



Estrategias y herramientas de apoyo para facilitar la aplicabilidad de los algoritmos bio-inspirados y otras metaheurísticas

Mohammed Mahrach, Gara Miranda, Coromoto León

Dpto. Ingeniería Informática y de Sistemas

Universidad de La Laguna

Avenida Astrofísico Francisco Sánchez, s/n.

Apto. 456, 38200, San Cristóbal de La Laguna,

Santa Cruz de Tenerife, España

mmahrach@ull.edu.es, gmiranda@ull.edu.es, cleon@ull.edu.es

Resumen—Actualmente existen diversas plataformas y entornos software para la optimización, con algoritmos evolutivos y otras metaheurísticas bio-inspiradas. Sin embargo, lograr una clara separación entre el problema y las implementaciones de los resolutores sigue siendo un desafío, debido a muchas dificultades tanto de representación del espacio de soluciones, como de búsqueda dentro de dicho espacio. En la mayoría de las estrategias de exploración, tanto la representación de la solución como la forma en que se pasa de una solución a otra, depende en gran medida del propio algoritmo. De esta forma, el uso de un algoritmo o estrategia concreta, indirectamente implica que el usuario debe proporcionar una implementación diferente y adecuada para cada caso, así como diferentes tipos de operadores de variación (por ejemplo, los operadores de cruce y mutación en los algoritmos evolutivos). Esto puede resultar una tarea bastante compleja y requiere de mucho tiempo y experiencia, pues no sólo hay que conocer las características del problema a resolver sino también los detalles de la técnica algorítmica a aplicar. Esto dificulta especialmente la aplicación de estas técnicas por parte de usuarios no expertos en este ámbito de investigación. Es por ello, que el principal objetivo de este proyecto es aumentar la aplicabilidad de los algoritmos “bio-inspirados” y otras metaheurísticas, para la resolución de problemas de optimización “multi-objetivo” en los que existen múltiples objetivos a optimizar simultáneamente. Para ello se pretende diseñar y desarrollar una interfaz con algunas de las metaheurísticas más extendidas, de forma que puedan ser utilizadas en entornos empresariales o por parte de usuarios no expertos.

Index Terms—Metaheurísticas, Algoritmos bio-inspirados, Optimización combinatoria, Optimización multi-objetivo, Ingeniería del software.

I. INTRODUCCIÓN

La resolución de problemas multi-objetivo mediante metaheurísticas es uno de los temas más investigados en el ámbito de las Ciencias de la Computación. Una de las principales dificultades a la hora de implementar metaheurísticas es cómo definir una representación general para las soluciones de los problemas, de modo que el espacio de búsqueda para el problema correspondiente pueda estar

completamente representado. Otra dificultad, una vez se ha definido el espacio de búsqueda que representa todas las posibles soluciones, es definir un mecanismo para explorar ese espacio. Este mecanismo de exploración debería definir los movimientos o criterios para seleccionar las próximas soluciones a tener en cuenta. Para superar estas dificultades, la misión de este trabajo es poder implementar optimizadores a un nivel de descripción abstracto no específico del problema, de modo que haya una separación entre el problema y su método de solución. De esta forma, se podría evitar la tarea de codificación de soluciones personalizadas a cada problema pues resulta bastante difícil y requiere de mucho tiempo y esfuerzo, especialmente para usuarios no expertos.

Este trabajo está organizado de la siguiente manera. Los antecedentes y estado actual del tema (tanto en lo que respecta a metaheurísticas y estrategias de optimización, como en relación a herramientas de Ingeniería del Software para el desarrollo de herramientas e interfaces de usuario) se identifican en la sección I. En la sección II se presentan los objetivos planteados y el interés de los mismos. Por su parte, en la sección III se incluye una breve descripción de la metodología a seguir y el plan de trabajo propuesto. Finalmente, en la sección IV se plantea la relevancia social y teórica de la investigación y los motivos por los cuales se lleva a cabo el estudio.

