UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

Centro de Investigación Operativa

Medición de la eficiencia y la productividad:

Aspectos computacionales

TESIS DOCTORAL.

Programa de Doctorado en Economía (DECiDE)

Presentada por:

Martín González Espinosa

Director: Juan Aparicio Baeza
Codirector: José Juan López Espín

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE
Programa de Doctorado en Economía (DECiDE)

Medición de la eficiencia y la productividad: Aspectos computacionales

Memoria presentada para optar al grado de
Doctor por la Universidad Miguel Hernández de Elche,
Realizada bajo la dirección de Juan Aparicio Baeza (director) y
José Juan López Espín (codirector).

Dedicado a mi madre,

mis padres, hermanos y familia.

Agradecimientos

Ha sido mucha gente la que he conocido a lo largo de estos años de investigación. Todo empezó como un proyecto a largo plazo, gracias a la confianza ciega de José Juan López Espín y Juan Aparicio Baeza, los cuales sin apenas conocerme, siempre han puesto la mano en el fuego por mí. He de decir que son dos de las personas que más me han valorado académicamente, y me han apoyado siempre a lo largo de todos estos años.

A nivel académico, también agradecer la compañía durante todos estos años a mi compañera Rocío Hernández, y a todos los que en mayor o menor medida me han ayudado a crecer como investigador, como Domingo Jiménez, Santi García, Antonio Peñalver, Mari Carmen Perea, María Belén Pérez, Javi Cuenca, y demás investigadores de la comunidad científica que han dejado su granito de arena en mi carrera.

También tengo que agradecer el apoyo y confianza de parte de toda mi familia, sobre todo mi madre María, mis padres Jose Luis y Miguel, mi hermana Marina, mi hermano Pablo, y mis primos Antonio y Carlos, los cuales siempre han estado a mi lado.

Gracias también al cariño, gran ayuda y confianza de todos mis amigos, en especial a Alvarito, Adri, Dowie, Roberto y todos los miembros de La Hermandad.

GRACIAS.

Estoy también agradecido, por la financiación económica proporcionada, al Ministerio de Economía, Industria y Competitividad y la Agencia Estatal de Investigación por los proyectos con código MTM2016-79765-P(AEI/FEDER,UE) y TIN2016-80565-R (AEI/FEDER, UE), a la Cátedra Santander de Eficiencia y Productividad de la Universidad Miguel Hernández de Elche y al Ministerio de Ciencia e Innovación y la Agencia Estatal de Investigación por el proyecto PID2019-105952GB-I00/ AEI / 10.13039/501100011033.

Tesis por compendio de publicaciones

La presente tesis se ha elaborado siguiendo la normativa de Estudios de Doctorado de la Universidad Miguel Hernández de Elche (aprobada el 20 de abril de 2016) para la presentación de tesis doctorales en la modalidad de compendio por publicaciones y mención internacional. Las referencias completas de los artículos que constituyen el cuerpo de la tesis son:

- *Aparicio, J., Cordero, J. M., Gonzalez, M., & Lopez-Espin, J. J. (2018). Using non-radial DEA to assess school efficiency in a cross-country perspective: An empirical analysis of OECD countries.*

- *Gonzalez M, López-Espín JJ, Aparicio J, Talbi E. 2022. A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis.* PeerJ Computer Science *8:e828*

- *González, M., López-Espín, J. J., Aparicio, J., & Giménez, D. (2018). A Parallel Application of Matheuristics in Data Envelopment Analysis. In International Symposium on Distributed Computing and Artificial Intelligence (pp. 172-179). Springer, Cham.*

- *González, M., López-Espín, J. J. & Aparicio (2020). A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers. Electronics, 9(9), 1541.*

El Dr. D. Juan Aparicio Baeza, director/a, y el Dr. D. José Juan López Espín, codirector/a de la tesis doctoral titulada **"Medición de la eficiencia y la productividad: Aspectos computacionales"**

INFORMA/N:

Que D./Dña. *"nombre y apellidos del/a estudiante"* ha realizado bajo nuestra supervisión el trabajo titulado **"Medición de la eficiencia y la productividad: Aspectos computacionales"** conforme a los términos y condiciones definidos en su Plan de Investigación y de acuerdo al Código de Buenas Prácticas de la Universidad Miguel Hernández de Elche, cumpliendo los objetivos previstos de forma satisfactoria para su defensa pública como tesis doctoral.

Lo que firmo/firmamos para los efectos oportunos, en Elche a 02 de Marzo de 2022

Director/a de la tesis	Codirector/a de la tesis
Dr. D. Juan Aparicio Baeza	Dr. D. José Juan López Espín

PROGRAMA de DOCTORADO INTERUNIVERSITARIO DEcIDE

COMISIÓN ACADÉMICA INTERUNIVERSITARIA

Cristina Vilaplana Prieto, secretaria de la Comisión Académica Interuniversitaria (CAI) del programa de doctorado en Economía DEcIDE,

HACE CONSTAR,

Que la Comisión Académica Interuniversitaria, en la reunión de 20 de diciembre 2021, dentro del punto número 3 de "Autorización para la presentación de tesis doctorales" se autorizó la presentación de la tesis "Medición de la eficiencia y la productividad: Aspectos computacionales" realizada por el estudiante Martín González Espinosa y dirigida por los profesores Juan Aparicio Baeza y José Juan López Espín.

Para que conste, firmo el presente documento en Murcia, a 13 de enero de 2022

Fdo: Cristina Vilaplana Prieto

Secretaria CAI Programa de Doctorado en Economía (DEcIDE)

Índice general

Summary

The purpose of efficiency and productivity problems is based on evaluating whether the use of the resources available (inputs) by a company or public institution (in general, any decision-making unit) corresponds or not with the optimal way of operating in such a way as to generate the largest possible number of outputs. To carry out this type of calculations, several mathematical models have already been proposed in the specialized literature that can be used, all of which are based on Mathematical Programming problems, and, in particular, some of them correspond to Mixed Integer Linear Programming problems (MILP). These types of problems combine several types of variables, continuous and discrete, in the same mathematical model as well as numerous restrictions, depending on the nature of the problem; features that can make the resolution process somewhat difficult. In addition, it is worth noting that these problems tend to be combinatorial in practice (NP-hard). Throughout this work, the analysis and study will focus on a field within the area of Operations Research called Data Envelopment Analysis (DEA), whose main objective is the estimation of production frontiers and the measurement of productive efficiency. Different optimization models belonging to this field will be put to the test in this thesis from a purely computational perspective, being solved through different techniques, both 2 exact and approximate, analyzing the performance and the difficulty of the same. The main objective of this work does not lie in the development and modeling of new problems in the field of DEA, but in how to achieve optimal solutions in a reasonable time for certain problems of a combinatorial nature, given that

being NP-hard type problems, as the size of the problem grows, so does the difficulty of obtaining optimal solutions, especially in a short time. At this point, we will focus on the study and design of approximation techniques, known in the literature as Metaheuristics, closely linked to Machine Learning or Artificial Intelligence methodologies. In addition to these methodologies, based on learning and improving the solutions obtained, parallelization techniques have also been incorporated, capable of efficiently reducing the time needed to obtain optimal solutions in complex problems. This thesis is structured around four articles:

- **Article 1**.-

Aparicio, J., Cordero, J. M., Gonzalez, M., & Lopez-Espin, J. J. (2018). Using non-radial DEA to assess school efficiency in a cross-country perspective: An empirical analysis of OECD countries.

In this first contribution, the authors formulate the problem to be addressed, applying it to a real database (PISA Report for OECD countries). In this work, the authors empirically exploit 3 non-radial efficiency measures of performance and the estimation of an educational production function based on DEA techniques. Specifically, the non-radial measures of DEA allow the identification of different levels of inefficiency for each outcome considered (Mathematical, Language and Science Competencies of the PISA report). The optimization model to be solved is expressed through Bi-level Mathematical Programming, and it is transformed into a mono-level using the Karush-Kuhn-Tucker conditions. L The resolution of the models is done through a standard optimizer from the literature, with the intention of obtaining optimal results regardless of the computation time used.

- **Article 2**.-

Gonzalez M, López-Espín JJ, Aparicio J, Talbi E. 2022. A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis. PeerJ Computer Science 8:e828

This second contribution studies the main difficulties when solving certain DEA models based on the concept of determining the distance (mathematical) to the production frontier from an interior point of a convex, introducing and developing optimization algorithms based on heuristic methods. In this work, the authors propose a decomposition based on MILP in which the optimization problem is divided into two hierarchical subproblems. In this way, a decomposition is investigated that separates the discrete and continuous variables, treating each sub-problem with different optimization methods. The method introduced is generic 4 and would allow the resolution of problems outside the scope of DEA. On the other hand, a metaheuristic focuses on the search space for discrete variables, while the exact method takes continuous variables into account. Therefore, the metaheuristic uses an indirect representation that encodes an incomplete solution to the problem, while the exact method is applied to decode the solution and generate a complete solution. Finally, a new hybrid algorithm (exact methods and metaheuristics) is developed within the field known as Matheuristics.

- **Article 3**.-

González, M., López-Espín, J. J., Aparicio, J., & Giménez, D. (2018). A Parallel Application of Matheuristics in Data Envelopment Analysis. In International Symposium on Distributed Computing and Artificial Intelligence (pp. 172-179). Springer, Cham.

In this third work, the study focuses on the optimization of the computation time used when solving the previously presented models. In this way, parallelization techniques capable of simultaneously solving different MILP models are introduced, and of optimizing each of them using all the resources available in the computing node. The document also focuses on the performance of different optimization software packages working in parallel. In particular, a comparison is made of two optimization software packages widely known in the literature (CPLEX and GUROBI), when they work by running several simultaneous instances, solving several problems at the same time.

- **Article 4**.-

González, M., López-Espín, J. J. & Aparicio (2020). A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers. Electronics, 9(9), 1541.

This work is a direct extension of article 3 proposed in this thesis. It focuses on the optimization of the parallel algorithm at the level of hardware resources. In this way, various studies are carried out using all the proposed levels of parallelism, modifying the assigned resources until an

optimal configuration is found for the proposed MILP problem. A more powerful machine has been used, with a greater number of processors to have a greater configuration margin. In addition, the performance study between different optimization packages (CPLEX and GUROBI) continues.

Resumen

La finalidad de los problemas de eficiencia y productividad se basan en evaluar si el uso de los recursos (entradas o *inputs*, en inglés) disponibles por parte de una empresa o institución pública (en general, cualquier unidad tomadora de decisiones) se corresponde o no con la forma óptima de operar de dicha entidad, generando la mayor cantidad de salidas posible (*outputs* en inglés). Para llevar a cabo este tipo de cálculos, varios modelos matemáticos han sido ya planteados en la literatura especializada que pueden ser utilizados, teniendo en común todos ellos que están basados en problemas de Programación Matemática, y, en particular, algunos de ellos se corresponden con problemas de Programación Matemática Lineal Mixta (*Mixed Integer Linear Programming* en inglés – MILP). Este tipo de problemas combinan en un mismo modelo matemático varios tipos de variables, continuas y discretas, así como numerosas restricciones, dependiendo de la naturaleza del problema, siendo estas restricciones características que pueden hacer que el proceso de resolución resulte ser algo difícil. Además, cabe destacar la característica de que estos problemas suelen ser en la práctica de tipo combinatorio (NP-duros).

A lo largo de este trabajo, el análisis y el estudio se va a centrar en un campo dentro del área de Investigación Operativa denominado Análisis Envolvente de Datos (*Data Envelopment Analysis* en inglés - DEA), cuyo principal objetivo es el de la estimación de fronteras de producción y la medición de la eficiencia productiva. Diferentes modelos de optimización

pertenecientes a este ámbito serán puestos a prueba en esta tesis desde una perspectiva puramente computacional, siendo resueltos a través de diferentes técnicas, tanto exactas como de aproximación, analizando el rendimiento y la dificultad del mismo.

El objetivo principal de este trabajo no reside en el desarrollo y modelado de nuevos problemas en el ámbito del DEA, sino en cómo conseguir soluciones óptimas y eficientes en un tiempo razonable para ciertos problemas de naturaleza combinatoria, dado que al ser problemas de tipo NP-duro, a medida que el tamaño del problema crece, también lo hace la dificultad de obtener soluciones óptimas, sobre todo en un tiempo reducido. En este punto, centraremos la atención en el estudio y diseño de técnicas de aproximación, conocidas en la literatura como Metaheurísticas, estando muy ligadas a metodologías de *Machine Learning* o *Artificial Inteligence*. Además de estas metodologías, basadas en el aprendizaje y la mejora de las soluciones obtenidas, también se han incorporado técnicas de paralelismo, capaces de reducir de forma eficiente el tiempo necesario para obtener soluciones óptimas en problemas complejos.

Esta tesis se estructura en torno a tres artículos:

- **Artículo 1**.-

Aparicio, J., Cordero, J. M., Gonzalez, M., & Lopez-Espin, J. J. (2018). Using non-radial DEA to assess school efficiency in a cross-country perspective: An empirical analysis of OECD countries.

En esta primera aportación, los autores formulan el problema a tratar, aplicándolo a una base de datos real (Informe PISA para los países de la OECD). En este trabajo los autores explotan de manera empírica medidas de rendimiento no radiales de eficiencia y la estimación de una función de producción educativa basada en técnicas DEA. Específicamente, las medidas no radiales de DEA permiten identificar diferentes niveles de ineficiencia para cada resultado considerado (Competencia Matemática, Lengua y Ciencias del informe PISA). El modelo de optimización a resolver se expresa a través de Programación Matemática Bi-nivel, y es transformado en mono-nivel mediante las condiciones de Karush-Kuhn-Tucker. La resolución de los modelos se hace a través de un optimizador estándar de la literatura, con la intención de obtener resultados óptimos sin importar el tiempo de cómputo utilizado.

- **Artículo 2**.-

Gonzalez M, López-Espín JJ, Aparicio J, Talbi E. 2022. A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis. PeerJ Computer Science 8:e828

En esta segunda aportación se estudian las principales dificultades a la hora de resolver ciertos modelos DEA basados en el concepto de determinación de la distancia (matemática) a la frontera de producción desde un punto interior de un convexo, introduciendo y desarrollando algoritmos de optimización basados en métodos heurísticos. En este trabajo, los autores proponen una descomposición basada en MILP en la que el problema de optimización se divide en dos subproblemas

jerárquicos. De esta forma, se investiga una descomposición que separa las variables discretas y continuas, tratando cada sub-problema con diferentes métodos de optimización. El método introducido es genérico y permitiría la resolución de problemas fuera del ámbito del DEA. Por otro lado, un metaheurístico se centra en el espacio de búsqueda de variables discretas, mientras que el método exacto tiene en cuenta las variables continuas. Por lo tanto, el metaheurístico utiliza una representación indirecta que codifica una solución incompleta para el problema, mientras que el método exacto se aplica para decodificar la solución y generar una solución completa. Finalmente, se desarrolla un nuevo algoritmo híbrido (métodos exactos y metaheurísticas) dentro del campo conocido como Matheurística.

- **Artículo 3**.-

González, M., López-Espín, J. J., Aparicio, J., & Giménez, D. (2018). A Parallel Application of Matheuristics in Data Envelopment Analysis. In International Symposium on Distributed Computing and Artificial Intelligence (pp. 172-179). Springer, Cham.

En este tercer trabajo, el estudio se centra en la optimización del tiempo de cómputo utilizado en los modelos previamente presentados. De este modo, se introducen técnicas de paralelismo capaces de resolver diferentes modelos MILP de forma simultánea, y de optimizar cada uno de ellos utilizando todos los recursos disponibles en el nodo de cómputo. El documento también se centra en el rendimiento de diferentes paquetes de software de optimización que trabajan en paralelo. En particular, se realiza una comparación de dos paquetes de

software de optimización ampliamente conocidos en la literatura (CPLEX y GUROBI), cuando trabajan ejecutando varias instancias simultáneas, resolviendo varios problemas al mismo tiempo.

- **Artículo 4**.-

González, M., López-Espín, J. J. & Aparicio (2020) A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers. Electronics (High-Performance Computer Architectures and Applications)

Este trabajo es una ampliación directa del artículo 3 propuesto en esta tesis. Se centra en la optimización del algoritmo paralelo a nivel de recursos de hardware. De esta forma, se realizan diversos estudios utilizando todos los niveles de paralelismo propuestos, modificando los recursos asignados hasta encontrar una configuración optima par ale problema MILP propuesto.

Se ha utilizado una máquina más potente, con mayor número de procesadores para tener mayor margen de configuración. Además, se continua con el estudio de rendimiento entre diferentes paquetes de optimización (CPLEX y GUROBI).

CAPÍTULO 1

Introducción

La presente tesis se enmarca dentro de la rama de la economía, centrada en concreto en la evaluación y cálculo de eficiencia y productividad, y de la rama de la algorítmica y la computación, en tanto a que se evalúa la forma más eficiente de resolver modelos matemáticos complejos de esta índole. Por tanto, mezcla dos campos diferenciados, pero que a su vez convergen a la hora de afrontar un problema real. Por esta parte, el trabajo realizado se basa tanto en el conocimiento del modelo a evaluar, como en la forma más óptima de llegar a su solución.

Dado que el modelo matemático utilizado ha sido previamente diseñado, no es el objetivo principal de este trabajo. El principal objetivo es conseguir resolverlo de la forma más eficiente posible, analizando diferentes técnicas encontradas en la literatura, tanto exactas como métodos de aproximación, y analizar al mismo tiempo el rendimiento a nivel de computación que este genera a la hora de ser resuelto.

Por tanto, la presente introducción va a desglosar las diferentes partes de la tesis, dando una breve introducción histórica, así como el estado del arte de cada uno de ellos, para más adelante, profundizar en los temas más relevantes estudiados a lo largo de esta tesis.

1.1. Medición de la Eficiencia y la productividad mediante el Análisis Envolvente de Datos

El Análisis Envolvente de Datos (*Data Envelopment Analysis* en inglés, DEA) es una metodología cuyo principal objetivo es el análisis de fronteras de producción y la medición de la eficiencia técnica. Sus orígenes se remontan al artículo seminal de Charnes et al. (1978) y los trabajos de Banker et al. (1984) y Charnes et al. (1985), si bien hay antecedentes en la literatura económica relacionada con la agricultura que presentan modelos similares (Boles (1966), Bressler (1966) y Seitz (1966)). En el relativamente corto periodo de tiempo transcurrido desde su nacimiento, esta metodología ha experimentado un rápido desarrollo, posicionándose como una importante herramienta de análisis en los estudios de eficiencia y productividad

Así, el DEA es una importante y altamente fructífera técnica de la Investigación Operativa y la Economía centrada en la medición y análisis de la eficiencia con que se producen bienes y se proporcionan servicios. La naturaleza de las actividades de producción susceptibles de estudio mediante esta técnica varía ampliamente: desde las industrias manufactureras de todo tipo a entidades sin ánimo de lucro dedicadas a proporcionar diversos servicios públicos. En este contexto, al agente objeto de la medición de la eficiencia se le suele denominar en inglés *Decision Making Unit* (DMU), con el propósito de enfatizar su grado de independencia a la hora de modificar sus niveles de producción (véase Aparicio, 2007).

historia del DEA comenzó con la tesis de Edwardo Rhodes, en la Carnegie Mellon University en Pittsburgh, dirigida por W.W. Cooper. En

aquel momento Rhodes trabajaba en la evaluación de programas educativos para estudiantes desaventajados que se aplicaban en escuelas públicas de los Estados Unidos subvencionadas por el Gobierno Federal. En particular, Rhodes y Cooper se centraron en el programa conocido como *Follow Through*. Este proyecto del Departamento de Educación de los Estados Unidos pretendía aplicar los principios estadísticos del diseño de experimentos para realizar un estudio a nivel nacional sobre el comportamiento de un conjunto de escuelas. La base de datos utilizada era lo suficientemente grande como para que problemas que se presentan con cierta asiduidad, tales como los grados de libertad, no representaran un serio inconveniente en el análisis, a pesar del elevado número de variables (inputs y outputs) involucradas. Sin embargo, la aplicación de las técnicas clásicas de carácter estadístico condujo a resultados poco satisfactorios mientras que, en contraposición, el DEA se reveló como una herramienta de enorme potencial. La parte más substancial del conjunto de fundamentos teóricos en los que se sustentó la tesis de Rhodes vieron la luz en un artículo publicado en 1978 por Abraham Charnes, William W. Cooper y Edwardo Rhodes con el título *Measuring the efficiency of decision making units* en *European Journal of Operational Research*.

Anteriormente al trabajo de Charnes, Cooper y Rhodes, fue Farrell con su artículo *The Measurement of Productive Efficiency* (publicado en 1957 en la revista *Journal of the Royal Statistical Society*) el autor más influyente en temas relacionados con la medición de la eficiencia y la productividad. Véase también Debreu (1951), Koopmans (1951), Shephard (1953) y Afriat (1974). A este respecto, Farrell era consciente de que para ser capaces de calcular una medida de eficiencia era necesario conocer previamente la forma explícita de la función de producción. Dado que, en la práctica, la frontera de producción nunca es conocida, Farrell sugirió que

esta función podría ser estimada a partir de una muestra de datos usando, alternativamente, una tecnología no paramétrica lineal a trozos, o bien, una función de producción paramétrica. Estas ideas condujeron, décadas más tarde, a dos metodologías claramente diferenciadas: el DEA y las fronteras estocásticas, respectivamente. Mientras que el DEA utiliza herramientas de la programación matemática, la aproximación a la medición de la eficiencia a través de fronteras estocásticas recurre a técnicas de carácter puramente econométrico.

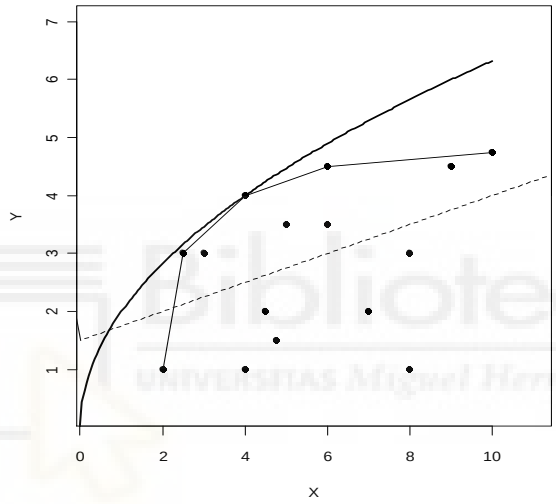

Figura 1. Representación de una frontera lineal a trozos.

En DEA se optimiza para cada DMU un modelo de programación matemática con el objetivo de estimar una frontera lineal a trozos, determinada por las DMUs eficientes en el sentido de Pareto; es decir, aquéllas no dominadas (véase la Figura 1, donde las DMUs vienen representadas por puntos en el plano en el caso de un input [X] y un output [Y]). A través del enfoque paramétrico, se asume que el modelo ajustado es aplicable a cada unidad en la muestra. Por el contrario, en DEA se optimiza de manera particular la medida de lo bien o mal que opera cada DMU.

1.2. Elementos básicos de DEA

De ahora en adelante supondremos que se desea evaluar el grado de eficiencia de un conjunto de n DMUs que consumen m inputs para producir s outputs. Concretamente, cada DMUj, que puede describirse como el par (X_j, Y_j), utiliza los inputs $X_j = (x_{1j}, \ldots, x_{mj})$ para producir los outputs $Y_j = (y_{1j}, \ldots, y_{sj})$. Asumiremos que tanto los inputs como los outputs son estrictamente positivos. Asimismo, $(0_m, 0_s)$ representará el vector de dimensión $m + s$ con todas sus componentes nulas.

A continuación, mostramos los modelos básicos DEA aparecidos en la literatura.

1.2.1. CCR (Charnes, Cooper y Rhodes)

Charnes et al. (1978) introdujeron una definición de eficiencia en forma de ratio que generalizaba la definición clásica del ratio entre un output y un input, usado en la Ingeniería y la Economía, al caso en el que existieran múltiples outputs e inputs. Sea la DMU_0 la entidad que deseamos evaluar, entonces el siguiente problema de Programación Lineal (PL), denominado CCR orientación input en su versión en forma de ratio, proporciona el grado de eficiencia relativa de dicha DMU:

$$
\begin{aligned}
Max \quad & \frac{\sum_{r=1}^{s} u_r y_{r0}}{\sum_{i=1}^{m} v_i x_{i0}} \\
s.a: \quad & \\
& \frac{\sum_{r=1}^{s} u_r y_{rj}}{\sum_{i=1}^{m} v_i x_{ij}} \leq 1, \quad j = 1, \ldots, n \\
& u_r \geq 0, \qquad r = 1, \ldots, s \\
& v_i \geq 0, \qquad i = 1, \ldots, m
\end{aligned}
\tag{1}
$$

El objetivo de (1) consiste en determinar unos pesos no negativos para inputs y outputs de forma tal que se maximice el grado de eficiencia (score) de la DMU_0. Obsérvese que dependiendo de la DMU que sea evaluada a través del anterior programa, los pesos óptimos obtenidos pueden variar. De hecho, será el escenario más habitual en la práctica.

La transformación de modelos de programación fraccional lineal en modelos lineales debida a Charnes y Cooper (1962) permite determinar el valor óptimo de (1) a través del siguiente problema de PL equivalente:

$$\begin{array}{ll} Max & \sum_{r=1}^{s} \mu_r y_{r0} \\ s.a: & \\ & \sum_{i=1}^{m} \nu_i x_{i0} = 1 \\ & -\sum_{i=1}^{m} \nu_i x_{ij} + \sum_{r=1}^{s} \mu_r y_{rj} \leq 0, \quad j = 1,\ldots,n \\ & \mu_r \geq 0, \quad r = 1,\ldots,s \\ & \nu_i \geq 0, \quad i = 1,\ldots,m \end{array} \tag{2}$$

Al modelo (2) se le conoce como formulación del modelo CCR en función de los multiplicadores. El problema dual asociado a dicho modelo es el siguiente, el cual se conoce como la formulación del modelo CCR en su versión de envoltura:

$$\begin{array}{ll} Min & \theta_0 \\ s.a: & \\ & \sum_{j=1}^{n} \lambda_j x_{ij} = \theta x_{i0} - s_{i0}^{-}, \quad i = 1,\ldots,m \\ & \sum_{j=1}^{n} \lambda_j y_{rj} = y_{r0} + s_{r0}^{+}, \quad r = 1,\ldots,s \\ & \lambda_j \geq 0, \quad j = 1,\ldots,n \\ & s_{r0}^{+} \geq 0, \quad r = 1,\ldots,s \\ & s_{i0}^{-} \geq 0, \quad i = 1,\ldots,m \end{array} \tag{3}$$

Si denotamos por $(\theta_0^*, \lambda_1^*, \ldots, \lambda_n^*, s_{10}^{-*}, \ldots, s_{m0}^{-*}, s_{10}^{+*}, \ldots, s_{s0}^{+*})$ a una solución óptima de (3), entonces, en primer lugar, puede probarse que

$0 < \theta_0^* \leq 1$ y, en segundo lugar, podemos identificar si la DMU_0 evaluada es eficiente o no, haciendo uso de la definición siguiente.

Definición 1.1: [Definición de Eficiencia de Farrell-Debreu] La DMU_0 es eficiente si, y sólo si, $\theta_0^* = 1$.

Obsérvese que la solución del modelo (3) involucra realizar reducciones simultáneas de todos los inputs sin alterar el mix, es decir, las proporciones en las que los recursos son utilizados en el proceso de producción, a través del coeficiente θ_0^*. De hecho, la resolución del modelo (3) determina un punto sobre la superficie de envoltura obtenido al reducir al máximo (al proyectar) y de forma equiproporcional todos los inputs de la DMU evaluada. El valor óptimo de la variable θ_0^* identifica el grado de ineficiencia. Por su parte, al resolver el programa (2) obtenemos la expresión del hiperplano soporte del conjunto de referencia empírico determinado por las condiciones lineales de (3) sobre el que se produce la proyección de la DMU_0. Estos hiperplanos permiten la estimación de precios sombra para inputs y outputs, así como tasas marginales de sustitución y transformación.

Otro subproducto del DEA, es la obtención a través de la resolución del modelo (3) de referentes y targets para la DMU_0. Los referentes, el conjunto de DMUs tales que $\lambda_j^* > 0$, constituyen para la entidad evaluada empresas modelo observadas (reales) que dicha unidad ineficiente debería imitar con el propósito de mejorar su rendimiento. Por otro lado, los targets son las coordenadas de la proyección sobre la frontera de la tecnología, es decir, $\left(\sum_{j=1}^n \lambda_j^* x_{1j}, \ldots, \sum_{j=1}^n \lambda_j^* x_{mj}, \sum_{j=1}^n \lambda_j^* y_{1j}, \ldots, \sum_{j=1}^n \lambda_j^* y_{sj}\right)$ y representan los niveles de inputs y outputs virtuales (no necesariamente

observados) que harían a la correspondiente DMU evaluada rendir de manera eficiente.

De forma análoga a como se actuó con el modelo (1), puede ser definido el modelo CCR de orientación output. En tal caso, la medida de eficiencia relativa obtenida resulta de la expansión máxima equiproporcional de los outputs de la DMU_0 que es factible para su nivel de inputs observado.

Ahora bien, al resolver (3) una DMU_0 puede ser evaluada como eficiente, siguiendo la Definición 1.1, y, sin embargo, es posible que alguno de sus inputs u outputs pueda ser mejorado sin empeorar alguno de los restantes inputs y outputs. O lo que es lo mismo, puede darse que $\theta_0^* = 1$ y, sin embargo, $\exists i' = 1, \ldots, m$ tal que $s_{i'0}^{-*} > 0$ y/o $\exists r' = 1, \ldots, s$ tal que $s_{r'0}^{+*} > 0$. Es por este motivo por el cual se hace necesario extender la definición de eficiencia técnica debida a Farrell y Debreu.

Lo más habitual en el DEA, a la hora de determinar el score de eficiencia y las holguras asociadas a alguna DMU_0, es utilizar un procedimiento en dos fases. En la primera fase se resuelve (3). Así, θ_0^* coincide con el valor de la eficiencia de tipo Farrell-Debreu para la DMU evaluada. Este valor de θ_0^* será incorporado en la segunda fase del método. En esta segunda fase, se resuelve el modelo de PL siguiente usando como variables $(\lambda_1, \ldots, \lambda_n, s_{10}^-, \ldots, s_{m0}^-, s_{10}^+, \ldots, s_{s0}^+)$:

$$
\begin{aligned}
Max \quad & \sum_{i=1}^{m} s_{i0}^{-} + \sum_{r=1}^{s} s_{r0}^{+} \\
s.a: \quad & \\
& \sum_{j=1}^{n} \lambda_j x_{ij} = \theta_0^* x_{i0} - s_{i0}^{-}, \quad i = 1, \ldots, m \\
& \sum_{j=1}^{n} \lambda_j y_{rj} = y_{r0} - s_{r0}^{+}, \quad r = 1, \ldots, s \\
& \lambda_j \geq 0, \quad j = 1, \ldots, n \\
& s_{i0}^{-} \geq 0, \quad i = 1, \ldots, m \\
& s_{r0}^{+} \geq 0, \quad r = 1, \ldots, s
\end{aligned}
\tag{4}
$$

El objetivo del modelo (4) es encontrar una solución que maximice la suma de las holguras asociadas a inputs y outputs, es decir, se maximiza la distancia ℓ_1^{m+s} del punto $(\theta_0^* x_{10}, \ldots, \theta_0^* x_{m0}, y_{10}, \ldots, y_{s0})$ a la frontera de la tecnología. Es fácil comprobar que el punto proyección asociado a una solución de este modelo se corresponde con un punto de Pareto. Por tanto, la primera fase del método proporciona la eficiencia de Farrell-Debreu, pero no necesariamente un vector de targets no dominado, mientras que la segunda fase determina un vector proyección de Pareto a partir de la proyección radial de la primera fase.

Koopmans (1951) introdujo en el área del análisis de actividades la definición extendida de eficiencia técnica que mostramos a continuación, relacionada con el concepto de dominancia de Pareto.

Definición 1.2: [Definición de Eficiencia de Pareto-Koopmans] Sea θ_0^* la solución óptima de la primera fase y $(s_{10}^{-*}, \ldots, s_{m0}^{-*}, s_{10}^{+*}, \ldots, s_{s0}^{+*})$ la solución óptima de la segunda fase, entonces la DMU_0 es eficiente si, y sólo si, $\theta_0^* = 1$, $s_{i0}^{-*} = 0$, $i = 1, \ldots, m$, y $s_{r0}^{+*} = 0$, $r = 1, \ldots, s$.

Una vez introducidas las dos definiciones de eficiencia más importantes de la literatura económica, mostraremos en detalle el modelo DEA que recibe el nombre de BCC, debido a Banker et al., 1984.

1.2.2. BCC (Banker, Charnes y Cooper)

En el artículo publicado en 1984 por Banker, Charnes y Cooper, se propone un enfoque alternativo de tipo axiomático para la evaluación de la eficiencia. La clave de este planteamiento axiomático reside en la caracterización del llamado conjunto de posibilidades de producción (o tecnología):

$$T = \{(X, Y): X \text{ puede producir Y}\}. \tag{5}$$

A partir de este conjunto T pueden definirse los conjuntos de posibilidades de inputs para cada Y como:

$$L(Y) = \{X: (X, Y) \in T\}, \tag{6}$$

y de forma análoga los conjuntos de posibilidades de outputs para cada X:

$$P(X) = \{Y: (X, Y) \in T\}. \tag{7}$$

Con respecto al conjunto T, Banker et al. (1984) asumen los postulados siguientes:

Postulado 1. [Convexidad] Si $(X_j, Y_j) \in T, j = 1, \ldots, n$, entonces

$$\left(\sum_{j=1}^{n} \lambda_j x_{1j}, \ldots, \sum_{j=1}^{n} \lambda_j x_{mj}, \sum_{j=1}^{n} \lambda_j y_{1j}, \ldots, \sum_{j=1}^{n} \lambda_j x_{sj}\right) \in T,$$

para cualquier $\lambda_j \geq 0, j = 1, \ldots, n$, tales que $\sum_{j=1}^{n} \lambda_j = 1$.

Postulado 2. [Ineficiencia o libre disponibilidad] Si $(X,Y) \in T$ y $X \leq \tilde{X}$, entonces $(\tilde{X},Y) \in T$. Por otro lado, si $(X,Y) \in T$ e $Y \geq \tilde{Y}$, entonces $(X,\tilde{Y}) \in T$.

Postulado 3. [No acotación de los rayos o rendimientos constantes a escala] Si $(X,Y) \in T$, entonces $(kX,kY) \in T$ para cualquier $k \geq 0$.

Postulado 4. [Mínima extrapolación] T es la intersección de todos los conjuntos $\bar{T}$ que cumplen los postulados 1, 2 y 3 sujetos a la condición de que cada una de las DMUs observadas cumplan que $(X_j,Y_j) \in \bar{T},\ j = 1,\ldots,n$. O lo que es lo mismo, T es el conjunto más reducido de R^{m+s} que verifica todos los postulados anteriores a la vez que contiene a los datos observados.

Por otra parte, el conjunto de posibilidades de producción T que cumple, a la vez, con los cuatro postulados anteriores puede ser caracterizado como:

$$T_{CRS} = \{(X,Y) \in R_+^{m+s} : X \geq \textstyle\sum_{j=1}^{n} \lambda_j X_j,\ Y \leq \sum_{j=1}^{n} \lambda_j Y_j,\ \lambda_j \geq 0, j = 1,\ldots,n\}, \quad (8)$$

donde $X \geq \sum_{j=1}^{n} \lambda_j X_j$ es equivalente a $x_i \geq \sum_{j=1}^{n} \lambda_j x_{ij},\ i = 1,\ldots,m$, e $Y \leq \sum_{j=1}^{n} \lambda_j Y_j$ equivale a $y_r \leq \sum_{j=1}^{n} \lambda_j y_{rj}, r = 1,\ldots,s$.

Obsérvese que el postulado 3 implica que la tecnología T en (8) exhibe rendimientos constantes a escala.

Si el conjunto T únicamente verifica los postulados 1, 2 y 4, entonces la tecnología queda caracterizada como:

$$T_{VRS} = \{(X,Y) \in R_+^{m+s} : X \geq \sum_{j=1}^{n} \lambda_j X_j\,,\ Y \leq \sum_{j=1}^{n} \lambda_j Y_j,\ \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \geq 0, j = 1,\ldots,n\}. \quad (9)$$

Si para esta nueva situación, que exhibe rendimientos variables a escala (al no considerar el postulado tercero), se deseara determinar el grado de eficiencia de una DMU_0, se llegaría al siguiente problema de PL:

$$\begin{array}{lll} Min & \theta_0 & \\ s.a: & & \\ & \sum_{j=1}^{n} \lambda_j x_{ij} = \theta_0 x_{i0} - s_{i0}^-, & i = 1,\ldots,m \\ & \sum_{j=1}^{n} \lambda_j x_{ij} = y_{r0} + s_{r0}^+, & r = 1,\ldots,s \\ & \sum_{j=1}^{n} \lambda_j = 1, & \\ & \lambda_j \geq 0, & j = 1,\ldots,n \\ & s_{i0}^- \geq 0, & i = 1,\ldots,m \\ & s_{r0}^+ \geq 0, & r = 1,\ldots,s \end{array} \quad (10)$$

que se conoce como la formulación del modelo BCC de orientación input en su versión de envoltura. La solución de (10) puede interpretarse de forma similar a la de (3), es decir, podemos utilizar igualmente las definiciones 1.1 y 1.2 para identificar a las DMUs eficientes en el sentido de Farrell-Debreu o Pareto-Koopmans, respectivamente. De hecho, la única diferencia entre el modelo CCR y el BCC reside en la restricción de convexidad de las variables de intensidad (lambdas).

La inclusión en el modelo DEA de la condición de convexidad induce que los rendimientos a escala de la tecnología sean variables (*Variable Returns to Scale* en inglés, VRS). En contraposición, si no incluimos la restricción de convexidad estaremos describiendo una tecnología que exhibe rendimientos constantes a escala (*Constant Returns to Scale* en inglés, CRS).

Por otra parte, nos gustaría señalar que el procedimiento en dos fases seguido para resolver el modelo CCR puede ser aplicado también al modelo BCC de manera análoga.

Geométricamente, la Figura que subyace de la expresión (9) es un poliedro en R^{m+s}. En la Figura 2 mostramos en dos dimensiones la típica tecnología DEA bajo rendimientos variables a escala (T cumple los postulados 1, 2 y 4). En ella puede visualizarse el conjunto de posibilidades de producción generado por las DMUs: 0, 1, 2, 3, 4 y 5 (Véase Aparicio, 2007).

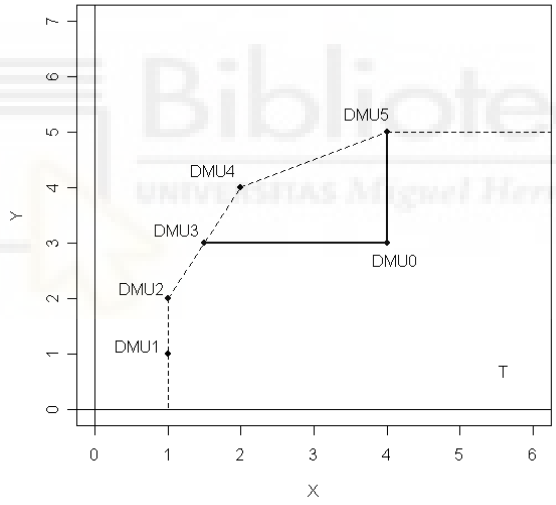

Figura 2. Representación de tecnología DEA bajo rendimientos variables a escala.

1.2.3. Modelo Aditivo

Hemos visto que los modelos CCR y BCC proporcionan medidas de eficiencia técnica en el sentido de Farrell-Debreu a través de θ_0^*. Ahora nos centraremos en el modelo aditivo, cuyos resultados responden al concepto económico de "no dominancia" de Pareto tal y como fue interpretado por Koopmans (1951).

El modelo aditivo fue introducido en Charnes et al. (1985) con la formulación siguiente (en su formato envolvente):

$$
\begin{aligned}
Max \quad & \textstyle\sum_{i=1}^{m} s_{i0}^{-} + \sum_{r=1}^{s} s_{r0}^{+} \\
s.a: \quad & \\
& \textstyle\sum_{j=1}^{n} \lambda_j x_{ij} = x_{i0} - s_{i0}^{-}, \quad i = 1,\dots,m \\
& \textstyle\sum_{j=1}^{n} \lambda_j y_{rj} = y_{r0} - s_{r0}^{+}, \quad r = 1,\dots,s \\
& \textstyle\sum_{j=1}^{n} \lambda_j = 1, \\
& \lambda_j \geq 0, \qquad j = 1,\dots,n \\
& s_{i0}^{-} \geq 0, \qquad i = 1,\dots,m \\
& s_{r0}^{+} \geq 0, \qquad r = 1,\dots,s
\end{aligned}
\tag{11}
$$

Al resolver el modelo (11) se maximiza la distancia ℓ_1^{m+s} desde la DMU_0 a la frontera eficiente. La DMU_0 será considerada eficiente, en el sentido de Pareto-Koopmans si, y sólo si, el valor óptimo de (11) es cero. En consecuencia, la DMU_0 será ineficiente si alguna holgura es estrictamente positiva, lo que, por otra parte, permite identificar las fuentes y la magnitud de la ineficiencia en los correspondientes inputs y outputs.

El conjunto factible de (11) coincide con el del modelo BCC. Por otra parte, para consultar la formulación del modelo aditivo bajo rendimientos constantes a escala y su versión de los multiplicadores consúltese el trabajo de Ali y Seiford, 1993.

En cuanto a las propiedades del modelo, el valor de la función objetivo de (11) depende de las unidades de medida. Por este motivo, ciertos autores han propuesto diferentes normalizaciones de la función objetivo de (11) (véase Cooper et al., 1999):

$$\sum_{i=1}^{m} \frac{s_{i0}^{-}}{x_{i0}} + \sum_{r=1}^{s} \frac{s_{r0}^{+}}{y_{r0}}, \tag{12}$$

dando lugar al modelo conocido como aditivo extendido.

