recognition of the inhibitor ag-1343 by hiv-1 protease: conformationally flexible docking by evolutionary programming. *Chem. Biol.*, 2:317–324, 1995.

[10] P. Green. Phrap sequence assembly program. *University of Washington, Seattle*, 1996.

[11] W. Huang and A. Madan. CAP3: A DNA Sequence Assembly Program. *Genome Research*, 9(9):868–877, 1999.

[12] G. Jones, P. Willett, R.C. Glen, A.R. Leach, and R. Taylor. Development and validation of a genetic algorithm for flexible docking. *Journal of Molecular Biology*, (267):727–748, 1997.

[13] H.G. Gauch Jr. *Scientific Method in Practice*. Cambridge, 1st edition, 2002.

[14] O. Karpenko, J. Shi, and Y. Dai. Prediction of mhc class ii binders using the ant colony search strategy. *Artificial Intelligence in Medicine*, 35(1):147–156, 2005.

[15] K. Kim and C.K. Mohan. Parallel hierarchical adaptive genetic algorithm for fragment assembly. In IEEE, editor, *The 2003 Congress on Evolutionary Computation, 2003. CEC03.*, volume 1, pages 600–607. 2003.

[16] G. Klimovsky. *Las desventuras del conocimiento científico. Una introducción a la epistemología*. A-Z editora, Bs.As., 1997.

[17] L. Li and S. Khuri. A Comparison of DNA Fragment Assembly Algorithms. In *Proc. of the 2004 International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences*, pages 329–335, Las Vegas, 2004.

[18] G. Luque, E. Alba, and S. Khuri. Chapter 16: Assembling DNA Fragments with a Distributed Genetic Algorithm. In *Parallel Algorithms for Bioinformatics*. Wiley, 2006.

[19] H. Mamitsuka. Finding the biologically optimal alignment of multiple sequences. *Artificial Intelligence in Medicine*, 35(1-2):9–18, 2005.

[20] P. Meksangsouy and N. Chaiyaratana. DNA fragment assembly using an ant colony system algorithm. In *The 2003 Congress on Evolutionary Computation, 2003. CEC03*, volume 3, pages 1756– 1763. IEEE. ISBN: 0-7803-7804-0, 2003.

[21] G. Minetti and E. Alba. Metaheuristic Assemblers of DNA strands: Noiseless and Noisy Cases. In *2010 IEEE Congress on Evolutionary Computation*, Barcelona, España, July 2010.

[22] G. Minetti, E. Alba, and G. Luque. Seeding strategies and recombination operators for solving the DNA fragment assembly problem. *Information Processing Letters*, 108(3):94–100, October 2008.

[23] G. Minetti, G. Leguizamón, and E. Alba. Assembling DNA Sequences Containing Noisy Information With Metaheuristic Algorithms. *Journal of Information Sciences, Elsevier (en evaluación)*, 2011.

[24] G. Minetti, G. Luque, and E. Alba. Variable Neighborhood Search as Genetic Algorithm Operator for DNA Fragment Assembling Problem. In *Eighth International Conference on Hybrid Intelligent Systems*, pages 714 – 719, Barcelona, Spain, September 2008.

[25] G. Minetti, G. Luque, G. Leguizamón, and E. Alba. A new Hybrid SA for Solving the DNA Fragment Assembly Problem. In *XXVIII Internacional Conference of the Chilean Computing Science Society (SCCC)*, pages 109 – 116, Santiago, Chile, November 2009.

[26] E. W. Myers. A whole-genome assembly of drosophila. *Science*, 287:219–2204, 2000.

[27] C. Notredame and D.G. Higgins. SAGA: sequence alignment by genetic algorithm. *Nucleic Acids Research*, 24(8):1515–1524, 1996.

[28] C. Notredame, L. Holm, and D.G. Higgins. COFFEE: an objective function for multiple sequence alignments. *Bioinformatics*, 14(5):407–422, 1998.

[29] R. J. Parsons, S. Forrest, and C. Burks. Genetic algorithms, operators, and dna fragment assembly. In *Machine Learning*, pages 11–33. Kluwer Academic Publishers, 1995.

[30] W.R. Pearson. Comparison of methods for searching protein sequence databases. *Protein Sci.*, 4:1145–1160, 1995.

[31] W.R. Pearson and D.J Lipman. Improved tools for biological sequence analysis. *Proc. Natl Acad. Sci. USA,85*, pages 2444–2448, 1998.

[32] P. Pevzner. *Computational molecular biology: An algorithmic approach*. The MIT Press, 2000.

[33] J.C. Prasad, S.R. Comeau, S. Vajda, and C.J. Camacho. Consensus alignment for reliable framework prediction in homology modeling. *Bioinformatics*, pages 1–10, 2003.

[34] M.I. Sadowski, J.H. Parish, and D.R. Westhead. Automated derivation and refinement of sequence length patterns for protein sequences using evolutionary computation. *BioSystems*, (81):247–254, 2005.

[35] F. Sanger, A.R. Coulson, G.F. Hong, D.F. Hill, and G.B. Petersen. Nucleotide Sequence of Bacteriophage Lambda DNA. *Journal of Molecular Biology*, 162(4):729–773, 1982.

[36] S. Schulze-Kremer. Genetic algorithms for protein tertiary structure prediction. *Parallel Problem Solving from Nature II*, pages 391–400, 1992.

[37] J. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology*. International Thomson Publishing, 20 park plaza, Boston, MA02116, 1999.

[38] F. Solis and R. Wets. Minimization by random search techniques. *Math. Ops. Research*, 6:10–30, 1981.

[39] G. G. Sutton, O. White, M. D. Adams, and A. R. Kerlavage. TIGR Assembler: A new tool for assembling large shotgun sequencing projects. *Genome Science and Technology*, pages 9–19, 1995.

[40] J.D. Thompson, D.G. Higgins, and T.J. Gibson. CLUSTALW: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, 22:4673–4680, 1994.

[41] J. Yang and C. Kao. A family competition evolutionary algorithm for automated docking of flexible ligands to proteins. *IEEE Transactions on Information Technology in Biomedicene*, 4(3):225–237, 2000.