II. ANTECEDENTES

La optimización combinatoria es una rama de la optimización en Matemáticas Aplicadas y en Ciencias de la Computación, relacionada con la Investigación Operativa, la Teoría de Algoritmos y la teoría de la Complejidad Computacional. El propósito de la Optimización Combinatoria es encontrar objetos matemáticos discretos que maximicen (o minimicen) una determinada función objetivo, de tal forma que las variables de dicha función representan las decisiones que se pueden tomar y que modifican el valor de esa función y estén sujetas a una serie de restricciones que algunas de las

variables deben satisfacer [1]. Generalmente, estos objetos se denominan estados y al conjunto de todos los posibles estados candidatos se le denomina espacio de búsqueda. La naturaleza de los estados y del espacio de búsqueda viene determinada por el tipo de problema y por la formulación que se haga del mismo.

Aquellos problemas que se centran en la optimización de una única función objetivo reciben el nombre de problemas mono-objetivo. Sin embargo, en multitud de problemas reales en ingeniería es necesario optimizar simultáneamente un conjunto de funciones objetivo. A este tipo de problemas se les denomina problemas multi-objetivo. En la optimización de problemas multi-objetivo, se podrían ponderar los distintos objetivos para obtener una única función y, por tanto, poder aplicar los métodos clásicos de resolución. Sin embargo, existe un conjunto de algoritmos especialmente diseñados para afrontar la resolución de los problemas multi-objetivo como tales. Estos algoritmos tienen en cuenta que es posible que no exista una única solución optimizando todos y cada uno de los objetivos. En tal caso, lo que se obtiene como resultado es un frente de Pareto de soluciones [2] o conjunto de soluciones no inferiores o no dominadas. De entre este conjunto de soluciones, un experto podrá elegir la que más convenga en cada caso particular y que mejor se adecua a las necesidades de la solución.

Los algoritmos de optimización combinatoria [3] se relacionan comúnmente con los problemas NP-completos. Para este tipo de problemas no existe ningún método que garantice la solución óptima en un tiempo polinomial ($NP = \text{Nondeterministic Polynomial}$). Se puede decir, entonces, que los algoritmos de optimización combinatoria resuelven instancias de problemas difíciles en general, explorando el espacio de soluciones para estas instancias. Para ello, dichos algoritmos deben reducir el tamaño efectivo del espacio de soluciones, realizando así una exploración del espacio de búsqueda mucho más eficiente. De esta forma, se contemplan dos tipos de algoritmos o métodos: los métodos exactos u óptimos, que encuentran la solución óptima pero que sólo son aplicables cuando el tamaño del problema es pequeño, y los métodos aproximados o heurísticos que, aunque no garantizan el óptimo, encuentran buenas soluciones en un tiempo razonable. Dentro de los métodos exactos se encuentran las búsquedas enumerativas o de fuerza bruta y los métodos de ramificación y acotación (*Branch and Bound*). Por otro lado, algunos de los métodos aproximados más conocidos se enumeran a continuación: búsquedas heurísticas, búsquedas locales, recocido simulado, búsqueda tabú, algoritmos genéticos, algoritmos evolutivos, colonias de hormigas, etc.

En general, el método a utilizar depende de la naturaleza del problema en particular, pero también de las características esperadas de la solución. En la mayoría de las aplicaciones reales que aparecen principalmente en entornos industriales dinámicos y competitivos, la diferencia entre usar un método sencillo e inmediato, o utilizar estrategias más sofisticadas para obtener soluciones de mayor calidad, puede ser determinante

para la supervivencia de una empresa. Normalmente, los usuarios están familiarizados con el problema que van a abordar, pero no tienen por qué conocer las diferentes técnicas de resolución existentes, ni las estrategias de paralelización más convenientes. Por lo tanto, suele ser conveniente utilizar herramientas que faciliten la labor de los usuarios en la resolución del problema. Estas herramientas (esqueletos, patrones o librerías) suelen dar soporte para la técnica algorítmica a utilizar, evitando que los usuarios tengan que implementar una solución desde el comienzo.