1.2.4. Otros conceptos

Por otra parte, tanto cuando hacemos uso de la definición de eficiencia de Farrell-Debreu como de Pareto-Koopmans de manera implícita estamos asumiendo la comparación de la DMU evaluada con respecto a los puntos de cierto subconjunto de la tecnología. En este sentido, y dada su relevancia, introducimos en primer lugar la definición de frontera débilmente eficiente (véase Briec,1998):

Definición 1.7: El conjunto $\partial^{W}(T) = \{(X,Y) \in T : U < X, V > Y \Rightarrow (U,V) \notin T\}$ es conocido como el conjunto eficiente débil de T o frontera débilmente eficiente.

Del anterior conjunto destaca cierto subconjunto de puntos como consecuencia de su importancia en la determinación de la eficiencia de tipo Pareto-Koopmans. Nos estamos refiriendo al subconjunto eficiente fuerte de T, o lo que es lo mismo, al conjunto de puntos no dominados de la tecnología. Su definición es la siguiente:

Definición 1.8: El conjunto

$$\partial^S(T) = \{(X,Y) \in T : U \leq X, V \geq Y, (U,V) \neq (X,Y) \Rightarrow (U,V) \notin T\}$$

es conocido como el conjunto Pareto-eficiente de T o frontera Pareto-eficiente.

$\partial^S(T)$ puede expresarse, equivalentemente, como la unión de las caras de la tecnología asociadas a hiperplanos soporte con todos los coeficientes (pendientes) estrictamente positivos (véase Chen et al. (2003)).

1.3. Cálculo de la distancia de un punto interior de la tecnología DEA a su frontera

De las secciones anteriores podemos concluir que en DEA es habitual, aunque sorprendente desde un punto de vista matemático, recurrir a la maximización, en lugar de minimización, de la suma de las desviaciones de la DMU evaluada al punto proyección en la frontera. Esto es así, tanto si se usa un modelo aditivo, o modelo aditivo extendido (con pesos), como si se recurre al modelo radial en primera instancia y al modelo aditivo en segunda etapa. Este tipo de procedimientos, llamados tradicionales, generan, en general, targets "alejados" de la unidad evaluada. Lo que contradice los principios mínimos del benchmarking a la hora de intentar no desincentivar a la unidad bajo estudio a alcanzar el objetivo de llegar a ser eficiente, es decir: posicionarse en la frontera de producción. La única justificación razonable para que este tipo de proceder haya sido y continúe siendo habitual en la literatura es su simplicidad computacional. El problema a resolver es lineal y fácilmente, por tanto, resoluble. El caso opuesto, es decir, aquel asociado a calcular la distancia "matemática" de un punto ineficiente a la frontera de la tecnología DEA resulta ser un problema

no convexo y de cálculo complejo, en general. Su complejidad se debe al hecho de que el problema se encuentra asociado al cálculo de la distancia de un punto interior de un convexo a la frontera (o parte de la frontera) de dicho convexo. Sin embargo, y a pesar de su mayor complejidad de cómputo, este tipo de aproximación proporciona los targets más fácilmente alcanzables por parte de la DMU analizada, si ésta deseara llegar a la frontera de producción y ser, por tanto, eficiente.

Por todo ello, parece interesante proponer modelos, probablemente más complejos que los tradicionales, pero en la línea de la determinación de distancias matemáticas a la frontera de producción del DEA.

En este sentido, la evaluación de la eficiencia de cada DMU_0 se obtiene generalmente como el resultado de su comparación con un punto proyección sobre el conjunto Pareto-eficiente de T. Las coordenadas de esta proyección serán los targets para la DMU_0. La aproximación que se mostrará a continuación para el cálculo de la distancia a la frontera, se encuentra fundamentada en la caracterización del conjunto de puntos Pareto-eficientes de T que dominan a la DMU_0. Dicha caracterización conduce a un conjunto de restricciones lineales que serán posteriormente implementadas a través de un problema de programación matemática, una vez haya sido prefijado un criterio de similitud entre los inputs/outputs observados y los targets. Este criterio se implementa en DEA normalmente mediante una distancia matemática o por medio del uso de una medida de eficiencia. Tanto en un caso como en otro, la nueva metodología nos conducirá a los targets más próximos.

Los resultados de esta sección serán desarrollados suponiendo rendimientos constantes a escala. No obstante, su extensión a rendimientos variables resulta trivial.

El teorema 1.1 nos proporciona la caracterización mencionada arriba.

Teorema 1.1: Sea D_0 el conjunto de puntos Pareto-eficientes en T que dominan a la DMU_0. Entonces,

$$(X,Y) \in D_0 \Leftrightarrow \exists\, \lambda_j, d_j \geq 0,\ b_j \in \{0,1\},\ j \in E,\ \nu_i \geq 1, i=1,...,m, \mu_r \geq 1,\ r=1,\ldots,s,$$
$$s_{i0}^- \geq 0,\ i=1,\ldots,m, \text{ y } s_{r0}^+ \geq 0,\ r=1,\ldots,s, \text{ tal que}$$

$$\begin{aligned}
X &= \textstyle\sum_{j\in E} \lambda_j X_j && && (13.1)\\
Y &= \textstyle\sum_{j\in E} \lambda_j Y_j && && (13.2)\\
\textstyle\sum_{j\in E} \lambda_j x_{ij} &= x_{i0} - s_{i0}^- && i=1,\ldots,m && (13.3)\\
\textstyle\sum_{j\in E} \lambda_j y_{rj} &= y_{r0} + s_{r0}^+ && r=1,\ldots,s && (13.4)\ (13)\\
-\textstyle\sum_{i=1}^{m} \nu_i x_{ij} + \sum_{r=1}^{s} \mu_r y_{rj} + d_j &= 0 && j \in E && (13.5)\\
d_j &\leq M b_j && j \in E && (13.6)\\
\lambda_j &\leq M(1-b_j) && j \in E && (13.7)
\end{aligned}$$

donde M es un escalar positivo suficientemente grande y E es el conjunto de DMUs eficientes extremas (puntos extremos del poliedro DEA). Para la prueba, véase Aparicio et al. (2007).

Consecuentemente, los puntos (X,Y) satisfaciendo las condiciones del Teorema 1.1 son aquellos de T que dominan a (X_0,Y_0) y que pueden expresarse como una combinación de unidades eficientes extremas localizadas sobre la misma cara de la frontera. Además, dado que todos los pesos de los hiperplanos que definen tales caras son estrictamente positivos, tenemos que los referentes asociados y sus correspondientes combinaciones lineales realmente pertenecerán a una cara eficiente.

Bajo el supuesto de VRS, basta añadir la típica condición de convexidad (la suma de las variables lambda igual a uno) al conjunto de restricciones en el teorema, y una variable libre al término de la izquierda de (13.5) en (13).

La importancia del anterior resultado recae en el hecho de que hemos sido capaces de representar el conjunto de puntos Pareto-eficientes que dominan a la DMU_0 por medio de condiciones lineales. Esto nos permitirá superar las dificultades de la no convexidad del conjunto factible, cuando enfocamos nuestros esfuerzos en minimizar la distancia a la frontera Pareto-eficiente. Adicionalmente, el resultado de la caracterización nos permitirá, en una única etapa, determinar los targets más cercanos a una DMU dada, una vez ha sido especificada la distancia o medida de eficiencia técnica a utilizar.

A continuación, formularemos diferentes modelos de programación matemática resultantes de la selección de varias de estas distancias o medidas de eficiencia. Para cada una de ellas, los targets correspondientes vendrán dados por las coordenadas del vector $\left(\sum_{j\in E}\lambda_j^* x_{1j}, \ldots, \sum_{j\in E}\lambda_j^* x_{mj}, \sum_{j\in E}\lambda_j^* y_{1j} \ldots, \sum_{j\in E}\lambda_j^* y_{sj}\right)$, donde el superíndice * significa que éstas son las soluciones óptimas para las variables de intensidad en el problema de optimización asociado. Obviamente, para cualquiera de los casos que vamos a estudiar, la distancia o medida de eficiencia sigue el enfoque de la distancia mínima a la frontera en DEA.

Si usásemos para evaluar proximidad la distancia ℓ_1^{m+s}, entonces el siguiente programa lineal entero mixto, que sigue la idea del modelo

aditivo tradicional pero en el cual se minimiza la función objetivo en lugar de maximizar, proporcionará los targets más próximos a la frontera Pareto-eficiente de acuerdo a esta distancia.

$$
\begin{aligned}
Min \quad & \sum_{i=1}^{m} s_{i0}^{-} + \sum_{r=1}^{s} s_{r0}^{+} \\
s.a: \quad & \\
& \sum_{j \in E} \lambda_j x_{ij} = x_{i0} - s_{i0}^{-} && i = 1, \ldots, m \quad (b.1) \\
& \sum_{j \in E} \lambda_j y_{rj} = y_{r0} + s_{r0}^{+} && r = 1, \ldots, s \quad (b.2) \\
& -\sum_{i=1}^{m} v_i x_{ij} + \sum_{r=1}^{s} \mu_r y_{rj} + d_j = 0 && j \in E \quad (b.3) \\
& v_i \geq 1 && i = 1, \ldots, m \quad (b.4) \\
& \mu_r \geq 1 && r = 1, \ldots, s \quad (b.5) \\
& d_j \leq M b_j && j \in E \quad (b.6) \\
& \lambda_j \leq M(1 - b_j) && j \in E \quad (b.7) \\
& b_j \in \{0,1\} && j \in E \quad (b.8) \\
& d_j \geq 0, \lambda_j \geq 0 && j \in E \quad (b.9) \\
& s_{i0}^{-} \geq 0 && i = 1, \ldots, m \quad (b.10) \\
& s_{r0}^{+} \geq 0 && r = 1, \ldots, s \quad (b.11)
\end{aligned}
\tag{14}
$$

Según el modelo (14), la caracterización de una DMU eficiente no cambia con respecto a dicha caracterización en el modelo aditivo tradicional.

Otras medidas DEA pueden ser, de la misma forma, adaptadas al contexto de determinar los targets más “cercanos” a la DMU evaluada. Un ejemplo relevante es de la medida conocida como Slacks-Based Measure (SBM) (Tone, 2001) o Enhanced Russell Graph (Pastor et al., 1999). En este caso, la función objetivo a utilizar, junto a las restricciones del modelo (14), sería:

$$Max \quad \frac{1-\frac{1}{m}\sum_{i=1}^{m}\frac{\bar{s_{i0}}}{x_{i0}}}{1+\frac{1}{s}\sum_{r=1}^{s}\frac{s_{r0}^{+}}{y_{r0}}} \tag{15}$$

1.4. Técnicas de optimización

Dada la naturaleza del problema propuesto, y la dificultad de resolverlo, se han valorado distintas técnicas para encontrar soluciones válidas (que cumplan todas las restricciones) y óptimas para las instancias evaluadas. Los problemas MILP son conocidos por ser problemas NP-duros, en los cuales, obtener una de las soluciones óptimas del problema es muy complejo.

Podemos distinguir entre dos tipos diferentes de métodos de optimización: los métodos de optimización exactos, que garantizan encontrar una solución óptima, y los métodos de optimización heurística, donde no tenemos garantía de que se encuentre una solución óptima pero sí la posibilidad de encontrar aproximaciones satisfactorias. Para los problemas NP-duros, los métodos exactos requieren de un alto esfuerzo de cómputo, incluso en problemas de tamaño medio, volviéndose a menudo intratables e imposibles de resolver. Por esto, se acude a métodos heurísticos, normalmente dependientes del problema, explotando las propiedades internas de este.

1.4.1. Métodos exactos

Este tipo de técnicas, aplicadas para problemas de tipo MILP, están basadas en técnicas de búsqueda por árboles. Estas técnicas buscan en todo el espacio de soluciones, resolviendo el problema de forma completa, mediante el estudio y análisis de sub-espacios (distintos valores de las variables implicadas). Algunas de las técnicas más conocidas en este ámbito son: dynammic programming, branch and X (branch and bound (Lawler et al., 1966), branch and cut (Gomory, 2010), branch and Price (Barnhart et al., 1998) or constraint programming (Apt, 2000). Los métodos basados en arboles de decisión nos permiten formular el proceso de obtención de soluciones como una secuencia de decisiones basadas en las variables discretas del problema (Figura 2). El espacio de búsqueda se explora construyendo dinámicamente un árbol cuyo nodo raíz representa el problema que se está resolviendo y todo su espacio de búsqueda asociado. Los nodos hoja son las posibles soluciones y los nodos internos son subproblemas del espacio total de la solución. Estas técnicas pierden efectividad a medida que dichas variables crecen en tamaño, ya que la cantidad de ramificaciones posibles aumenta exponencialmente.

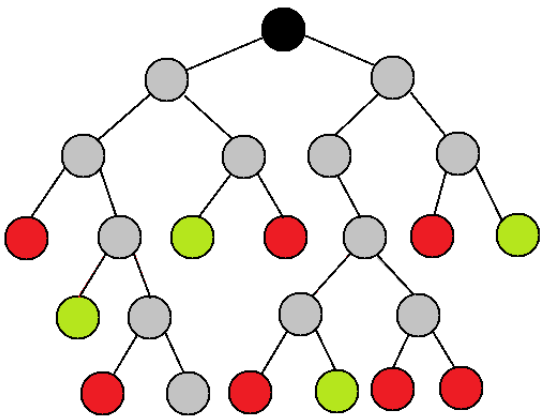

Figura 2. Algoritmo Branch and Bound.

Los métodos exactos son muy eficientes en pequeñas instancias de problemas difíciles (tamaños pequeños del problema). En el momento que las instancias crecen de tamaño, también lo hace de forma exponencial el número de combinaciones posibles en sus variables (espacio de soluciones) y, por tanto, la cantidad de cálculos necesarios para obtener la solución óptima, siendo imposible en muchos casos.

1.4.2. Técnicas de aproximación

Los métodos heurísticos han sido estudiados y evaluados como buenos candidatos para resolver problemas difíciles de optimización, obteniendo buenas soluciones (óptimas o cercanas al óptimo) utilizando un esfuerzo (tiempo) considerable. Sin embargo, estos métodos normalmente son totalmente dependientes del problema que se va a evaluar, requiriendo un coste adicional en su diseño, así como siendo únicamente válidos para dicho problema. Con la intención de solventar este problema, y conseguir un método óptimo a un nivel más general, aparecieron las denominadas Metaheurísticas (Glover 1977). Estás técnicas siguen la optimización por aproximación de las heurísticas, pero a nivel más abstracto, pudiendo adaptarlas a cualquier tipo de problema.

1.4.2.1. Metaheurísticas

A diferencia de los métodos exactos, las metaheurísticas son capaces de encontrar soluciones satisfactorias a problemas de gran tamaño en un tiempo razonable. Esto no quiere decir que sea la solución óptima, pero sí que se aproxime a ella de manera eficiente, dado que los métodos exactos, para estos tamaños grandes, no son capaces de resolverlos.

Los problemas de optimización se encuentran en todos los ámbitos, haciendo que estos sean complejos y costosos de resolver. Algunas de las áreas donde encontramos la aplicación de estas Metaheurísticas son:

- Ingeniera de diseño, diseño VLSI, aerodinámica, dinámica de fluidos, telecomunicaciones y robótica.
- Machine learning y data mining en bioinformática y biología computacional.
- Modelado de sistemas, simulación e identificación química, física y procesado de control e imagen.

Inicialmente, el concepto de heurística, con el cual se optimizaban problemas específicos (dado que estas técnicas son dependientes del problema), fue introducido por (Polya and Conway, 1957) A continuación, aparecieron técnicas como el algoritmo simplex (Dantzig, 1947), el cual puede ser visto como un algoritmo de búsqueda local para problemas de programación lineal. Otras técnicas, como los denominados Greedy Heuristics (Edmonds, 1971) o algoritmos de colonias (Colorni et al., 1992) fueron desarrollados posteriormente, siendo aplicados a numerosos problemas de optimización. En la actualidad, se encuentran numerosos algoritmos metaheurísticos, cada uno de ellos con sus características particulares, como pueden ser los algoritmos genéticos introducidos por Holland et al., 1962 y Holland, 1992, los algoritmos de enjambres de partículas (Kennedy and Eberhart, 1942), algoritmos de búsqueda tabú (Glover, 1986) o los de la búsqueda del mejor vecino (Hansen and Mladenovic, 1999).

De esta forma, dado el abanico tan grande de posibilidades, se debe definir un criterio base para el diseño de estas técnicas. Para ello, se debe tener muy en cuenta dos parámetros: la exploración del espacio de

soluciones (diversificación) y la profundidad de búsqueda de la mejor solución (intensificación). De esta forma, a la hora de diseñar un algoritmo metaheurístico, se debe decidir que es más interesante, explorar un mayor número de soluciones, ampliando el espacio de soluciones, o determinar únicamente ciertas zonas del espacio y buscar muy exhaustivamente dentro de ellas. Otros criterios para la clasificación de estas técnicas, son los siguientes:

- **Inspiración**: Muchos algoritmos metaheurísticos están inspirados en procesos de la naturaleza (algoritmos evolutivos, algoritmos de colonias, de enjambres de partículas). Otros únicamente se basan en procesos artificiales de búsqueda y reconocimiento.
- **Uso de memoria**: Este criterio separa aquellas metaheurísticas que necesitan un conocimiento de las soluciones previas, y por tanto tiene que almacenar toda la información recogida de forma on-line, de otras que independizan más las soluciones encontradas, y únicamente hacen esfuerzos en cada una de ellas, sin necesidad de conocer las otras zonas exploradas.
- **Deterministas**: Existen metaheurísticas que toman decisiones deterministas durante la optimización del problema, mientras que otras aplican reglas aleatorias durante algún proceso interno. De esta forma, en las Metaheurísticas deterministas, una solución inicial siempre obtiene la misma solución final, mientras que las no deterministas pueden finalizar en diferentes soluciones partiendo de la misma base inicial.
- **Población**: Ciertas metaheurísticas pueden estar basadas en una única solución (*local search*, por ejemplo), manipulándola y transformándola durante el proceso de optimización, mientras que otras utilizan una población de soluciones para elaborar la

optimización. Esta población de soluciones evoluciona a medida que la metaheurística trabaja, explorando y diversificando el espacio de soluciones.

El diseño y la implementación de una metaheurística, dependerá en gran medida de la naturaleza del problema a evaluar. De esta forma, se decidirá la población inicial del algoritmo, así como si es más interesante intensificar pocas soluciones, dado que todas se mueven en un espacio reducido, o explorar de forma amplia el espacio de soluciones, ya que este es mucho mayor y es necesario diversificar.

1.4.2.2. Metaheurísticas basadas en población

El presente trabajo se centra principalmente en las Metaheurísticas basadas en poblaciones. En estos algoritmos, una población inicial de soluciones es generada, normalmente de forma aleatoria (pudiendo incorporar alguna funcionalidad de reajuste), e incluida en un proceso iterativo, el cual incluye ciertas funcionalidades que mejoran y optimizan las soluciones creadas inicialmente. La población sufre constantes cambios, donde algunos individuos son reemplazados por otros mejores (Figura 3).

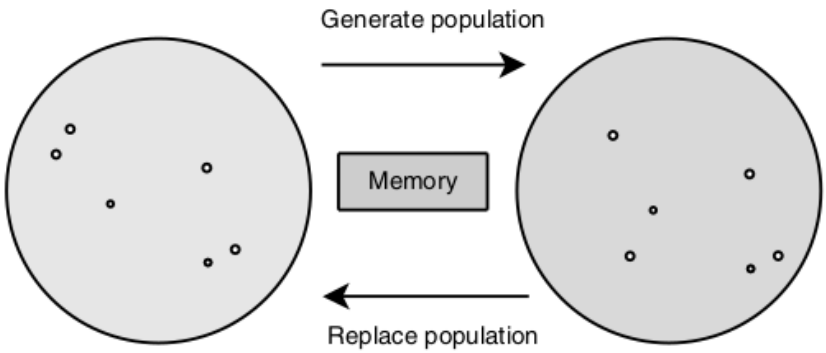

Figura 3. Esquema principal de Metaheurísticas basadas en poblaciones.

La mayoría de los algoritmos basados en poblaciones, están basados en procesos de la naturaleza, donde las especies (soluciones) evolucionan y mutan a través del tiempo, adaptando nuevas características que las hace mejores a sus predecesores. De esta forma, podemos explicar estos algoritmos a un alto nivel, separando sus funcionalidades en dos grupos: fase de generación y fase de reemplazo. Durante la fase de generación, una nueva población de soluciones es creada, intentando expandirse lo máximo posible a lo largo del espacio de soluciones. A continuación, durante la fase de reemplazo, ciertas soluciones son seleccionadas y tratadas. Esta segunda fase se repite hasta que se cumple cierto criterio de parada. El algoritmo 1 muestra esta explicación a alto nivel.

Algoritmo 1: Esquema a alto nivel de un metaheurístico basado en poblaciones.

$P = P_{0;}$ /* Generación de la población inicial */
$t = 0;$
Repetir
    Generar (P'_t) /* Generación de una nueva población */
    P_{t+1} = Selección $(P'_t \cup P_t)$ /* Nueva población combinada */
    $t = t + 1;$
Hasta cumplir criterio de parada
Salida: Mejor solucion(es) encontrada(s).

Independientemente del metaheurístico basado en población utilizado, todos coinciden en que la inicialización del algoritmo (población inicial) es una parte crucial, que afecta en gran medida a la eficiencia y la efectividad del metaheurístico. De esta forma, se debe encontrar una población inicial lo suficientemente diversificada, con la intención de no converger de forma prematura a un óptimo local. Existen numerosas

formas de generar la población inicial, siendo las más comunes la generación pseudo-aleatoria, la diversificación por regiones, o las heurísticas.

1.4.2.3. Algoritmos Evolutivos

Los algoritmos evolutivos (EA) son los metaheurísticos basados en poblaciones más comunes en la literatura. Estos algoritmos son ampliamente utilizados en problemas reales y complejos de muchos ámbitos (optimización continua o combinatoria, modelado e identificación de sistemas, diseño de ingeniería, machine learning, etc). Estos metaheurísticos representan una clase de algoritmos de tipo iterativo, en donde se simula la evolución natural de especial tal y como la conocemos (Figura 4).

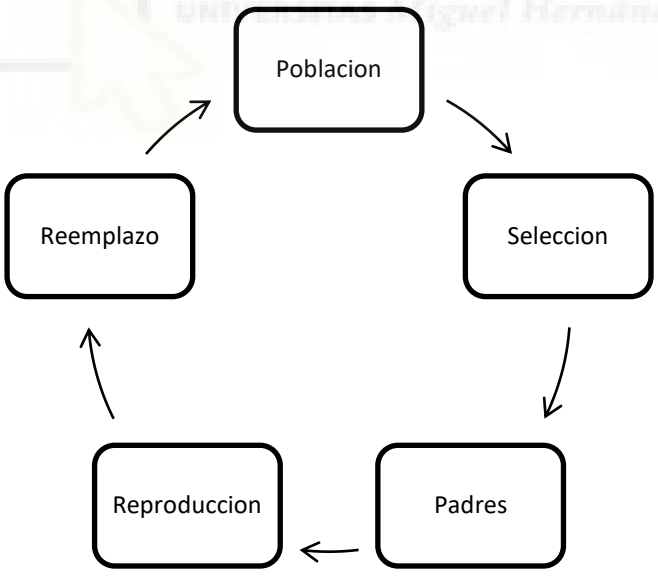

Figura 4. Generación de algoritmos evolutivos.

Por tanto, este algoritmo está basado en la evolución de una población de individuos, normalmente generada de forma aleatoria. Cada individuo representa una solución factible al problema planteado,

codificado a través de los valores internos de sus variables y restricciones, así como de la función objetivo evaluada (fitness). De esta forma, en función del problema a tratar, pueden existir conjuntos de soluciones válidas/factibles (cumplen con todas las restricciones del modelo) o no válidas (incumplen alguna restricción). Para todas aquellas soluciones válidas, la calidad de la solución viene definida por el valor obtenido en la función objetivo.

Al ser un algoritmo iterativo, cada iteración selecciona ciertos individuos de la población. Algunos EA (Evolutionary Algorithm) aplican probabilidades a la hora de seleccionar individuos, dando mayor peso a aquellas soluciones con mejor valor de fitness. Estos individuos seleccionados son utilizados para formar parte de las funciones de reproducción, aquellas que se encargan de generar nuevas soluciones a partir de las seleccionadas previamente, utilizando técnicas como el cruce o la mutación. Finalmente, las nuevas soluciones son incluidas o descartadas en la población, en función de su valor de fitness. Este proceso completo representa una generación, y estas generaciones se repiten hasta que se cumpla cierta condición de fin, la cual puede ser estática (fijada al inicio del algoritmo) o dinámica (adaptándose a las necesidades del proceso). El algoritmo 2 muestra el funcionamiento de dichos algoritmos evolutivos.

Algoritmo 2: Esquema estándar de un algoritmo evolutivo

Generar (P_0); /* Generación de la población inicial */
Repetir
Ordenar (P_0) /* Organizar individuos por fitness */
P_1 = Selección (P_0);
P_2 = Reproducir (P_1);
P_3 = Evaluar/Mejorar (P_2);
P_0 = Reemplazar (P_3);

Hasta cumplir criterio de parada
Salida: Mejor solucion(es) encontrada(s).

Dentro de esta clasificación de Metaheurísticas, los algoritmos genéticos son los más famosos y ampliamente utilizados dentro de los EA. Estos algoritmos fueron desarrollados en primera instancia para entender ciertos procesos adaptativos de la naturaleza (Holland, 1992). Posteriormente, fueron utilizados y desarrollados para ser aplicados en problemas de optimización y machine learning (Goldberg, 1989). Estos algoritmos aplican un operador de cruce a dos soluciones previamente seleccionadas, y un operador de mutación que aleatoriamente modifica cierto individuo para fomentar la diversificación de soluciones (ampliando el espacio de soluciones por fuerza bruta), ambos operadores utilizados dentro de la función de reproducción.

Los componentes principales para el diseño de este tipo de algoritmos, los cuales han sido estudiados y desarrollados a lo largo del trabajo son:

1. **Representación:** El termino *cromosoma* está extendido en la literatura, y en la comunidad que trabaja con estos algoritmos, para referirse a la solución codificada (conjunto de variables y función objetivo).
2. **Población inicial:** Como previamente se ha indicado, la población inicial es un punto crítico del algoritmo. Las soluciones iniciales suelen venir de procesos pseudoaleatorios, o de heurísticos diseñados expresamente para el problema evaluado.
3. **Función objetivo:** El termino *fitness* hace referencia al valor de la función objetivo, valor a través del cual se clasifican los *cromosomas* y se determina la calidad de estos.

4. **Selección:** Este apartado determina que *cromosomas* son seleccionados para ser utilizados en procesos posteriores, y por qué ciertos individuos deben sobrevivir a sus predecesores.
5. **Reproducción:** Consiste en el diseño de los operadores de cruce y mutación, a través de los cuales se generarán los nuevos individuos de la población.
6. **Criterio de parada:** Un operador importante es cómo se va a decidir que el algoritmo finalice el proceso iterativo.

1.4.2.4. Metaheurísticas Hibridas

Durante los últimos años, el diseño de algoritmos metaheurísticos ha alcanzado su límite, de forma que todas las estructuras posibles utilizando los criterios base para el diseño de estos algoritmos son finitas. De esta forma, se han empezado a plantear formas de mejorar estos algoritmos, siendo una de ellas la combinación entre estos mismos algoritmos, o incluyendo entre sus funciones otras técnicas complementarias, como inteligencia artificial, análisis de datos o métodos exactos.

Este trabajo se centra en la utilización de metaheurísticas híbridas, de forma que los algoritmos evolutivos presentados previamente son combinados con métodos exactos. Estos últimos son integrados para la resolución de problemas lineales, previamente descompuestos del problema original.

Las metaheurísticas y los métodos exactos son estrategias de optimización complementarias en términos de calidad de la solución y tiempo de cómputo. Existen varias formas de clasificar estas

combinaciones (Talbi, 2009), en función de la utilización de cada uno de los algoritmos:

- **Low-Level Relay Hybrid:** En este conjunto de algoritmos, la metaheurística está incluida dentro del método exacto en forma de metaheurística basada en individuos o viceversa. De esta forma, el algoritmo interno es utilizado para mejorar la capacidad de búsqueda de soluciones óptimas de forma única e independiente.
- **Low-Level Teamwork Hybrids:** Este conjunto engloba todas aquellas metaheurísticas basadas en poblaciones, en donde alguno de las funciones de búsqueda es reemplazada por otro algoritmo. La cooperación entre ambos es importante pero no crucial, ya que cada uno de ellos puede trabajar de forma independiente, pese a que la solución conjunta de ambos mejore la calidad de la solución final.
- **High-Level Relay Hybrids**: En este caso, se pierde la independencia de los algoritmos, ya que se requiere una cooperación estricta entre los algoritmos combinados. En estos algoritmos híbridos, las técnicas incluidas son ejecutadas secuencialmente. En esta clase de algoritmos se incluyen los términos de preproceso y postproceso, en donde cierta información es compartida entre las dos familias de algoritmos para poder conseguir una solución válida (factible).

El algoritmo desarrollado en esta tesis se engloba dentro de los algoritmos High-Level Relay Hybrids. Más concretamente, el algoritmo híbrido desarrollado establece la metaheurística como algoritmo maestro, el cual comparte información con el método exacto para conseguir una solución completa del problema (Figura 5). Para esto, el problema principal es descompuesto, utilizando la naturaleza de las variables como criterio de descomposición.

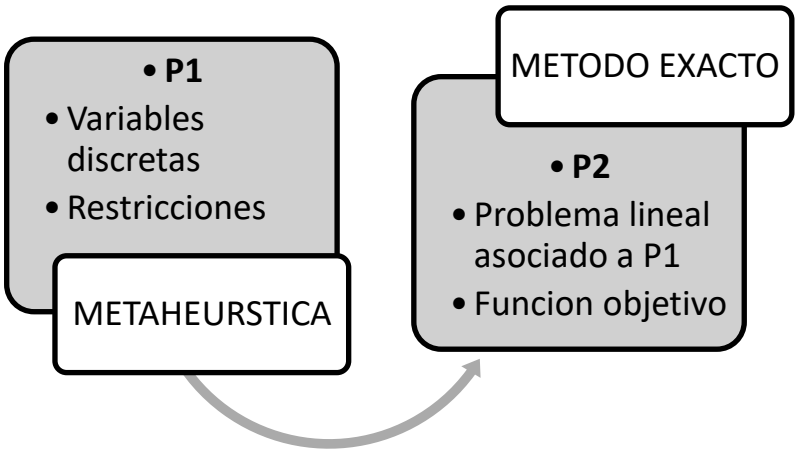

Figura 5. Algoritmo de tipo High-Level Relay Hybird.

La estrategia utilizada divide el problema en función de las variables, separando las variables de decisión y sus restricciones (set X) de las variables continuas (set Y). La metaheurística se encarga de fijar las variables del set X y el método exacto encuentra una solución al set Y utilizando la información enviada por la metaheurística. De esta forma, aplicando el algoritmo a un problema MILP, se puede resolver la parte discreta por medio de la metaheurística, explorando el espacio de soluciones con un estudio interno de las variables, y resolviendo el problema lineal asociado a cada una de las soluciones obtenidas mediante un método exacto. Este método híbrido, que combina metaheurísticas y métodos exactos, es denominado Matheurística (Boschetti et al (2009) y Blesa et al. (2009)). Este tipo de algoritmos está siendo cada vez más estudiado e implementado en problemas reales, ya que aprovechan las ventajas de dos potentes algoritmos (metaheurísticas y métodos exactos) en un solo marco conceptual, como podemos estudiar en Singh et al., 2013, Pinheiro et al., 2018 o Singh y Rossi, 2013.

1.4.2.5. Hiper-Metaheurísticas

Como hemos comentado, los métodos heurísticos para resolver problemas de optimización complejos son totalmente dependientes del problema a evaluar, siendo así muy limitados en su utilización. En un nivel más abstracto, también hemos estudiado las denominadas metaheurísticas, algoritmos de optimización independientes del problema, encargados de analizar y explorar el espacio de soluciones disponible para cada problema. De esta forma, una metaheurística puede ser aplicada a un gran número de problemas de diferente ámbito, sin importar (a priori) la naturaleza de sus variables o su función objetivo, En función del problema, se decidirá qué familia de metaheurísticas puede ser la más óptima, pero sin conocer, de forma exacta, cuál será la mejor metaheurística para cada problema. De esta forma, podemos encontrar ciertos automatismos a un mayor nivel de abstracción, capaces de decidir qué metaheurística se ajusta mejor a nuestro problema, llegando incluso a modificarla en tiempo de ejecución, para adaptarla de la mejor forma posible. Estas técnicas de automatización y diseño de algoritmos de búsqueda son denominadas hiper-metaheurísticas, situándose a un nivel superior de las denominadas metaheurísticas.

La idea principal es desarrollar algoritmos más versátiles y genéricos que muchas de las implementaciones de metodologías de búsqueda actuales.

Algunos de los enfoques más utilizados están basados en la configuración automática de los parámetros internos de los algoritmos evolutivos. Durante su ejecución, los algoritmos se adaptan en base a los conocimientos obtenidos por la generación de soluciones previas. Algunos ejemplos en la literatura los podemos encontrar en Eiben et al., 2007 o Rechenberg, 1989.

En esta tesis, la hiper-metaheurística que se ha desarrollado utilizando el principio previamente comentado (configuración automática de parámetros) pero no únicamente centrado en los algoritmos evolutivos. Además, la hiper-metaheurística que nos ocupa, está incluida dentro de los algoritmos de aprendizaje *online*. Una hiper-matheurística es un algoritmo de aprendizaje cuando utiliza algunos recursos como feedback durante el proceso de búsqueda. Según el feedback obtenido durante el aprendizaje, podemos distinguir entre el aprendizaje online y offline. En la hiper-matheurística del aprendizaje online, el aprendizaje tiene lugar mientras el algoritmo está resolviendo una instancia de un problema, mientras que en la hiper-matheurística del aprendizaje offline, la idea es reunir conocimiento en forma de reglas o programas, a partir de un conjunto de instancias de entrenamiento, que con suerte se generalizará para resolver otras instancias.

En la hiper-metaheurística desarrollada, inicialmente se deben definir los parámetros a estudiar durante el estudio. En nuestro caso, dado que ciertas metaheurísticas comparten funcionalidades semejantes, hemos desarrollado un esquema parametrizado (Almeida et al., 2013) capaz de generar, en función de los valores de los parámetros, Metaheurísticas de diversos tipos, como algoritmos evolutivos, búsquedas dispersas (Scatter Search) o Greedy Randomized Adaptive Search Procedure (GRASP). Estos tres algoritmos mencionados, comparten ciertas funcionalidades, pudiendo generar nuevas metaheurísticas en puntos intermedios de las iniciales, combinando sus herramientas internas. En la Figura X se muestra el espacio de soluciones disponible para la hiper-metaheurística desarrollada.

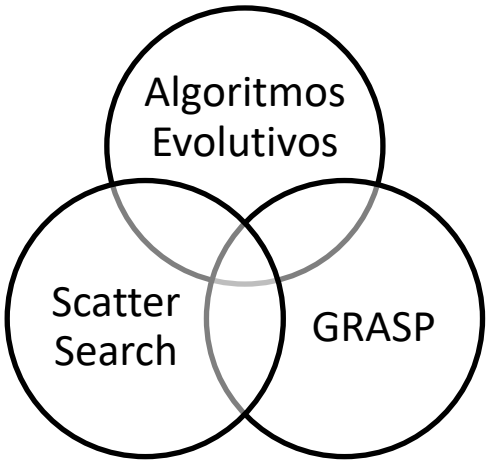

Figura 6. Espacio de soluciones metaheurísticas para la hiper-metaheurística desarrollada en la tesis.

1.4.3. Paralelismo

Dada la complejidad de los problemas planteados en esta tesis, y del gran número de problemas que se deben resolver, se han estudiado y desarrollado ciertas técnicas de optimización, ya no basadas en la calidad de la solución, sino en el tiempo de ejecución requerido para obtener dichas soluciones.

Al mismo tiempo que se desea optimizar la calidad de las soluciones (fitness), el tiempo también es un recurso importante. Minimizar el tiempo de cómputo mejora el rendimiento del algoritmo, y permite obtener soluciones a problemas muy complejos. Para esto, se han utilizado técnicas de computación paralela, una forma de cómputo en la que se ejecutan, de forma simultánea, varias instrucciones encargadas de resolver un problema común. Cualquier problema transformado a lenguaje informático está organizado en un flujo de trabajo secuencial, en donde las instrucciones siguen un orden de ejecución establecido. Normalmente, hay partes de este

flujo que resultan independientes de otras, permitiendo ser ejecutadas de forma simultánea sin que interfieran entre ellas.

La dificultad de la computación paralela radica en el estudio de estas partes independientes. La parte más crítica de estos algoritmos radica en dividir el flujo de trabajo principal en diferentes secciones, intentando ejecutar el mayor número posible de estas al mismo tiempo, unificando coherentemente las soluciones de cada una de las secciones para conseguir una solución al problema dado. La Figura 7 muestra este tipo de descomposiciones y recombinaciones.

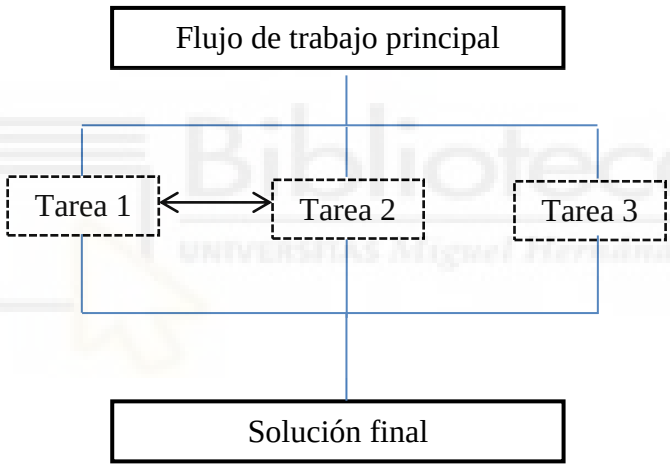

Figura 7. Esquema básico de paralización por tareas.

Dependiendo de la naturaleza de las tareas a realizar, estas múltiples tareas deben intercambiar información entre sí en algún punto, comunicándose de alguna manera. En función de la estrategia de comunicación, los algoritmos paralelos pueden ser clasificados de la siguiente forma:

- **Memoria compartida:** Las múltiples tareas a evaluar son distribuidas en hilos (threads), ejecutándose en núcleos de procesamiento donde todos ellos tienen acceso a la misma área de

memoria. De esta forma, todas las tareas programadas acceden a los mismos registros de memoria, no necesitando comunicación directa entre ellos para compartir información. Además, en este paradigma de programación, se definen ciertas regiones de paralelismo dentro de un flujo de trabajo secuencial. Uno de los estándares de programación paralela más extendido en la comunidad, y el utilizado durante este trabajo, ha sido el lenguaje OpenMP (Figura 8).

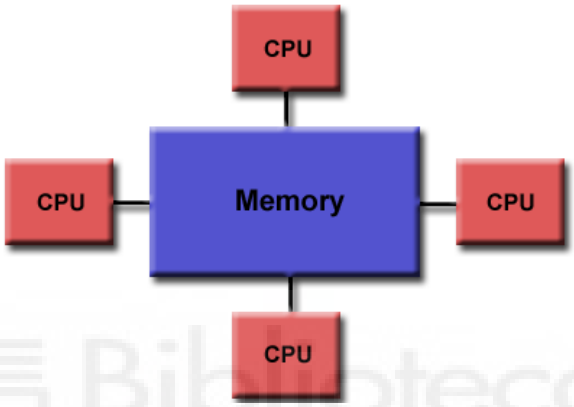

Figura 8. Paradigma de paralelismo basado en memoria compartida (OpenMP).

- **Memoria distribuida:** Las tareas (o procesos) participantes en este paradigma no comparten recursos de memoria del sistema. Por tanto, todo lo ejecutado entre ellos es completamente independiente, salvo si existe una comunicación directa entre ellos. De esta forma, en este paradigma existe un medio para intercambiar datos entre los procesos mediante paso de mensajes. Para ello, se realizan peticiones individuales entre los núcleos de procesamiento a la espera de recibir respuestas con los datos solicitados, usando habitualmente (al igual que en esta tesis) *Message Passing Interface* o MPI (Figura 9). Por tanto, para este tipo de programación paralela es necesario que los procesos estén conectados entre a través de una red.

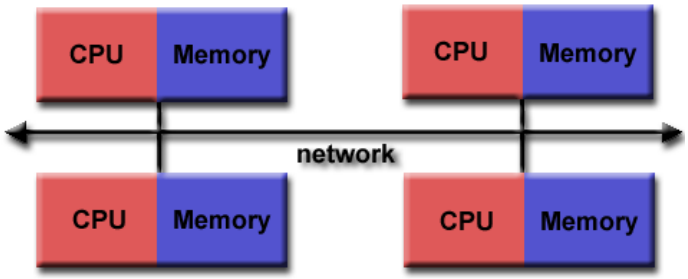

Figura 9. Paradigma de paralelismo basado en memoria distribuida (MPI).

Ambos paradigmas presentados, pueden ser combinados en algoritmos híbridos en donde ciertas funcionalidades del código trabajen en memoria compartida porque así lo requieren, mientras que otros procesos más independientes pueden hacerlo en memoria distribuida y compartir su información únicamente en ciertos puntos.

En esta tesis, al tener como objetivo resolver un modelo DEA específico ligado a la determinación de la distancia más corta, en donde múltiples problemas de Programación Matemática tienen que ser resueltos (tantos como número de DMUs), se aprovecha el paradigma de memoria distribuida para esta tarea. Al ser problemas independientes, no se requiere acceder a la misma memoria para cada uno de los problemas, sino que simplemente con una conexión (paso de mensajes) final se recopilan las soluciones de todos los problemas, pudiendo así ver el resumen de todas las soluciones en un mismo archivo o memoria (Martin et al., 2018).

Por otro lado, para optimizar cada uno de los problemas de forma interna, se incluye el paradigma de memoria compartida para agilizar cada una de las funcionales del algoritmo matheurístico desarrollado. El estudio del rendimiento en paralelo de los algoritmos exactos estará basado en cómo se comportan al ejecutarse múltiples instancias simultáneas, tanto en memoria compartida como distribuida. Estos algoritmos incluyen paralelización interna que ayuda a la resolución de problemas complejos,

pero dada la naturaleza de nuestro problema, y el desarrollo de la descomposición del modelo, no es objetivo de estudio (el tiempo que tardan los algoritmos exactos en revolver los modelos LP descompuestos son muy bajos y se estudian en Martin et al, 2020. Algunos de los algoritmos exactos más comunes en la literatura, y que van a ser utilizados en el desarrollo del método matheurístico son CPLEX (Cplex, I.I (2009)) y GUROBI (Gurobi Optimization, I. (2014)).

Esta combinación hibrida de paradigmas paralelos (Katouda y Nakajima, 2013) pretende optimizar los recursos del sistema (Drosinos y Koziris, 2004) de forma que se aprovechen todos los procesos posibles de la forma más coherente posible, pudiendo separar regiones de memoria para todas aquellas tareas independientes para resolver un mayor número de problemas simultáneamente (Wang y Jandhyala, 2008). La Figura 10 muestra el esquema de un algoritmo paralelo híbrido, donde los dos paradigmas estudiados trabajan conjuntamente.

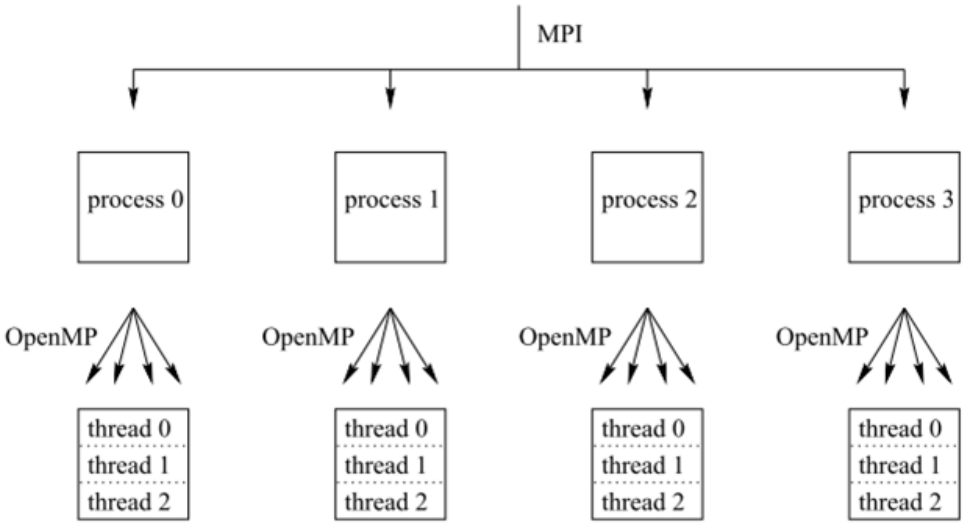

Figura 10. Esquema de paralelización híbrido MPI + OpenMP.

CAPÍTULO 2

Resumen de las aportaciones

2.1. Using non-radial DEA to assess school efficiency in a cross-country perspective: An empirical analysis of OECD countries.

2.1.1. Introducción

En este trabajo utilizamos datos de los países de la OCDE (Organización para la Cooperación y el Desarrollo Económicos) que participan en PISA (*Programme for International Student Assessment*) 2012 para evaluar la eficiencia de las escuelas en un marco internacional. En el análisis, y en contraste con las aplicaciones anteriores, consideramos que las escuelas podrían concentrar sus esfuerzos en mejorar los resultados en una dimensión del producto educativo en mayor medida que en la otra. Para hacer esto, confiamos en medidas de rendimiento no radiales de eficiencia y la estimación de una función de producción educativa basada en técnicas de Análisis Envolvente de Datos (DEA). Específicamente, las

medidas no radiales de DEA permiten identificar diferentes niveles de ineficiencia para cada *output* considerado (lectura y matemáticas).

La participación de la mayoría de las naciones en estudios comparativos internacionales a gran escala en educación ha brindado a los investigadores bases de datos internacionales ricas y extensas que pueden usarse para evaluar el desempeño y la efectividad de los sistemas educativos. Como resultado, los estudios de educación comparativa se han vuelto cada vez más populares en las ciencias de la educación hoy en día (Gustafsson, 2008), ya que los investigadores pueden ver el mundo entero como un laboratorio natural para ver las múltiples formas en que los factores sociales, las políticas y las prácticas educativas pueden variar en diferentes países (Bray y Thomas, 1995).

La mayoría de los estudios que adoptan una perspectiva entre países se sitúan dentro del campo de la investigación de la efectividad educativa, que explora los principales determinantes del logro educativo utilizando un enfoque econométrico para estimar una ecuación en forma de función de producción educativa (ver Hanushek, 1979; Todd y Wolpin, 2003; Creemers y Kyriakides 2008). Este capítulo de la literatura investiga cómo las entradas están estadísticamente relacionadas con las salidas. Sin embargo, también debe considerarse la existencia potencial de un nivel inesperado de ineficiencia en el desempeño de los estudiantes, las escuelas o los sistemas educativos (Levin, 1974).

Hemos explorado la existencia potencial de estas compensaciones entre lectura y matemáticas en una evaluación del desempeño de las escuelas de todos los países de la OCDE que participan en PISA 2012 adoptando un

marco internacional. Para este fin, confiamos en medidas de eficiencia no radiales, que no requieren un aumento proporcional en todas las salidas consideradas, por lo tanto, podemos calcular diferentes proyecciones en la frontera para cada salida incluida en la función de producción. Esta posibilidad nos permite detectar si algunas escuelas pueden ser más eficientes en promover la competencia de sus estudiantes en lectura, mientras que otras escuelas podrían ser más propensas a mejorar los resultados en matemáticas. Del mismo modo, el enfoque propuesto también puede ser útil para explorar otras posibles compensaciones entre los resultados educativos, como la relación entre las habilidades cognitivas y no cognitivas (Cordero et al., 2016) o la desigualdad educativa y el rendimiento promedio (Thieme et al., 2012; Giménez et al., 2017).

Para la resolución del modelo planteado, se ha utilizado una metodología reciente, basada en la aplicación de la versión "output-oriented" de la medida Russell, que determina los objetivos más cercanos a través de a determinación de la distancia (mínima) a la frontera Pareto-eficiente en DEA, basada en un modelo de Programación Lineal Bi-nivel (Aparicio et al. , 2016). Además, con el objetivo de satisfacer la monotonicidad, se propone una corrección de este modelo basada en Ando et al. (2012).

2.1.2.Metodología

Cualquier medida en DEA que no adopte reducciones equi-proporcionales de entradas o salidas es considerada "no radial". De hecho, un inconveniente bien conocido de las medidas radiales es la arbitrariedad en la imposición de objetivos en la frontera. En particular, en el contexto de la educación, algunas unidades (escuelas, por ejemplo) podrían verse

tentadas, directa o indirectamente, a actualizar alguna dimensión específica, como la ciencia o las matemáticas, debido, por ejemplo, a las características culturales y tradiciones inherentes al país donde están ubicados geográficamente. Además, desde el punto de vista de la DMU (la escuela en nuestro contexto), se podría estar interesado en determinar la forma más fácil de ser clasificado como técnicamente eficiente.

En cuanto a la existencia de modelos orientados no radiales, DEA otorga a los profesionales una caja de herramientas llena de posibilidades. El primer enfoque a este respecto se debió a Färe et al. (1985), quienes introdujeron las medidas de entrada y salida de Russell de eficiencia técnica. Después de eso, se definieron otras medidas orientadas y no radiales que buscaban más flexibilidad que la proporcionada por las medidas radiales.

Con respecto a las medidas de Russell, queremos resaltar que hay dos paradigmas claramente diferentes para su determinación. Por un lado, tenemos el enfoque tradicional, que está asociado con la identificación de objetivos exigentes. Los objetivos son específicamente las coordenadas del punto de proyección en la frontera de la tecnología y, por lo tanto, representan niveles de operación que harían que la DMU evaluada funcionara de manera eficiente. Esta primera filosofía es seguida por la definición original de las medidas de entrada y salida de Russell (Färe et al., 1985), donde el esfuerzo técnico total requerido por una DMU para ser técnicamente eficiente es maximizado en lugar de minimizado. Una propuesta reciente (Aparicio et al., 2016) ha sugerido determinar los objetivos eficientes más cercanos a través de las medidas orientadas de Russell, minimizando, en lugar de maximizar, el esfuerzo técnico correspondiente para llegar a la frontera. Este segundo enfoque sigue una línea de investigación bien conocida en la literatura DEA relacionada con

la determinación de la menor distancia a la frontera eficiente, la identificación de objetivos más cercanos y la aplicación del Principio de Mínima Acción en Física.

Además, implementar en la práctica el enfoque basado en la determinación de objetivos más cercanos no es fácil desde un punto de vista computacional (Briec, 1997). Esta dificultad es consecuencia de la complejidad de determinar la menor distancia a la frontera de una tecnología DEA (un conjunto poliédrico) desde un punto interior (DMU ineficiente). Por esta razón, Aparicio et al. (2016) introducen una nueva metodología para implementar este enfoque en el contexto de las medidas orientadas. Su método se basa en la Programación Lineal Bi-nivel (LBP). La formulación general de un problema de programación lineal de dos niveles (BLP) es la siguiente:

$$
\begin{aligned}
&\underset{z,t}{Min} && c_1 z + d_1 t \\
&s.t. && \\
& && A_1 z + B_1 t \leq b_1, \\
& && \underset{t}{Min} c_2 z + d_2 t \\
& && s.t. \\
& && A_2 z + B_2 t \leq b_2, \\
& && z \geq 0, t \geq 0
\end{aligned}
\qquad (15)
$$

El modelo (15) consta de dos subproblemas. Por un lado, el problema de decisión del nivel superior y, por otro lado, el problema del nivel inferior, que aparece como una restricción en el modelo (15). Ambos problemas están conectados de forma dependiente, en donde ambos se ven afectados por el otro subproblema.

Incluso para un problema de programación de dos niveles donde todas las funciones sean lineales, como en (15), el modelo a resolver es no convexo y NP-duro. Esta complejidad es la razón por la cual se han

propuesto muchas técnicas diferentes en la literatura para estudiar los aspectos computacionales de los problemas de programación de dos niveles. Una posibilidad es transformar el problema original en un único problema de optimización mediante la aplicación de las conocidas condiciones de optimización de Karush-Kuhn-Tucker (KKT) del problema de nivel inferior (Shi et al., 2005a).

Aparicio et al. (2016) propuso el siguiente modelo para calcular la medida de salida de Russell basada en la filosofía de menor distancia para el DMU_0:

$$
\begin{aligned}
&\underset{\phi,\lambda,s^+,\gamma}{Min} && \tfrac{1}{s}\textstyle\sum_{r=1}^{s}\phi_r && && (16.1)\\
&s.t. && && && \\
& && \textstyle\sum_{j\in E_{VRS}}\lambda_j x_{ij} \le x_{i0}, && i=1,\dots,m && (16.2)\\
& && \textstyle\sum_{j\in E_{VRS}}\lambda_j y_{rj} \ge \phi_r y_{r0}, && r=1,\dots,s && (16.3)\\
& && \textstyle\sum_{j\in E_{VRS}}\lambda_j = 1, && && (16.4)\\
& && \textstyle\sum_{r=1}^{s} s_r^+ = 0, && && (16.5)\\
& && \underset{s^+,\gamma}{Max}\textstyle\sum_{r=1}^{s} s_r^+ && && (16.6)\\
& && s.t. && && \\
& && \textstyle\sum_{j\in E_{VRS}}\gamma_j x_{ij} \le x_{i0}, && i=1,\dots,m && (16.7)\\
& && \textstyle\sum_{j\in E_{VRS}}\gamma_j y_{rj} = \phi_r y_{r0} + s_r^+, && r=1,\dots,s && (16.8)\\
& && \textstyle\sum_{j\in E_{VRS}}\gamma_j = 1, && && (16.9)\\
& && \phi_r \ge 1, \lambda_j, s_r^+, \gamma_j \ge 0, && \forall r,j && (16.10)
\end{aligned}
\qquad (16)
$$

Con el objetivo de garantizar una monotonicidad débil cuando la medida anterior se calcula sobre la frontera Pareto-eficiente, necesitamos modificar (16). Para hacer eso, debemos incluir en las restricciones un punto genérico tal que esté dominado por la unidad evaluada en el sentido de Pareto. Entonces, el problema de programación bi-nivel correspondiente para evaluar la DMU_0 sería el siguiente.

$$
\begin{aligned}
&\underset{\phi,\lambda,y,s^+,\gamma}{Min} && 1+\frac{1}{s}\sum_{r=1}^{s}\frac{t_r}{y_{r0}} && && (17.1)\\
&s.t. && \textstyle\sum_{j\in E_{VRS}}\lambda_j x_{ij}\le x_{i0}, && i=1,\dots,m && (17.2)\\
& && \textstyle\sum_{j\in E_{VRS}}\lambda_j y_{rj}\ge y_r+t_r, && r=1,\dots,s && (17.3)\\
& && y_r\le y_{r0}, && r=1,\dots,s && (17.4)\\
& && \textstyle\sum_{j\in E_{VRS}}\lambda_j=1, && && (17.5)\\
& && \textstyle\sum_{r=1}^{s}s_r^+=0, && && (17.6)\\
& && \underset{s^+,\gamma}{Max}\ \textstyle\sum_{r=1}^{s}s_r^+ && && (17.7)\\
& && s.t. && && \\
& && \textstyle\sum_{j\in E_{VRS}}\gamma_j x_{ij}\le x_{i0}, && i=1,\dots,m && (17.8)\\
& && \textstyle\sum_{j\in E_{VRS}}\gamma_j y_{rj}=y_r+t_r+s_r^+, && r=1,\dots,s && (17.9)\\
& && \textstyle\sum_{j\in E_{VRS}}\gamma_j=1, && && (17.10)\\
& && t_r,\lambda_j,y_r,s_r^+,\gamma_j\ge 0, && \forall r,j && (17.11)
\end{aligned}
\qquad (17)
$$

A continuación se procede a transformar como un modelo de un nivel, sustituyendo el problema de nivel inferior por sus condiciones de optimización de Karush-Kuhn-Tucker (KKT) correspondientes (Shi et al., 2005). En consecuencia, (17.7) - (17.10) debe sustituirse por (18.1) - (18.9).

$$
\begin{aligned}
&\textstyle\sum_{j\in E_{VRS}}\gamma_j x_{ij}+l_i=x_{i0}, && i=1,\dots,m && (18.1)\\
&\textstyle\sum_{j\in E_{VRS}}\gamma_j y_{rj}=y_r+t_r+s_r^+, && r=1,\dots,s && (18.2)\\
&\textstyle\sum_{j\in E_{VRS}}\gamma_j=1, && && (18.3)\\
&-\textstyle\sum_{i=1}^{m}\eta_i x_{ij}+\sum_{r=1}^{s}\mu_r y_{rj}+\psi\ +\tau_j=0, && j\in E_{VRS} && (18.4)\\
&-\eta_i+e_i=0, && i=1,\dots,m && (18.5)\\
&\mu_r\ge 1, && r=1,\dots,s && (18.6)\\
&\gamma_j\tau_j=0, && j\in E_{VRS} && (18.7)\\
&l_i e_i=0, && i=1,\dots,m && (18.8)\\
&l_i,\eta_i,\mu_r,\tau_j,e_i\ge 0, && \forall i,j && (18.9)
\end{aligned}
\qquad (18)
$$

2.1.3.Datos y variables

Utilizamos datos comparativos sobre las escuelas que operan en los 34 países de la OCDE que participan en la encuesta PISA (Programa para la Evaluación Internacional de Estudiantes) de 2012 de la OCDE. La encuesta se realiza cada tres años, a partir de 2000. Se selecciona un mínimo de 150 escuelas en cada país. Posteriormente, 35 estudiantes de 15 años son seleccionados al azar de cada escuela para participar en la encuesta. Los datos se recopilaron entre marzo y mayo de 2012 para los países del hemisferio norte y de mayo a agosto de 2012 para los países del hemisferio sur.

Nuestro conjunto de datos final comprende un número total de 11,767 escuelas distribuidas entre países como se informa en la Tabla 1. Como se explicó anteriormente, el número mínimo de escuelas participantes en cada país debe ser 150, aunque en nuestra muestra tenemos el caso excepcional de Luxemburgo, donde hay Son solo 42 escuelas participantes. Asimismo, en varios países la muestra es muy grande debido a la existencia de muestras representativas para diferentes regiones dentro del país (por ejemplo, Australia, Canadá, Italia, México, España).

Country	Schools	Country	Schools
Australia	775	Japan	191
Austria	191	Korea	156
Belgium	287	Luxembourg	42
Canada	885	Mexico	1471
Chile	221	Netherlands	179
Czech Republic	297	New Zealand	177
Denmark	341	Norway	197
Estonia	206	Poland	184
Finland	311	Portugal	195
France	226	Slovak Republic	231
Germany	230	Slovenia	338
Greece	188	Spain	902
Hungary	204	Sweden	209
Iceland	134	Switzerland	411
Ireland	183	Turkey	170
Israel	172	United Kingdom	507
Italy	1194	USA	162
TOTAL		11,767	

Tabla 1. Número de escuelas en cada país.

Las variables de salida están representadas por las calificaciones promedio obtenidas por los estudiantes que pertenecen a la misma escuela en las dos competencias más relevantes: lectura y matemáticas. La selección de entradas es una decisión difícil, ya que en el conjunto de datos hay una extensa lista de indicadores potenciales que pueden considerarse. En nuestro estudio empírico, utilizamos la inversa de la relación alumno-docente, es decir, el número de docentes por (cien) alumnos (TEACHERS) y un índice que representa la calidad de los recursos escolares (SCMATEDU) creado por los analistas de PISA a partir de las respuestas dadas por directores de las escuelas con respecto a varios aspectos (computadoras, software educativo, calculadoras, libros, recursos audiovisuales o equipo de laboratorio). Además, también consideramos el estado socioeconómico promedio de los estudiantes en la escuela (ESCS) como un aporte adicional. ESCS representa el Estado Económico, Social y

Cultural, y proporciona una medida de los antecedentes familiares que incluye los niveles más altos de ocupación de los padres, recursos educativos y posesiones culturales en el hogar. Dado que los valores originales de SCMATEDU y ESCS presentaron valores positivos y negativos, todos ellos fueron reescalados para mostrar valores positivos.

Finalmente, también hemos seleccionado algunas variables contextuales para explorar si el desempeño de las unidades evaluadas puede verse afectado por el entorno educativo o el tipo de gestión escolar.

La Tabla 2 muestra las estadísticas descriptivas para todas estas variables (salidas, entradas y factores contextuales).

Variable	Type	Mean	Std. dev.	Min	Max
READING	Output	480.15	68.49	98.23	782.37
MATHS	Output	482.00	70.06	158.39	734.68
ESCS	Input	4.27	0.76	0.01	6.09
SCMATEDU	Input	3.59	1.04	0.008	5.576
TEACHERS	Input	9.76	16.01	0.098	1075.27
School factors					
PRIVATE	Contextual	0.169	0.374	0	1
RURAL	Contextual	0.328	0.469	0	1
PCGIRLS	Contextual	0.479	0.179	0	1
REPEATERS	Contextual	0.160	0.230	0	1
COMPETITION	Contextual	0.725	0.447	0	1
MATHHOMEWORK	Contextual	0.391	0.150	0	1
MATHEXAMS	Contextual	0.440	0.142	0	1
MATHCLASSES	Contextual	0.519	0.134	0	1
ENJOYMATHS	Contextual	0.261	0.136	0	1
READPLEASURE	Contextual	0.323	0.072	0.207	0.582

Tabla 2. Estadísticas descriptivas de las variables incluidas en el análisis.

2.1.4.Resultados

La metodología desarrollada nos permite determinar los puntos de proyección en la frontera para cada salida sin suponer que deben ser proporcionales. Sin embargo, inicialmente los presentamos en forma de las medidas tradicionales no radiales de Russell, es decir, como el promedio de las tasas proporcionales de expansión de las salidas (*outputs*), de modo que puedan ser más fáciles de interpretar y comparar con otros estudios empíricos que usan medidas radiales tradicionales. En la Tabla 3 mostramos los valores promedio de esas medidas resumidas por país.

Country	Maths	Reading	Efficiency score
Korea	551	534	1.2032
Japan	534	535	1.2574
Turkey	437	461	1.2758
Poland	527	527	1.2833
Estonia	518	514	1.3003
Ireland	498	520	1.3034
Netherlands	517	505	1.3082
New Zealand	497	511	1.3106
Canada	507	509	1.3193
Germany	507	500	1.3300
Mexico	410	419	1.3433
USA	479	496	1.3433
Czech Rep.	500	496	1.3446
Spain	491	490	1.3487
Finland	508	511	1.3491
Switzerland	514	492	1.3500
UK	488	496	1.3552
Australia	492	500	1.3683
Portugal	479	480	1.3690
Belgium	508	500	1.3763
France	489	498	1.3768
Norway	491	505	1.3911
Austria	488	475	1.4051
Denmark	484	482	1.4054
Luxembourg	490	486	1.4085
Sweden	483	486	1.4106
Chile	432	446	1.4118
Hungary	461	472	1.4167
Italy	476	476	1.4306
Slovak Rep.	473	450	1.4336
Israel	464	484	1.4383
Iceland	488	478	1.4386
Greece	440	461	1.4985
Slovenia	461	438	1.5223
MEAN	488	489	1.3668

Tabla 3. Scores de eficiencia.

Las Figuras 11 y 12 ilustran que existe una relación negativa entre eficiencia y desempeño en matemáticas y lectura en todos los países. En términos generales, los países donde el puntaje promedio de PISA es más alto, también tienden a mostrar niveles más bajos de ineficiencia. Del mismo modo, también vale la pena señalar que algunos países con puntajes de eficiencia promedio similares tienen niveles de desempeño muy diferentes.

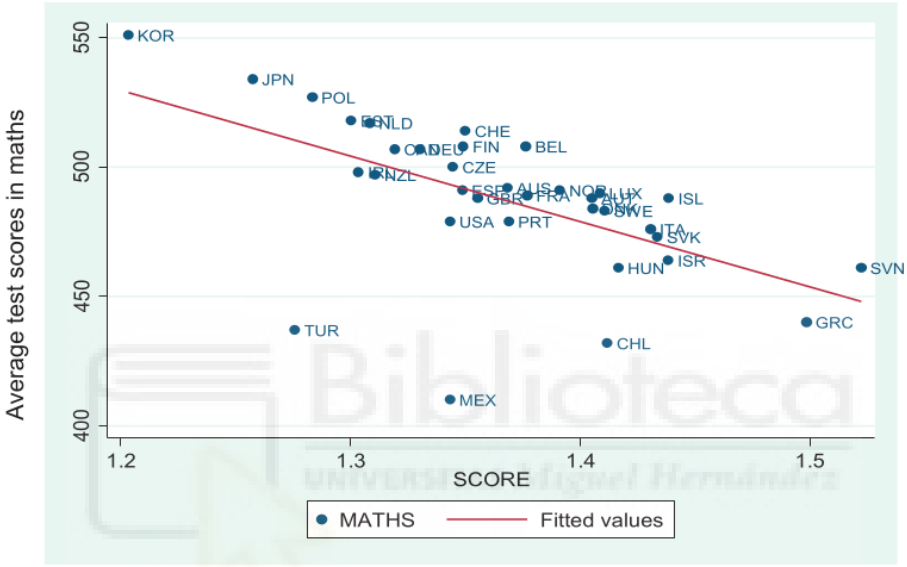

Figura 11. Resultados (en media) de la eficiencia en matemáticas.

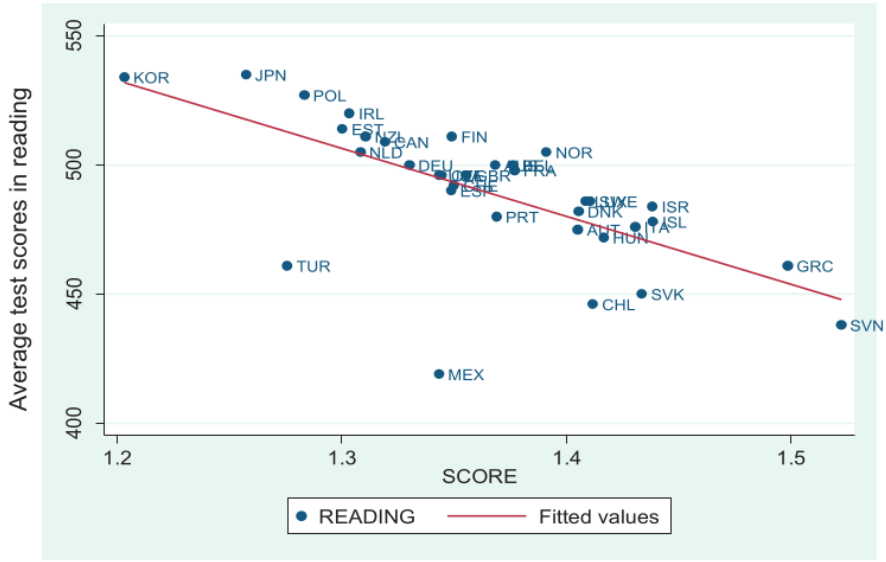

Figura 12. Resultados (en media) de la eficiencia en lectura.

Además, nos gustaría explorar la proyección (relación entre el valor objetivo y el valor real) para cada salida calculada con el modelo propuesto desarrollado por Aparicio et al. (2016) antes de calcular las expansiones de producción promedio. De acuerdo con los valores mostrados en la Tabla 4, existen divergencias entre los valores de proyección de lectura y matemática que no son visibles cuando solo se muestran los valores promedio.

Country	Average efficiency score	Maths (M)	Reading (R)	Difference (R-M)
Korea	1.2032	1.1702	1.2362	6.6010
Japan	1.2574	1.2311	1.2838	5.2645
Turkey	1.2758	1.3206	1.2311	−8.9429
Poland	1.2833	1.2569	1.3097	5.2763
Estonia	1.3003	1.2709	1.3297	5.8840
Ireland	1.3034	1.3049	1.3020	−0.2934
Netherlands	1.3082	1.2738	1.3426	6.8833
New Zealand	1.3106	1.3022	1.3190	1.6772
Canada	1.3193	1.3029	1.3357	3.2730
Germany	1.3300	1.3004	1.3596	5.9190
Mexico	1.3433	1.3778	1.3087	−6.9126
USA	1.3433	1.3517	1.3349	−1.6784
Czech Rep.	1.3446	1.3144	1.3748	6.0459
Spain	1.3487	1.3163	1.3810	6.4710
Finland	1.3491	1.3467	1.3516	0.4875
Switzerland	1.3500	1.2974	1.4026	10.5229
UK	1.3552	1.3486	1.3617	1.3109
Australia	1.3683	1.3534	1.3832	2.9825
Portugal	1.3690	1.3285	1.4095	8.1024
Belgium	1.3763	1.3410	1.4115	7.0555
France	1.3768	1.3505	1.4031	5.2527
Norway	1.3911	1.4110	1.3712	−3.9798
Austria	1.4051	1.3777	1.4324	5.4627
Denmark	1.4054	1.3801	1.4306	5.0557
Luxembourg	1.4085	1.3814	1.4356	5.4181
Sweden	1.4106	1.3879	1.4333	4.5357
Chile	1.4118	1.4545	1.3692	−8.5327
Hungary	1.4167	1.3981	1.4352	3.7096
Italy	1.4306	1.3957	1.4655	6.9814
Slovak Rep.	1.4336	1.3649	1.5024	13.7524
Israel	1.4383	1.4506	1.4260	−2.4580
Iceland	1.4386	1.4433	1.4339	−0.9408
Greece	1.4985	1.4915	1.5055	1.4084
Slovenia	1.5223	1.4488	1.5957	14.6966
TOTAL	1.3655	1.3435	1.3780	

Tabla 4. Valores de proyección de lectura y matemática.

La posibilidad de medir la eficiencia de las escuelas utilizando un punto de referencia internacional, y considerando los desplazamientos radiales y no radiales para medir la distancia a la frontera, permite identificar que algunas escuelas podrían centrarse más en una dimensión del ámbito educativo. En particular, de acuerdo con la forma del gráfico que se muestra en la Figura 13, en términos globales hay mayores ineficiencias en la lectura. De hecho, 26 de los 34 países de la OCDE presentan valores más altos en esta competencia y, en la mayoría de los casos, son superiores al 5%. Por lo tanto, parece que la mayoría de las escuelas de todo el mundo presentan niveles más altos de eficiencia en el dominio de las matemáticas.

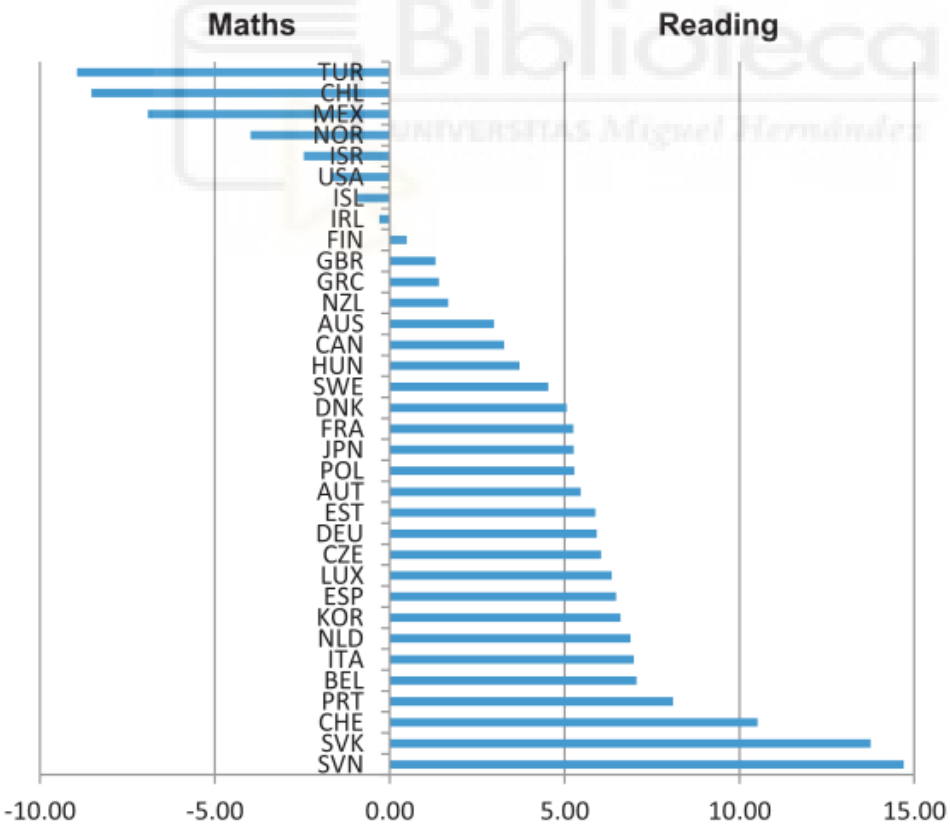

Figura 13.

2.1.5.Conclusión

En este documento, se ha utilizado datos de los países de la OCDE que participan en PISA 2012 para evaluar la eficiencia de las escuelas en un marco internacional. Para el desarrollo del análisis empírico consideramos que las escuelas podrían concentrar sus esfuerzos en mejorar más los resultados en una dimensión del producto educativo que en otra. Para hacer esto, confiamos en medidas de rendimiento no radiales de eficiencia, que son particularmente interesantes en el contexto de la educación, ya que permiten identificar diferentes niveles de ineficiencia para cada producto analizado. En particular, aplicamos una metodología que determina los objetivos más cercanos y la menor distancia a la frontera Pareto-eficiente en DEA basada en la Programación Lineal Bi-nivel.

Aunque es cierto que existen muchas medidas no radiales alternativas en DEA, aquellas basadas en la determinación de la distancia mínima arrojan información útil de evaluación comparativa. Entre el conjunto de medidas DEA no radiales asociadas con la determinación de la menor distancia, optamos por la medida de Russell orientada a la salida. Sin embargo, reconocemos que otras medidas basadas en distancias mínimas también podrían usarse en la aplicación empírica.

Nuestros hallazgos indican que se pueden lograr mejoras potenciales más grandes en el dominio de la lectura que en las matemáticas. En términos globales, detectamos niveles más altos de ineficiencia en la competencia de lectura, por lo tanto, parece que las escuelas de todo el mundo están concentrando sus esfuerzos en matemáticas, tal vez porque el dominio de las matemáticas generalmente se considera uno de los predictores más fuertes de resultados positivos en la educación de los jóvenes, como su capacidad para participar en la educación postsecundaria y sus ganancias futuras esperadas.

2.2. A Hyper-Matheuristic Approach for Solving Mixed Integer Linear Optimization Models in the context of Data Envelopment Analysis

2.2.1. Introducción

En esta aportación se desarrolla una metodología de trabajo para resolver problemas de optimización pertenecientes al área de Programación Lineal Entera-Mixta (MILP), donde el coste de computo puede ser elevado, dependiendo de la estructura interna del problema; como sucede en el caso del modelo de optimización DEA utilizado para la determinación del punto de proyección eficiente más próximo a la unidad evaluada. En primer lugar, se ha desarrollado una metodología de trabajo para encontrar soluciones aproximadas en un tiempo razonable para problemas MILP. El algoritmo desarrollado tiene en cuenta una descomposición del problema, en donde el problema principal se descompone en dos subproblemas jerárquicos, ya que es más fácil resolverlos por separado utilizando diferentes familias de algoritmos de optimización. Esta descomposición se basa en la naturaleza de las variables del problema: continuas y discretas. De esta forma, se propone la utilización de una metodología basada en la combinación de algoritmos metaheurísticos y métodos exactos, llamada Matheurística, la cual ha sido utilizada en la literatura en casos como en Raidl, 2015.

Por otro lado, también se aplica en este artículo un enfoque basado en una estrategia Hyper-Matheurística. Esta metodología consiste en diseñar un esquema parametrizado, capaz de seleccionar de forma dinámica

la metaheurística utilizada dentro de la Matheurística y, por tanto, encontrar una metaheurística óptima para el problema estudiado. De esta forma, gracias a la combinación del esquema parametrizado, en donde cada función de la metaheurística cuenta con unos contadores específicos, y la Hyper-heurística, desarrollamos una Hyper-Matheurística capaz de configurar la Matheurística interna (combinación de metaheurística y método exacto) para obtener el mejor rendimiento en cada problema a evaluar. Por último, la metodología aquí desarrollada será ilustrada a través de la resolución del modelo de DEA bajo el Principio de Mínima Acción.

2.2.2. Descomposición de variables en MILP

Debido a la dificultad a la hora de resolver problemas MILP, y a la gran cantidad de tiempo de computo necesario para obtener soluciones válidas y óptimas del problema, se ha decidido incorporar una técnica de descomposición basada en la naturaleza de las variables del modelo; siguiendo el principio de codificación indirecta (Talbi, 2009). De esta forma, se ha dividido el problema principal en dos subproblemas, resolviendo cada uno de ellos con técnicas diferentes. En particular, uno de los subproblemas será tratado como problema maestro (P1) mientras que el segundo lo hará como problema esclavo (P2), de forma que:

- El problema maestro P1 contendrá todas las variables discretas, así como las restricciones que incluyan a éstas. Este tipo de variables y sus restricciones son muy complejas de resolver utilizando métodos exactos, dependiendo de la estructura interna del problema (que podría resultar de naturaleza altamente combinatoria), por lo que se propone encontrar soluciones válidas mediante técnicas de aproximación (Metaheurísticas). De esta forma, se encontrarán

soluciones válidas para el problema dando valores únicamente a las variables discretas.

- El problema esclavo P2 contendrá el resto de las variables (continuas) así como todo el resto de restricciones y la función objetivo. Con la información proporcionada por el problema P1, se configurarán las restricciones y las variables necesarias. Este nuevo subproblema se considera de Programación Lineal (LP) siendo sencillo de resolver mediante un método exacto. Este subproblema es el encargado de decodificar la solución incompleta del problema P1, obteniendo así una solución para el problema completo planteado inicialmente.

La descomposición descrita queda reflejada gráficamente en la Figura 14.

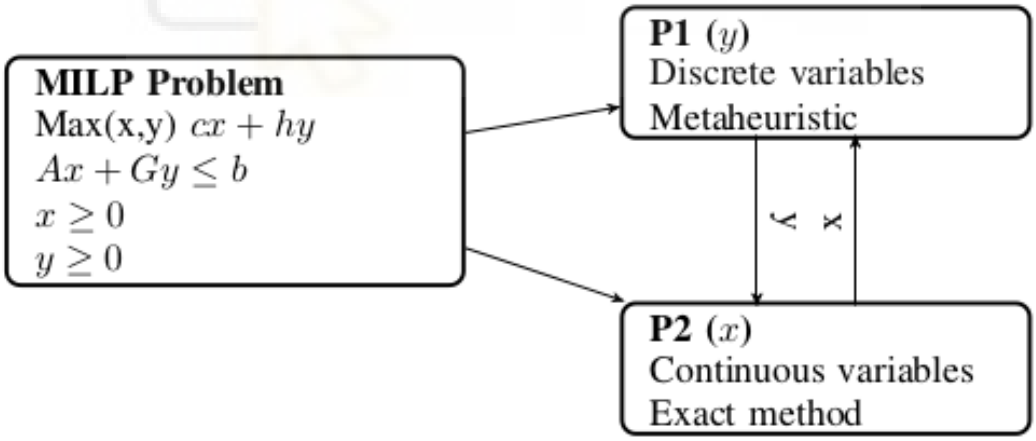

Figura 14. Descomposición del problema principal en dos subproblemas.

2.2.3. Matheurística

La combinación entre metaheurística y métodos exactos que se presenta surge de la necesidad de simplificar los modelos matemáticos que son difíciles de resolver con cualquiera de estas técnicas. De esta forma, cualquier modelo matemático puede dividirse en diferentes subproblemas, tomando la naturaleza de sus variables como criterio principal.

El algoritmo matheurístico propuesto está diseñado para ser utilizado principalmente por metaheurísticas basadas en poblaciones, donde se genera un conjunto de soluciones iniciales, procesándolas mediante diferentes pasos, como la selección, la recombinación y el reemplazo. No obstante, el diseño del algoritmo también permite la utilización de metaheurísticas basadas en individuos si estas son incluidas dentro de alguna funcionalidad interna, como la mejora o la combinación de soluciones. Una vez generadas las soluciones iniciales, siendo éstas válidas/factibles para el subproblema P1, son derivadas al subproblema P2, fijando así el problema LP y siendo resuelto mediante un método exacto. De esta forma, como podemos observar, las dos técnicas de optimización están en continua conexión, compartiendo información para poder obtener soluciones completas del problema principal. Así el algoritmo metaheurístico interno genera las variables discretas (resolviendo el subproblema P1), compartiendo esta información con el método exacto para fijar valores para las variables continúas (resolviendo el subproblema P2). La Figura 15 muestra cómo el problema principal se divide en dos problemas más pequeños y cómo estos comparten información entre ellos.

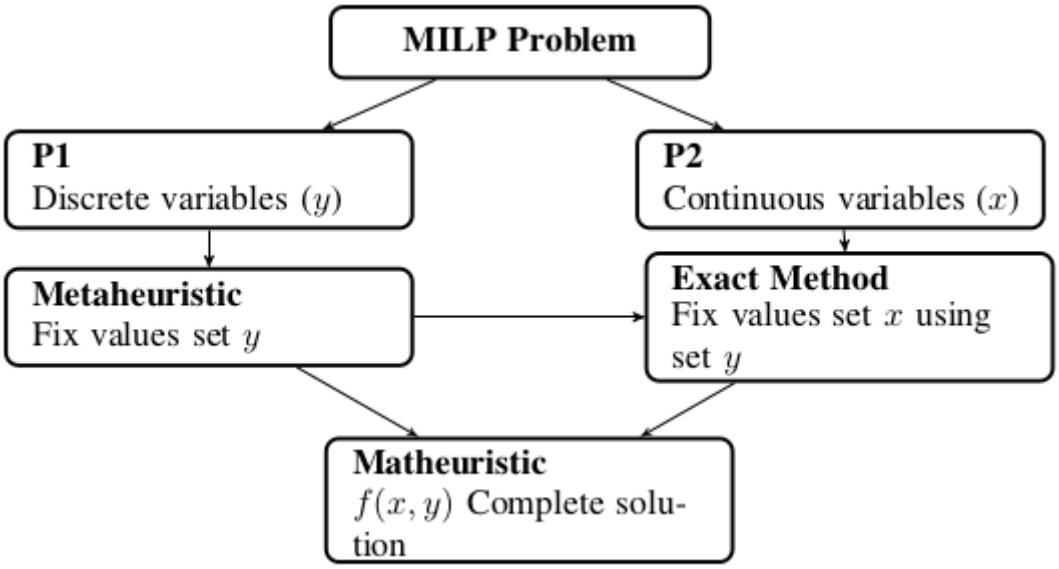

Figura 15. Estructura del algoritmo metaheurístico.

Todas las funciones incorporadas en la metaheurística se ejecutan secuencialmente:

- **Población inicial**: este paso se utiliza para generar la población inicial, fijando valores para las variables discretas en P1. Este primer paso no realiza una exploración exhaustiva del espacio de soluciones, sino que fija ciertos puntos iniciales aleatorios, para utilizarlos como puntos de partida.