De entre este tipo de herramientas, los esqueletos algorítmicos se presentan como un conjunto de procedimientos que constituyen el armazón con el que desarrollar programas para resolver un problema dado utilizando una técnica particular. Este software presenta declaraciones de clases vacías que el usuario ha de rellenar para adaptarlo a la resolución de un problema concreto. Los esqueletos proporcionan modularidad al diseño de algoritmos, lo que supone una gran ventaja respecto a la implementación directa del algoritmo desde el principio, no sólo en términos de reutilización de código sino también en cuanto a metodología y claridad de conceptos. El problema de estas herramientas es que la definición del problema se adapta al tipo de técnica a aplicar, con lo cual, si el usuario desea resolver el problema mediante la aplicación de técnicas diferentes, tendrá que definir el problema de formas distintas o bien implementar funciones adaptadas a cada una de las estrategias a utilizar.

Por otro lado, el diseño de software modular y reusable orientado a objetos es muy complejo. Existen técnicas para el diseño de patrones que no son nada sencillas y que no siempre obtienen la reusabilidad de los objetos esperada. El diseño de patrones está, en este sentido, muy relacionado con los esqueletos algorítmicos. Ambos pretenden obtener código modular, portable y, por supuesto, lo más reutilizable posible. Sin embargo, los patrones facilitan descripciones de objetos y clases que se adecúan para resolver problemas generales de diseño en un contexto en particular, mientras que los esqueletos algorítmicos proporcionan un mecanismo para resolver problemas aplicando un determinado tipo de técnica algorítmica. Para la implementación de este tipo de software puede utilizarse cualquier lenguaje de programación, aunque en general, los orientados a objetos son los más empleados. En la fase de diseño es preciso partir de un metamodelo, precisamente un metamodelo para problemas de optimización [4], para capturar, en términos abstractos, las características esenciales de un problema de optimización, el metamodelo puede servir como forma de almacenamiento para problemas de optimización, a partir de los cuales se pueden generar problemas de optimización reales, ejecutables.

Teniendo en cuenta todo lo anterior, en la actualidad existen diversas herramientas o marcos de trabajos conocidas como “frameworks” que ofrecen este tipo de esqueletos o librerías para optimización de problemas en base a técnicas preestablecidas. Algunos de los más conocidos son EasyLocal [5], ParadisEO [6] y ParadisEO-MOEO [7], MaLLBa [8], jMetal [9], GAMES [10], OR-Tools [11],



etc. Algunos se centran en optimización mono-objetivo y otros en optimización multi-objetivo, incluso algunas permiten hibridación de metaheurística de bajo nivel [12]. También se diferencian en el tipo de técnicas que proporcionan y en la usabilidad o dificultad de uso que presentan de cara a los usuarios finales no expertos.

Una herramienta que podría ser un punto de partida para poder abordar los objetivos plantados de este trabajo es METCO (*Metaheuristics-based Extensible Tool Cooperative Optimization*) [13]. METCO es un marco de trabajo paralelo para la resolución de problemas de optimización multi-objetivo que proporciona implementaciones de algunos de los algoritmos evolutivos y otras metaheurísticas más conocidas en el ámbito de la optimización multi-objetivo: NSGAI [14], SPEA2 [15], etc. La interfaz de METCO permite al usuario establecer características específicas del problema que necesita solucionar (representación de la solución, objetivos, operadores genéticos, etc.), pudiendo especificar una amplia gama de opciones de configuración para adaptar el comportamiento del software a varios modelos diferentes.

El reto de este trabajo sería lograr aún más separación entre el problema y las implementaciones de los resolutores. Sin embargo, las dificultades están presentes tanto en la representación del espacio de búsqueda, como en la implementación de las búsquedas dentro del mismo espacio. En la mayoría de las estrategias de búsqueda, tanto la representación del espacio de búsqueda como la forma de exploración, dependen de la implementación del propio algoritmo. Esto es, el uso de un algoritmo o estrategia de búsqueda diferente, implican una codificación de soluciones diferente y adecuada para cada caso, además de un conjunto de operadores de búsqueda adaptados para cada caso, lo cual dificulta la interacción entre el usuario y la interfaz planteada para este trabajo.

Debido a lo expuesto en el párrafo anterior, Carlos M. Fonseca [16] ha propuesto un modelo conceptual para la implementación de un software para optimización que resuelve la problemática de la separación entre el problema y los métodos de búsqueda, de tal forma que los resolutores puedan escribirse independientemente de los problemas a los que se aplicarán y que, por su parte, los problemas se puedan formular sin pensar en los operadores de búsqueda que serán empleados por los resolutores.