- **Mejora**: algunas de las soluciones factibles (dado que no toda las soluciones obtenidas son válidas) se modifican para mejorar el valor de fitness (función objetivo) utilizando el algoritmo de búsqueda de del mejor vecino, propuesto por Mladenović, 1997. Las soluciones infactibles se mejoran tratando de transformarlas en soluciones válidas, haciendo que se cumplan la mayor cantidad de restricciones.

- **Selección**: las soluciones se ordenan. En primer lugar, se ordenan las soluciones válidas de mayor a menor valor de la función objetivo (fitness). A continuación, se incluyen las soluciones no válidas (no se ha podido encontrar valores para las soluciones continuas dada la solución de P1) ordenadas al azar. Se selecciona un porcentaje fijo de soluciones para ser usadas en las funciones posteriores.

- **Combinación**: el algoritmo incluye una función de cruce, en donde las soluciones seleccionadas previamente son emparejadas de dos en dos, combinando los valores de sus variables discretas tratando de obtener soluciones heredadas con mejores características. Únicamente se combinan las variables discretas, de forma que con el nuevo resultado se resuelve el nuevo problema P2, comprobando si la solución es mejor o peor que la de sus predecesores.

- **Diversificación**: el algoritmo incluye una diversificación basada en el método *Edge Recombination* (ER) introducido por Laporte et al., 1997. El objetivo de los métodos de diversificación es impulsar la búsqueda en nuevas regiones del espacio de búsqueda. Aquí, la heurística de búsqueda tabú se reinicia periódicamente con una nueva solución obtenida mediante la recombinación de dos soluciones visitadas previamente durante la búsqueda.

2.2.4. Hiper-Matheurística

Proponemos un algoritmo Hiper-Matheurístico para encontrar la metaheurística más adecuada para el problema evaluado. Esta metaheurística "óptima", al ser combinada con un método exacto, generará la mejor solución para cada problema en el menor tiempo posible. Este

algoritmo busca en todo el espacio metaheurístico para encontrar el mejor diseño. Para conseguir esto, el algoritmo genera un número determinado de metaheurísticas (a través de un esquema parametrizado) que prueba sobre el problema principal obteniendo valores de fitness en cada caso. Además, el método Hiper-Matheurístico incluye un aprendizaje online, en donde ciertas metaheurística son modificadas para optimizar los resultados durante la ejecución de las mismas. De esta forma, el algoritmo es capaz de aprender de las metaheurísticas evaluadas previamente para diseñar las siguientes de una forma más precisa.

Para poder llevar a cabo el diseño dinámico de metaheurísticas, es necesario incluir un esquema parametrizado capaz de establecer, mediante ciertos parámetros, cierta intensificación a las funcionalidades internas de la metaheurística. De esta forma, seríamos capaces de diseñar tanto un algoritmo genético (inicialización, selección, cruce, mutación) como un algoritmo GRASP (inicialización, mejora, intensificación) sin tener que modificar el código. Los parámetros diseñados para esta tarea quedan reflejados en la Tabla 5.

Function	Parameters	Description
Initialize	INEIni	Initial number of elements
	FNEIni	Final number of elements selected for the iterations
	PEIIni	Percentage of elements to improve in the initialization
	IIEIni	Intensification of the improvement in the initialization
End Condition	MNIEnd	Maximum number of iterations
	NIREnd	Maximum number of iterations without improving
Selection	NBESel	Number of best feasible solutions selected
	NWESel	Number of infeasible solutions selected
Combination	PBBCom	Number of combinations between feasible solutions
	PWWCom	Number of combinations between infeasible solutions
Improve	PEIImp	Percentage of crossover elements to be improved
	IIEImp	Intensification of the improvement
	PEDImp	Percentage of elements to diversify
	IIDImp	Intensification of the improvement to diversify elements

Tabla 5. Parámetros utilizados en cada una de las funcionalidades internas de la matheurística.

Al mismo tiempo, gracias a este esquema, la Hiper-Matheurística es capaz de generar metaheurísticas híbridas, las cuales incluyen características de varias de ellas, sin ser necesariamente una metaheurística exacta como las que aparecen en la literatura.

Ya se ha utilizado y experimentado un esquema parametrizado en Almeida et al., 2013, donde se consideraron otros parámetros, como las combinaciones entre soluciones válidas y no válidas. El número y el significado de los parámetros cambiarían si se consideran otras metaheurísticas básicas o si las funciones básicas se implementan de manera diferente, pero los parámetros considerados aquí nos permiten generar y experimentar automáticamente diferentes metaheurísticas y combinaciones de ellas para mejorar los resultados obtenidos.

2.2.5. Datos experimentales

Los experimentos realizados en esta aportación están estrictamente relacionados con el rendimiento del algoritmo presentado. De esta forma, se va a evaluar tanto la eficiencia de la descomposición del problema, comparando los resultados de resolver un problema de forma global frente a hacerlo en forma descompuesta, como el funcionamiento de la Hiper-Matheurística, asignando valores a los indicadores del esquema parametrizado y encontrando la configuración óptima para el problema evaluado. Todos los experimentos medirán la calidad de la solución obtenida tanto en término de fitness (valor de la función objetivo) como en tiempo de cómputo.

$$
\begin{aligned}
& \max \left\{ \beta_k - \tfrac{1}{m} \textstyle\sum_{i=1}^{m} \frac{t_{ik}^-}{z_{ik}} \right\} \\
\text{s.t.} \quad & \\
& \beta_k + \tfrac{1}{s} \textstyle\sum_{r=1}^{s} \frac{t_{rk}^+}{q_{rk}} \leq 1 && (c.1) \\
& -\beta_k - \tfrac{1}{s} \textstyle\sum_{r=1}^{s} \frac{t_{rk}^+}{q_{rk}} \leq -1 && (c.2) \\
& -\beta_k z_{ik} + \textstyle\sum_{j=1}^{n} \alpha_{jk} x_{ij} + t_{ik}^- \leq 0 \quad \forall i = 1, \ldots, m && (c.3) \\
& \beta_k z_{ik} - \textstyle\sum_{j=1}^{n} \alpha_{jk} x_{ij} - t_{ik}^- \leq 0 \quad \forall i && (c.4) \\
& -\beta_k q_{rk} + \textstyle\sum_{j=1}^{n} \alpha_{jk} y_{rj} - t_{rk}^+ \leq 0 \quad \forall r && (c.5) \\
& \beta_k q_{rk} - \textstyle\sum_{j=1}^{n} \alpha_{jk} y_{rj} + t_{rk}^+ \leq 0 \quad \forall r && (c.6) \\
& -\textstyle\sum_{i=1}^{m} \nu_{ik} z_{ij} + \textstyle\sum_{r=1}^{s} \mu_{rk} q_{rj} + d_{jk} \leq 0 \quad \forall j && (c.7) \\
& \textstyle\sum_{i=1}^{m} \nu_{ik} z_{ij} - \textstyle\sum_{r=1}^{s} \mu_{rk} q_{rj} - d_{jk} \leq 0 \quad \forall j && (c.8) \\
& -\nu_{ik} \leq -1 \quad \forall i && (c.5) \\
& -\mu_{rk} \leq -1 \quad \forall r && (c.6) \\
& -d_{jk} \leq -M b_{jk} \quad \forall j && (c.7) \\
& \alpha_{jk} \leq M(1 - b_{jk}) \quad \forall j && (c.8) \\
& b_{jk} = 0, 1 \quad \forall j && (c.9) \\
& -\beta_k \leq 0 && (c.10) \\
& -t_{ik}^- \leq 0 \quad \forall i && (c.11) \\
& -t_{rk}^+ \leq 0 \quad \forall r && (c.12) \\
& -d_{jk} \leq 0 \quad \forall j && (c.13) \\
& -\alpha_{jk} \leq 0 \quad \forall j && (c.14)
\end{aligned}
\tag{16}
$$

En particular, el objetivo del problema a resolver (16) en este apartado consiste en la determinación del grado de eficiencia técnica de un

conjunto de DMUs, bajo el contexto del principio de Mínima Acción. En concreto, el modelo a resolver se corresponde, en este caso, a la medida *Slacks-Based Measure* (SBM) modificada de (Aparicio et al., 2007), que se corresponde con el modelo *graph* de la medida usada en nuestra aportación 1. El modelo matemático aparece a continuación en este texto. Con respecto a los datos, en nuestras simulaciones, los inputs y outputs de cada una de las n DMU se generan al azar, pero teniendo en cuenta que la función de producción que gobierna el proceso productivo es una función Cobb-Douglas (Cobb et al., 1928). Para todos los experimentos, el software IBM ILOG CPLEX Optimization Studio (CPLEX) será utilizado como método exacto (*Branch and Bound + Simplex*), utilizando la versión 12.1.

En primer lugar, se quiere estudiar si la descomposición de un problema en varios subproblemas, según la metodología estudiada previamente, supone una mejora a la hora de resolver este tipo de problemas. De esta forma, se va a comparar, utilizando el problema de DEA planteado, dos formas de resolverlo. Uno de los métodos estará basado en una heurística, diseñada exclusivamente para este problema e introducida en González et al., 2015. Este método (*Heuristic Method*) no realiza ningún tipo de descomposición, sino que trata el problema de forma global, intentando resolver todas las restricciones de manera conjunta. Por otro lado, se resolverá el problema mediante la Matheurística propuesta en esta aportación (Matheuristic). Para ambos métodos, únicamente se van a evaluar las soluciones obtenidas en la fase de Inicialización, contando ésta con 100 individuos, sin utilizar ninguna de las otras funciones propias de las metaheurística (Mejora, Selección, Cruce y Diversificación). Además, se van a incluir dos casos especiales con el fin de aportar una conclusión más robusta al experimento. Uno de los casos se trata de utilizar el mismo

método matheurístico, pero aumentando el número de soluciones iniciales a 1000 (Matherustic 1000). El segundo caso será un método híbrido, en donde la mitad de los problemas a evaluar serán resueltos por la matheurística, mientras que la otra mitad serán resueltos por el método heurístico.

Recordar que cada instancia constará de *n* problemas a resolver (DMUs), en donde cada DMU será resuelta mediante alguno de los 4 métodos. El resultado final será el promedio de las mejores soluciones obtenidas por cada DMU. El problema utilizado en este experimento cuenta con las siguientes características: *m = 3, n = 50, s = 1*.

size			Matheuristic 100		Heuristic Method		Hybrid method		Matheuristic (1000)	
m	n	s	% val.	fitness	% val.	fitness	% val.	fitness	% val.	fitness
2	100	1	$18_{5.16}$	0.7605	$94.96_{6.47}$	0.685	$56.946_{8.24}$	0.413	$6.032_{1.02}$	0.7609
3	50	1	$1.32_{1.61}$	0.4748	$78.85_{5.90}$	0.777	$43.441_{7.82}$	0.5661	$1.221_{0.77}$	0.7974
3	100	1	$0.55_{1.37}$	0.2713	$66.12_{5.17}$	0.6171	$32.539_{5.23}$	0.48	$0.758_{0.52}$	0.6662
4	100	1	$0.29_{0.54}$	0.1544	$60.71_{6.72}$	0.588	$25.015_{6.12}$	0.414	$0.305_{0.82}$	0.5515
5	100	1	$0.33_{0.81}$	0.1974	$58.66_{8.36}$	0.501	$39.485_{9.08}$	0.398	$0.371_{0.15}$	0.5164

Tabla 6. Promedio del porcentaje de soluciones válidas y valor de la función objetivo durante el proceso de inicialización para los diferentes métodos de optimización y variando el parámetro INFIni.

La Tabla 6 muestra los valores obtenidos, tanto el promedio de fitness obtenido como el porcentaje de soluciones válidas encontradas (en promedio) en todas las DMUs evaluadas. Con este experimento podemos comprobar cómo no es trivial obtener soluciones válidas utilizando el método matheurístico propuesto, ya que el espacio de soluciones es muy grande a medida que el tamaño del problema crece, sobre todo trabajando únicamente con las variables discretas. Por otro lado, vemos como el método heurístico aumenta en gran medida este parámetro, dado que está específicamente diseñado para este problema. Sin embargo, lo importante

de este experimento es ver cómo, a medida que aumentamos la población inicial, y sin hacer ningún tratamiento posterior (el cual mejorará sustancialmente las soluciones), las soluciones obtenidas por la matheurística mejoran en fitness a las obtenidas por el método heurístico.

A continuación, procede utilizar el algoritmo matheurístico completo, tal cual lo hemos descrito previamente, y comprobar su rendimiento sobre este mismo problema. Para esto, se van a utilizar los mismos métodos del experimento anterior, pero incluyendo parámetros para el resto de funcionalidades. Se han establecido los siguientes parámetros: INEIni = 100/1000, FNEINI = 50, IIEIni = 10, PEIIni = 10 NBESel = 15, NWESel = 15, PBBCom = 25, PWWCom = 25, PEEImp = 10, IIEImp = 5, PEDImp = 5, IIDImp = 5, NIREnd = 5, MNIEnd = 10. Los parámetros han sido elegidos de forma que la metaheurística generada sea lo más versátil posible (incluya todas las funcionalidades) y no requiera de excesivo tiempo de computo. Los resultados en términos de fitness quedan reflejados en la Figura 16, mientras que los tiempos resultantes en la Figura 17.

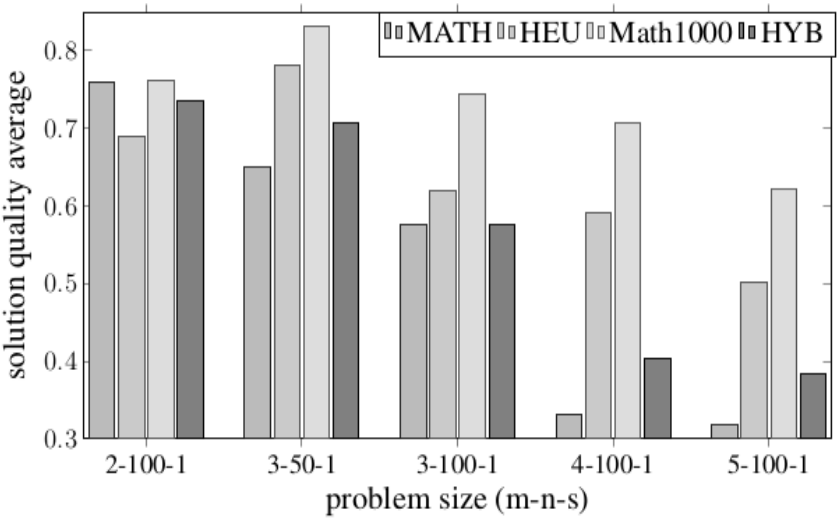

Figura 16. Valores de la función objetivo obtenidos por los métodos propuestos en la tabla 6.

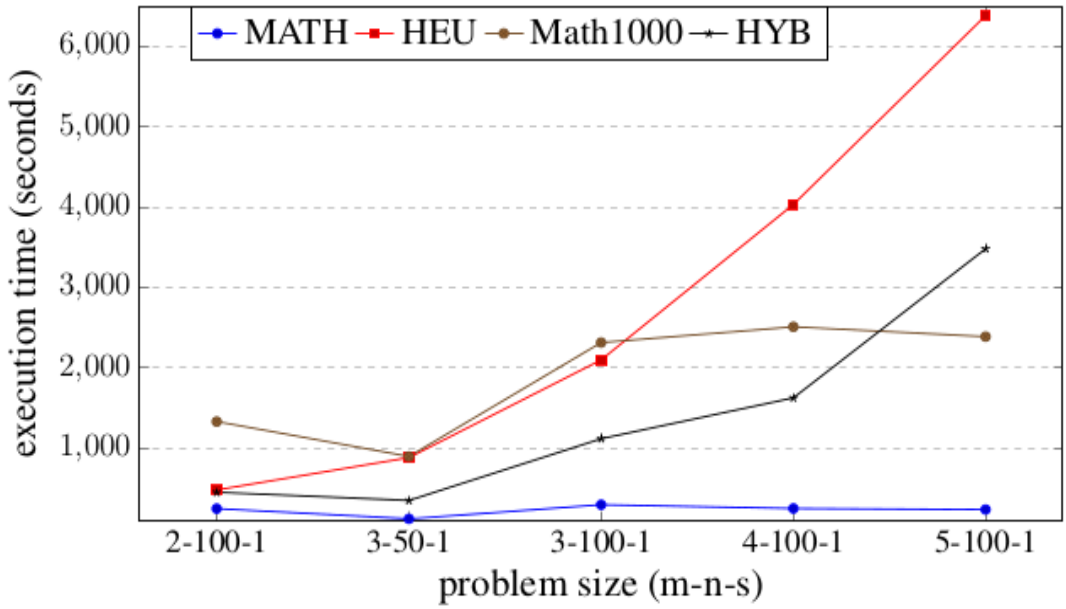

Figura 17. Tiempos de cómputo obtenidos por cada uno de los métodos propuestos en la tabla 6.

Estudiando ambas gráficas, podemos observar cómo, en términos de fitness, la Matheurística con alto nivel de población inicial, supera en todos los tamaños de problema al resto de métodos, y además, en términos de tiempo, no es extremadamente alto, siendo muy inferior al utilizado por el método heurístico. Este método heurístico, en cuanto al fitness, se acerca en gran medida al algoritmo matheurístico, pero es mucho peor en términos de tiempo, ya que este método tiene que resolver un problema mucho más complejo. Además, a medida que el problema aumenta en tamaño, el método heurístico se separa más del algoritmo matheurístico en términos de fitness. Por tanto, concluimos que la descomposición del problema es óptimo, y que tanto en calidad de fitness como en tiempo de computo, es mucho mejor que resolverlo de forma completa.

Una vez demostrada la eficiencia del método matheurístico, toca configurar la metaheurística interna (variando los indicadores del esquema parametrizado) para encontrar la que genere mejores resultados. Para el diseño automático de los parámetros, se trabaja a partir de ciertos esquemas

metaheurísticos generales, como Algoritmo Genetico (GA), Scatter Search (SS) o Greedy Randomized Adaptive Search Procedure (GRASP). Los tres esquemas propuestos utilizan las funcionalidades descritas previamente. Todas las metaheurísticas generadas por la Hiper-Matheurística heredarán algunas características de las metaheurísticas generales y combinarán otras. Las metaheurísticas propuestas para ser ejecutadas en los experimentos se muestran en la Tabla 7.

Los valores más bajos y más altos de los parámetros de estas metaheurísticas se usan para limitar los valores de todos los parámetros dentro de la Hiper-Matheurística. Por esto, todos los metaheurísticos creados serán generados usando estos límites.

Metaheuristic	IINEIni	FNEIni	PEIIni	IIEIni	NBESel	NWESel	PBBCom
EA	300	150	0	0	100	0	50
GR	500	1	100	20	0	0	0
SS	100	50	50	5	25	25	25
Hyper	100/500	1/150	0/100	0/20	0/100	0/25	0/50

Metaheuristic	PWWCom	PEIImp	IIEImp	PEDImp	IDEImp	MNIEnd	NIREnd
EA	0	0	0	10	10	10	5
GR	0	0	0	0	0	10	5
SS	25	50	10	0	0	10	5
Hyper	0/25	0/50	0/10	0/10	0/10	0/10	0/5

Tabla 7. Valores de los parámetros para los tres esquemas metaheurísticos propuestos y los limites especificados para la generación automática.

El diseño de la metaheurística óptima se realiza entrenando sobre un único tamaño de problema, extrapolando los resultados obtenidos sobre el resto de tamaños, dado que la naturaleza del problema sigue siendo la misma. Se ha seleccionado el tamaño $m = 3, n = 50, s = 1$ como problema de entrenamiento. El entrenamiento se ha realizado ejecutando la Hiper-

Matheurística con 100 configuraciones distintas y 100 combinaciones entre éstas (funcionando como una metaheurística a un nivel superior). En la Tabla 8 podemos ver los parámetros obtenidos para la mejor metaheurística (mejor fitness obtenido).

Metaheuristic	IINEIni	FNEIni	PEIIni	IIEIni	NBESel	NWESel	PBBCom
Hyper	310	61	13	17	12	10	22

Metaheuristic	PWWCom	PEIImp	IIEImp	PEDImp	IDEImp	MNIEnd	NIREnd
Hyper	25	12	8	3	7	6	5

Tabla 8. Valores de los parámetros encontrados para la mejor metaheurística (Mbest), encontrados por la Hiper-Matheurística.

A continuación, se van a mostrar los valores tanto de fitness como de tiempo para los 3 esquemas metaheurísticos propuestos (GA, SS y GRASP), la mejor metaheurística obtenida por la Hiper-Matheurística (Mbest), y la matheurística previa con 1000 elementos (caso que había sido el mejor hasta ahora). En la Figura 18 podemos observar los valores de fitness, mientras que en la Figura 19 podemos ver los tiempos requeridos para obtener la solución final.

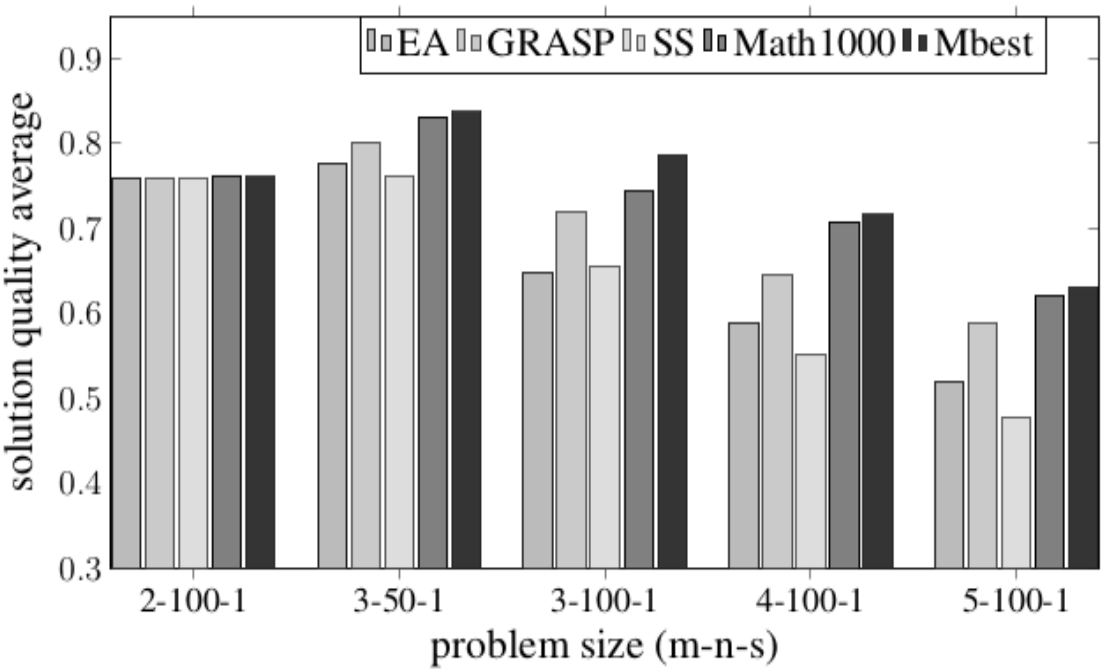

Figura 18. Fitness obtenido por cada una de las Metaheurísticas propuestas (EA, GRASP y SS), por el meto matheurístico previo (Math1000) y por la Hiper-Matheurística (Mbest).

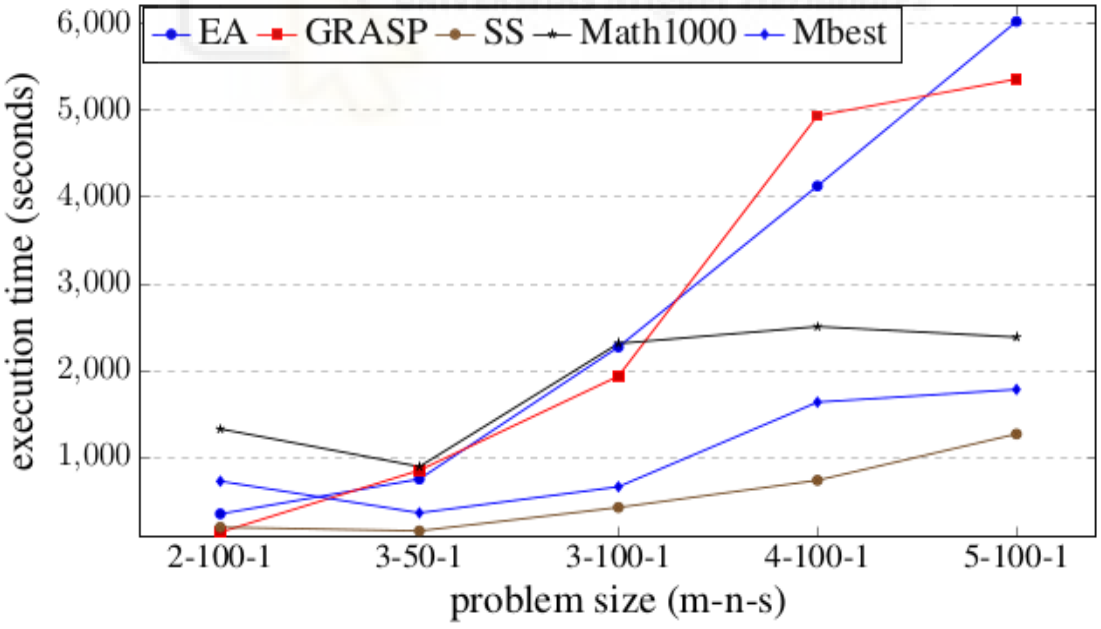

Figura 19. Tiempo consumido por cada método para resolver el problema.

2.2.6. Conclusión

En este trabajo, hemos propuesto una estrategia de descomposición eficiente para problemas de optimización de MILP. Se ha desarrollado una descomposición jerárquica basada en la naturaleza de las variables de decisión (continua versus discreta) y la complejidad de los subproblemas. La metaheurística explora una codificación incompleta que representa sólo variables de decisión discretas. La codificación de soluciones se completa para las variables de decisión continuas resolviendo un problema lineal de manera exacta.

Este marco, así creado, ha demostrado su eficacia en la resolución de problemas tipo MILP. Además, se ha desarrollado una metodología Hiper-Matheurística sobre el esquema metaheurístico parametrizado. Permite el diseño y la configuración automáticos de una plantilla flexible y genérica para metaheurísticas basadas en la población. Se han obtenido resultados satisfactorios en términos de calidad de la solución y tiempo de ejecución.

Una de las líneas futuras que nos planteamos, ligadas a este trabajo, es aplicar esta metodología Hiper-Matheurística a otros problemas de optimización de la vida real formulados como MILP, así como la generalización del esquema de descomposición propuesto para otras familias de problemas de optimización, en el que sólo la parte continua del problema es lineal y fácil de resolver mediante un algoritmo exacto. Otra perspectiva interesante es investigar el diseño en paralelo y la implementación de esta metodología Hiper-Matheurística en algoritmos paralelizables, utilizando arquitecturas heterogéneas compuestas por grupos de múltiples núcleos y GPU (unidades de procesamiento de gráficos).

2.3. A Parallel Application of Matheuristics in Data Envelopment Analysis

2.3.1. Introducción

En esta aportación se analiza el coste computacional necesario para resolver el problema de DEA propuesto en capítulos anteriores, de forma que se estudia el rendimiento de diversos paquetes de optimización encontrados en la literatura (CPLEX y GUROBI). Además, con el fin de mejorar la velocidad y optimizar la resolución de este tipo de problemas, se desarrolla un algoritmo paralelizado, específicamente diseñado para problemas MILP de este tipo, en donde un conjunto de subproblemas deben ser evaluados para obtener una solución final. Estos subproblemas están definidos por cada una de las DMUs evaluadas en el modelo DEA, necesarias para formar la frontera de eficiencia, así como la distancia de todas aquellas que se encuentren por debajo de dicha frontera hasta la misma.

En primer lugar, se va a utilizar el modelo DEA propuesto en artículos previos (como en el Artículo 1 de esta memoria de tesis) como modelo tipo a resolver. Este problema cuenta con la particularidad de que, para obtener la solución final, es necesario resolver multitud de problemas MILP, los cuales son independientes unos de otros a la hora de ser resueltos. Esta propiedad permite ejecutarlos totalmente en paralelo, sin necesidad de compartir, a priori, información entre ellos.

Como consecuencia de esto, es posible incluir en el algoritmo paradigmas de paralización para dotar de mayor velocidad a la resolución

de los problemas. Por tanto, en segundo lugar, se va a desarrollar un algoritmo paralelizado, el cual incluirá a la Matheurística desarrollada en González et al., 2017, en donde se utilizan tanto metaheurísticas como métodos exactos para resolver los problemas.

Por último, como análisis computacional, se van a utilizar diferentes métodos exactos (paquetes de optimización encontrados en la literatura). Esto permitirá comprobar el rendimiento de estos paquetes cuando se lanzan múltiples instancias simultáneamente, y cómo responde el hardware en el que se ejecutan.

2.3.2. Algoritmo Paralelo

Dada la naturaleza del problema DEA propuesto, y la configuración de la Matheurística utilizada para resolverlo (combinando Metaheurísticas y métodos exactos) propuesta en González et al. (2017), es posible diseñar un algoritmo paralelo híbrido, incluyendo distintos niveles de paralelismo (con distintos paradigmas). Estos niveles están asociados a distintos paradigmas de computación, cada uno asociado a una etapa del algoritmo concreta, en función de las necesidades.

El algoritmo paralelo propuesto se organiza en dos niveles claramente diferenciados. El primero de ellos se encuentra en el nivel superior, y se encarga de ejecutar las diferentes DMUs del problema DEA en paralelo, dado que dichos problemas son independientes unos de otros. Este primer nivel optimiza el tiempo de computo. El segundo nivel se encuentra asociado a la Matheurística, paralelizando las funciones internas de dicho algoritmo. Este segundo nivel optimiza el fitness de cada uno de los problemas (mejorando la solución final de la función objetivo).

Para la implementación de todos estos niveles de paralelismo se han escogido los dos paradigmas más conocidos de la literatura, ambos destinados a optimización de recursos de la CPU con la finalidad de ejecutar procesos de forma simultánea: OpenMP y MPI.

- **OpenMP**: Este primer paradigma es un modelo de memoria compartida, en donde las ejecuciones, llamadas *threads,* se comunican utilizando variables compartidas (en los mismos registros de memoria). En este tipo de modelos de programación, la sincronización es crítica, dado que varias ejecuciones simultaneas acceden a los mismos registros de memoria, y debe existir una concurrencia en la información. Las funciones de paralelización de memoria compartida se han incluido dentro de las funciones propias del algoritmo, lo que las hace más rápidas. Principalmente, estas mejoras se han incluido en la función de inicialización, donde es posible distribuir la generación de soluciones entre los diferentes subprocesos, y en las funciones de mejora y cruce, donde estas tareas se pueden dividir en funciones más pequeñas. Este modelo de paralelismo se ha introducido con la intención de mejorar los bucles internos del algoritmo metaheurístico principal. Por lo tanto, en todas las partes del algoritmo donde se deben evaluar muchos modelos (inicialización, mejora, cruce), y siempre que la evaluación de los modelos sea independiente entre sí, se introduce este tipo de paralelismo.

- **MPI**: Esta técnica de paralelismo está basada en el paso de mensajes entre procesos, ya que estos se ejecutan de manera aislada entre ellos, siendo totalmente independientes (distintos registros de

memoria). De esta forma, no existe necesidad de una sincronización estricta a priori. Las ejecuciones paralelas, llamadas *procesos*, son capaces de compartir información a través de sentencias de programación. Las funciones de paso de mensajes se han incluido en un nivel superior a la matheurística. Este paradigma optimiza el tiempo de computación. Este nivel de paralelismo se encuentra en un nivel más alto que el esquema de memoria compartida. De esta manera, el número de modelos a evaluar se divide en partes iguales en los diferentes hilos. Por lo tanto, el tiempo de cálculo disminuye a medida que aumenta el número de núcleos. En general, la cantidad de DMUs a evaluar del modelo DEA se dividen de forma equitativa entre los recursos del sistema. Si no es posible dividir todas las DMU en grupos iguales, las unidades restantes se asignarán aleatoriamente a los diferentes recursos.

El algoritmo propuesto queda reflejado en la Figura 20. En esta imagen se hace también referencia a la distribución de recursos, dado que al dividir los problemas en diferentes recursos, llamados procesos para MPI, cada uno de estos procesos usará un número determinado de recursos, *threads* para OpenMP, de forma que, si llamamos q a los recursos (procesos) utilizados por MPI, y p a los utilizados por OpenMP (*threads*), los recursos totales de nuestro sistema (T) tienes que cumplir que $T = q * p$. De esta forma, cada proceso podría hacer referencia a un hardware especifico, el cual se encargará, utilizando el resto de sus recursos, de resolver ciertos problemas (DMUs), haciendo lo mismo el resto de hardware disponible.

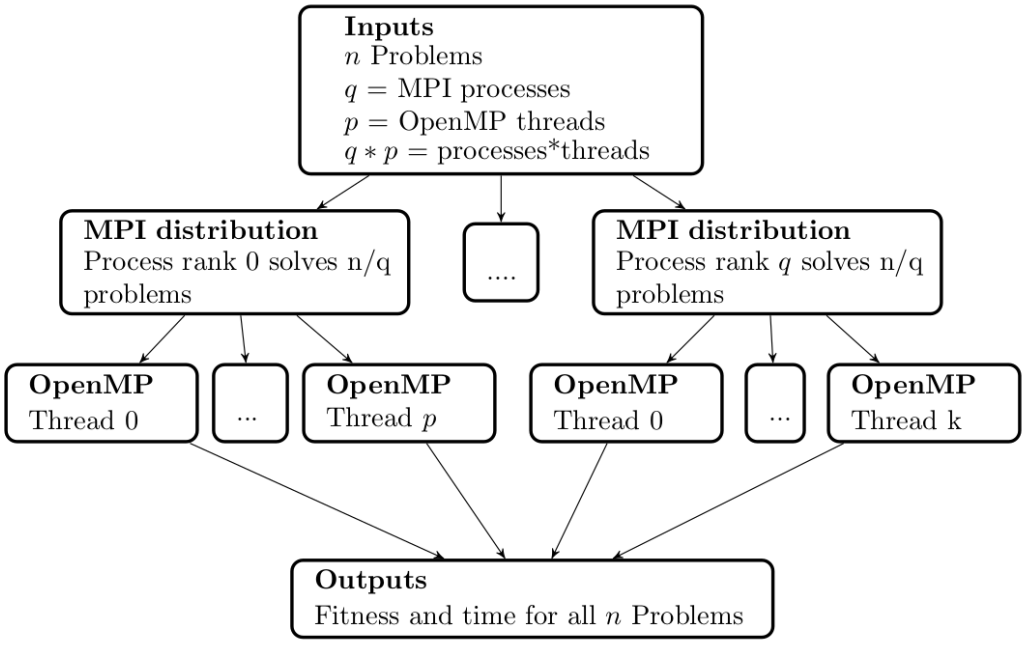

Figura 20. Algoritmo paralelo híbrido.

Dada la naturaleza del problema, también es posible intercambiar la utilización de los paradigmas paralelos y, por tanto, el método de memoria compartida OpenMP podría ser utilizado para dividir los diferentes problemas (DMUs) a solucionar. De esta forma, es necesario experimentar con ambos al primer nivel de paralelismo, para analizar si la forma especificada previamente es la correcta.

2.3.3. Datos Experimentales

Los experimentos están destinados a analizar la efectividad y rendimiento del esquema paralelo desarrollado, así como comprobar el funcionamiento de los distintos paquetes de optimización (GUROBI y CPLEX) incorporados a este algoritmo.

En primer lugar, se ha desarrollado un experimento para comprobar la efectividad de los dos paradigmas de programación propuestos. Para esto, se han utilizado ambos con la misma funcionalidad: dividir las DMUs a resolver en grupos iguales. De esta forma, vamos a realizar el nivel superior de paralelismo con los dos paradigmas, con la intención de comprobar cuál de ellos es más efectivo en esta tarea; mientras que la matheurística queda de momento en secuencial.

Al mismo tiempo, se van a evaluar los dos paradigmas de programación propuestos (MPI y OpenMP) cuando están siendo ejecutados con diferentes paquetes de optimización (CPLEX y GUROBI) y estos ejecutan múltiples instancias de forma simultánea. Para este experimento se ha fijado un problema con tamaño *m=3, s=2 y n=50*, y unos parámetros para el esquema metaheurístico tal que *Population = 310, Combination = 49, Improvement = 17, Diversification = 10, EndCondition = 10*.

Además, este primer experimento fue ejecutado sobre un sistema AMD Phenom II X6 1075T CPU (hexacore) a 3 GHz con 16 GBytes of RAM, memoria cache L1 y L2 de 64 KBytes y 512 KBytes respectivamente, y una memoria cache L3 de 6 MBytes compartida por todos los núcleos.

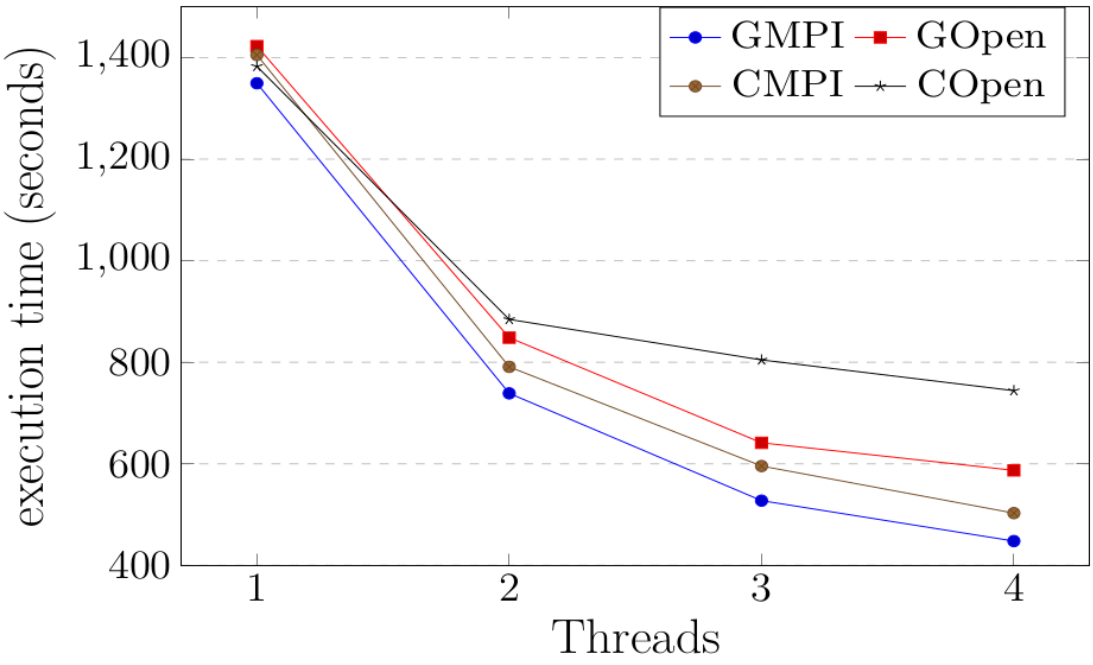

Figura 21. Comparación entre diferentes paradigmas de paralelismo utilizados en el mismo nivel del esquema paralelizado con varios paquetes de optimización (CPLEX y GUROBI).

La Figura 21 muestra cómo, a medida que aumenta el número de recursos, el tiempo de conmutación decrece para todas las combinaciones establecidas. Sin embargo, también se ve un decrecimiento de la mejora en tiempo a medida que aumenta el número de recursos utilizado, y esto es dado que, a medida que más recursos utilizamos para dividir los problemas en grupos, los grupos cada vez son más parecidos en número de problemas, y por tanto el tiempo de computo se parece cada vez más. Como conclusión primera de este experimento, podemos asegurar que MPI es más óptimo para la tarea de división de problemas, haciendo que cada uno de ellos se resuelva y utilice únicamente ciertos recursos de memoria independientes.

2.3.4. Conclusión

La aplicación del principio de mínima acción en la DEA es un tema de relevancia en la literatura reciente de la DEA. Sin embargo, es bien sabido que desde un punto de vista computacional, esto generalmente se ha abordado con enfoques inadecuados, asociados con problemas combinatorios NP-hard.

Los algoritmos paralelos son buenas soluciones para resolver este tipo de problemas. Esto se debe a que se debe resolver una gran cantidad de problemas independientes, y esos problemas se pueden dividir y resolver por diferentes hilos de forma independiente. Se debe tener en cuenta el rendimiento en paralelo de los paquetes de optimización utilizados. En este artículo se han evaluado dos de los más comunes en la literatura, CPLEX y GUROBI, comprobando su rendimiento en paralelo. El algoritmo paralelo propuesto en este artículo funciona de manera óptima con ambos optimizadores y con diferentes paradigmas paralelos.

Para futuros trabajos, proponemos el uso de otros optimizadores gratuitos, comprobando su rendimiento con los optimizadores más potentes del mercado. Además, también es deseable incorporar mejoras en el paralelismo, de modo que las funciones internas del algoritmo metaheurístico se puedan ejecutar con mayor rapidez y precisión.

2.4. A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers

2.4.1. Introducción

Como continuación directa del artículo presentado en González et al. (2018), se ha desarrollado una ampliación del análisis en paralelo del algoritmo previamente desarrollado. De esta forma, se ha profundizado en mayor medida en la calidad de este algoritmo, así como encontrar la configuración más óptima de recursos. El objetivo de este artículo es el de evaluar el rendimiento de ambos niveles de paralelismo (presentados en González et al., 2018), aplicando los diferentes paradigmas de programación paralela, y de conseguir la configuración de recursos más óptima en este tipo de problemas MILP.

En el artículo previo (González et al., 2018) se estudiaron los diferentes paradigmas de paralización, aplicándolos únicamente en el primer nivel de paralelismo (división de problemas). Gracias a estos experimentos se demostró como MPI era más eficiente en esta tarea, dejando a OpenMP la tarea de optimizar el algoritmo matheurístico interno. En este nuevo artículo se hereda esta información, y se centra en evaluar la combinación de ambos niveles con todos los recursos del sistema. Además, se ha utilizado una máquina con mayor número de recursos (hasta 40 threads).

Una vez evaluado ambos niveles dando todos los recursos disponibles, se pretende dividir dichos recursos entre ambos niveles, con el objetivo de conseguir el mínimo número de interferencias entre las instancias, mejorando así el tiempo de cómputo.

2.4.2. Algoritmo Matheurístico Paralelo

Para entender mejor este método paralelo (González et al., 2018) mostrado en la figura 22, y sobre todo dónde se incluyen estas técnicas de paralelismo, se va a proceder a representar el algoritmo metaheurístico diseñado en González et al. (2017), haciendo referencia a las secciones en paralelo. El algoritmo 3 muestra su funcionamiento. Se puede observar cómo aparecen las sentencias ***do in parallel*** en todos aquellos bucles que se quieren paralelizar, de forma que algunos estarán paralelizados a través de MPI mientras que otros en OpenMP. Como hemos comentado, en principio MPI estaría ligado a la separación de problemas (DMUs) en diferentes procesos, mientras que OpenMP se va a utilizar para optimizar los bucles internos de la matheurística. De esta forma, el primer bucle **for** *k=1 to TotalProblems* ***do in parallel*** separaría los diferentes problemas en diversos procesos mediante MPI. El resto de bucles estarán paralelizados utilizando OpenMP, con la intención de minimizar el tiempo de cómputo necesario para cada problema.

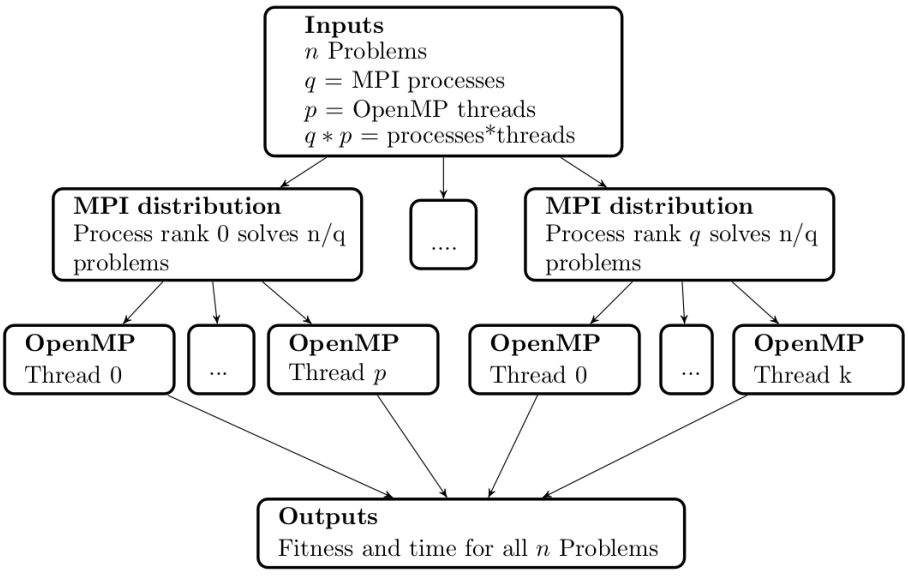

Figura 22. Algoritmo paralelo desarrollado en Martin et al. 2018..

input : $Problems(x, y)$
output: Best solution in each of the problems

Fix the metaheuristic parameters;
for $k = 1$ **to** $TotalProblems$ **do in parallel**
  //Create S_k *set of solutions for problem k-th*;
  for $j = 1$ **to** $Population_k$ **do in parallel**
    Fix discrete variables vd_j *of problem* $P1$;
    Obtain continuous variables vc_j *solving* $P2$ *through exact method*;
    $S_k \leftarrow [Solution_j := (vd_j, vc_j)]$;
    if $Solution_j$ *is not feasible* **then**
      Improve $Solution_j$ *using the best neighbourhood algorithm*;
    end
  end
  do
    //Select SS_k *subset of* S_k *such as* $|SS_k| > 1$;
    for $w = 1$ **to** $Combination_k$ **do in parallel**
      Select $s_1, s_2 \in S_k$ *randomly* ;
      Combine vd_1 *and* vd_2 *the discrete variables of* s_1 *and* s_2 *saving as the discrete variables of a new solution* s_w;
      Obtain vc_w *continuous variables of* s_w *solving* $P2$;
      if $Fitness(w) > Fitness(r_1)$ & $Fitness(w) > Fitness(r_2)$ **then**
        $SS_k \leftarrow s_w$;
      end
    end
    //Improve SS_k *subset of* S_k;
    for $w = 1$ **to** $Improve_k$ **do in parallel**
      Select $s_w \in SS_k$ *randomly* ;
      REPEAT:;
      Modify vd_w *using the best neighbourhood algorithm and obtain* vc_w *solving* $P2$ *through exact method*;
      UNTIL $Fitness(s_w)$ *increase achieve EndConditions*;
    end
    //Diversify SS_k *subset of* S_k;
    for $w = 1$ **to** $Diversification_k$ **do in parallel**
      Select $s_w \in SS_k$ *randomly* ;
      Modify randomly vd_w *of* s_w;
      Obtain vc_w *solving* $P2$ *through exact method*;
    end
    Include SS_k *in* S_k;
  while *not EndCondition*;
  $BestSolution_k \leftarrow s \in S_k$ *such as* $Fitness(s) \geq Fitness(w) \forall w \in S_k$;
end

Algoritmo 3. Esquema matheurístico paralelizado.

2.4.3. Datos experimentales

Todos los experimentos realizados, así como en el artículo previo (González et al., 2018) han sido realizados sobre el mismo modelo DEA. También se van a utilizar los mismos paquetes de optimización (CPLEX y GUROBI), aprovechando así los experimentos previamente realizados para reforzar su estudio. El cambio más importante en estos nuevos experimentos es la utilización de una máquina mucho más potente, que nos permite evaluar ambos niveles de paralelismo por separado, y dividir recursos entre ambos. El hardware utilizado es un nodo DELL PowerEdge R730 con 2 Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30 GHz, con 20 cores (40 threads) a 2.4 GHz y 25 MB de memoria SmartCache.

En este artículo, varios tamaños de problema para el modelo DEA son fijados, generando aleatoriamente las entradas y salidas, de forma que los experimentos no sean dependientes del tamaño del problema. Los tamaños elegidos son los siguientes:

- Tamaño 1: m = 3/n = 50/s = 1
- Tamaño 2: m = 4/n = 50/s = 2
- Tamaño 3: m = 5/n = 250/s = 3

Antes de comenzar a evaluar el rendimiento del algoritmo paralelo, se ha realizado un experimento previo para analizar el comportamiento de los paquetes de optimización cuando se ejecutan de forma paralela. Cuando hablamos de ejecutar un optimizador de forma paralela no nos referimos a sus funcionalidades internas, sino a ejecutar varias instancias del mismo optimizador de forma simultánea. De esta forma, podemos estudiar el tiempo de cómputo de un único problema, en función de la cantidad de

instancias que se estén ejecutando en el mismo momento en la misma máquina.

La figura 23 muestra el tiempo que tarda cada paquete de optimización en resolver un único problema (en milisegundos) cuando, al mismo tiempo, existen diferentes instancias realizando el mismo trabajo. La cantidad de instancias dependerá del número de procesadores utilizados.

El tiempo siempre esta medido sobre el mismo procesador (número 0) resolviendo el mismo problema. En este experimento únicamente se va a evaluar un problema referente al modelo DEA (DMU 1 de n). En la figura 23 se puede observar cómo, a medida que aumenta el número de instancias simultáneas, el tiempo de cómputo aumenta cada vez en mayor medida. Esto implica que el número de instancias que se pueden ejecutar en paralelo de estos optimizadores no es ilimitado, llegando en ciertos momentos a colapsar entre ellas.

Analizando la figura 23, podemos asegurar que, a partir de 10 instancias simultaneas, el tiempo de cómputo comienza a aumentar, siendo ineficiente el uso de más recursos.

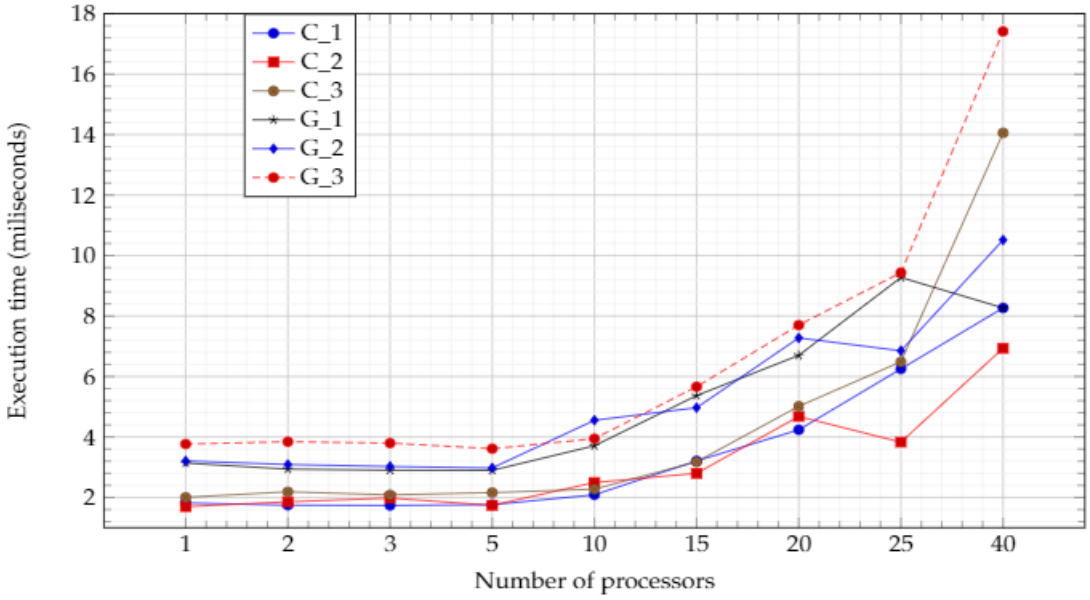

Figura 23. Comparativa del tiempo de cómputo para diferentes tamaños de problema y diferentes paquetes de optimización en la evaluación de un único problema.

Una vez comprobado y asegurado que el paradigma MPI va a ser utilizado en la capa superior de paralelismo (separación de problemas en diferentes núcleos) y, por tanto, OpenMP será utilizado en la optimización interna de la Matheurística, procedemos a comprobar la eficiencia de ambos niveles de forma separada. Para este experimento se va a utilizar únicamente el problema de tamaño 1, ya que entendemos que el resto de tamaños se comportarán de igual forma. Al mismo tiempo, se ha decidido asignar todos los recursos a cada uno de los niveles por separado, y comprobar así si ambos niveles generan un beneficio en cuanto al tiempo de cómputo, reduciéndolo a medida que aumentan los recursos utilizados. Además, se han comprobado ambos modelos de programación paralela con los dos paquetes de optimización propuestos. En este experimento se ha utilizado un hardware mayor, para así poder analizar el rendimiento a una escala mayor. De esta forma, contamos con las siguientes configuraciones para este experimento:

- MPI [CPLEX]
- MPI [GUROBI]
- OpenMP [CPLEX]
- OpenMP [GUROBI]

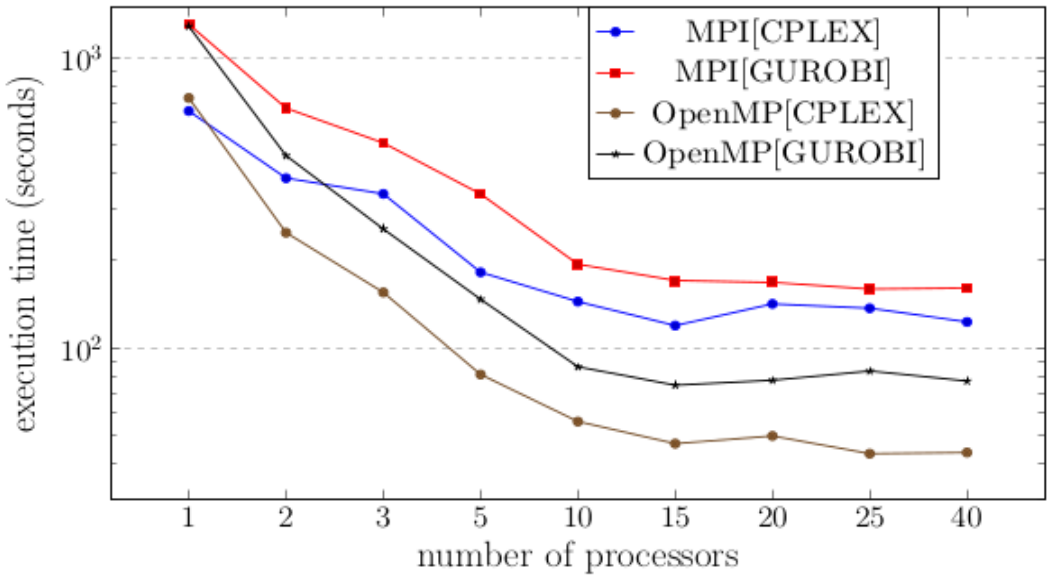

Figura 24. Comparativa entre diferentes niveles de paralelismo y optimizadores.

Como se aprecia en la Figura 24, la optimización es clara en todos los niveles y optimizadores, disminuyendo a medida que aumenta el número de recursos, como ya pasaba en experimentos anteriores. Además, podemos ver cómo el nivel de división de problemas en diferentes núcleos (MPI) converge antes que OpenMP. Esto confirma lo visto en el primer experimento. Otro dato importante es ver cómo el nivel interno de la Matheurística, el destinado a OpenMP, ejerce una gran optimización sobre el tiempo de cómputo. Podemos concluir con este experimento que ambos niveles son importantes, y que su ejecución simultánea generará una optimización mayor. Para conocer cuál sería el reparto óptimo de recursos entre los distintos niveles, se ha procedido a realizar ciertas configuraciones y analizar los tiempos obtenidos. La Tabla 9 muestra los resultados.

Hybrid Parallel Configuration							
GUROBI	CPLEX	MPI	OpenMP	GUROBI	CPLEX	MPI	OpenMP
$223.7383_{16.4439}$	$137.3334_{3.3673}$	2	2	$104.2315_{8.9342}$	$66.4368_{5.7361}$	3	3
$81.3666_{5.8771}$	$59.2839_{1.9190}$	4	3	$\mathbf{59.6567}_{2.1628}$	$45.1889_{2.8821}$	4	5
$\mathbf{68.5390}_{3.6255}$	$51.4602_{3.0931}$	5	4	$\mathbf{56.6961}_{2.7679}$	$\mathbf{40.6429}_{1.5087}$	3	8
$\mathbf{53.2853}_{2.2711}$	$\mathbf{42.5253}_{1.9582}$	5	8	$\mathbf{70.2143}_{6.3089}$	$54.2284_{2.3153}$	6	4
$\mathbf{52.8286}_{2.9948}$	$\mathbf{40.0809}_{2.4266}$	2	20	$\mathbf{71.9496}_{4.2597}$	$75.3947_{4.6624}$	20	2
$\mathbf{49.8065}_{1.9282}$	$\mathbf{37.3163}_{2.4220}$	3	13	$75.7056_{4.4348}$	$62.6644_{2.2626}$	13	3
$\mathbf{50.8675}_{2.3173}$	$\mathbf{39.1378}_{1.8670}$	4	10	$\mathbf{70.8639}_{4.0900}$	$56.7864_{2.6158}$	10	4

Tabla 9. Comparación entre diferentes configuraciones de MPI y OpenMP en un modo híbrido.

La Tabla 9 confirma todo lo dicho anteriormente. El tiempo de cómputo se reduce en mayor medida destinando pocos recursos a MPI (2-4 núcleos) y el resto de recursos a OpenMP (10-20 threads). Además, también podemos apreciar en esta tabla como CPLEX es el optimizador que mejor reacciona a este tipo de algoritmos paralelos, en donde existen numerosas instancias del software ejecutándose simultáneamente.

2.4.4. Conclusión

En este artículo se continua con el algoritmo paralelo propuesto en Martin et al., 2018. Se amplía la investigación proponiendo nuevos experimentos, con los cuales se demuestra la eficacia del algoritmo, sobre todo, configurando de forma óptima los recursos del sistema.

De esta forma, también se estudia el comportamiento de los paquetes de optimización CPLEX y GUROBI cuando se ejecutan de forma simultánea, tanto en memoria compartida como distribuida.

Para concluir, se demuestra cómo, lo más eficiente es destinar muchos recursos a la optimización interna de la matheurística, sin pasar por alto que el nivel superior de MPI también es importante.

CAPÍTULO 3

Conclusions

While it is true that there are many alternative non-radial measurements in DEA, those based on the determination of the least distance provide useful benchmarking information. Among the set of non-radial DEA measurements associated with the determination of the least distance, we opted for the output-oriented Russell measure. However, we acknowledge that other least distance based efficiency measures could also be used in empirical applications.

Therefore, this thesis uses an MILP mathematical model included within NP-hard optimization problems, based on combinations of elements, this being a complex problem when evaluating at a computational level. An efficient decomposition strategy has been developed for MILP optimization problems; a hierarchical decomposition based on the nature of the decision variables (continuous versus discrete) and the complexity of the subproblems. For this, the characteristics of the metaheuristic models are used to explore an incomplete encoding, representing only discrete decision variables.

On the other hand, the encoding of solutions is completed for the decision continuous variables by solving a linear problem exactly. This framework, thus created, has proven its worth in solving MILP-type problems. Furthermore, a Hyper Matheuristic methodology has been

developed on the parameterized metaheuristic scheme. It allows automatic design and configuration of a flexible and generic template for population-based metaheuristics. Satisfactory results have been obtained in terms of solution quality and runtime. Finally, and given the amount of time required to solve this type of problem with the proposed techniques, parallel algorithms are proposed as good solutions to solve this type of problem. This is because a large number of independent problems have to be solved, and those problems can be distributed and solved by different threads independently.

The parallel performance of the optimization packages used must be taken into account. In this article, two of the most common in the literature, CPLEX and GUROBI, have been evaluated, checking their performance in parallel optimization. The parallel algorithm proposed in this article works optimally with both optimizers and with different parallel paradigms. In this way, the behavior of the CPLEX and GUROBI optimization packages is also studied when they are executed simultaneously, both in shared and distributed memory.

Conclusiones

Si bien es cierto que existen muchas medidas no radiales alternativas en DEA, las que se basan en la determinación de la distancia mínima proporcionan información útil de evaluación comparativa. Entre el conjunto de medidas DEA no radiales asociadas a la determinación de la distancia mínima, optamos por la medida de Russell orientada a la salida. Sin embargo, reconocemos que otras medidas de eficiencia basadas en la distancia mínima también podrían usarse en aplicaciones empíricas.

Por lo tanto, esta tesis utiliza un modelo matemático MILP incluido dentro de los problemas de optimización NP-duro, basados en combinaciones de elementos, siendo este un problema complejo a nivel computacional, sobre todo a la hora de evaluar las posibles soluciones. Se ha desarrollado una estrategia de descomposición eficiente para problemas de optimización MILP; una descomposición jerárquica basada en la naturaleza de las variables de decisión (continuas y discretas) y la complejidad de los subproblemas. Para ello, se utilizan las características de los modelos metaheurísticos para explorar una codificación incompleta, representando solo variables de decisión discretas.

Por otro lado, la codificación de soluciones se completa para las variables continuas de decisión al resolver un problema lineal de forma exacta. Este desarrollo, así creado, ha demostrado su valor para resolver problemas de tipo MILP. Además, se ha desarrollado una metodología

hipermateurística sobre el esquema metaheurístico parametrizado. Permite el diseño y configuración automáticos de una plantilla flexible y genérica para metaheurísticas basadas en población. Se han obtenido resultados satisfactorios en términos de calidad de solución y tiempo de ejecución.

Finalmente, y dada la cantidad de tiempo necesario para resolver este tipo de problemas con las técnicas propuestas, los algoritmos paralelos se proponen como buenas soluciones para resolver este tipo de problemas en un tiempo razonable. Esto se debe a que se debe resolver una gran cantidad de problemas independientes, y esos problemas pueden ser distribuidos y resueltos por diferentes subprocesos de forma independiente.

Se debe tener en cuenta el rendimiento paralelo de los paquetes de optimización utilizados. En el último artículo propuesto se han evaluado dos de los más habituales en la literatura, CPLEX y GUROBI, comprobando su rendimiento en optimización paralela. El algoritmo paralelo propuesto en este artículo funciona de manera óptima tanto con optimizadores como con diferentes paradigmas paralelos. De esta forma, también se estudia el comportamiento de los paquetes de optimización CPLEX y GUROBI cuando se ejecutan simultáneamente, tanto en memoria compartida como distribuida.

CAPÍTULO 4

Separatas

- *Aparicio, J., Cordero, J. M., Gonzalez, M., & Lopez-Espin, J. J. (2018). Using non-radial DEA to assess school efficiency in a cross-country perspective: An empirical analysis of OECD countries.*

Contents lists available at ScienceDirect

journal homepage: [redacted]

Using non-radial DEA to assess school efficiency in a cross-country perspective: An empirical analysis of OECD countries☆

Juan Aparicio[a], Jose M. Cordero[b,*], Martin Gonzalez[a], Jose J. Lopez-Espin[a]

[a] Center of Operations Research (CIO), University Miguel Hernandez of Elche (UMH), 03202 Elche (Alicante), Spain
[b] Department of Economics, University of Extremadura (UEX), 06006 Badajoz, Spain

ARTICLE INFO

Article history:
Received 23 January 2017
Accepted 21 July 2017
Available online 26 July 2017

Keywords:
Data envelopment analysis
Non-radial measures
Least distance
Education

ABSTRACT

In this paper we use data from OECD countries participating in PISA 2012 to assess the efficiency of schools in a cross-country framework. In the analysis, and in contrast to previous applications, we consider that schools might concentrate their efforts on improving the results in one dimension of the educational output to a greater extent than in the other. To do this, we rely on non-radial efficiency measures of performance and the estimation of an educational production function based upon Data Envelopment Analysis (DEA) techniques. Specifically, DEA non-radial measures allow for identifying different levels of inefficiency for each output considered (reading and maths). In particular, we apply a non-radial measure based on Ando et al. [5] and Aparicio et al. [12]. Our results show that the majority of schools in OECD countries tend to be less efficient in reading than in mathematics.

1. Introduction

The participation of the majority of nations in international large-scale comparative studies in education has provided researchers with rich and extensive cross-national databases that can be used to assess the performance and effectiveness of educational systems. As a result, comparative education studies have become increasingly popular in education sciences today [49], since researchers can look at the entire world as a natural laboratory to view the multiple ways in which societal factors, educational policies and practices may vary across countries [19].

Most studies adopting a cross-country perspective are situated within the field of educational effectiveness research, which explores the main determinants of educational achievement using an econometric approach to estimate an equation in the form of an educational production function (see [32,51,78]). This strand of literature investigates how inputs are statistically related with outputs. However, the potential existence of an unexpected level of inefficiency in the performance of students, schools or educational systems also needs to be considered [57]. In this sense, the existing constraints of resources faced by most countries and the great amount of national income devoted to educational costs, policy makers and researchers have become increasingly concerned with assessing the efficiency of schools, although until now most of existing literature has been devoted to assessing schools operating in the same country or region.[1]

To the best of our knowledge, only few studies have applied frontier methods to micro data from those international datasets to evaluate the performance of educational systems using a cross-country approach. This line of research includes several works using data that has been aggregated at a country level from different samples of countries participating in international tests such as PISA [1,14,43,76,81]; Agasisti [3,45] or TIMSS [44]. Likewise, we can also find studies that compare the performance of educational systems in different countries using data at school level. For example, Sutherland et al. [75] study the performance of schools from 30 OECD countries participating in PISA 2003; Agasisti and Zoido [4] derives efficiency measures for more than 8600 schools in 30 countries using PISA 2012 data comparing efficiency scores and measures of equity; Cordero et al. [31] evaluates performance using the metafrontier framework to compare and decompose the technical efficiency of primary schools from 16 European countries participating in PIRLS 2011. Finally, De Jorge and Santín [34] and Deutsch et al. [37] use PISA data at a student level to estimate the efficiency of EU and Latin American countries, respectively.

Those studies predominantly use nonparametric techniques like DEA or FDH [24,36]. These methods are generally based on Farrell–Debreu radial efficiency measures, i.e. they reflect the ability of the unit to increase different outputs (e.g. test scores in maths

☆ This manuscript was processed by Associate Editor Chien-Ming Chen.

* Corresponding author.
E-mail address: jmcordero@unex.es (J.M. Cordero).

[1] Recent literature reviews on efficiency in education include De Witte and López-Torres [35], Johnes [55] and [48].

http://dx.doi.org/10.1016/j.omega.2017.07.004
0305-0483/

Table 1
Ranking of OECD countries according to results in maths and reading.

Country	Maths	Rank	Reading	Rank
Korea	554	1	536	2
Japan	536	2	538	1
Switzerland	531	3	509	12
Netherlands	523	4	511	10
Estonia	521	5	516	7
Finland	519	6	524	3
Canada	518	7	523	5
Poland	518	8	518	6
Belgium	515	9	509	11
Germany	514	10	508	13
Austria	506	11	490	21
Australia	504	12	512	9
Ireland	501	13	523	4
Slovenia	501	14	481	29
Denmark	500	15	496	18
New Zealand	500	16	512	8
Czech Rep.	499	17	493	19
France	495	18	505	14
UK	494	19	499	16
Iceland	493	20	483	28
Luxembourg	490	21	488	24
Norway	489	22	504	15
Portugal	487	23	488	25
Italy	485	24	490	20
Spain	484	25	488	23
Slovak Rep.	482	26	463	32
USA	481	27	498	17
Sweden	478	28	483	27
Hungary	477	29	488	22
Israel	466	30	486	26
Greece	453	31	477	30
Turkey	448	32	475	31
Chile	423	33	441	33
Mexico	413	34	424	34

and reading) equiproportionately. Nevertheless, schools sometimes might concentrate their efforts on improving the results in one dimension of the educational outcome more than in other one, thus there might be trade-offs between outputs that cannot be identified through radial efficiency measures. This intuition arises from the fact that there is relevant cross-country divergence in test results between maths and reading. Table 1 illustrates this evidence by showing the ranking of OECD countries in these subjects. Although, they are essentially similar, it is possible to observe some countries with relatively better results in maths (Switzerland, Netherlands, Austria or Slovenia) or in reading (Ireland, New Zealand or United States).

In this paper we would like to explore the potential existence of these trade-offs between reading and maths in an assessment of the performance of schools from all OECD countries participating in PISA 2012 adopting a cross-national framework. For this purpose, we rely on non-radial efficiency measures, which do not require an equiproportional increase in all the considered outputs, thus we can calculate different projections on the frontier for each output included in the production function. This possibility allows us to detect whether some schools may be more efficient in promoting their students´proficiency in reading, while other schools could be more prone to enhance the results in mathematics. Likewise, the proposed approach can also be useful to explore other potential trade-offs between educational outcomes such as the relationship between cognitive and non-cognitive skills [30][2] or educational inequality and average achievement [45,76].[3] Moreover, since we adopt a cross-national framework, we can derive some interesting insights about the average performance of schools from the same country, making it possible to construct different ranking of countries according to the levels of efficiency demonstrated in promoting different educational outcomes. This is an unusual approach in the literature on efficiency measurement in the educational sector, since most studies tend to analyze the performance of schools treating test scores as the unifying outcome that needs to be improved.

Some research in the literature has sought to construct different non-radial efficiency measures such as the Russell measure [38,66], the additive model [25], the slacks-based measure [79] or the Multi-directional Efficiency Analysis (MEA) approach of Bogetoft and Hougaard [18] and Asmild et al. [15]. In our case, we apply a recent methodology, fundamentally based on the application of the 'output-oriented' version of the Russell measure that determines the closest targets and the least distance to the strongly efficient frontier in DEA, based on Bilevel Linear Programming [12]. In addition, with the aim of satisfying monotonicity, a correction of this model is proposed based on Ando et al. [5]. Given that Ando et al. [5] did not show how to implement their approach without previously determining the explicit characterization of the set of points belonging to the strongly efficient frontier, we show for the first time in this paper how this methodology can be implemented in practice.

Moreover, we examine the potential determinants of existing divergence in the setting of production targets across schools and countries using a two-stage approach (DEA and regression). Among potential drivers of schools´performance we distinguish between school factors in the surrounding context and variables representing students´attitudes toward mathematics and reading with the aim of exploring whether schools with students devoting more time to one of those subjects tend to concentrate their effort in promoting that subject or the other one.

The contribution of this paper is threefold. First, as we are aware, there is no paper in the literature devoted to the estimation of technical efficiency in education for OECD countries that applies non-radial measures. So, in this sense, the empirical application that we present is original. Second, our analysis allows us to determine, for the first time, which dimension (in particular, between reading and maths) presents more technical inefficiency for schools in OECD countries. Third, as Aparicio [13] argues, current methodologies associated with the determination of the least distance in DEA lack real applications. Therefore, this paper represents an example of the use of this type of techniques, which permits the implementation of the Principle of Least Action in DEA [9].

The remainder of the paper is structured as follows. Section 2 describes the methodology. Section 3 explains the main characteristics of the data and the variables selected for the empirical analysis. Section 4 presents the main results. Finally, the paper ends with some concluding remarks in Section 5.

2. Methodology

Data Envelopment Analysis was introduced by Charnes et al. [24] under constant returns to scale for multiple inputs and outputs and later extended by Banker et al. [16] to variable returns to

[2] The relationship between those dimensions of educational outcomes is usually difficult to explore due to the difficulties of establishing a standard definition for non-cognitive skills. However, recently, the PISA survey has added some component tests designed to capture aspects of non-cognitive skills including openness, locus of control, and motivation [66].

[3] In their empirical analysis of the performance of different countries participating in PISA, Giménez et al. [45] claim that "*for a given country, the results of the educational process should not be constrained to the knowledge students acquire at school, but should also include other outcomes such as the standard deviation of test scores (an undesirable outcome of the educational process, in terms of educational inequality)*".

scale. Nevertheless, Farrell [39] was the first in showing, for a single output and multiple inputs, how to estimate an isoquant enveloping all the observations, implementing the seminal ideas of Shephard [71] regarding the input distance function. In all these cases, the technical efficiency assessment of a Decision Making Unit (DMU) is based upon an 'oriented' measure of distance, which identifies a point on the isoquant of the technology with the same mix of inputs (input orientation) or outputs (output orientation) of that of the evaluated unit. The conservation of this mix in movements toward the boundary of the technology is the characteristic that gives a radial measure.

However, many real-life situations require non-radial measures of technical efficiency to be used. Any measure in DEA that does not adopt equi-proportional reductions of inputs or outputs is non-radial. Indeed, a well-known drawback of radial measures is the arbitrariness in imposing targets on the isoquant preserving the mix within inputs or within outputs, depending on the selected orientation of the model, when the firm's very reason to change its input/output levels might often be the desire to actually change that mix due to their differing opportunity costs (see [23], and [68]). In particular, in the context of education, some units (schools, for example) could be tempted, directly or indirectly, to upgrade some specific dimension, like science or mathematics, due to, for example, cultural characteristics and traditions inherent to the country where they are geographically located. Additionally, from a DMU point of view, it may mainly be interested in the easiest way of being classified as technically efficient (especially for public "production" like schools, and the regulatory pressure that could arise from being classified as inefficient rather than efficient). This type of benchmarking strategy will be the focus of our contribution in this paper.

Regarding the existence of non-radial oriented models, DEA endows practitioners with a toolbox full of possibilities. The first approach in this respect was due to Färe et al. [38], who introduced the Russell input and output measures of technical efficiency. After that, other oriented and non-radial measures were defined seeking more flexibility than that provided by the radial measures as, for example, the directional distance function [22] or the weighted additive models (see [47] or [28]). In contrast to these two last types of technical efficiency measures, the Russell input and output measures present some interesting properties. First, the directional distance function does not correspond to the Pareto–Koopmans definition of technical efficiency [56]. This implies that it ignores the possible existence of slacks associated with the projected points on the boundary of the technology. In other words, the directional distance function neglects some additional sources of technical inefficiency. Conversely, the Russell measures always generate non-dominated projection points in the corresponding input/output space. Second, in contrast to the weighted additive models, which aggregate slacks by a weighted scheme, the interpretation of the Russell measures is easier. In particular, the value of the Russel output measure can be interpreted as the average of proportional rates of output expansion needed to be technically efficient.

Regarding the Russell measures, we want to highlight that there are two clearly different paradigms for determining them nowadays. On the one hand, we have the traditional approach, which is associated with the identification of demanding targets. The targets are specifically the coordinates of the projection point on the boundary of the technology and thus represent levels of operation that would make the evaluated DMU perform efficiently. This first philosophy is followed by the original definition of the Russell input and output measures [38], where the total technical effort required by a DMU to become technically efficient is maximized instead of minimized, thereby generating the furthest projection points on the frontier. On the other hand, a recent proposal

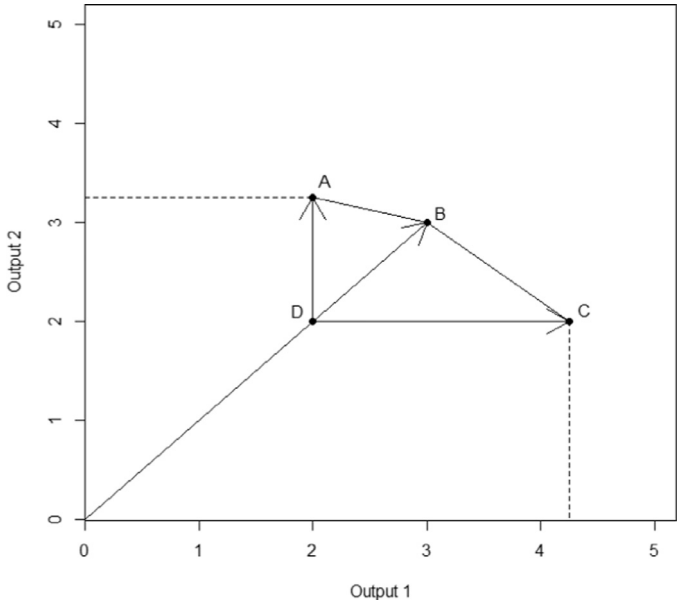

Fig. 1. Differences between radial and non-radial measures.

[12] has suggested determining the closest efficient targets through the oriented Russell measures, minimizing, instead of maximizing, the corresponding technical effort in order to reach the frontier. This second approach follows a well-known line of research in the DEA literature related to the determination of the least distance to the efficient frontier, the identification of closest targets and the application of the Principle of Least Action (see, for example, [7–11,21,40,67,68], Fukuyama et al. [42] and Ruiz and Sirvent [69]). Next, we graphically illustrate the main differences between the Russell measures under the two philosophies and their comparison with the radial measure (see Fig. 1). To do that, we focus our simple example on the Russell output measure of technical efficiency.

In Fig. 1, an output production set in two dimensions is represented. In this simple example, the only technically inefficient DMU is unit D = (2, 2). If it is assessed by the output-oriented radial measure, then unit B = (3, 3) is its corresponding projection point with a radial score of 1.5. This same value can be interpreted in terms of the Russell measures as the average of output changes associated with the radial projection: $1.5 = 0.5 \cdot (\frac{3}{2} + \frac{3}{2})$. In the case of the traditional 'non-radial' Russell output measure, the projection point on the isoquant would be unit C (4.25, 2), which produces an average of proportional rates of output expansion equals $0.5 \cdot (\frac{4.25}{2} + \frac{2}{2}) = 1.5625$. However, there is also an alternative projection point on the frontier that yields the least value for the average of output expansions. We are referring to unit A (3.25, 2): $0.5 \cdot (\frac{2}{2} + \frac{3.25}{2}) = 1.3125 < 1.5 < 1.5625$. This last projection corresponds to the approach by Aparicio et al. [12], which, in general, determines the Pareto-efficient point more easily achievable for the evaluated DMU. Note that, for DMU D, it is possible to be technically efficient improving output 2 only by 62.5%, instead of enhancing output 1 and output 2 by 50% (radial projection) or increasing output 1 only by 112.5% (traditional Russell projection).

Other key issue in the measurement of technical inefficiency is the satisfaction of some interesting properties. One of them is monotonicity. Monotonicity relates the notion of efficiency to Pareto optimality. Specifically, if unit A dominates unit B, in the Pareto sense, then the measure of technical inefficiency associated to A should be less than the measure of technical inefficiency of B. Briec [21] proved that Hölder distance functions meet weak

monotonicity over the weakly efficient frontier. Ando et al. [5] were the first in showing that Hölder distance functions do not meet strong monotonicity on the strongly efficient frontier and suggested a solution for satisfying weak monotonicity on the strongly efficient frontier. Later, Aparicio and Pastor [8] proved that the output-oriented version of the Russell measure is a well-defined efficiency measure, satisfying strong monotonicity on the strongly efficient frontier, if efficiency is evaluated with respect to an extended facet production possibility set based on Full Dimensional Efficient Facets (FDEF) instead of the standard DEA technology.

Indeed, Aparicio and Pastor [10] showed that this drawback of the Hölder norms is associated with the dimensionality of the strongly efficient frontier. On the other hand, Fukuyama et al. [41,42] showed an alternative to the extension of FDEFs for endowing least distances with the property of strong monotonicity. Their approach is based on transforming the traditional definition of the measure introducing in the corresponding optimization model an auxiliary point that is dominated by the evaluated unit, and that it is used to calculate the least distance to the strongly efficient frontier. Finally, Ando et al. [6] proved the strong monotonicity property for the input and output oriented DEA models based on the Hölder norms in the context of two inputs and two outputs, respectively, as well as the strong monotonicity for non-oriented DEA models in two dimensions (one input – one output). Additionally, they reviewed the minimum distance inefficiency measures based on the extended facet approach, and discussed the problems of unmeasurability and overestimation. All this means that the least distance measure proposed in Aparicio et al. [12], as defined originally, satisfies neither weak nor strong monotonicity. Nevertheless, according to the above discussion, the technology or the measure could be modified in order to try to meet the property. In particular, in this paper, we opt for the transformation of the inefficiency measure *a la* Ando et al. [5] in order to correct the problem since the modification of the traditional DEA technology requires to detect the set of all FDEFs, if they exist. Additionally, given that Ando et al. [5] did not provide the way of implementing their approach without an explicit description of the strongly efficiency frontier, we show for the first time how it can be carried out using an implicit characterization of this set of the frontier. Thanks to the correction of the original model of Aparicio et al. [12], we can be sure that the proposed measure in this paper satisfies at least weak monotonicity in the output-oriented context.

Definition 1. Let $I : R^m_+ \times R^s_+ \to R_+$ be an output-oriented inefficiency index. I satisfies weak monotonicity if $I(x, y) \leq I(x, \tilde{y})$ for all feasible vectors (x, y) and $(x, \tilde{y})$ with $y \geq \tilde{y}$.

Moreover, implementing in practice the approach based on the determination of closest targets is not easy from a computational point of view [20]. This difficulty is consequence of the complexity of determining the least distance to the frontier of a DEA technology (a polyhedral set) from an interior point (inefficient DMU). For this reason, Aparicio et al. [12] introduce a new methodology to implement this approach in the context of the DEA-oriented measures. Their method is based upon Linear Bi-level Programming (LBP).

A Bilevel Programming model refers to a mathematical programming problem where one of the constraints is an optimization problem. This theory has been successfully applied to model different real situations with a common feature: the existence of a hierarchical structure (see [85]). A Bilevel Programming problem where both the objective functions and the constraints are linear is called a Bilevel Linear Programming problem. Denote by $z \in Z \subset R^p$ and $t \in T \subset R^q$ the decision variables corresponding to the first and second level, respectively. The general formulation of a Bilevel Linear Programming (BLP) problem is as follows:

$$\begin{aligned} &\underset{z,t}{Min} && c_1 z + d_1 t \\ &s.t. \\ &&& A_1 z + B_1 t \leq b_1, \\ &&& \underset{t}{Min}\ c_2 z + d_2 t \\ &&& s.t. \\ &&&\quad A_2 z + B_2 t \leq b_2, \\ &&& z \geq 0,\ t \geq 0 \end{aligned} \tag{1}$$

Program (1) consists of two subproblems. On the one hand, the higher level decision problem and, the other hand, the lower level decision problem, which appears as a constraint in (1). Both problems are connected in a way that the higher problem sets parameters influencing the lower level problem and the higher problem, in turn, is affected by the outcome of the lower level problem.

It is known that even for the Bilevel Programming problem where all the functions are linear, like in (1), the model to be solved is non-convex and NP-hard. This complexity is the reason why many different techniques have been proposed in the literature to study the computational aspects of Bilevel Programming problems. The formulation of optimality conditions for this type of problems usually starts with a suitable reformulation of the problem as a one-level model. One possibility is to transform the original problem into a single optimization problem by applying the well-known Karush–Kuhn–Tucker (KKT) optimality conditions of the lower level problem [72].

Regarding the solutions of a BLP problem, $(z^*, t^*) \geq 0$ is a feasible solution of (1) if t^* is an optimal solution of the lower level program with $z = z^*$ and, at the same time, $A_1 z^* + B_1 t^* \leq b_1$. In this way, (z^*, t^*) is an optimal solution if additionally $c_1 z^* + d_1 t^* \leq c_1 z + d_1 t$ for all feasible solution (z, t) of (1), being $c_1 z^* + d_1 t^*$ the corresponding optimal value of the BLP problem.