El diagrama de clase en la Figura 1 representa el metamodelo de la propuesta, donde podemos apreciar que la separación entre el problema y el resolutor se logra al reconocer que las estructuras del vecindario son una parte integral de la definición de problema, teniendo en cuenta que los optimizadores deben formularse en un nivel abstracto, independientemente de las estructuras vecinales específicas y, por lo tanto, de sus detalles de implementación. Este modelo ofrece una representación de la solución basada en la estructura del espacio de búsqueda y la evaluación de la solución en este espacio.

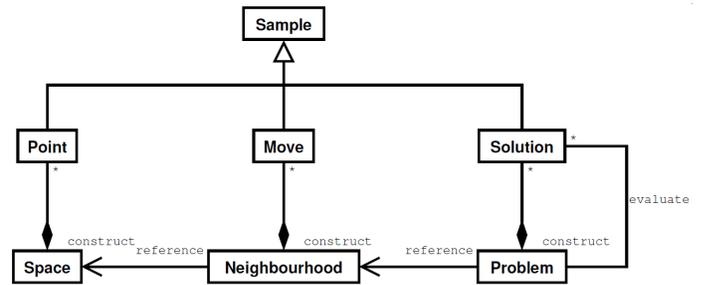


Figura 1. Diagrama de clase del Metamodelo de Carlos M. Fonseca [16]

III. OBJETIVOS CONCRETOS E INTERÉS DE LOS MISMOS

En general, la calidad de las soluciones obtenidas al resolver un problema dependerá de la técnica algorítmica aplicada, de la eficiencia de la implementación de la técnica en cuestión, de su parametrización y, por supuesto, del conocimiento específico del problema que el usuario introduzca en el algoritmo. El proporcionar implementaciones eficientes y flexibles de técnicas algorítmicas, permite a los usuarios centrarse únicamente en el propio problema, evitándoles tener que introducirse en áreas distintas a las de su dominio. Es por lo que en este trabajo nos queremos centrar principalmente en los objetivos siguientes:

- Analizar diferentes estrategias de optimización y tipologías de problemas para definir una representación del espacio de búsqueda de forma que las operaciones de modificación de soluciones no dependan directamente de la propia estrategia a utilizar, sino que tengan una definición mucho más general y extensible a diferentes técnicas.
- Diseñar y desarrollar una herramienta que permita a los usuarios entender el esquema básico de funcionamiento de algunas de las metaheurísticas y algoritmos bio-inspirados más conocidos. Además de facilitar la comprensión de los esquemas, la herramienta también permitirá aplicar dichos esquemas a la resolución de problemas concretos y reales.
- Evitar obtener una herramienta que profundice demasiado en aspectos de implementación o en cuestiones muy particulares de cada uno de los algoritmos y que se centre en la especificación del problema a resolver.
- Incluir en la herramienta funcionalidades que permitan la especificación y configuración de algoritmos a través de una interfaz visual, intuitiva y de fácil comprensión.
- Integrar en la herramienta funcionalidades que permitan, no sólo el proceso de resolución del problema, sino también la configuración y validación de los algoritmos y de los resultados obtenidos

La hipótesis que se plantea se centra fundamentalmente en que los métodos de optimización bio-inspirados ofrecen grandes posibilidades para la resolución de problemas en el ámbito empresarial, pero debido a su complejidad intrínseca

y también a sus particularidades a la hora de configurarlos y adaptarlos a cada uno de los problemas, no se consigue extender su uso fuera del ámbito de la investigación convencional. Es por ello que se plantea la necesidad de diseñar e implementar nuevas herramientas que ayuden a mejorar la aplicabilidad de estas técnicas de optimización.

IV. METODOLOGÍA Y PLAN DE TRABAJO

La metodología a seguir viene dada por el método científico. En primer lugar, se llevará a cabo un exhaustivo análisis de las referencias bibliográficas relacionadas, clasificando las investigaciones para ordenar el estado del arte. Para ello se realizarán observaciones, que consisten en analizar la abundante bibliografía existente en el contexto en el que nos hemos situado. A continuación, se establecerán criterios de organización de la información seleccionada, clasificándola según su impacto, profundidad de los estudios y área de aplicación. Posteriormente, se procederá a analizar la forma actual de trabajo para detectar todas las debilidades y carencias existentes y cómo podrían ser evitadas. Una vez hecho esto, se estudiarán las distintas alternativas que puedan ayudarnos a superar estas debilidades.