Aparicio et al. [12] proposed the following model for calculating the Russell output measure based on the least distance philosophy for the DMU_0:

$$\begin{aligned} &\underset{\phi,\lambda,s^+,\gamma}{Min} && \frac{1}{s}\sum_{r=1}^{s}\phi_r && && (2.1) \\ &s.t. \\ &&& \sum_{j\in E_{VRS}} \lambda_j x_{ij} \leq x_{i0}, && i = 1,\ldots,m && (2.2) \\ &&& \sum_{j\in E_{VRS}} \lambda_j y_{rj} \geq \phi_r y_{r0}, && r = 1,\ldots,s && (2.3) \\ &&& \sum_{j\in E_{VRS}} \lambda_j = 1, && && (2.4) \\ &&& \sum_{r=1}^{s} s_r^+ = 0, && && (2.5) \end{aligned} \tag{2}$$

$$\begin{aligned} &\underset{s^+,\gamma}{Max} && \sum_{r=1}^{s} s_r^+ && && (2.6) \\ &s.t. \\ &&& \sum_{j\in E_{VRS}} \gamma_j x_{ij} \leq x_{i0}, && i = 1,\ldots,m && (2.7) \\ &&& \sum_{j\in E_{VRS}} \gamma_j y_{rj} = \phi_r y_{r0} + s_r^+, && r = 1,\ldots,s && (2.8) \\ &&& \sum_{j\in E_{VRS}} \gamma_j = 1, && && (2.9) \\ &&& \phi_r \geq 1,\ \lambda_j,\ s_r^+,\ \gamma_j \geq 0, && \forall r, j && (2.10) \end{aligned}$$

where $x_j = (x_{1j}, \ldots, x_{mj}) \in R^m_{++}$ denotes the inputs and $y_j = (y_{1j}, \ldots, y_{sj}) \in R^s_{++}$ denotes the outputs for a sample of $j = 1, \ldots, n$ observations (DMUs), and E_{VRS} denotes the set of extreme efficient units in the case of assuming variable returns to scale (VRS).

In (2), (2.1)–(2.4) coincide with the constraints of the traditional Russell output measure of technical efficiency ([38], p. 149) except

for the fact that the objective function is minimized instead of maximized as happens with the original definition of the Russell measure, while the lower level problem, (2.6)–(2.9) is an output-oriented version of the additive model in DEA [25] for evaluating the Pareto-efficiency of $(\phi_1 y_{10}, ..., \phi_s y_{s0})$. Constraint (2.5) ensures that the optimal $(\phi_1^* y_{10}, ..., \phi_s^* y_{s0})$ is not dominated.

Note that model (3) is mathematically equivalent to a model where (2.1) is changed by $1 + Min\{\frac{1}{s}\sum_{r=1}^{s}\frac{t_r}{y_{r0}}\}$, (2.3) substituted by $\sum_{j\in E_{VRS}}\lambda_j y_{rj} \geq y_{r0} + t_r$ and (2.8) by $\sum_{j\in E_{VRS}}\gamma_j y_{rj} = y_{r0} + t_r + s_r^+$, with $t_r \geq 0$, $r = 1, ..., s$.

Now, with the aim of guaranteeing weak monotonicity when the above measure is calculated on the strongly efficient frontier, we need to modify (2) *a la* Ando et al. [5]. To do that, we need to include in the constraints a generic point $(y_1, ..., y_s) \in R_+^s$ such that it is dominated by the assessed unit in the sense of Pareto, i.e., $y_{r0} \geq y_r$, for all $r = 1, ..., s$. The corresponding Bilevel Programming problem for evaluating DMU_0 would be then as follows.

$$
\begin{aligned}
\underset{\phi,\lambda,y,s^+,\gamma}{Min}\quad & 1 + \frac{1}{s}\sum_{r=1}^{s}\frac{t_r}{y_{r0}} && && (3.1)\\
s.t.\quad & \sum_{j\in E_{VRS}}\lambda_j x_{ij} \leq x_{i0}, && i = 1, \ldots, m && (3.2)\\
& \sum_{j\in E_{VRS}}\lambda_j y_{rj} \geq y_r + t_r, && r = 1, \ldots, s && (3.3)\\
& y_r \leq y_{r0}, && r = 1, \ldots, s && (3.4)\\
& \sum_{j\in E_{VRS}}\lambda_j = 1, && && (3.5)\\
& \sum_{r=1}^{s} s_r^+ = 0, && && (3.6)\\
\underset{s^+,\gamma}{Max}\quad & \sum_{r=1}^{s} s_r^+ && && (3.7)\\
s.t.\quad & & & & & \\
& \sum_{j\in E_{VRS}}\gamma_j x_{ij} \leq x_{i0}, && i = 1, \ldots, m && (3.8)\\
& \sum_{j\in E_{VRS}}\gamma_j y_{rj} = y_r + t_r + s_r^+, && r = 1, \ldots, s && (3.9)\\
& \sum_{j\in E_{VRS}}\gamma_j = 1, && && (3.10)\\
& t_r,\ \lambda_j,\ y_r, s_r^+,\ \gamma_j \geq 0, && \forall r, j && (3.11)
\end{aligned}
\tag{3}
$$

Regarding the computation of (3), even for the Bilevel Programming problem where all the functions are linear, as in (3), the model to be solved is non-convex and NP-hard [27]. In this paper, as in Aparicio et al. [12], we resort to reformulate (3) as a one-level model, substituting the lower level problem by its corresponding Karush–Kuhn–Tucker (KKT) optimality conditions [72]. Accordingly, (3.7)–(3.10) must be substituted by (4.1)–(4.9).

$$
\begin{aligned}
& \sum_{j\in E_{VRS}}\gamma_j x_{ij} + l_i = x_{i0}, && i = 1, \ldots, m && (4.1)\\
& \sum_{j\in E_{VRS}}\gamma_j y_{rj} = y_r + t_r + s_r^+, && r = 1, \ldots, s && (4.2)\\
& \sum_{j\in E_{VRS}}\gamma_j = 1, && && (4.3)\\
& -\sum_{i=1}^{m}\eta_i x_{ij} + \sum_{r=1}^{s}\mu_r y_{rj} + \psi + \tau_j = 0, && j \in E_{VRS} && (4.4)\\
& -\eta_i + e_i = 0, && i = 1, \ldots, m && (4.5)\\
& \mu_r \geq 1, && r = 1, \ldots, s && (4.6)\\
& \gamma_j \tau_j = 0, && j \in E_{VRS} && (4.7)\\
& l_i e_i = 0, && i = 1, \ldots, m && (4.8)\\
& l_i,\ \eta_i,\ \mu_r,\ \tau_j,\ e_i \geq 0, && \forall i, j && (4.9)
\end{aligned}
\tag{4}
$$

Constraints (4.7)–(4.8) are not linear. Nevertheless, constraints of this nature are not difficult to be implemented through Special Ordered Sets (SOS)[4] [17].

Finally, as we are interested in exploring potential contextual or environmental variables that might affect the performance of the evaluated schools, in a second stage we estimate different Tobit regression models. This approach has been extensively used by in previous literature to analyze dependent variables subject to a known upper or lower bound like our non-radial measures. We are aware that this conventional regression model has been criticized by Simar and Wilson [73], who argue that it might yield biased estimations because the efficiency scores estimated in the first stage are serially correlated. They address this issue by proposing two algorithms that incorporate the bootstrap procedure in a truncated regression model that allow for valid inference while simultaneously generating standard errors and confidence intervals for the efficiency estimates conventional regression methods like Tobit.[5]

The main problem of this approach is that it only considers the radial term of the efficiency in the second stage, thus we decided to use Tobit regressions in our empirical analysis. This choice is supported by two main reasons. First, the consistency of the estimation in a second-stage regression increases as the sample size enlarges, thus the potential bias disappears in large-size samples [88]. Since in our empirical application we have more than 10,000 observations, we consider that our estimates can be consistent. Second, our dataset contains five different plausible values drawn from the estimated distribution of results of each school. Therefore, we are already working with a certain confidence interval from the beginning and there would be no need to apply methods based on resampling like bootstrapping [34]. The estimated results from the Tobit regressions are extremely useful to identify what exogenous variables have a significant influence on both the average efficiency measure of performance and the output-specific score as well as whether the direction of this influence is positive or negative.

3. Data and variables

This section includes an empirical illustration with real data by applying the methodology proposed in the previous section. In particular, we use comparative data about schools operating in the 34 OECD countries participating in the OECD's PISA (*Programme for International Student Assessment*) 2012 survey. The survey takes place every three years, starting in 2000, thus PISA 2012 represents the fifth wave of this study. For each assessment, one of reading, mathematics and science is chosen as the major domain and given greater emphasis. In 2012, the major domain was mathematics (as well as in 2003) and two additional competences were also assessed for the first time (problem solving and financial literacy).

This survey uses a two-stage stratified design sampling [83]. In the first stage of sampling, schools having age-eligible students are sampled systematically with probabilities proportional to the school size. A minimum of 150 schools is selected in each country. Subsequently, 35 15 year-old students are randomly selected from each school to participate in the survey. Data were collected between March and May 2012 for countries in the northern hemisphere and May-August 2012 for countries in the southern hemisphere.

One of the main advantages of using PISA data is that this study does not evaluate cognitive abilities or skills through using one single score but each student receives five different scores (plausible

[4] SOS is a way to specify that a pair of variables cannot take strictly positive values at the same time and is a technique related to using special branching strategies. Traditionally, SOS was used with discrete and integer variables, but modern optimizers, like for example CPLEX, use also SOS with continuous variables.

[5] See Simar and Wilson [73,74] for a technical description of the model.

Table 2
Dataset composition: number of schools in each country.

Country	Schools	Country	Schools
Australia	775	Japan	191
Austria	191	Korea	156
Belgium	287	Luxembourg	42
Canada	885	Mexico	1471
Chile	221	Netherlands	179
Czech Republic	297	New Zealand	177
Denmark	341	Norway	197
Estonia	206	Poland	184
Finland	311	Portugal	195
France	226	Slovak Republic	231
Germany	230	Slovenia	338
Greece	188	Spain	902
Hungary	204	Sweden	209
Iceland	134	Switzerland	411
Ireland	183	Turkey	170
Israel	172	United Kingdom	507
Italy	1194	USA	162
TOTAL		11,767	

values) that represent the range of abilities that a student might reasonably have (see [65] for details). Specifically, the dataset provides measures on students' performance based upon pupils' responses to different test booklets, each of each of which includes only a limited number of test questions. Thus, it is difficult to make claims about individual performance with great accuracy. Using a complex process based on item response theory model, the survey organizers produce test scores for participants taking into account the difficulty of each test question.[6] Plausible values can therefore be defined as random values drawn from this distribution of proficiency estimates [60,86].

In addition, the survey collects a great volume of data about other factors potentially related to those results, such as variables representing student's background, school environment or educational provision. This information comes from the responses given to different questionnaires completed by students and school principals. From these data, it is possible to extract a great amount of information referred to the main determining factors of educational performance.

Our final dataset comprises a total number of 11,767 schools distributed across countries as reported in Table 2. As explained above, the minimum number of participating schools in each country must be 150, although in our sample we have the exceptional case of Luxembourg, where there are only 42 participating schools. Likewise, in several countries the sample is very large due to the existence of representative samples for different regions within the country (e.g. Australia, Canada, Italy, Mexico, Spain).

The output variables are represented by the average test scores achieved by students belonging to the same school in the two most relevant competences: reading and mathematics. Those test scores are obtained from the five plausible values, which have an international mean of 500 and standard deviation of 100. Following the recommendations made by survey organizers [63], our empirical analyses have been performed independently on each of these five plausible values. Due to space restrictions, we only present the estimations obtained with that first value, although the results of the empirical analysis are quite similar to the others.

The selection of inputs is a tough decision, since in the dataset there is an extensive list of potential indicators that can be considered. In this sense, most empirical papers attempting to measure efficiency of schools usually include some measures of human and capital resources [35,84]. In our empirical study we use the inverse of the student–teacher ratio, i.e., the number of teachers per (hundred) students (TEACHERS) and an index representing the quality of school resources (SCMATEDU) created by PISA analysts from the responses given by school principals regarding several aspects (computers, educational software, calculators, books, audio visual resources or laboratory equipment). Moreover, we also consider the average socio-economic status of students in the school (ESCS) as an additional input, since students are the "raw material" to be transformed through the learning process.[7] ESCS stands for the Economic, Social and Cultural Status, and provides a measure of family background that includes the highest levels of parents' occupation, educational resources and cultural possessions at home. Since the original values of SCMATEDU and ESCS presented positive and negative values, all of them were rescaled to show positive values.[8]

Finally, we have also selected some contextual variables in order to explore whether the performance of the evaluated units may be affected by the educational environment or the type of school management. In the following lines we provide a brief explanation about each of the variables selected and the expected direction of its influence according to previous literature.

- School ownership. Recent literature provides some empirical studies assessing whether the public or private nature of the school may affect their level of efficiency. Regarding this issue, in the literature we can find evidence that supports the idea of better performance in private schools [33,59] while others do not find enough evidence to justify this superiority [46,58]. In our case, we have included this information using a dummy variable taking a value of one if the school is private and zero if it is public.
- School location. This is a common factor usually considered as a potential factor affecting the efficiency of schools (see [35]), since whether a school is located in a main urban area or in a less densely populated area may affect its scale of operations and/or its ability to attract teaching staff. In order to take this into account we have defined a dummy variable that takes value one if the institution is located in a small town with less than 15,000 inhabitants.
- Percentage of girls. Various studies have demonstrated that girls usually perform better than boys in reading [62], while in mathematics the situation is the opposite [54]. Since we are interested in exploring whether some schools may concentrate their effort on one competence, this percentage might be a relevant factor in order to explain those potential behaviors.
- Percentage of repeaters. Different meta-analyses and literature reviews about the practice of retaining students have concluded that it has a negative effect on achievement (e.g. [52,53,87]). In our framework, we explore whether schools having a higher proportion of retained students may be more efficient.
- Competition with (at least) another school in the same neighborhood. The reference framework is the idea that the presence of more schools in a certain area should raise the performance of schools operating in that area as a response to pressures from nearby competitors [2,61], thus we expect a positive influence of this dummy variable on efficiency levels.

Moreover, we also take into account several variables representing students´attitudes toward mathematics, the domain about which there is more available information, in order to test whether the orientation of students toward one subject might be relevant in

[6] See Von Davier and Sinharay [82] for further details.

[7] This is a common practice in several recent papers attempting to measure the efficiency of schools (e.g. [3,4,33,70,77]).

[8] The rescaling process was made by adding up the minimum value to all the original values of the variables. This transformation does not alter the efficient frontier (or empirical production function) and hence the associated DEA model is translation invariant.

Table 3
Descriptive statistics of variables included in the analysis.

Variable	Type	Mean	Std. dev.	Min	Max
READING	Output	480.15	68.49	98.23	782.37
MATHS	Output	482.00	70.06	158.39	734.68
ESCS	Input	4.27	0.76	0.01	6.09
SCMATEDU	Input	3.59	1.04	0.008	5.576
TEACHERS	Input	9.76	16.01	0.098	1075.27
School factors					
PRIVATE	Contextual	0.169	0.374	0	1
RURAL	Contextual	0.328	0.469	0	1
PCGIRLS	Contextual	0.479	0.179	0	1
REPEATERS	Contextual	0.160	0.230	0	1
COMPETITION	Contextual	0.725	0.447	0	1
MATHHOMEWORK	Contextual	0.391	0.150	0	1
MATHEXAMS	Contextual	0.440	0.142	0	1
MATHCLASSES	Contextual	0.519	0.134	0	1
ENJOYMATHS	Contextual	0.261	0.136	0	1
READPLEASURE	Contextual	0.323	0.072	0.207	0.582

order to explain potential behaviors of schools concentrating their effort in that subject or another one. Specifically, we select four representative variables: (i) proportion of students that work hard on math homework[9]; (ii) proportion of students that declare to be prepared for maths exams; (iii) proportion of students that usually pay attention in maths classes[10]; and (iv) proportion of students that declare to enjoy maths. Finally, in order to supplement the extensive information available about activities related to mathematics, we have also retrieved some information about reading habits represented by the percentage of students reading for enjoyment at least 30 min a day.[11] Table 3 reports the descriptive statistics for all these variables (outputs, inputs and contextual factors).

4. Results

In this section we report the results obtained applying the closest target approach to the entire sample of schools across all countries. As explained in Section 2, this methodology allows us to determine projection points on the frontier for each output without assuming that they should be equiproportional. However, initially we present them in the form of the traditional non-radial Russell measures, i.e. as the average of the proportional rates of output expansion, so that they can be easier to interpret and compare with other empirical studies using traditional radial measures. Specifically, in Table 4 we show the average values of those "artificial" measures summarized by country. It is worth noting that those scores are presented in the form of Farrell output-oriented efficiency measures, thus values equal to one indicate that a unit is efficient, while values greater to one reflect the percentage of inefficiency.

The mean efficiency score *a la* Russell of the entire sample is 1.367, although the mean values by countries vary substantially, ranging from an inefficiency level of 20% for Korean schools to 52% for Slovenian ones. If we compare those values with average test scores in maths and reading for each country, also shown in Table 4, we observe that there is not always a straightforward relationship between results and efficiency. For instance, Turkey is among the best performers in terms of (aggregate) efficiency

Table 4
Efficiency scores versus average results in maths and reading.

Country	Maths	Reading	Efficiency score
Korea	551	534	1.2032
Japan	534	535	1.2574
Turkey	437	461	1.2758
Poland	527	527	1.2833
Estonia	518	514	1.3003
Ireland	498	520	1.3034
Netherlands	517	505	1.3082
New Zealand	497	511	1.3106
Canada	507	509	1.3193
Germany	507	500	1.3300
Mexico	410	419	1.3433
USA	479	496	1.3433
Czech Rep.	500	496	1.3446
Spain	491	490	1.3487
Finland	508	511	1.3491
Switzerland	514	492	1.3500
UK	488	496	1.3552
Australia	492	500	1.3683
Portugal	479	480	1.3690
Belgium	508	500	1.3763
France	489	498	1.3768
Norway	491	505	1.3911
Austria	488	475	1.4051
Denmark	484	482	1.4054
Luxembourg	490	486	1.4085
Sweden	483	486	1.4106
Chile	432	446	1.4118
Hungary	461	472	1.4167
Italy	476	476	1.4306
Slovak Rep.	473	450	1.4336
Israel	464	484	1.4383
Iceland	488	478	1.4386
Greece	440	461	1.4985
Slovenia	461	438	1.5223
MEAN	488	489	1.3668

despite their results being relatively poor. Something similar also occurs for Mexico, which is placed in a high position in the rank of countries according to the efficiency of their schools despite their schools have the worst results in both competences. In contrast, some countries with relatively good results in both reading and math are placed in a worst position in terms of efficiency (e.g. Finland or Norway), which indicates that schools operating in those countries are not sufficiently exploiting their resources.

Figs. 2 and 3 illustrate that there is a negative relationship between efficiency and performance in maths and reading across countries. In general terms, countries where the average PISA score is higher, also tend to show lower levels of inefficiency. Likewise, it is also worth noting that some countries with similar average efficiency scores have very different levels of performance (see the case of Portugal and Finland).

Even more interesting than this evidence, which can also be derived using a traditional DEA approach, we would like to explore the projection (ratio between the target value and the actual value) for each output calculated with the proposed model developed by Aparicio et al. [12] before calculating the average output expansions. According to the values shown in Table 5, there are divergences between reading and maths projection values that are not visible when only average values are reported. Thus, for example, schools operating in Turkey, Chile and Mexico present significantly higher values in maths, which indicates that they need to make a greater effort to improve results in this area, since they are performing relatively well in reading. Turkish schools in fact present the lower averaged projection values among all the OECD countries in reading. The opposite occurs in Slovenia, Slovak Republic or Switzerland, where schools present significantly higher

[9] A large number of studies have generated conflicting findings about effects of homework on educational achievement [29,80].

[10] These indicators can be interpreted as proxies of student engagement, which has been traditionally identified as a key factor associated with better student achievement [26].

[11] Since PISA 2012 does not include specific information about reading activities and attitudes, we have drawn this information from PISA 2009. Given that the participating schools were not the same in that year, we have calculated those indicators aggregated at country level.

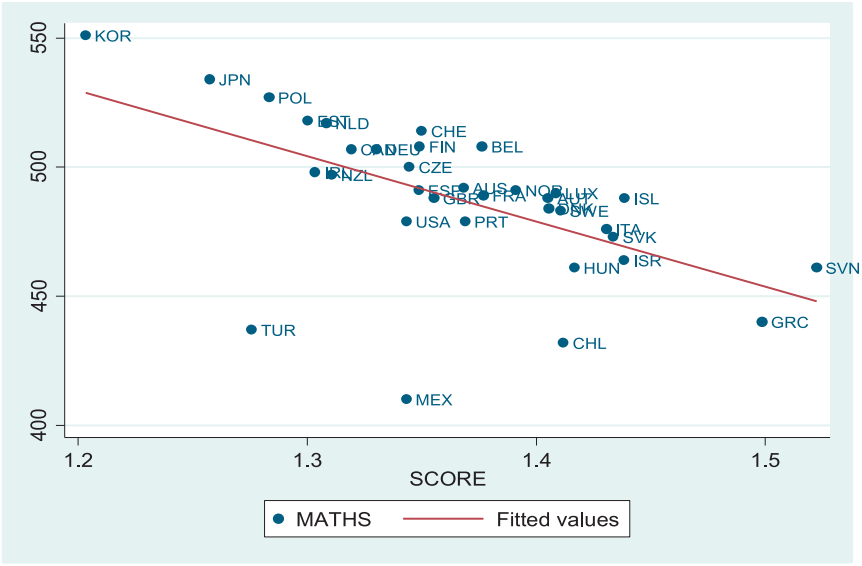

Fig. 2. Relationship between efficiency and maths across countries.

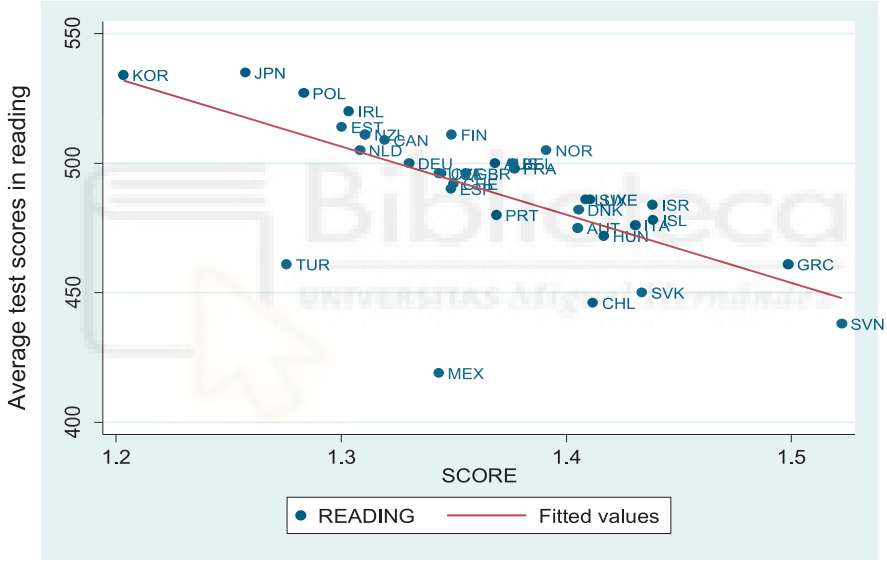

Fig. 3. Relationship between efficiency and reading across countries.

values in their projections for reading than in maths. Actually, this divergence is above 10 points, so the distance to the frontier for schools operating in those countries are, on average, 10% further in the reading competency.

The possibility of measuring the efficiency of schools using an international benchmark and considering both radial and non-radial displacements to measure the distance to the frontier allows for identifying that some schools might focus more on one dimension of the educational outcome. In particular, according to shape of the graph shown in Fig. 4, in global terms there are higher inefficiencies in reading. Indeed, 26 out of the 34 OECD countries present higher values in this competency and, in most cases, they are higher than 5%. Therefore, it seems that most part of schools around the world present higher efficiency levels in math proficiency. Nevertheless, there are some exceptions. In addition to the countries mentioned above (Turkey, Chile and Mexico), which are the poorest countries in the sample, some of the richest countries such as Norway or United States also present higher values of inefficiency in maths.

Finally, with the aim of exploring some potential characteristics of countries or schools that might explain why some schools present higher levels of inefficiency in maths or reading we have estimated two different Tobit regression models, one for each output.[12] Specifically, Table 6 reports the estimation results from the regression using math expansion as dependent variable and Table 7 reports the estimates for reading expansion as dependent variable. Since both measures are bounded from below at the value of one, a positive (negative) coefficient suggests a negative (positive) impact of the corresponding exogenous variable. Moreover, a coefficient is only statistically significant when both the lower

[12] In our estimation we used data about the five efficiency scores obtained with each of the five plausible values of the output variables in order to obtain more consistent results. Specifically, we follow the procedure recommended by survey organizers, which consists of estimating each regression model five times (once using each of the plausible values), thus we have five separate parameter estimates and five estimates of the sampling error and, subsequently, we take the average of the five estimates (See [65], p. 44 for details).

Table 5
Projections for each output dimension.

Country	Average efficiency score	Maths (M)	Reading (R)	Difference (R-M)
Korea	1.2032	1.1702	1.2362	6.6010
Japan	1.2574	1.2311	1.2838	5.2645
Turkey	1.2758	1.3206	1.2311	−8.9429
Poland	1.2833	1.2569	1.3097	5.2763
Estonia	1.3003	1.2709	1.3297	5.8840
Ireland	1.3034	1.3049	1.3020	−0.2934
Netherlands	1.3082	1.2738	1.3426	6.8833
New Zealand	1.3106	1.3022	1.3190	1.6772
Canada	1.3193	1.3029	1.3357	3.2730
Germany	1.3300	1.3004	1.3596	5.9190
Mexico	1.3433	1.3778	1.3087	−6.9126
USA	1.3433	1.3517	1.3349	−1.6784
Czech Rep.	1.3446	1.3144	1.3748	6.0459
Spain	1.3487	1.3163	1.3810	6.4710
Finland	1.3491	1.3467	1.3516	0.4875
Switzerland	1.3500	1.2974	1.4026	10.5229
UK	1.3552	1.3486	1.3617	1.3109
Australia	1.3683	1.3534	1.3832	2.9825
Portugal	1.3690	1.3285	1.4095	8.1024
Belgium	1.3763	1.3410	1.4115	7.0555
France	1.3768	1.3505	1.4031	5.2527
Norway	1.3911	1.4110	1.3712	−3.9798
Austria	1.4051	1.3777	1.4324	5.4627
Denmark	1.4054	1.3801	1.4306	5.0557
Luxembourg	1.4085	1.3814	1.4356	5.4181
Sweden	1.4106	1.3879	1.4333	4.5357
Chile	1.4118	1.4545	1.3692	−8.5327
Hungary	1.4167	1.3981	1.4352	3.7096
Italy	1.4306	1.3957	1.4655	6.9814
Slovak Rep.	1.4336	1.3649	1.5024	13.7524
Israel	1.4383	1.4506	1.4260	−2.4580
Iceland	1.4386	1.4433	1.4339	−0.9408
Greece	1.4985	1.4915	1.5055	1.4084
Slovenia	1.5223	1.4488	1.5957	14.6966
TOTAL	1.3655	1.3435	1.3780	

Table 6
Determinants of total inefficiency in maths (Tobit regression).

Variable	Coefficient	SE	(95% Confidence interval)	
			Low	High
PRIVATE	−0.0252***	0.0040	−0.0421	−0.0261
RURAL	0.00288	0.0034	0.0037	0.0170
PCGIRLS	−0.0237***	0.0083	−0.1091	−0.0762
REPEATERS	0.282***	0.0066	0.3049	0.3312
COMPETITION	−0.0177***	0.0037	−0.0281	−0.0135
MATHHOMEWORK	0.0871***	0.0131	0.0145	0.0661
MATHEXAMS	−0.199***	0.0135	−0.2277	−0.1742
MATHCLASSES	−0.134***	0.0149	−0.2037	−0.1450
ENJOYMATHS	−0.0444***	0.0128	−0.0514	−0.0008
READPLEASURE	−0.0378	0.0265	0.0110	0.1156

***$p < 0.01$, **$p < 0.05$, *$p < 0.1$

Table 7
Determinants of total inefficiency in reading (Tobit regression).

Variable	Coefficient	SE	(95% Confidence interval)	
			Low	High
PRIVATE	−0.0431***	0.0047	−0.0523	−0.0339
RURAL	0.0180***	0.0039	0.0103	0.0257
PCGIRLS	−0.162***	0.0097	−0.1807	−0.1427
REPEATERS	0.355***	0.0077	0.3393	0.3697
COMPETITION	−0.0239***	0.0043	−0.0323	−0.0155
MATHHOMEWORK	0.0053	0.0152	−0.0351	0.0245
MATHEXAMS	−0.204***	0.0158	−0.2347	−0.1728
MATHCLASSES	−0.216***	0.0173	−0.2495	−0.1816
ENJOYMATHS	−0.0099	0.0149	−0.0392	0.0193
READPLEASURE	0.1458***	0.0308	−0.0393	0.0816

***$p < 0.01$, **$p < 0.05$, *$p < 0.1$

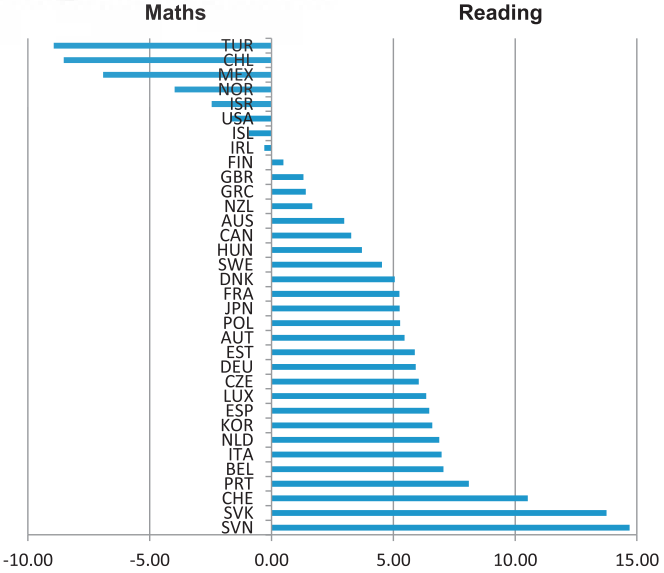

Fig. 4. Average levels of inefficiency per competency (R – M).

bound and upper bound of the confidence interval have the same sign.

The results shown in Tables 6 and 7 reveal that school factors are almost identically associated with both dependent variables, thus it seems that the aforementioned existing divergences between reading and mathematics across schools are not related to the school environment. Thus, in both cases being a private school, having a higher proportion of girls and having competition are

positively associated with efficiency measures, whereas schools placed in a rural area and, more importantly, having a higher proportion of repeaters are negatively related to efficiency in both competences.[13]

In contrast, we observe some differences in variables representing attitudes toward math and reading. For instance, reading for pleasure is a key factor as a determinant of the reading expansion, but we find no relationship between this variable and math efficiency. Nevertheless, these results should be interpreted cautiously because this information was retrieved at a country level due to the lack of information related to reading activities in the original dataset. Similarly, we find that math enjoyment affects the expansion in maths, but it is not related to reading efficiency. These results reveal the existence of a certain substitution effect between subjects with regard to the preferences of students that might affect their performance and, subsequently, the efficiency demonstrated by schools with regard to each dimension of the output. Another divergence arises for working hard on math homework, which is found to be negatively associated with math efficiency, but it is not a significant factor for reading. One explanation of the negative relation is that students with lower levels of proficiency in reading tend to put more time into homework because of necessity and pressure from parents and teachers [50]. This result suggests that assigning more homework to students does not seem to be the appropriate strategy that school should apply to achieve higher levels of efficiency neither in mathematics nor in reading.

Finally, we find that both indicators related to school engagement, i.e. being prepared for math exams and paying attention in maths classes, are positively and significantly related to both dependent variables. Although the degrees of engagement depends in a certain extent of intrinsic motivation of students, schools have the capacity of influencing this type of behaviors, thus promoting student engagement appears to be a good way to be more efficient.

5. Concluding remarks

In this paper we have used data from OECD countries participating in PISA 2012 to assess the efficiency of schools in a cross-country framework. In our empirical analysis we consider that schools might concentrate their efforts on improving more the results in one dimension of the educational output than in other. To do this, we rely on non-radial efficiency measures of performance, which are particularly interesting in the context of education, since they allow for identifying different levels of (in) efficiency for each output analyzed. In particular, we apply a methodology recently developed by Aparicio et al. [12] that determines the closest targets and the least distance to the strongly efficient frontier in DEA based on Bilevel Linear Programming.

Although it is true that many alternative non-radial measures exist in DEA, those based on the determination of the least distance yield useful benchmarking information. In particular, this type of measures can be useful from the point of view of practice, for example to managers in their decision making. It is especially important for firms/units which seek to achieve superior performance results as soon as possible. Indeed, the distance which forms the basis of the technical inefficiency measure in this paper generates targets that are easily achievable by units. Among the set of non-radial DEA measures associated with the determination of the least distance, we opt for the output-oriented Russell measure. Nevertheless, we recognize that other measures based on least distances could be also used in the empirical application. The evaluation of the impact of the utilization of different measures is outside the scope of this paper, although it could be a good avenue for further research.

Our findings indicate that larger potential improvements may be achieved in reading proficiency than in maths. In global terms we detect higher levels of efficiency in the reading competency, thus it seems that schools around the world are concentrating their efforts in maths, maybe because math proficiency is generally considered as one of the strongest predictors of positive outcomes for young adults, such as their ability to participate in post-secondary education and their expected future earnings [64]. This might entail neglecting reading. Actually, around three quarters of the studied countries present greater inefficiency values in this competency and, in most cases, this divergence exceeds 5%. Even so, there are some exceptions, such as schools operating in poor countries like Turkey, Chile or Mexico, which perform relatively better in reading than in maths.

We have also examined some potential determinants associated with each dimension of efficiency by estimating one regression for each output projection. The results reveal that divergences detected between the two dimensions of the output are mainly explained by students´attitudes, while school characteristics do not appear to be differential factors. In particular, we identify divergences with regard to the reading habits of students as well as homework assignments, whereas variable representing student engagement has a positive effect on both educational outcomes. However, this approach does not allow for a causal interpretation of results, but it allows a future line of research based on the search of the causes of inefficiency to be flagged up.

These findings provide some interesting insights into the analysis of determinants of educational attainment using a cross-country approach. However, further future research will be needed to explore some of the results discussed here in greater depth. For instance, the proposed analysis could be replicated using data at pupil level in order to test whether the origin of the existing divergence in terms of inefficiency between subjects might come from intrinsic characteristics of the students that affect their performance instead of the activities carried out by schools.

Acknowledgments

Juan Aparicio and Martin Gonzalez thank the financial support from the Spanish Ministry for Economy and Competitiveness (Ministerio de Economía, Industria y Competitividad), the State Research Agency (Agencia Estatal de Investigacion) and the European Regional Development Fund (Fondo Europeo de DEsarrollo Regional) under grant MTM2016-79765-P (AEI/FEDER, UE). Additionally, Jose M. Cordero also acknowledges the support from the Spanish Ministry for Economy and Competitiveness through grant ECO2014-53702-P and Junta de Extremadura through grant IB16171.

References

[1] Afonso A, Aubyn MS. Cross-country efficiency of secondary education provision: a semi-parametric analysis with non-discretionary inputs. Econ Model 2006;23(3):476–91.
[2] Agasisti T. The efficiency of Italian secondary schools and the potential role of competition: a data envelopment analysis using OECD-PISA2006 data. Educ Econ 2013;21(5):520–44.
[3] Agasisti T. The efficiency of public spending on education: an empirical comparison of EU countries. Eur J Educ 2014;49(4):543–57.
[4] Agasisti T, Zoido P. *The efficiency of secondary schools in an international perspective: preliminary results from PISA 2012* (No. 117). OECD Publishing; 2015.
[5] Ando K, Kai A, Maeda Y, Sekitani K. Least distance based inefficiency measures on the pareto-efficient frontier in DEA. J Oper Res Soc Jpn 2012;55:73–91.
[6] Ando K, Minamide M, Sekitani K. Monotonicity of minimum distance inefficiency measures for data envelopment analysis. Eur J Oper Res 2017;260(1):232–43.
[7] Aparicio J, Ruiz JL, Sirvent I. Closest targets and minimum distance to the pareto-efficient frontier in DEA. J Prod Anal 2007;28:209–18.

[13] The negative relationship of repeaters might explain why countries like Japan or Korea, where the practice of retention is almost inexistent (see [68]), are among the top performers in terms of both achievement and efficiency.

[8] Aparicio J, Pastor JT. A well-defined efficiency measure for dealing with closest targets in DEA. Appl Math Comput 2013;219:9142–54.
[9] Aparicio J, Mahlberg B, Pastor JT, Sahoo BK. Decomposing technical inefficiency using the principle of least action. Eur J Oper Res 2014;239:776–85.
[10] Aparicio J, Pastor JT. Closest targets and strong monotonicity on the strongly efficient frontier in DEA. Omega 2014;44:51–7.
[11] Aparicio J, Pastor JT. On how to properly calculate the Euclidean distance-based measure in DEA. Optimization 2014;63(3):421–32.
[12] Aparicio J, Cordero JM, Pastor JT. The determination of the least distance to the strongly efficient frontier in data envelopment analysis oriented models: modelling and computational aspects. Omega 2016. http://dx.doi.org/10.1016/j.omega.2016.09.008i.
[13] Aparicio J. A survey on measuring efficiency through the determination of the least distance in data envelopment analysis. J Centr Cath 2016;9(2):143–67.
[14] Aristovnik A, Obadić A. Measuring relative efficiency of secondary education in selected EU and OECD countries: the case of Slovenia and Croatia. Technol Econ Dev Econ 2014;20(3):419–33.
[15] Asmild M, Hougaard JL, Kronborg D, Kvist HK. Measuring Inefficiency via potential improvements. J Prod Anal 2003;19(1):59–76.
[16] Banker RD, Charnes A, Cooper WW. Some models for estimating technical and scale inefficiencies in data envelopment analysis. Manage Sci 1984;30:1078–92.
[17] Beale EML, Tomlin JA. Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables. In: Lawrence J, editor. Proceedings of the fifth international conference on operational research. Tavistock Publications; 1970. p. 447–54.
[18] Bogetoft P, Hougaard JL. Efficiency evaluations based on potential (non-proportional) improvements. J Prod Anal 1999;12:233–47.
[19] Bray M, Thomas RM. Levels of comparison in educational studies: different insights from different literatures and the value of multilevel analyses. Harvard Educ Rev 1995;65(3):472–90.
[20] Briec W. Minimum distance to the complement of a convex set: duality result. J Optim Theory Appl 1997;93:301–19.
[21] Briec W. Hölder distance function and measurement of technical efficiency. J Prod Anal 1998;11:111–31.
[22] Chambers R, Chung Y, Färe R. Benefit and distance functions. J Econ Theory 1996;70:407–19.
[23] Chambers RG, Mitchell T. Homotheticity and non-radial changes. J Prod Anal 2001;15 31-19.
[24] Charnes A, Cooper WW, Rhodes E. Measuring the efficiency of decision making units. Eur J Oper Res 1978;2(6):429–44.
[25] Charnes A, Cooper WW, Golany B, Seiford L, Stutz J. Foundations of data envelopment analysis for Pareto–Koopmans efficient empirical production functions. J Econometr 1985;30:91–107.
[26] Christenson SL, Reschly AL, Wylie C. Handbook of research on student engagement. Springer Science & Business Media; 2012.
[27] Colson B, Marcotte P, Savard G. An overview of bilevel optimization. Ann Oper Res 2007;153:235–56.
[28] Cook WD, Hababou M. Sales performance measurement in bank branches. Omega 2001;29(4):299–307.
[29] Cooper H, Robinson JC, Patall EA. Does homework improve academic achievement? A synthesis of research, 1987–2003. Rev Educ Res 2006;76:1–62.
[30] Cordero JM, Muñiz M, Polo C. The determinants of cognitive and non-cognitive educational outcomes: empirical evidence in Spain using a Bayesian approach. Appl Econ 2016;48(35):3355–72.
[31] Cordero JM, Santín D, Simancas R. Assessing European primary school performance through a conditional nonparametric model. J Oper Res Soc 2017;68(4):364–76.
[32] Creemers B, Kyriakides L. The dynamics of educational effectiveness: a contribution to policy, practice and theory in contemporary schools. Abingdon, Oxon: Routledge; 2008.
[33] Crespo-Cebada E, Pedraja-Chaparro F, Santín D. Does school ownership matter? An unbiased efficiency comparison for regions of Spain. J Prod Anal 2014;41(1):153–72.
[34] De Jorge J, Santín D. Determinantes de la eficiencia educativa en la Unión Europea. Hacienda Pública Española 2010(193):131–55.
[35] De Witte K, López-Torres L. Efficiency in education. A review of literature and a way forward. J Oper Res Soc 2017;68(4):339–63.
[36] Deprins D, Simar L, Tulkens H. Measuring labor inefficiency in post offices. In: Marchand M, Pestieau P, Tulkens H, editors. The performance of public enterprises: concepts and measurements. Amsterdam, North-Holland; 1984. p. 243–67.
[37] Deutsch J, Dumas A, Silber J. Estimating an educational production function for five countries of Latin America on the basis of the PISA data. Econ Educ Rev 2013;36:245–62.
[38] Färe R, Grosskopf S, Lovell CAK. The measurement of efficiency of production. Kluwer Nijhof Publishing; 1985.
[39] Farrell MJ. The measurement of productive efficiency. J R Stat Soc, Ser A: Gen 1957;120:253–81.
[40] Fukuyama H, Maeda Y, Sekitani K, Shi J. Input-output substitutability and strongly monotonic p-norm least-distance DEA measures. Eur J Oper Res 2014;237:997–1007.
[41] Fukuyama H, Masaki H, Sekitani K, Shi J. Distance optimization approach to ratio-form efficiency measures in data envelopment analysis. J Prod Anal 2014;42:175–86.
[42] Fukuyama H, Hougaard JL, Sekitani K, Shi J. Efficiency measurement with a nonconvex free disposal hull technology. J Oper Res Soc 2016;67(1):9–19.
[43] Giambona F, Vassallo E, Vassiliadis E. Educational systems efficiency in European Union countries. Stud Educ Eval 2011;37(2):108–22.
[44] Giménez V, Prior D, Thieme C. Technical efficiency, managerial efficiency and objective-setting in the educational system: an international comparison. J Oper Res Soc 2007;58(8):996–1007.
[45] Giménez V, Thieme C, Prior D, Tortosa-Ausina E. An international comparison of educational systems: a temporal analysis in presence of bad outputs. J Prod Anal 2017;47(1):83–101.
[46] Goldhaber DD. Public and private high schools: Is school choice an answer to the productivity problem? Econ Edu Rev 1996;15(2):93–109.
[47] Grifell-Tatje E, Lovell CAK, Pastor JT. A quasi-malmquist productivity index. J Prod Anal 1998;10:7–20.
[48] Grosskopf S, Hayes KJ, Taylor LL. Efficiency in education: research and implications. Appl Econ Perspect Policy 2014;36(2):175–210.
[49] Gustafsson JE. Effects of international comparative studies on educational quality on the quality of educational research. Eur Educ Res J 2008;7(1):1–17.
[50] Gustafsson JE. Causal inference in educational effectiveness research: a comparison of three methods to investigate effects of homework on student achievement 1. School Eff School Improv 2013;24(3):275–95.
[51] Hanushek EA. Conceptual and empirical issues in the estimation of educational production functions. J Human Resour 1979;14:351–88.
[52] Holmes CT. Grade-level retention effects. A meta-analysis of research studies. In: Shepard L, Smith ML, editors. Flunking grade: research and policies on retention. London: The Falmer Press; 1989. p. 16–33.
[53] Huddleston AP. Achievement at whose expense? A literature review of test-based grade retention policies in U.S. school. Educ Policy Anal Arch 2014;22(18):1–31.
[54] Hyde JS, Fennema E, Lamon SJ. Gender differences in mathematics performance: a meta-analysis. Psychol Bull 1990;107(2):139.
[55] Johnes J. Operational research in education. Eur J Oper Res 2015;243(3):683–96.
[56] Koopmans TC. Analysis of production as an efficient combination of activities. Activity analysis of production and allocation. Koopmans TC, editor. New York: John Wiley; 1951.
[57] Levin H. Measuring the efficiency in educational production. Public Finance Q 1974;2:3–24.
[58] Mancebon MJ, Muñiz MA. Private versus public high schools in Spain: disentangling managerial and programme efficiencies. J Oper Res Soc 2008;59(7):892–901.
[59] McEwan PJ. The effectiveness of public, catholic, and non-religious private schools in Chile's voucher system. Educ Econ 2001;9(2):103–28.
[60] Mislevy RJ, Beaton AE, Kaplan B, Sheehan KM. Estimating population characteristics from sparse matrix samples of item responses. J Educ Meas 1992;29(2):133–61.
[61] Misra K, Grimes PW, Rogers KE. Does competition improve public school efficiency? A spatial analysis. Econ Educ Rev 2012;31(6):1177–90.
[62] Mullis IM, Kennedy A, Foy P. IEAś progress in international reading literacy study in primary schools in 40 countries PIRLS 2006 International Report. TIMMS and PIRLS International Study Center, Lynch School of Education, Boston College; 2007.
[63] OECD. PISA data analysis Manual, SPSS. 2nd ed. Paris: OECD Publishing; 2009.
[64] OECD. PISA 2012 Results: what students know and can do – student performance in mathematics, reading and science, vol. I. Paris: PISA, OECD Publishing; 2013.
[65] OECD. PISA 2012 technical report. Paris: PISA, OECD Publishing; 2014.
[66] Pastor JT, Ruiz JL, Sirvent I. An enhanced DEA Russell graph efficiency measure. Eur J Oper Res 1999;115(3):596–607.
[67] Pastor JT, Aparicio J. The relevance of DEA benchmarking information and the least-distance measure: comment. Math Comput Model 2010;52:397–9.
[68] Portela MCAS, Castro P, Thanassoulis E. Finding closest targets in non-oriented DEA models: the case of convex and non-convex technologies. J Prod Anal 2003;19:251–69.
[69] Ruiz JL, Sirvent I. Common benchmarking and ranking of units with DEA. Omega 2016;65:1–9.
[70] Santín D, Sicilia G. Measuring the efficiency of public schools in Uruguay: main drivers and policy implications. Latin Am Econ Rev 2015;24(1):1–28.
[71] Shephard RW. Cost and production functions. Princeton, N.J.: Princeton University Press; 1953.
[72] Shi C, Lu J, Zhang G. An extended Kuhn–Tucker approach for linear bilevel programming. Appl Math Comput 2005;162:51–63.
[73] Simar L, Wilson PW. Estimation and inference in two-stage, semi-parametric models of production processes. J Econometr 2007;136(1):31–64.
[74] Simar L, Wilson PW. Two-stage DEA: caveat emptor. J Prod Anal 2011;36(2):205–18.
[75] Sutherland D, Price R, Gonand F. Improving public spending efficiency in primary and secondary education. OECD J: Econ Stud 2009;2009(1):1–30.
[76] Thieme C, Giménez V, Prior D. A comparative analysis of the efficiency of national educational systems. Asia Pacific Educ Rev 2012;13:1–15.
[77] Thieme C, Prior D, Tortosa-Ausina E. A multilevel decomposition of school performance using robust nonparametric frontier techniques. Econ Educ Rev 2013;32:104–21.
[78] Todd PE, Wolpin KI. On the specification and estimation of the production function for cognitive achievement. Econ J 2003;113(485):3–33.

[79] Tone K. A slacks-based measure of efficiency in data envelopment analysis. Eur J Oper Res 2001;130(3):498–509.
[80] Trautwein U, Köller O. The relationship between homework and achievement: still much of a mystery. Educ Psychol Rev 2003;15:115–45.
[81] Verhoeven M, Gunnarsson V, Carcillo S. Education and health in G7 countries: achieving better outcomes with less spending; 2007. IMF Working Papers p. 1–50.
[82] Von Davier M, Sinharay S. Analytics in international large-scale assessments: item response theory and population models. In: Rutkowski L, Von Davier M, Rutkowski D, editors. Handbook of international large-scale assessment: background, technical issues, and methods of data analysis. London: CRS Press; 2013. p. 155–74.
[83] Willms JD, Smith T. A manual for conducting analyses with data from TIMSS and PISA; 2005. Report prepared for UNESCO Institute for Statistics.
[84] Worthington AC. An empirical survey of frontier efficiency measurement techniques in education. Educ Econ 2001;9(3):245–68.
[85] Wu DD. Bilevel programming data envelopment analysis with constrained resource. Eur J Oper Res 2010;207:856–64.
[86] Wu M. The role of plausible values in large-scale surveys. Stud Educ Eval 2005;31(2-3):114–28.
[87] Xia N, Kirby SN. Retaining students in grade: a literature review of the effects of retention on students' academic and nonacademic outcomes; 2009. Technical Report No. 678.
[88] Xue M, Harker PT. Overcoming the inherent dependency of DEA efficiency scores: a bootstrap approach. Wharton Financial Institutions Center, University of Pennsylvania; 1999. *Unpublished Working Paper*.

Separatas

- *Gonzalez M, López-Espín JJ, Aparicio J, Talbi E. 2022. A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis. PeerJ Computer Science 8:e828.*

PeerJ Computer Science

A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis

Martin Gonzalez[1], Jose J. López-Espín[1], Juan Aparicio[1] and El-Ghazali Talbi[2]

[1] Center of Operations Research, Universidad Miguel Hernández de Elche, Elche, Spain
[2] INRIA, Université des Sciences et Technologies de Lille (Lille I), Lille, France

ABSTRACT

Mixed Integer Linear Programs (MILPs) are usually NP-hard mathematical programming problems, which present difficulties to obtain optimal solutions in a reasonable time for large scale models. Nowadays, metaheuristics are one of the potential tools for solving this type of problems in any context. In this paper, we focus our attention on MILPs in the specific framework of Data Envelopment Analysis (DEA), where the determination of a score of technical efficiency of a set of Decision Making Units (DMUs) is one of the main objectives. In particular, we propose a new hyper-matheuristic grounded on a MILP-based decomposition in which the optimization problem is divided into two hierarchical subproblems. The new approach decomposes the model into discrete and continuous variables, treating each subproblem through different optimization methods. In particular, metaheuristics are used for dealing with the discrete variables, whereas exact methods are used for the set of continuous variables. The metaheuristics use an indirect representation that encodes an incomplete solution for the problem, whereas the exact method is applied to decode the solution and generate a complete solution. The experimental results, based on simulated data in the context of Data Envelopment Analysis, show that the solutions obtained through the new approach outperform those found by solving the problem globally using a metaheuristic method. Finally, regarding the new hyper-matheuristic scheme, the best algorithm selection is found for a set of cooperative metaheuristics ans exact optimization algorithms.

Subjects Algorithms and Analysis of Algorithms, Optimization Theory and Computation
Keywords Hyper-matheuristic, Metaheuristics, Exact methods, Mixed integer problems, MILP decomposition, Mathematical optimization

Submitted 1 April 2021
Accepted 3 December 2021
Published 20 January 2022

Corresponding author
Martin Gonzalez,
mgonzalez@umh.es

Academic editor
Nicholas Higham

Additional Information and Declarations can be found on page 21

DOI **10.7717/peerj-cs.828**

INTRODUCTION

Mixed Integer Linear Programs (MILPs) address mathematical optimization problems involving two families of variables: discrete and continuous ones. Both the objective function as well as the constraints are linear. This family of optimization problems appears for many real-life applications in various domains. Indeed, many real problems can be formulated using MILP models, for example: packing, knapsack, inventory, production planning, location, resource allocation, routing and scheduling problems, to name but a few (*Winston & Goldberg, 2004*). This large applicability has led to an increased interest in

the development of efficient algorithms for solving this general and popular class of optimization problems.

MILP models are generally NP-hard problems. Approximation algorithms have been developed in response to the impossibility of solving a great variety of important optimization problems. Very often, one is confronted with the fact that the problem is NP-hard, making it really difficult to obtain an optimal solution in a reasonable time (*Hochba, 1997*). For all intents and purposes, we use two families of algorithms to solve MILPs: exact algorithms and heuristics. The exact methods (*e.g.* branch and bound, branch and cut, branch and price) are generally applicable, but they have been proven to be laborious for large or more complex problems. When instances become too large or difficult for exact methods, heuristics and particularly metaheuristics are often used. A metaheuristic is a high-level procedure to select among different heuristics. We can examine two types of metaheuristics: single solution algorithms (*e.g.* local search, tabu search) and algorithms based on population (*e.g.* evolutionary algorithms, swarm optimization) (*Talbi, 2009*). Metaheuristics do not, however, generally guarantee that the best solutions are found. Thus, the combination of metaheuristics and exact optimization algorithms can offer a more efficient and effective resolution method (*Talbi, 2016*). A general classification of this hybridization is discussed by *Jourdan, Basseur & Talbi (2009)*, and some examples can be found in the literature (*Pradenas et al., 2013*; *Li et al., 2012*).

In this paper, a new hyper-matheuristic methodology, based on the matheuristic previously introduced in *González et al. (2017)*, is developed to find solutions for MILP models in the context of Data Envelopment Analysis (DEA) (*Vanderbeck & Wolsey, 2010*). Nowadays, DEA is one of the most used non-parametric techniques in Economics and Engineering to measure technical efficiency from a data sample of firms. Regarding our methodology, the matheuristic allows for MILP-based decomposition, where the main problem is broken down into two hierarchical subproblems, since it is easier to solve them separately using distinct categories of optimization algorithms (*Raidl, 2015*). This breakdown is based on the characteristics of the continuous and discrete decision variables. The hyper-matheuristic methodology is proposed from this matheuristic. The hyper-heuristic concept (*Pillay & Banzhaf, 2009*) has been applied in the context of hybrid heuristics (*Wang et al., 2019*; *Li et al., 2020*) to find the best combination of heuristics. In this work, we propose a generalization of the hyper-heuristic methodology for matheuristics that combines exact algorithms and metaheuristics, which is called the hyper-matheuristic approach.

The aim of the proposed hyper-matheuristic methodology is to find the best combination between metaheuristics and exact methods for a given type of MILP models within the framework of DEA. For this, some input instances of the problem are evaluated with several iterations of the algorithm, training the hyper-matheuristic and obtaining a good solution for any problem. Within the hyper-matheuristic, several parameters have been established to generate many different metaheuristics. The functionalities of the generated metaheuristics, which depend on the values of these parameters, where metaheuristics like Evolution Algorithm (EA) (*Holland, 1973*), Scatter Search (SS) (*Glover, Laguna & Marti, 2003*), Tabu Search (TS) (*Glover, 1997*) or Greedy Randomized Adaptive

Search Procedure (GRASP) (*Resende & Ribeiro, 2003*) can be generated automatically. All of these parameters take different values. These values are studied in the experiments, having set certain limit values, and they represent inputs for the algorithm.

The matheuristic method was designed by examining the various synergies between metaheuristics and exact methods, in order to find the best combination for resolving MILP problem. A list of existing approaches combining exact methods and metaheuristics for MILP optimization can be found in *Puchinger & Raidl (2005)*:

- Collaborative combinations: self-contained optimization algorithms exchange information extracted from the search. The different algorithms are independent. There is no direct relationship to the internal workings of the algorithms. Exact method and heuristic algorithms can be executed sequentially, interwoven or in parallel.
- Integrative combinations: in these types of algorithms, hybridization addresses the functional composition of a single optimization method. A given internal function of an optimization algorithm is replaced by another optimization algorithm.

The matheuristic algorithm (*González et al., 2017*) used in this paper for the proposed hyper-matheuristic employs an integrative combination, where by the metaheuristic supplies information to the exact method, which solves the problem and returns some new information to the metaheuristic. The basic concept is to break the problem down into much smaller subproblems which can be accurately solved using cutting-edge mathematical programming algorithms. The variables and the constraints are divided up into two sets, which break the main problem down into two hierarchical subproblems: the metaheuristic determines the decision variables in one set and the exact method optimizes the problem in the other. In the literature, there are some works where certain exact techniques are improved using approximation techniques (metaheuristics), as in the case of *Poojari & Beasley (2009)*, where Bender's decomposition is optimized through a genetic algorithm, integrating the latter as a seed generator for decomposition.

Moreover, a hyper-metaheuristic scheme has been included in the proposed methodology for an autonomous design of metaheuristics. Certain design parameters define the characteristics of each metaheuristic, and these are framed into different search components: Initialize, Improvement, Selection, Combination and Stopping Criteria. In this work, the hyper-metaheuristic methodology (*González et al., 2017*) has been generalized to matheuristics, in which exact optimization is combined with a set of metaheuristics.

The main contributions in this paper are based on the development of a general methodology in terms of optimization algorithms, being capable of solving MILP problems in the context of DEA. A MILP-based decomposition is studied that combines metaheuristics and exact methods in a single algorithm called a matheuristic. A final algorithm is implemented to obtain the best combination of those algorithms previously mentioned, called hyper-matheuristic.

The paper is organized as follows: In "MILP-Based Decomposition", we present the proposed breakdown of MILP problems. In "Matheuristic Methodology" we detail the

matheuristic strategy that combines linear continuous programming and discrete metaheuristics. "Hyper-Matheuristic Methodology", focuses on the hyper-matheuristic methodology in which an automatic design of optimization algorithms is carried out. In "Experimental Results", we provide some computational experiments on a MILP problem. Finally, in "Conclusions and Future Works", we conclude and point out some future works.

MILP-BASED DECOMPOSITION

In this section, we will handle general notions within the field of MILP models and the developments will be as general as possible. Nevertheless, our approach will be exclusively tuned and tested with problems from Data Envelopment Analysis in "Experimental Results". We are aware that the technique could be used with other types of MILP models. However, we cannot guarantee the validity of the new approach in those cases. In this respect, further research in this line would be necessary. Let us consider the following linear problem (LP) (1):

$$\max\{\boldsymbol{cx} : \boldsymbol{Ax} \leq \boldsymbol{b}, \boldsymbol{x} \geq 0, \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}\} \tag{1}$$

where $\boldsymbol{A}$ is a $m \times n$ matrix, $\boldsymbol{c}$ a n-dimensional row vector, $\boldsymbol{b}$ a m-dimensional column vector, and $\boldsymbol{x}$ a n-dimensional column vector of continuous variables. If we add the constraint that certain variables must take integer values, we have a MILP (2), that can be written as:

$$\begin{array}{l} \max \boldsymbol{cx} + \boldsymbol{hy} \\ \boldsymbol{Ax} + \boldsymbol{Gy} \leq \boldsymbol{b} \\ \boldsymbol{x} \geq 0, \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}} \\ \boldsymbol{y} \geq 0, \boldsymbol{y} \in \mathbb{Z}^{\boldsymbol{p}} \end{array} \tag{2}$$

where $\boldsymbol{A}$ is again a $m \times n$ matrix, $\boldsymbol{G}$ is $m \times p$ matrix, $\boldsymbol{h}$ is a p row-vector, and $\boldsymbol{y}$ is a p column-vector of integer variables.

An MILP problem is defined as one where discrete (y) variables, which are restricted to integer values, and continuous variables (x), and which can be assigned any value on a given continuous interval, are combined with integrality constraints. The integrality constraints allow MILP models to capture the discrete nature of some decisions. For example, a binary variable can be used to decide whether or not any action needs to be taken.

Using MILP solvers to resolve large-scale and complex instances is inefficient in terms of the time spent determining the solution to the problem. Indeed, large MILPs are often difficult to resolve using exact methods, owing to the complexity of the combinatorial nature of the discrete part of the problem. One way to solve large MILPs is to break them down into smaller subproblems, so they can be solved individually. Problem decomposition techniques comprise an approach that is particularly aimed at solving very large and difficult problems. The basic idea is to solve such a large problem by solving a set of smaller problems, combining their solutions to obtain an optimal one for the main MILP problem. For each subproblem, if the optimality criterion is satisfied, the current

feasible solution is judged to be the optimal solution of the original MILP problem. If the optimality criterion is not satisfied, other values for the variable in the subproblems are assumed, and the procedure is repeated (*Yokoyama & Ito, 2000*).

Popular decomposition techniques

The objective of decomposition techniques is to tackle large-scale problems which cannot be solved by exact optimization algorithms such as MIP solvers (*Ralphs & Gelati, 2010*). From the integer programming point of view, there are two types of decomposition approaches that exploit the problem structure: constraint decomposition and variable decomposition (*Vanderbeck & Wolsey, 2010*).

In constraint decomposition techniques, a compact problem is created by the insertion of constraints to obtain a better approximation by eliminating a part of the feasible space that does not contain integer solutions. Outer approximation (cutting plane methods) (*Kelley, 1960*) or inner approximation (Dantzig-Wolfe method) (*Vanderbeck, 2000*; *Ruszczynski, 1989*) are the most popular ones.

In variable decomposition techniques, the decision variables of the problem are generally separated into two groups and the problem is solved in two steps. Bender's decomposition represents one of the most popular variable decomposition approaches for solving integer programming problems (*Bender's, 1962*), being less popular than branch-and-cut, but really common in the literature. This decomposition technique is based on a cycle of two steps in each iteration. At the first step, a subset of integer variables is selected and their values are found. Then, the second step finds an optimal solution for the rest of the continuous variables according to the values allocated to the first subset of variables. In each iteration, some constraints are modified in the sub-problems to improve the solution. This approach has been applied to many problems such as routing, scheduling, network design and planning (*Rahmaniani et al., 2017*).

From the metaheuristic point of view, one can use either a variable or data decomposition by using some problem features. In variable decomposition techniques, the problem is decomposed into subproblems of similar size following a variable decomposition (for example, time decomposition in scheduling problems). After the resolution of the subproblems, the global solution is constructed from the sub-optimal partial solutions obtained. The subproblems can be solved in independent or hierarchical ways. In data decomposition techniques, the input data (*e.g.* geographical space) of the general problem is divided into different partitions. Then, the problem is solved using all the partitions and the final solutions are aggregated from the sub-solutions obtained from the different partitions. For instance, some clustering algorithms can be applied to partition a geographical space into different regions for routing problems (*Reimann, Doerner & Hartl, 2004*; *Taillard, 1993*).

Variable-based decomposition of MILPs

The main drawback of data-based decomposition is its specificity to the target optimization problem. In our work, a more general decomposition predicated on the type of variables (discrete *vs* continuous) and the complexity of the generated subproblems is

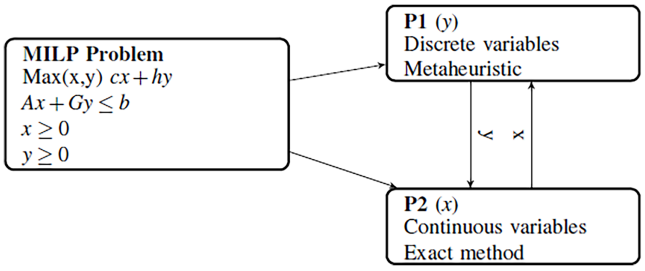

Figure 1 **Variable decomposition of MILP problem into two subproblems.**
Full-size DOI: 10.7717/peerj-cs.828/fig-1

carried out. The main problem is decomposed into two different hierarchical subproblems, following the principles of indirect encoding (*Talbi, 2009*). The master problem is associated to the discrete variables. A solution is encoded using an incomplete solution for the problem in which only the discrete variables are handled. For each solution of the master problem, the subproblem will fix the continuous variables of the solution. It can be seen as decoding the incomplete solution given by the master problem to generate a complete solution for the problem. Then, the constraints associated to the optimization problem are handled by the subproblem and will guarantee the validity of the solution that is decoded. Compared to Bender's decomposition approach, the master problem, including the variables and the constraints, is not modified at each iteration. The subproblems solved at each iteration depend on the sub-solution generated at the master problem.

Figure 1 shows how a general MILP problem is broken down into two hierarchical subproblems of different complexities:

- The master problem (P1) contains the discrete variables and is difficult to solve efficiently with an exact method. Then, metaheuristic approaches are more efficient to solve the master problem. In this paper, the hyper-heuristic methodology with a set of adaptive metaheuristics H_i (i = 1,…,k) is used to solve the master problem.
- The subproblem (P2), including the continuous variables, is a linear continuous problem (LP), and easy to solve using an exact linear solver. This subproblem decodes the incomplete solution of the master problem to obtain a complete solution for the problem.

MATHEURISTIC METHODOLOGY

The combination between metaheuristics and exact methods that is presented arises from the need to simplify mathematical models that are difficult to solve by any of these techniques. In this way, any mathematical model can be divided into different subproblems, taking the nature of its variables as the main criteria. In this paper, we propose to divide the model into two different subproblems where one of them must be a linear problem. In this case, it is easier to solve the linear problem using exact methods, and we can study the other subproblem using metaheuristic methods. Figure 2 shows how the decomposition is developed and how both methods collaborate.

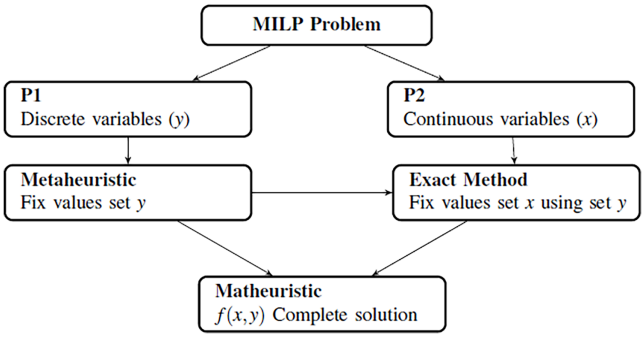

Figure 2 **Generation of metaheuristics and structure of the algorithm.**
Full-size DOI: 10.7717/peerj-cs.828/fig-2

The matheuristic algorithm is designed to be mainly used by population-based metaheuristics, in which a set of solutions are randomly generated and processed by different steps such as selection, recombination and replacement. However, single solution-based metaheuristics such as local search and tabu search can also be used.

After generating of the initial population using a metaheuristic, an exact method is employed to solve the subproblems generated. This method involves the use of relaxation or decomposition techniques of the mathematical model. Relaxation methods consist of relaxing a strict requirement in the target optimization problem (*Sadykov et al., 2019*). This method comprises disregarding the integrality constraints of an integer program and solving it using LP solvers. To do so, the metaheuristic generates the discrete variables (solving the subproblem P1), and provides this information to the exact method to fix values for the continuous ones (solving the subproblem P2). Figure 2 shows how the main problem is divided into two smaller problems.

All the functions incorporated in the metaheuristic are executed sequentially:

- **Initial population:** This step is used to generate the initial population (*Population*), fixing values for the discrete variables in P1.
- **Improvement:** Some of the feasible solutions (*TotalImprove*) are modified to improve the fitness value using the Variable neighborhood search algorithm proposed by (*Mladenović & Hanse, 1997*). The infeasible solutions are improved trying to transform them into feasible ones.
- **Selection:** We first sort the valid solutions, in decreasing order of fitness, followed by the invalid ones, which are ordered randomly. We then select a percentage of solutions to be used in the Crossover and Diversification function.
- **Crossover:** The algorithm includes a crossover function, which combines a certain number of pairs of solutions (*Combination*), which are chosen randomly from those previously selected.
- **Diversification:** A diversification based on edge recombination (ER) (*Laporte, Potvin & Quilleret, 1997*) is included, where the aim of diversification methods is to drive the search in new regions of the solution space. Here, the tabu search heuristic is periodically

restarted with a new solution obtained through recombination of two elite solutions previously visited during the search (*Diversification*).

Depending on the metaheuristic selected, some internal functions may or may not be executed. For example, an EA does not use the improvements, or a GRASP does not employ the crossover function. At this point, a huge number of metaheuristics can be used to solve the problem. There are many options to determine the metaheuristic used, but in this article it has been decided to use a parameterized scheme, where certain parameters give value to the internal functionalities. This technique will be further detailed in "Experimental Results". For each problem, the metaheuristic selected solves the problem P1. Then, the exact method is used to solve the problem P2 and obtain the initial complete population. Algorithm 1 shows how the algorithm works.

Of paramount importance is the handling of the constraints in the proposed decomposition methodology. The feasibility of the solutions strongly depends on the values obtained in the discrete variables of problem P1. That is because the constraints in the problem P2 are created using the values of the discrete variables. Therefore, the solutions of the problem P2 can be feasible or infeasible. A linear program is infeasible if there is no solution that satisfies all of the constraints at the same time. We evaluated and classified the infeasible solutions generated by the exact method by assigning them a value based on certain parameters of the exact method. This parameter relates to the number of restrictions not met by these solutions, and is modeled using a numeric value. We assign this fitness penalty-based value to infeasible solutions. When this value is close to 0 it means that the solution is close to being feasible, and implies that it needs fewer changes than other infeasible ones.

From the initial population, a number of elements from both groups are selected (feasible and infeasible solutions) and used to generate new solutions.

We first sort the valid solutions, in decreasing order of fitness, followed by the invalid ones, which are ordered randomly. We select a percentage of solutions to be combined and mutated. We select the best solutions (with the highest fitness) from the valid set while the solutions from the invalid set are selected randomly.

The algorithm includes a combination function that combines pairs of solutions, which are randomly chosen from those selected previously. The pair of solutions must belong to the same group, whereby valid solutions are combined with valid solutions and invalid solutions with invalid ones. These combinations generate new solutions that inherit some characteristics from their parents, and, for all the combinations, the algorithm only uses the discrete variables from P1. The remaining the variables are obtained by solving P1 using the exact method. To execute these combinations, a multi-point crossover operator was developed which generates an offspring by copying its genes from the parents, and which are chosen according to a randomly constructed crossover mask. We use this mask, which contains ones and zeros randomly generated with the same probability to generate new discrete variables. We determine the selected values from each of the two solutions by the mask in each position, taking the value from the first solution if there is 1 in the mask, or, if not, from the second one.

Algorithm 1 **Matheuristic algorithm.**

REQUIRE: MILP problem (x,y)

ENSURE: Best solution $[max\ f(x,y)]$

//The selected metaheuristic could be that proposed in section 4 or another from the literature; 2 Fix the metaheuristic parameters (*Population*, *Combination*, *TotalImprove*, *Diversification*)

//Create S set of solutions;

for $j = 1$ **to** *Population* **do**

Fix discrete variables y_j of problem $P1$;

Obtain continuous variables x_j solving $P2$ using the exact method;

$S \leftarrow [Solution_j := (y_j, x_j)]$;

if $Solution_j$ *is not feasible* **then**

Improve $Solution_j$ using the Variable neighborhood search algorithm (*Mladenović & Hanse, 1997*);

end

end

do

//Crossover SS subset of S such as $|SS| > 1$;

for $w = 1$ **to** *Combination* **do**

Parents $\leftarrow$ *RandomSelect*(s_1 and s_2 from S);

$y_w \leftarrow$ Crossover(y_1 from s_1, y_2 from s_2);

$x_w \leftarrow$ *ExactMethod*(P2,y_w);

s_w: = (y_w,x_w);

if $Fitness(s_w) > Fitness(s_1)$ or $Fitness(s_w) > Fitness(s_2)$ **then**

$SS \leftarrow s_w$;

end

end

//Improve SSI subset of SS;

for $w = 1$ **to** *TotalImprove* **do**

Select $s_w \in SS$ randomly;

REPEAT:;

Modify y_w using the best neighbourhood algorithm and obtain x_w solving $P2$ using the exact method;

UNTIL $Fitness(s_w)$ increase or achieve *EndConditions*;

end

//Diversify SSD subset of SSI;

for $w = 1$ **to** *Diversification* **do**

Select $s_w \in SSI$ randomly ;

Modify randomly y_w of s_w;

Obtain x_w solving $P2$ using the exact method;

end

Include *SSD* in *S*

while *not EndCondition*;

;

$BestSolution_k \leftarrow s \in S$ such as $Fitness(s) \geq Fitness(w)\ \forall\ w \in S$;

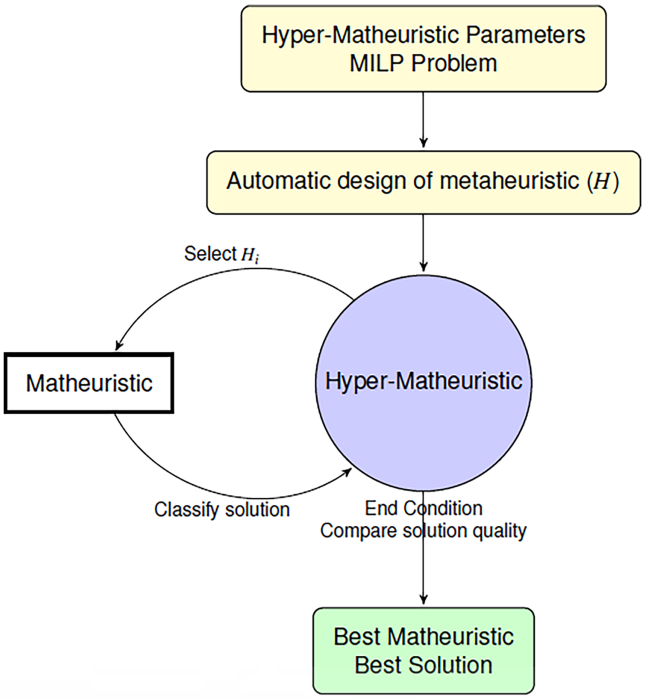

Figure 3 **Algorithm to find the best matheuristic using the hyper-matheuristic method.**
Full-size DOI: 10.7717/peerj-cs.828/fig-3

We also evaluated and improved all these new generated solutions in order to maximize the number of feasible solutions. Those steps of the algorithm are repeated a given number of times. Algorithm 1 is a schematic representation of the main matheuristic algorithm. This algorithm defines the extent to which the metaheuristic and the exact method are involved.

HYPER-MATHEURISTIC METHODOLOGY

In this work, a hyper-matheuristic framework is developed to generalize the matheuristic scheme proposed in *González et al. (2017)*. We develop a hyperheuristic method on top of the matheuristic to find the best Metaheuristic in terms of fitness (optimal solution). For that, a set of metaheuristics (H) is created. In each iteration, the metaheuristic used in the matheuristic (H_i) is updated in an adaptive way, obtaining new possible solutions and time values.

We propose a hyper-matheuristic algorithm to find the best suited metaheuristics, that combined with an exact method, generates the best solution for each problem in the shortest time possible (Fig. 3). This algorithm searches the whole metaheuristic space to find the best design. Then, two levels of metaheuristics are developed. One of them uses the objective function for each problem to evaluate the solutions (matheuristic), and the other one uses the average of all the objective values obtained and the time used to evaluate all the metaheuristics used (hyper-matheuristic).

Table 1 Parameters used in each basic function of the parameterized metaheuristic scheme.

Function	Parameters	Description
Initialize	INEIni	Initial number of elements
	FNEIni	Final number of elements selected for the iterations
	PEIIni	Percentage of elements to improve in the initialization
	IIEIni	Intensification of the improvement in the initialization
End condition	MNIEnd	Maximum number of iterations
	NIREnd	Maximum number of iterations without improving
Selection	NBESel	Number of best feasible solutions selected
	NWESel	Number of infeasible solutions selected
Combination	PBBCom	Number of combinations between feasible solutions
	PWWCom	Number of combinations between infeasible solutions
Improve	PEIImp	Percentage of crossover elements to be improved
	IIEImp	Intensification of the improvement
	PEDImp	Percentage of elements to diversify
	IIDImp	Intensification of the improvement to diversify elements

The next subsections describe how the hyper-matheuristic framework works. First, the parameterized scheme to generate all the metaheuristics is introduced. Then, the hyper-matheuristic methodology is detailed.

A parameterized scheme of metaheuristics

This scheme is included in the hyper-matheuristic and offers the possibility of generating and analyzing a large number of combinations between different metaheuristics. Depending on the problem evaluated, there is a large number of metaheuristic algorithms that can obtain good solutions. The objective of the scheme is to offer the possibility of using different metaheuristics for each problem, as well as being able to generate hybrid metaheuristics. This scheme is included in the work so as not to particularize a complete metaheuristic, but to use its most interesting functionalities. According to certain parameters, the parameterized scheme is able to generate hybrid metaheuristics that shares information from general schemes, like EA, GRASP or SS. This scheme has the possibility of designing a large number of metaheuristics in a general way, varying the value of all the parameters inside.

In this work, the main search components used for the design of a metaheuristic are: Initialize (*i.e.* initialization of the population), Improvements, Selection, Combinations (*e.g.* crossover) and Stopping Criteria. For each search component, several parameters are included. Table 1 summarizes the parameters used to generate some well-known and diverse sets of metaheuristics automatically.

A parameterized scheme was already employed and tested in *Almeida et al. (2013)*, where other parameters such as combinations between feasible and infeasible solutions were examined. The number and meaning of the parameters would vary if other basic metaheuristics were evaluated or if the basic functions were executed differently, but the parameters considered here allow us to automatically generate and experiment with

different metaheuristics and combinations of them in order to improve the results obtained.

A large number of combinations can be considered simply by selecting different values for the considered parameters. The best metaheuristic with the parameterized scheme can be obtained by generating all the possible combinations of the parameters and by applying them to some small training set of problem instances. In this way, the generated combination of the various metaheuristics, given by the values of the parameters, is that which gives the best results in terms of the training set and can be deemed to be a satisfactory metaheuristic for the problem under consideration. There are many possible combinations of the parameters in the parameterized metaheuristic scheme, obtaining the best metaheuristic for the training set is an expensive optimization problem, and therefore suitable for the hyper-matheuristic.

Hyper-matheuristic

To obtain the best metaheuristic that provides the best objective quality in less search time, a metaheuristic has been developed at a higher level called hyper-matheuristic. This new metaheuristic is developed to generate, evaluate and improve different types of metaheuristics (set H_i). The developed hyper-matheuristic makes it possible to design a set of metaheuristics H_i automatically. Then, the hyper-matheuristic is a metaheuristic included on top of the matheuristic algorithm that is able to select the best value of the parameters in the Parameterized Scheme, with the aim of designing an efficient metaheuristic that, combined with an exact method, provides the best objetive funcion value. The generated metaheuristics depend on the value of the parameters in the parameterized scheme. Depending on the solution quality obtained for each problem and the search time used, the hyper-matheuristic is able to adapt itself modifying the next metaheuristic to improve on the previous one. For that, this algorithm saves all the information about all metaheuristics in H_i, thus it learns every step online. In order to have an initial reference, the EA, GRASP and SS methods have been established as diverse metaheuristic prototypes, so that all the metaheuristics generated by the hyper-matheuristic will have certain functionalities of those general metaheuristic frameworks. To determine the number of metaheuristics evaluated for each problem, as well as the number of applied improvements and changes, a certain number of parameters have been designed to control these aspects:

- NIM_EA: Number of initial metaheuristics generated from evolutionary algorithms (EA).
- NIM_SS: Number of initial metaheuristics generated from scatter search (SS).
- NIM_GRASP: Number of initial metaheuristics generated from greedy randomized adaptive search procedure (GRASP).
- NIM: Total number of initial metaheuristics
- (NIM_EA+NIM_SS+NIM_GRASP).
- NFM: Number of new metaheuristics created by recombination.

Algorithm 2 **Hyper-matheuristic algorithm.**

REQUIRE: Hyper-matheuristic parameters, MILP Problem
ENSURE: *Hi*, Best metaheuristic.
1 *Fix limits for all the parameters in the scheme (upper and lower bound);*
2 **for** i = 1 **to** *NIM* **do**
3 *Generate matheuristic* H_i;
4 **End**
5 **for** i = 1 **to** *NIM* **do**
6 *Solve problem using the matheuristic in* H_i;
7 **for** k = 1 **to** i **do**
8 **if** (H_i *solution quality*) < (H_k *solution quality*) **then**
9 *Compare parameters between the metaheuristics*;
10 *Improve the matheuristic* H_i *modifying the parameters with more differences*;
11 **end**
12 **end**
13 **end**
14 **for** i = *NIM* **to** *NIM* + *NFM* **do**
15 *Select two matheuristics randomly from those generated in step 3*;
16 *Combine their parameters to create a new metaheuristic*;
17 *Execute steps from line 6 to 12*;
18 **end**
19 *Classify all the solutions by quality and time;*
20 *Select the best matheuristic (which maximize fitness/time), and save parameters;*

First, a given number of metaheuristics (NIM) based on EA, GRASP and SS are generated in the reference set. In order to achieve the desired metaheuristic functionalities, values are assigned to the parameters in Table 1, establishing values between 0 and a preset limit that has been evaluated during the experiments. The established limits for each parameter can be modified in each execution to study the variances in the solution quality and the search time obtained. In each iteration, the MILP problem is solved using a metaheuristic included in H_i, providing values of solution quality and search time.

When the initial population is created, all the created metaheuristics are selected to be improved. Several improved functions are developed to modify the metaheuristic parameters, increasing the parameters associated to some functionalities such as the initial population, improvements or crossovers while decreasing others at the same time. With these improved functions, the metaheuristics are evaluated and compared depending on their functionalities. The new improved metaheuristics are added to the reference set. With this reference set, the crossover function is executed. Hence, new metaheuristics are created, where their parameters are selected from the metaheuristics in the reference set. The parameters from the metaheuristics of a better quality have a higher probability of being selected. Algorithm 2 shows how the proposed hyper-matheuristic methodology works.

EXPERIMENTAL RESULTS

We performed various experiments to analyze the effectiveness of the hyper-matheuristic methodology for a given MILP problem coming from the DEA literature. To be specific, the aim of the solved problem is to ascertain the technical efficiency of a set of n firms (in general, a set of Decision Making Units - DMUs), which use m inputs to generate s outputs. To that end, we will apply a well-known non-parametric technique, called Data Envelopment Analysis (DEA) (*Charnes, Cooper & Rhodes, 1978*). The MILP problem to be solved must be executed for each DMU of the sample of observations. Specifically, we focus our analysis on the Slacks-Based Measure developed in *Aparicio, Ruiz & Sirvent (2007)*, since it must be computed by the MILP. With regard to the data, in our simulations the m inputs and s outputs of each of the n DMUs are generated randomly but bearing in mind that the well-known Cobb-Douglas function (*Cobb & Douglas, 1928*) is the function governing the production situation.