Finalmente, vendrá una fase de experimentación en la que se implementarán las propuestas realizadas y se comprobará su funcionamiento tanto con bancos de pruebas como con problemas reales. Es deseable que para la fase de experimentación se utilicen algunos de los problemas más referenciados en la literatura de forma que otros investigadores puedan comprender fácilmente las aportaciones realizadas. Para demostrar o refutar la hipótesis planteada, se han de analizar los resultados obtenidos para los distintos problemas con técnicas matemáticas y estadísticas. Por último, se presentarán las conclusiones, indicando en qué condiciones las hipótesis realizadas son válidas y de utilidad.

Uno de los pilares en los que se sustenta el método científico es la reproducibilidad, es decir, la capacidad de repetir un determinado experimento. Este pilar se basa, esencialmente, en la descripción correcta y detallada de todo el entorno de experimentación y de los resultados obtenidos. Al trabajar con software informático, la reproducibilidad se ve facilitada enormemente si el mismo está disponible. Por ello, los desarrollos realizados durante esta investigación serán liberados a la comunidad científica bajo licencia GNU-GPL.

Teniendo en cuenta la metodología anterior, el plan de trabajo será el siguiente:

Tarea 1. Revisión bibliográfica: El objetivo de esta Tarea es abordar el estudio de los métodos, algoritmos y problemas previamente propuestos en la literatura, así como las metodologías de programación y los criterios para medir los rendimientos y calidad de las soluciones encontradas. En esta Tarea se realizará una puesta en contexto y una revisión bibliográfica. También se realizarán búsquedas en bases de datos de información relacionada. Adicionalmente, se programará la asistencia a cursos y seminarios de formación en el área.

Tarea 2. Identificación de los recursos computacionales: Las actividades de este proyecto requieren la utilización de recursos hardware y software especiales. Por lo tanto, será necesario la identificación de plataformas de cómputo masivo, así como la realización de una formación específica sobre su uso. De esta forma, durante el proceso de validación de la herramienta se podrá hacer un uso intensivo de recursos computacionales, acelerando así todo el proceso de análisis de las herramientas diseñadas.

Tarea 3. Diseño y desarrollo de un software de optimización: El software que se desarrolle deberá incorporar funcionalidades que implementen algunas de las metaheurísticas y algoritmos bio-inspirados más utilizados por la comunidad científica. Además, se prestará especial atención al proceso de configuración y validación de los algoritmos, tratando de automatizar en la medida de lo posible todo el proceso de análisis estadístico de los resultados. Una vez finalizados los desarrollos, el software será liberado, mejorando así la reproducibilidad de los resultados y haciendo accesible este tipo de técnicas al entorno empresarial.

Tarea 4. Validación con bancos de problemas: Las funcionalidades implementadas en la herramienta serán inicialmente comprobadas con algunos problemas de referencia de la literatura. Además, se aprovechará para generar la documentación y los tutoriales de referencia que luego sirvan para difundir la herramienta.

Tarea 5. Aplicación a problemas reales: La herramienta se aplicará también a problemas reales, intentando optimizarlos todo lo posible. Se compararán los resultados con los obtenidos haciendo uso de otras herramientas o marcos de trabajos existentes en la comunidad científica. En este caso, se prestará especial atención a los problemas relacionados con el ámbito de la salud. En particular, se trabajará con el problema de la planificación de menús saludables y adaptados a cada persona.

Tarea 6. Documentación y difusión de los resultados: El software desarrollado incluirá la documentación técnica correspondiente. Para la difusión de los resultados se utilizarán los cauces habituales de publicación en congresos y revistas.

En la Tabla I se recoge la estructura del plan de tesis y la planificación del tiempo para la ejecución de las tareas de investigación enumeradas anteriormente.

V. RELEVANCIA DE LA INVESTIGACIÓN

Este proyecto de tesis se integra en un Proyecto de I+D+i del Programa Estatal de Investigación, Desarrollo e Innovación orientada a los Retos de la Sociedad, titulado “*Técnicas de Optimización Meta-heurísticas Aplicadas a la Salud*” (TIN2016-78410-R). El proyecto está enfocado a realizar avances con clara aplicación de interés social de la Informática y tiene como objetivo estudiar y proponer extensiones y mejoras de modelos de optimización mono y multi-objetivo y aplicarlas a estudios relacionados con la salud. Se compararán los resultados con los obtenidos haciendo uso de otras herramientas o marcos de trabajos existentes en la comunidad científica. En este caso, se prestará especial



Cuadro I
PLAN DE ACTIVIDADES

	1-3	4-6	7-9	10-12	13-15	16-18	19-21	22-24	25-27	28-30	31-33	34-36
Tarea 1												
Tarea 2												
Tarea 3												
Tarea 4												
Tarea 5												
Tarea 6												

Unidad de tiempo : meses

atención a los problemas relacionados con el ámbito de la salud. En particular, se trabajará con el problema de la planificación de menús saludables y adaptados a cada persona.

Además, esta investigación se enmarca en el área de la “Estrategia de Especialización Inteligente 2014-2020 (RIS3) de Canarias” [17], denominada “Agenda Digital”, en concreto, está totalmente englobada en la subárea “Crecimiento digital” pues el objetivo final de la Tesis es desarrollar una herramienta que permita que las empresas y los usuarios no expertos puedan acceder a servicios avanzados de la sociedad de la información. En concreto, se pretende hacer extensible a las empresas herramientas de optimización avanzada que permitirán gestionar de forma más eficiente los recursos disponibles en las empresas. Este es un objetivo estratégico directamente relacionado con el crecimiento digital pues, de esta forma, se podrá impulsar la competitividad y la internacionalización de nuestras empresas.

También está relacionado con la Acción COST “European Cooperation in Science and Technology”, CA15140 “Improving Applicability of Nature-Inspired Optimisation by Joining Theory and Practice” [18]. La existencia de esta acción se basa en que las estrategias de búsqueda y optimización inspiradas en la naturaleza son fáciles de implementar y de aplicar a nuevos problemas. Sin embargo, para lograr un buen rendimiento, es necesario ajustarlos al problema en cuestión y, para ello, generalmente es necesario tener un profundo conocimiento sobre el problema, pero también sobre la técnica en cuestión. Los fundamentos teóricos para la comprensión de tales enfoques se han construido con gran éxito en los últimos veinte años, pero existe una gran desconexión entre la base teórica y las aplicaciones prácticas. El desarrollo de poderosas herramientas analíticas, la comprensión significativa de las limitaciones generales de los diferentes tipos de métodos de optimización inspirados en la naturaleza y el desarrollo de perspectivas más prácticas para el análisis teórico han traído avances impresionantes al campo teórico del campo. Sin embargo, hasta ahora el impacto en el lado de la aplicación ha sido limitado y pocas personas en las diversas áreas potenciales de aplicación se han beneficiado de estos avances. Es por ello que el objetivo principal de la acción COST es cerrar esta brecha y mejorar la aplicabilidad de todo tipo de métodos de optimización inspirados en la naturaleza. Su objetivo es hacer que los conocimientos teóricos sean más accesibles y prácticos mediante la creación de una plataforma donde los teóricos y los profesionales puedan

reunirse e intercambiar ideas y necesidades, desarrollando directrices sólidas y soporte práctico para el desarrollo de aplicaciones basadas en conocimientos teóricos, desarrollando marcos teóricos impulsados por las necesidades reales que surgen de las aplicaciones prácticas, formando a los jóvenes investigadores en este campo, ampliando la participación en la investigación en curso sobre cómo desarrollar y aplicar métodos de optimización robustos inspirados en la naturaleza en diferentes áreas de aplicación.

Teniendo en cuenta lo anterior, creemos que la necesidad de desarrollar el tipo de herramientas que se plantean en esta investigación está justificado y detectado por los expertos del ámbito y, por lo tanto, sería una importante aportación para conseguir una mejora en la aplicabilidad y universalidad de este tipo de técnicas, beneficiando, no sólo a la comunidad científica involucrada sino también a potenciales usuarios y empresas que podrían gestionar más eficientemente sus recursos y procesos de producción. El hecho de que la investigación esté relacionada con el ámbito de esta acción COST supone una excelente oportunidad para que el doctorando pueda acceder a una formación muy específica y avanzada en el tema, así como para iniciar contactos y posibles colaboraciones con investigadores de reconocido prestigio en el ámbito de la optimización bio-inspirada.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el proyecto TOMAS (TIN2016-78410-R) en el marco del Plan Nacional de I+D+i del Ministerio de Economía, Industria y Competitividad, Desarrollo e Innovación orientada a los Retos de la Sociedad.

El trabajo de Moahmmmed Mahrach ha sido financiado a través del Gobierno de Canarias mediante la Agencia Canaria de investigación, Innovación y sociedad de la información, por la subvención total de la tesis titulada “Mejorando La Aplicabilidad De Metaheurísticas y Algoritmos Bio-inspirados: Aplicaciones y herramientas” en el marco del Programa Pre-doctoral de Formación del Personal Investigador mediante la ayuda con identificador TESIS2018010095.

REFERENCIAS

- [1] Guangquan Zhang, Jie Lu, and Ya Gao. Multi-Level Decision Making. 82(4), 2015.
- [2] E Alba, B Dorronsoro, F Luna, A J Nebro C León, G Miranda, and C Segura. Metaheurísticas multiobjetivo para optimizar el proceso de difusión en MANETs metropolitanas. *Computer*, (November 2014).
- [3] Christian Blum and Andrea Roli. Metaheuristics in combinatorial optimization. *ACM Computing Surveys*, 35(3):268–308, sep 2003.
- [4] Ibrahim Assourocko and Peter Owen Denno. A metamodel for optimization problems. Technical report, 2016.
- [5] Luca Di Gaspero and Andrea Schaerf. EASYLOCAL++: an object-oriented framework for the flexible design of local-search algorithms. *Software: Practice and Experience*, 33(8):733–765, jul 2003.
- [6] S. Cahon, N. Melab, and E. G. Talbi. ParadisEO: A framework for the reusable design of parallel and distributed metaheuristics. *Journal of Heuristics*, 10(3):357–380, may 2004.
- [7] Arnaud Liefvooghe, Matthieu Basseur, Laetitia Jourdan, and El-Ghazali Talbi. ParadisEO-MOEO: A Framework for Evolutionary Multi-objective Optimization. In *Evolutionary Multi-Criterion Optimization*, pages 386–400. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
- [8] Enrique Alba, Gabriel Luque, J. García-Nieto, G. Ordóñez, and G. Leguizamón. MALLBA: a software library to design efficient optimisation algorithms. *International Journal of Innovative Computing and Applications*, 1(1):74–85, 2007.
- [9] Juan J. Durillo and Antonio J. Nebro. jMetal: A Java framework for multi-objective optimization. *Advances in Engineering Software*, 42(10):760–771, oct 2011.
- [10] Herman De Beukelaer, Guy F Davenport, Geert De Meyer, and Veerle Fack. JAMES: A modern object-oriented java framework for discrete optimization using local search metaheuristics. In *4th International Symposium and 26th National Conference on Operational Research*, number JUNE, pages 134–138, 2015.
- [11] Google. Google Optimization Tools, 2016.
- [12] Siti Musliha Mat-rasid, Mohamad Razali Abdullah, Hafizan Juahir, Universiti Sultan, Zainal Abidin, Ahmad Bisyrri, and Husin Musawi. Research Article Special Issue. (January), 2018.
- [13] Multi-objective Optimization. Metco: a parallel plugin-based framework for multi-objective optimization. 18(4):569–588, 2009.
- [14] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [15] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pages 95–100, 2001.
- [16] C C Vieira and C M Fonseca. A unified model of optimisation problems. page 1537, 2007.
- [17] Equipo de Gestión de la RIS3 de Canarias. Estrategia de especialización inteligente de canarias 2014-2020. <https://www3.gobiernodecanarias.org/aciisi/ris3/>, Junio 2018. (Accessed on 06/19/2018).
- [18] Cost — improving applicability of nature-inspired optimisation by joining theory and practice (imappnio). http://www.cost.eu/COST_Actions/ca/CA15140, Noviembre 2015. (Accessed on 06/19/2018).