Let us assume that data on m inputs and s outputs for n DMUs are observed. For the jth DMU these are represented by $z_{ij} \geq 0$, $i = 1,\ldots,m$ and $q_{rj} \geq 0$, $r = 1,\ldots,s$. The DEA model that should be solved as follows:

$$\max \left\{ \beta_k - \frac{1}{m} \sum_{i=1}^{m} \frac{t_{ik}^-}{z_{ik}} \right\}$$

s.t.

$$\begin{aligned}
&\beta_k + \frac{1}{s} \sum_{r=1}^{s} \frac{t_{rk}^+}{q_{rk}} \leq 1 && (c.1)\\
&-\beta_k - \frac{1}{s} \sum_{r=1}^{s} \frac{t_{rk}^+}{q_{rk}} \leq -1 && (c.2)\\
&-\beta_k z_{ik} + \sum_{j=1}^{n} \alpha_{jk} x_{ij} + t_{ik}^- \leq 0 \quad \forall i = 1,\ldots,m && (c.3)\\
&\beta_k z_{ik} - \sum_{j=1}^{n} \alpha_{jk} x_{ij} - t_{ik}^- \leq 0 \quad \forall i = 1,\ldots,m && (c.4)\\
&-\beta_k q_{rk} + \sum_{j=1}^{n} \alpha_{jk} y_{rj} - t_{rk}^+ \leq 0 \quad \forall r = 1,\ldots,s && (c.5)\\
&\beta_k q_{rk} - \sum_{j=1}^{n} \alpha_{jk} y_{rj} + t_{rk}^+ \leq 0 \quad \forall r = 1,\ldots,s && (c.6)\\
&-\sum_{i=1}^{m} v_{ik} z_{ij} + \sum_{r=1}^{s} \mu_{rk} q_{rj} + d_{jk} \leq 0 \quad \forall j = 1,\ldots,n && (c.7)\\
&\sum_{i=1}^{m} v_{ik} z_{ij} - \sum_{r=1}^{s} \mu_{rk} q_{rj} - d_{jk} \leq 0 \quad \forall j = 1,\ldots,n && (c.8)\\
&-v_{ik} \leq -1 \quad \forall i = 1,\ldots,m && (c.5)\\
&-\mu_{rk} \leq -1 \quad \forall r = 1,\ldots,s && (c.6)\\
&-d_{jk} \leq -M b_{jk} \quad \forall j = 1,\ldots,n && (c.7)\\
&\alpha_{jk} \leq M(1 - b_{jk}) \quad \forall j = 1,\ldots,n && (c.8)\\
&b_{jk} = 0, 1 \quad \forall j = 1,\ldots,n && (c.9)\\
&-\beta_k \leq 0 && (c.10)\\
&-t_{ik}^- \leq 0 \quad \forall i = 1,\ldots,m && (c.11)\\
&-t_{rk}^+ \leq 0 \quad \forall r = 1,\ldots,s && (c.12)\\
&-d_{jk} \leq 0 \quad \forall j = 1,\ldots,n && (c.13)\\
&-\alpha_{jk} \leq 0 \quad \forall j = 1,\ldots,n && (c.14)
\end{aligned} \tag{3}$$

where M is a large, positive number. For this specific MILP problem, the vector of continuous variables x consists of (β_k, t^-_{ik}, t^+_{rk}, d_{jk} and α_{jk}), while the vector of integer variables consists exclusively of b_{jk}.

The number of feasible solutions obtained in the initial population using the algorithm put forward in this paper is studied for various population sizes and optimization methods. In the first experiment, the methods proposed above for solving the main problem are evaluated, comparing the results of solving the problem globally (heuristic) with those obtained by applying the decomposition of the problem (matheuristic). In addition, a hybrid method in which both techniques are executed in a cooperative way is added. After that, the obtained solution quality with basic metaheuristics with parameters similiar of those of EA, GRASP or SS is compared with hybrid metaheuristics generated automatically with a hyper-matheuristic using the parameterized scheme shown in "A Parameterized Scheme of Metaheuristics". The hyper-matheuristic is trained with many instances to obtain a satisfactory metaheuristic for any problem size.

Finally, we compare the solution quality and the execution time obtained using the satisfactory metaheuristic with those obtained using other metaheuristics. For all the experiments, we use the IBM ILOG CPLEX Optimization Studio (CPLEX). The experiments are executed in a parallel NUMA node with 4 Intel hexa-core Nehalem-EX EC E7530, with 24 cores, at 1.87 GHz and 32 GB of RAM. The environment used to run the application is a Centos 8 operating system, using C code to develop the algorithm. Additionally, the Intel C++ Compiler was used and the Intel MKL Libraries were included.

MILP-decomposition vs global problem solving

To evaluate the proposed decomposition strategy, several experiments have been carried out, in which the focus is made on each of the algorithm stages. First, the generation of the initial population was evaluated. Then the algorithm was executed using several generation methods: heuristic method and matheuristic method. The heuristic method used is proposed in *González et al. (2015)* where a problem dependent algorithm is developed. This heuristic does not use a decomposition variable method. Therefore, it is a good option to compare it with the matheuristic algorithm used in this paper. The hybrid method is a combination of the proposed heuristic in *González et al. (2015)* and exact methods, where the number of problems to evaluate (DMUs) is divided in two groups. One of the groups is solved by the exact method, and the other is solved by the heuristic method. In this experiment, the parameters have been fixed, being the same in all the methods used. The population size (INEIni) is set by default to 100. In addition, another implementation of the exact method has been included, where the number of initial solutions (INEIni) has been increased to 1,000 to analyze the impact of this parameter.

Table 2 shows the average of the objective values and the percentage of feasible solutions obtained in the Initialization step of Algorithm 1, according to the method used and using several values for the initial population. The experiment shows that it is non-trivial to obtain feasible solutions using the matheuristic method, because the search space is huge, and the metaheuristic needs to make a great effort to obtain satisfactory values for the discrete variables. When the initial population grows, the number of feasible solutions

Table 2 Average of percentage of feasible solutions and fitness solution obtained at the initialization step for the different optimization methods and varying the INEIni parameter.

Size			Matheuristic 100		Heuristic method		Hybrid method		Matheuristic (1,000)	
m	*n*	*s*	% val.	Fitness	% val.	Fitness	% val.	Fitness	% val.	Fitness
2	100	1	$18_{5.16}$	0.7605	$94.96_{6.47}$	0.685	$56.946_{8.24}$	0.413	$6.032_{1.02}$	0.7609
3	50	1	$1.32_{1.61}$	0.4748	$78.85_{5.90}$	0.777	$43.441_{7.82}$	0.5661	$1.221_{0.77}$	0.7974
3	100	1	$0.55_{1.37}$	0.2713	$66.12_{5.17}$	0.6171	$32.539_{5.23}$	0.48	$0.758_{0.52}$	0.6662
4	100	1	$0.29_{0.54}$	0.1544	$60.71_{6.72}$	0.588	$25.015_{6.12}$	0.414	$0.305_{0.82}$	0.5515
5	100	1	$0.33_{0.81}$	0.1974	$58.66_{8.36}$	0.501	$39.485_{9.08}$	0.398	$0.371_{0.15}$	0.5164

increases (comparison between matheuristic with population size of 100 and 1,000). Moreover, the solution quality value greatly depends on the initial population. When the matheuristic is used with a population size of 1,000 solutions, the value of the fitness of the obtained solutions using the matheuristic improves compared to the other methods. Thus, it can be observed that, for the same initial population value, the heuristic method obtains a higher number of feasible solutions but with a lower quality. So, obtaining feasible solutions is a really difficult task in terms of the solution space. Metaheuristics need a huge effort to find feasible solutions in the set of discrete variables that satisfy all the constraints in the continuous variables. Additionally, the quality of the solutions suggests that only a few solutions are needed to obtain the optimal one. Now, it is the moment to evaluate the complete matheuristic algorithm and compare fitness and the amount of computational time required.

Matheuristic in the parameterized scheme of metaheuristics

Before executing the complete algorithm and developing the best-found hyper-matheuristic, the algorithm has been evaluated with fixed parametrized scheme values. The metaheuristics used for the two methods are configured to be as versatile as possible (with all internal functions), using low values to improve the execution time. In order to make fair time comparisons, the values of the various parameters for the different optimization methods are set to the same values: INEIni = 100, FNEINI = 50, IIEIni = 10, PEIIni = 10 NBESel = 15, NWESel = 15, PBBCom = 25, PWWCom = 25, PEEImp = 10, IIEImp = 5, PEDImp = 5, IIDImp = 5, NIREnd = 5, MNIEnd = 10. The solution quality is shown in the Fig. 4, and the execution time in Fig. 5.

The results obtained show that there is a high correlation between the execution time and the quality of the obtained solutions. It can be observed that using the matheuristic with a low number of initial solutions (INEIni), the solution quality obtained is the lowest, but the solution is found in the least time. On the other hand, it is observed that the matheuristic is much faster than the heuristic method (see Fig. 5). However, the solution quality obtained by the heuristic method for these low initial population values is better than those obtained by the matheuristic. This leads to the conclusion that, in order to obtain an efficient hyper-matheuristic method in the following steps, the heuristic method must be discarded. The heuristic method would only be optimal for small problem sizes.

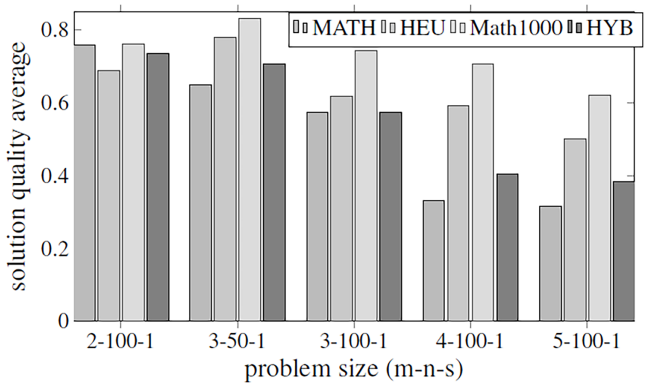

Figure 4 Objective values obtained by the methods proposed in Table 2 using all the parameters of Table 1. The methods compared are matheuristic (MATH), heuristic (HEU), hybrid method (HYB) and matheuristic using a population size of 1,000 (Math1000). Full-size DOI: 10.7717/peerj-cs.828/fig-4

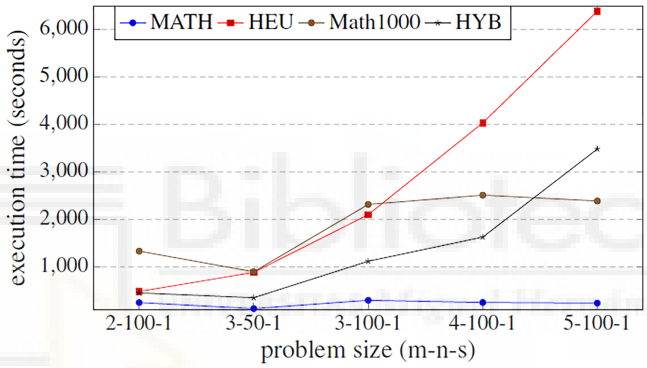

Figure 5 Comparison of the execution time function (in seconds) of the problem size. The methods compared are matheuristic (MATH), heuristic (HEU), hybrid method (HYB) and matheuristic 1,000 (Math1000). Full-size DOI: 10.7717/peerj-cs.828/fig-5

Since we want to make an algorithm independent of the type and size of problem, it was preferred not to use this method and only design the hyper-matheuristic including the matheuristic as such. This experiment concludes with the comparison between the fitness obtained by the feasible solution and the computational costs required. The figures explain that the initial population is a critical parameter, the computational time during which the size of the problem grows being a great limitation. Both values (fitness and time) must be evaluated and taken into account in the following experiments.

Hyper-matheuristic vs general metaheuristics

To develop the hyper-matheuristic methodology, some general metaheuristics are used. The automatic design of the best hyper-matheuristic within this space of algorithms is the main issue. All the metaheuristics generated by the hyper-matheuristics will inherit some characteristics from the general metaheuristics, and will combine others. The proposed metaheuristics to be executed in the experiments are shown in Table 3. The lowest and the highest values of the parameters of these metaheuristics are used to limit the values of

Table 3 Values of the parameters for the three basic metaheuristics considered and the hyper-matheuristic limits.

Metaheuristic	IINEIni	FNEIni	PEIIni	IIEIni	NBESel	NWESel	PBBCom
EA	300	150	0	0	100	0	50
GR	500	1	100	20	0	0	0
SS	100	50	50	5	25	25	25
Hyper	100/500	1/150	0/100	0/20	0/100	0/25	0/50
Metaheuristic	PWWCom	PEIImp	IIEImp	PEDImp	IDEImp	MNIEnd	NIREnd
EA	0	0	0	10	10	10	5
GR	0	0	0	0	0	10	5
SS	25	50	10	0	0	10	5
Hyper	0/25	0/50	0/10	0/10	0/10	0/10	0/5

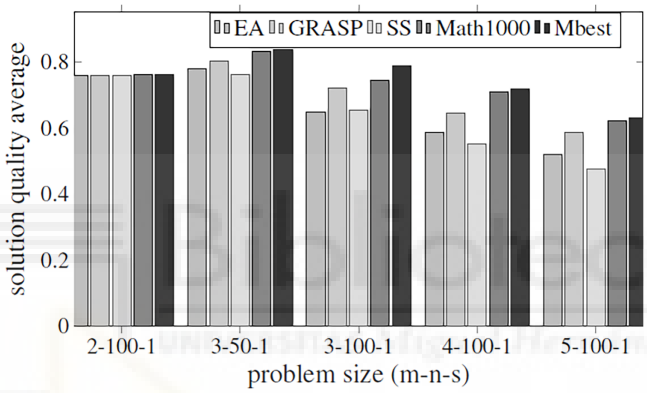

Figure 6 Fitness values obtained by the three basic metaheuristics, an exact method and the hyper-matheuristic (Mbest), varying the problem size. Full-size DOI: 10.7717/peerj-cs.828/fig-6

all the parameters within the hyper-matheuristic. Then, all the metaheuristics will generated using these limits.

The aim of this experiment is to be able to set the values for all parameters using them as default values for any problem size. A specific problem has been established (3/50/1) to train the hyper-matheuristic and get a good configuration. This training has been performed by running the hyper-matheuristic for this problem with 100 different metaheuristics and 100 combinations of these, where each evaluated metaheuristic has been tested 10 times for each model, obtaining the average value for each metaheuristic. At the end of the experiment, the best average quality of the 200 parameter settings (which means 200 different metaheuristics) has been obtained. For the solution quality evaluation of each metaheuristic, the average quality of all executions and DMUs has been taken into account, as well as the average time required to terminate the search. In conclusion, a ratio between solution quality and search time (*fitness*/*time*) is used to rank the metaheuristics according to the solution quality obtained and the search time. In Fig. 6, all the obtained qualities are compared between the three general metaheuristics proposed (EA, GRASP and SS), the best metaheuristic obtained by the hyper-matheuristic (Mbest) and the matheuristic method with initial population of 1,000 solutions (Math1000).

Table 4 Values of the parameters for the best metaheuristic (Mbest) found by the hyper-matheuristic in the matheuristic.

Metaheuristic	IINEIni	FNEIni	PEIIni	IIEIni	NBESel	NWESel	PBBCom
Hyper	310	61	13	17	12	10	22
Metaheuristic	PWWCom	PEIImp	IIEImp	PEDImp	IDEImp	MNIEnd	NIREnd
Hyper	25	12	8	3	7	6	5

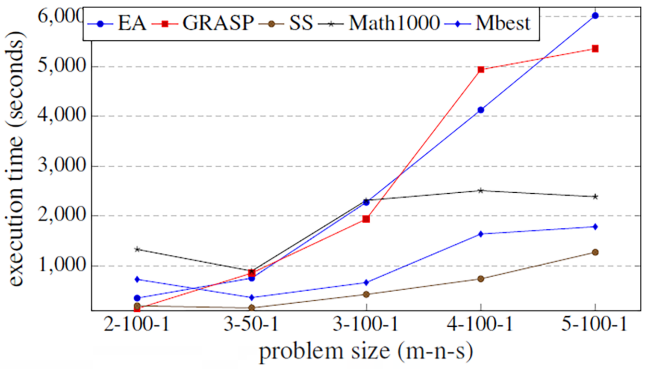

Figure 7 Comparison of the execution time (in seconds) between the three basic metaheuristics, the exact method and the hyper-matheuristic (Mbest) function of the problem size.
Full-size DOI: 10.7717/peerj-cs.828/fig-7

Figure 6 shows that the values obtained for the parameters in the hyper-matheuristics always improve the solution quality obtained with the other metaheuristics. The results obtained (Table 4) show that all the functions developed are used, but they do not need high values. It can be observed that for the three proposed metaheuristics, the GRASP strategy obtains better fitness solution. On the other hand, SS has the lowest initial population solutions and then lower probability of obtaining the best solution, despite incorporating all the functionalities. The fitness solution obtained with the Mbest is quite similar to the quality obtained by the matheuristic using the value of 1,000 in the initial solution set, but if the execution time is compared (Fig. 7), the Mbest is always faster than the matheuristic with the default parameter values used in the previous experiments. We can conclude that the trained hyper-matheuristic is able to obtain a good configuration of parameters, improving the quality and time for any problem size.

Bender's decomposition evaluation

In this subsection, our approach and that based on solving the MILP problem by the optimizer and Bender's decomposition are compared. For this, a last experiment was carried out, where based on a fixed size ($m = 3$, $s = 1$, $n = 100$), different problems generated have been evaluated obtaining, for both techniques, the optimal value of the objective function. Figure 8 shows a comparison between these two techniques (Bender's and Mbest). In particular, it graphically illustrates the objective function value for 20 simulated problems by means of bars. The bar is not drawn if the corresponding technique does not find any solution. Something that only happens for Bender's decomposition.

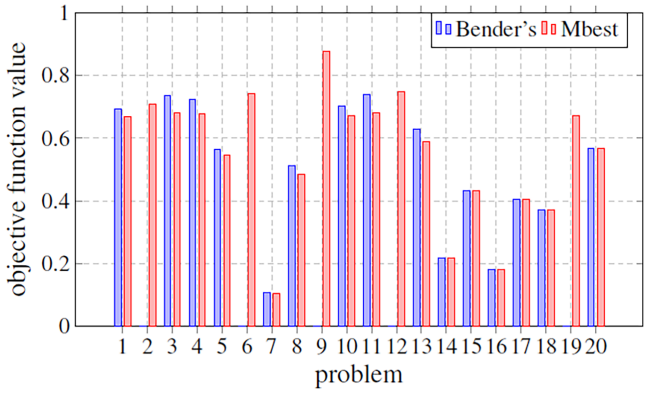

Figure 8 Comparison of the objective function value in several problems with a fixed size (m = 3; s = 1; n = 100). The methods compared are Bender's Decomposition (Bender's) and the best matheuristic found by the hyper-matheuristic (Mbest). Full-size DOI: 10.7717/peerj-cs.828/fig-8

Figure 8 shows how, when both techniques find an optimal solution, Bender's and Mbest get very similar values. For some problems, Bender's decomposition does not find any feasible solution. This contrasts with the Mbest technique, which is able to find solutions for all the cases studied. From this, it can be deduced that the proposed technique is valid for all the problems evaluated and that, in addition, it finds solutions very close to the optimum, since very similar results are obtained when comparing them with those obtained by an exact method through Bender's decomposition. Therefore, we can see that the hyper-matheuristic introduced fulfils the proposed objective satisfactorily, being a complementary technique to those already known in the literature.

Our conclusions associated with the computational experience are limited by the type of optimization problem that was analyzed, within the context of Data Envelopment Analysis. Further evaluations of the new methodology on different types of optimization problems (such as packing, knapsack, inventory, *etc.*) would be needed to lead to more robust conclusions.

CONCLUSIONS AND FUTURE WORKS

In this paper, we have developed an efficient decomposition strategy for MILP optimization problems in the context of Data Envelopment Analysis (DEA). We have developed a hierarchical decomposition based on the nature of the decision variables (continuous *vs* discrete) and the complexity of the subproblems. An incomplete encoding representing only discrete decision variables is explored by the metaheuristics. The encoding of solutions is completed for the continuous decision variables by solving a linear problem exactly.

This matheuristic framework has shown its validity in solving MILP problems in the framework of DEA. Moreover, we developed a hyper-matheuristic methodology on top of the parameterized metaheuristic scheme. It allows the automatic design and configuration of a flexible and generic template for population-based metaheuristics. Satisfactory results have been obtained in terms of solution quality and execution time. Other computational intelligence algorithms could be used to solve the problems, like the

monarch butterfly optimization (MBO) (*Wang, Deb & Cu, 2019*) or the earthworm optimization algorithm (EWA) (*Wang, Deb & Coelho, 2018*).

One of the future research lines of this paper is to apply this hyper-matheuristic methodology to other real-life optimization problems formulated as MILP, such as unit commitment problems in power energy systems and demand side management in smart grids. Another perspective consists in the generalization of the proposed decomposition scheme for other families of optimization problems, in which only the continuous part of the problem is linear and easy to solve using an exact algorithm. Indeed, the most important feature of the proposed decomposition scheme is the complexity of the subproblems generated by metaheuristics and solved by exact algorithms. Another interesting perspective is to investigate the parallel design and implementation of the hyper-matheuristic methodology. Indeed, the proposed decomposition strategy is suitable to be deployed on heterogeneous parallel architectures composed of clusters of multiple cores and GPUs (Graphics Processing Units).

As a limitation, we point out that the proposed algorithm has been proved in only one kind of MILP problem, within the Data Envelopment Analysis field. Also, the algorithm developed in this paper is just applicable to MILP problems which include both discrete and continuous variables. An interesting future line of research would be to apply the new methodology to different specific types of optimization problems: packing, knapsack, inventory, production planning, location, resource allocation, routing or scheduling problems, to name but a few. This analysis would allow us to shed light on the adequacy of the new approach for solving very different optimization problems with varied structures. Another possible line of further research would consist of incorporating Bender's decomposition to our approach to improve the computational time or, even consider Bender's method as a new feature in the hyper-matheuristic.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This work was supported by the Spanish Ministry of Science and Innovation and the State Research Agency under grant PID2019-105952GB-I00/AEI/10.13039/501100011033. We also have financial support from the Spanish Ministry for Economy and Competitiveness (Ministerio de Economía, Industria y Competitividad), the State Research Agency (Agencia Estatal de Investigacion) and the European Regional Development Fund (Fondo Europeo de Desarrollo Regional) under grant TIN2016-80565-R (AEI/FEDER, UE). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Spanish Ministry of Science and Innovation and the State Research Agency: PID2019-105952GB-I00/AEI/10.13039/501100011033.

Spanish Ministry for Economy and Competitiveness (Ministerio de Economía, Industria y Competitividad).
State Research Agency (Agencia Estatal de Investigacion).
European Regional Development Fund (Fondo Europeo de Desarrollo Regional): TIN2016-80565-R (AEI/FEDER, UE).

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Martin Gonzalez conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Jose J. López-Espín conceived and designed the experiments, performed the experiments, analyzed the data, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Juan Aparicio conceived and designed the experiments, authored or reviewed drafts of the paper, and approved the final draft.
- El-Ghazali Talbi conceived and designed the experiments, analyzed the data, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The code is available in the Supplemental Files.

Supplemental Information

Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj-cs.828#supplemental-information.

REFERENCES

Almeida F, Giménez D, López-Espn JJ, Pérez-Pérez M. 2013. Parameterized schemes of metaheuristics: basic ideas and applications with genetic algorithms, scatter search and GRASP. *IEEE Transactions on Systems, Man, and Cybernetics* **43(3)**:570–586 DOI 10.1109/TSMCA.2012.2217322.

Aparicio J, Ruiz JL, Sirvent I. 2007. Closest targets and minimum distance to the Pareto-efficient frontier in DEA. *Journal of Productivity Analysis* **28(3)**:209–218 DOI 10.1007/s11123-007-0039-5.

Bender's J. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4(1)**:238–252 DOI 10.1007/BF01386316.

Charnes A, Cooper WW, Rhodes E. 1978. Measuring the efficiency of decision making units. *European Journal of Operational Research* **2(6)**:429–444 DOI 10.1016/0377-2217(78)90138-8.

Cobb CW, Douglas PH. 1928. A theory of production. *The American Economic Review* **18(1)**:139–165.

Glover F. 1997. Tabu search and adaptive memory programming-advances, applications and challenges. In: *Interfaces in Computer Science and Operations Research*. Boston: Springer, 1–75.

Glover F, Laguna M, Marti R. 2003. Scatter search and path relinking: advances and applications. In: *Handbook of Metaheuristics*. Boston: Springer, 1–35.

González M, López Espín JJ, Aparicio J, Giménez D, Talbi E. 2017. A parameterized scheme of metaheuristics with exact methods for determining the principle of least action in data envelopment analysis. In: *Program of the 2017 IEEE Congress on Evolutionary Computation,* Piscataway: IEEE.

González M, López-Espn JJ, Aparicio J, Giménez D, Pastor JT. 2015. Using genetic algorithms for maximizing technical efficiency in data envelopment analysis. In: *Proceedings of the International Conference on Computational Science, ICCS 2015*. Reykjavík, Iceland, **51**:374–383.

Hochba DS. 1997. Approximation algorithms for NP-hard problems. *ACM Sigact News* **28(2)**:40–52 DOI 10.1145/261342.571216.

Holland JH. 1973. Genetic algorithms and the optimal allocation of trials. *SIAM Journal on Computing* **2(2)**:88–105 DOI 10.1137/0202009.

Jourdan L, Basseur M, Talbi EG. 2009. Hybridizing exact methods and metaheuristics: a taxonomy. *European Journal of Operational Research* **119(3)**:620–629 DOI 10.1016/j.ejor.2007.07.035.

Kelley JE. 1960. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics* **8(4)**:703–712 DOI 10.1137/0108053.

Laporte G, Potvin JY, Quilleret F. 1997. A tabu search heuristic using genetic diversification for the clustered traveling salesman problem. *Journal of Heuristics* **2(3)**:187–200 DOI 10.1007/BF00127356.

Li J, Lei H, Alavi AH, Wang GG. 2020. Elephant herding optimization: variants, hybrids, and applications. *Mathematics* **8(9)**:1415 DOI 10.3390/math8091415.

Li X, Yalaoui F, Amodeo L, Chehade H. 2012. Metaheuristics and exact methods to solve a multiobjective parallel machines scheduling problem. *Journal of Intelligent Manufacturing* **23(4)**:1179–1194 DOI 10.1007/s10845-010-0428-x.

Mladenović N, Hanse P. 1997. Variable neighborhood search. *Computers & Operations Research* **24(11)**:1097–1100 DOI 10.1016/S0305-0548(97)00031-2.

Pillay N, Banzhaf W. 2009. A study of heuristic combinations for hyper-heuristic systems for the uncapacitated examination timetabling problem. *European Journal of Operational Research* **197(2)**:482–491 DOI 10.1016/j.ejor.2008.07.023.

Poojari CA, Beasley JE. 2009. Improving Bender's decomposition using a genetic algorithm. *European Journal of Operational Research* **199(1)**:89–97 DOI 10.1016/j.ejor.2008.10.033.

Pradenas L, Garcés J, Parada V, Ferland J. 2013. Genotype-phenotype heuristic approaches for a cutting stock problem with circular patterns. *Engineering Applications of Artificial Intelligence* **26(10)**:2349–2355 DOI 10.1016/j.engappai.2013.08.003.

Puchinger J, Raidl GR. 2005. Combining metaheuristics and exact algorithms in combinatorial optimization: a survey and classification. In: Mira J, Álvarez JR, eds. *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach. IWINAC 2005. Lecture Notes in Computer Science.* Vol. 3562. Berlin: Springer, 41–53.

Rahmaniani R, Crainic TG, Gendreau M, Rei W. 2017. The Bender's decomposition algorithm: a literature review. *European Journal of Operational Research* **259(3)**:801–817 DOI 10.1016/j.ejor.2016.12.005.

Raidl GR. 2015. Decomposition based hybrid metaheuristics. *European Journal of Operational Research* **244(1)**:66–76 DOI 10.1016/j.ejor.2014.12.005.

Ralphs TK, Gelati MV. 2010. Decomposition methods for integer programming. In: *Wiley Encyclopedia of Operations Research and Managemenr Science*. Hoboken: John Wiley & Sons.

Reimann M, Doerner K, Hartl RF. 2004. D-ants: savings based ants divide and conquer the vehicle routing problem. *Computers and Operations Research* **31(3)**:563–591 DOI 10.1016/S0305-0548(03)00014-5.

Resende MG, Ribeiro CC. 2003. Greedy randomized adaptive search procedures. In: *Handbook of Metaheuristics*. Boston: Springer, 219–249.

Ruszczynski A. 1989. An augmented Lagrangian decomposition method for the block diagonal linear programming problems. *Operations Research Letters* **8(5)**:287–294 DOI 10.1016/0167-6377(89)90055-2.

Sadykov R, Vanderbeck F, Pessoa A, Tahiri I, Uchoa E. 2019. Primal heuristics for branch and price: the assets of diving methods. *INFORMS Journal on Computing* **31(2)**:251–267 DOI 10.1287/ijoc.2018.0822.

Taillard E. 1993. Parallel iterative search methods for vehicle routing problems. *Networks* **23(8)**:661–673 DOI 10.1002/(ISSN)1097-0037.

Talbi EG. 2009. *Metaheuristics: from design to implementation*. Hoboken: Wiley.

Talbi EG. 2016. Combining metaheuristics with mathematical programming, constraint programming and machine learning. *Annals of Operations Research* **240(1)**:171–215 DOI 10.1007/s10479-015-2034-y.

Vanderbeck F. 2000. On Dantzig-Wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. *Operations Research* **48(1)**:111–128 DOI 10.1287/opre.48.1.111.12453.

Vanderbeck F, Wolsey LA. 2010. Reformulation and decomposition of integer programs. In: Junger M, Liebling T, Naddef D, Nemhauser G, Pulleyblank W, Reinelt G, Rinaldi G, Wolsey L, eds. *50 Years of Integer Programming 1958-2008*. Berlin: Springer, 435–502.

Wang GG, Deb S, Coelho LDS. 2018. Earthworm optimisation algorithm: a bio-inspired metaheuristic algorithm for global optimisation problems. *International Journal of Bio-Inspired Computation* **12(1)**:1–22 DOI 10.1504/IJBIC.2018.093328.

Wang GG, Deb S, Cu Z. 2019. Monarch butterfly optimization. *Neural Computing and Applications* **31(7)**:1995–2014 DOI 10.1007/s00521-015-1923-y.

Wang GG, Gandomi AH, Alavi AH, Gong D. 2019. A comprehensive review of krill herd algorithm: variants, hybrids and applications. *Intelligence Review* **51(1)**:119–148 DOI 10.1007/s10462-017-9559-1.

Winston WL, Goldberg JB. 2004. *Operations research: applications and algorithms*. Pacific Grove: Thomson Brooks/Cole.

Yokoyama R, Ito K. 2000. A novel decomposition method for MILP and its application to optimal operation of a thermal storage system. *Energy Conversion and Management* **41(16)**:1781–1795 DOI 10.1016/S0196-8904(00)00002-9.

Separatas

- *González, M., López-Espín, J. J., Aparicio, J., & Giménez, D. (2018). A Parallel Application of Matheuristics in Data Envelopment Analysis. In International Symposium on Distributed Computing and Artificial Intelligence (pp. 172-179). Springer, Cham.*

A Parallel Application of Matheuristics in Data Envelopment Analysis

Martín González[1], Jose J. López-Espín[1], Juan Aparicio[1], and Domingo Giménez[2]

[1] Center of Operations Research, Miguel Hernández University of Elche, Alicante, Spain

[2] Department of Computer Science and Systems, University of Murcia, Spain

Abstract. Data Envelopment Analysis (DEA) is a non-parametric methodology for estimating technical efficiency and benchmarking. In general, it is desirable that DEA generates the efficient closest targets as benchmarks for each assessed unit. This may be achieved through the application of the Principle of Least Action. However, the mathematical models associated with this principle are based fundamentally on combinatorial NP-hard problems, difficult to be solved. For this reason, this paper uses a parallel matheuristic algorithm, where metaheuristics and exact methods work together to find optimal solutions. Several parallel schemes are used in the algorithm, being possible for them to be configured at different stages of the algorithm. The main intention is to divide the number of problems to be evaluated in equal groups, so that they are resolved in different threads. The DEA problems to be evaluated in this paper are independent of each other, an indispensable requirement for this algorithm. In addition, taking into account that the main algorithm uses exact methods to solve the mathematical problems, different optimization software has been evaluated to compare their performance when executed in parallel. The method is competitive with exact methods, obtaining fitness close to the optimum with low computational time.

1 Introduction

Data Envelopment Analysis (DEA) is a mathematical programming, non-parametric technique commonly used to measure the relative performance of a set of homogeneous processing units, which use several inputs to produce several outputs. These operating units are usually called Decision Making Units (DMUs) in recognition of their autonomy in setting their input and output levels. Thanks to being a non-parametric technique, DEA does not need to suppose a particular functional form for the production function, technical efficiency may be easily evaluated with multiple inputs and outputs and it also produces relevant benchmarking information from a managerial point of view. In particular, DEA provides both input and output efficient targets, the coordinates of the projection point on the estimated efficient frontier, and represents levels of operation that can make the corresponding inefficient DMU perform efficiently.

978-1-5090-4601-0/17/$31.00

Traditional DEA measures maximize the total technical effort associated with the evaluated unit in order to reach the efficient frontier. Instead, it seems more natural to assume that inefficient DMUs apply a Principle of Least Action, a well-known law in physics, with the aim of being technically efficient. Otherwise, inefficient units would need to make an extra effort, decreasing inputs and/or increasing outputs, to reach the frontier. The application of this 'natural' Principle of Least Action is linked to the determination of the closest targets on the efficient frontier of the corresponding DEA production possibility set. This drawback of traditional DEA measures has aroused increasing interest among researchers to develop new models capable of yielding achievable targets. Examples are the papers by Briec and Lesourd [2], Pastor and Aparicio [3], Aparicio and Pastor [1,4,5,6] and Aparicio et al. [7].

The application of the Principle of Least Action has been recently studied from a metaheuristic perspective (Benavente et al. [8], López-Espín et al. [9] and González et al. [10]). In [8,9] heuristics were used to generate valid solutions for a subset of restrictions of the problem, while in [10] all the constraints are incorporated, the heuristics are improved, and new ones are developed, thereby generating initial populations of solutions that satisfy all constraints.

Our paper takes up where González et al. [10] left off in the application of metaheuristics to the approach in [1]. The improvement of previous heuristics for the generation of valid solutions is a possible option, but greatly limits the search for valid solutions for large problem sizes, because when the number of variables grows, the number of valid solutions decreases. Exact methods can also be used to solve these problems. The main drawback of these methods is the great amount of time needed to solve a NP-hard problem. When the problem grows, the number of possible combinations between variables increases exponentially.

The contributions of this work include the development of a parallel algorithm that belongs to the class of hybrid metaheuristics [11]. New parallel features have been included in the matheuristic developed in [12]. The algorithm developed is focused on the need to solve multiple simultaneous models. This is due to the DEA problem that concerns us, in which numerous models must be analyzed for each DMU evaluated. The aim is to separate the number of DMUs to be evaluated in the most efficient way in the different available threads. For this, message-passing (MPI) and shared memory (OpenMP) programming have been considered.

The remainder of the paper is organized as follows. In Section 2, a brief introduction to the main notions associated with Data Envelopment Analysis is presented, and existing approaches for determining closest targets are outlined. The working problem is also presented in this section. The parallel algorithm used to generate and improve valid solutions is studied in Section 3. In Section 4, the results of some experiments are summarized. Section 5 concludes the paper and outlines some possible lines of research.

2 Data Envelopment Analysis and the Problem to be Solved

DEA involves the use of mathematical programming to construct a non-parametric piecewise surface over the data in the input-output space. Technical efficiency measures associated with the performance of each DMU are then calculated relative to this surface, as a distance from it.

Before solving the mathematical programming model, we introduce some notations. Let us assume that data on m inputs and s outputs for n DMUs are observed. For the j-th DMU, these are represented by $x_{ij} \geq 0$, $i = 1, \ldots, m$, and $y_{rj} \geq 0$, $r = 1, \ldots, s$.

One of the models that can be solved by applying the Principle of Least Action in DEA is that by [1]:

$$\max \left\{ \beta_k - \frac{1}{m} \sum_{i=1}^{m} \frac{t_{ik}^-}{x_{ik}} \right\}$$

s.t.

$$\begin{aligned}
\beta_k + \frac{1}{s} \sum_{r=1}^{s} \frac{t_{rk}^+}{y_{rk}} &= 1 && (c.1)\\
-\beta_k x_{ik} + \sum_{j=1}^{n} \alpha_{jk} x_{ij} + t_{ik}^- &= 0 \ \forall i && (c.2)\\
-\beta_k y_{rk} + \sum_{j=1}^{n} \alpha_{jk} y_{rj} - t_{rk}^+ &= 0 \ \forall r && (c.3)\\
-\sum_{i=1}^{m} \nu_{ik} x_{ij} + \sum_{r=1}^{s} \mu_{rk} y_{rj} + d_{jk} &= 0 \ \forall j && (c.4)\\
\nu_{ik} &\geq 1 \ \forall i && (c.5)\\
\mu_{rk} &\geq 1 \ \forall r && (c.6)\\
d_{jk} &\leq M b_{jk} \ \forall j && (c.7)\\
\alpha_{jk} &\leq M(1 - b_{jk}) \ \forall j && (c.8)\\
b_{jk} &= 0, 1 \ \forall j && (c.9)\\
\beta_k &\geq 0 && (c.10)\\
t_{ik}^- &\geq 0 \ \forall i && (c.11)\\
t_{rk}^+ &\geq 0 \ \forall r && (c.12)\\
d_{jk} &\geq 0 \ \forall j && (c.13)\\
\alpha_{jk} &\geq 0 \ \forall j && (c.14)
\end{aligned} \tag{1}$$

The definition and interpretation of the decision variables and constraints of the model 1 can be found in [1].

One weakness of the approach in model 1 is that it uses a "big M" in (c.7) and (c.8). These constraints allow us to link d_{jk} to α_{jk} by means of the binary variable b_{jk}. The value of M can be calculated if and only if all the facets that define the DEA technology are previously determined. Unfortunately, the identification of all these facets is a combinatorial NP-hard problem. This weakness will be overcome in the new approach introduced here, since the new methodology does not need to resort to a big M to obtain the desired result.

3 Parallel Algorithm

In order to improve the performance of the algorithm developed in [12], it has been parallelized at different levels. Some of the objectives to parallelize the algorithm are related to the difficulty of solving problems such as the one proposed

in section 2, where a lot of computing time is taken to find satisfactory solutions. So, the proposed parallelization models are intended to reduce computation time to solve these models and improve the fitness of the solutions obtained. For this, both shared memory (OpenMP) and message-passing (MPI) schemes have been used, since it is possible to use them either separately or in combination. The metaheuristic parallelised in this scheme is based on generating initial solutions to the given model, and improving them with the intention of finding satisfactory solutions. For this, the following scheme is followed: Initialization, Improvement, Selection, Crossing and Diversification. Each parallelization models tries to improve the solution quality or computational time:

- OpenMP: Shared memory parallelization functions have been included within the algorithm's own functions, making them faster. Mainly, these improvements have been included in the initialization function, where it is possible to distribute the generation of solutions between the different threads, and in the improvement and crossing functions, where these tasks can be divided into smaller functions. This model of parallelism has been introduced with the intention of improving the internal loops of the main metaheuristic algorithm. Therefore, in all parts of the algorithm where many models must be evaluated (initialization, improvement, crossing), whenever the evaluation of the models are independent of each other, this type of parallelism is introduced.
- MPI: Message-passing functions have been included over and above metaheuristics. These functions optimize the computing time and the fitness of the final solutions. This level of parallelism is found in a higher level than the shared memoria scheme. In this way, the number of models to be evaluated are divided in equal parts in the different threads. Thus, the computation time is decreased as the number of cores increases.If it is not possible to divide all the DMUs into equal groups, the remaining units will be assigned randomly to the different cores.

The OpenMP scheme can also be used at the same level as the MPI scheme. A comparison between these two schemes is developed in the results of the experiment. This comparison is made to decide where implement each of the schemes. Figure 1 shows how the algorithm works and all the possible configurations.

In the literature, there are several mathematical methods that can solve both mixed integer linear programming (MILP) and linear programming (LP) problems. In the algorithm developed in this paper, a MILP-based decomposition is used to divide the main DEA problem, which is difficult to solve, into smaller LP-type problems that are easier to solve. In this regard, an exact method able of optimally solving numerous LP problems is needed. For this task, two exact methods that work optimally are evaluated to measure their performance in combination with our parallel algorithm. The software packages used are CPLEX and GUROBI.

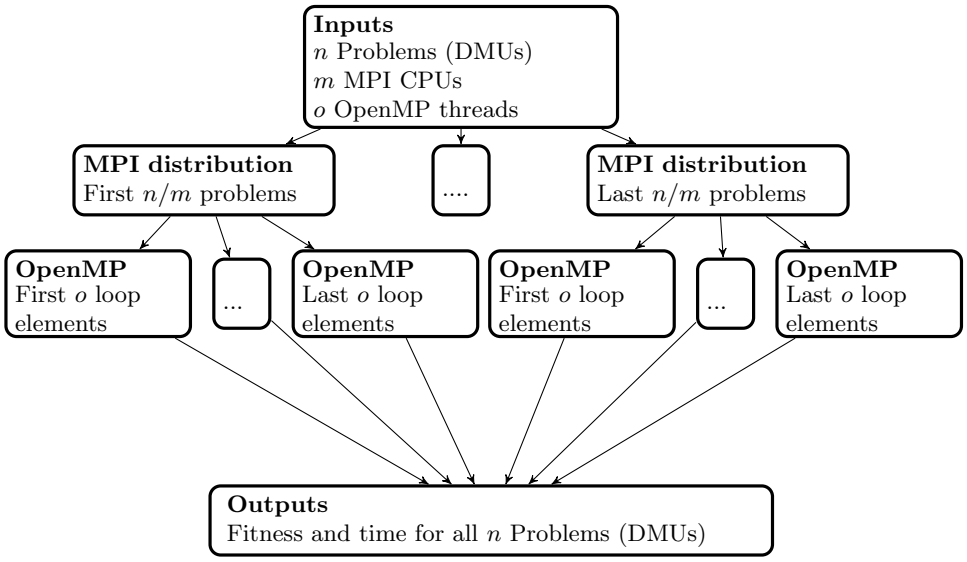

Fig. 1: Generation of metaheuristics and structure of the algorithm.

4 Experimental Results

Experiments were conducted to analyze the effectiveness of the parallel scheme developed. Two different exact methods are compared in terms of fitness and time, working in parallel. The performance of each exact method is also evaluated. For all the experiments, the IBM ILOG CPLEX Optimization Studio (CPLEX) and GUROBI are used. The system used in the experiments is a AMD Phenom II X6 1075T CPU (hexacore) at 3 GHz with 16 GBytes of RAM, private L1 and L2 caches of 64 KBytes and 512 KBytes respectively, and a L3 cache of 6 MBytes shared by all cores. For all the experiments, a standard problem has been taken with the following dimensions: 3 inputs, 2 outputs and 50 DMUs. Regarding the data, in our simulations the m inputs and s outputs of each of the n DMUs are generated at random but taking into account that the production function that governs the production situation is the well-known Cobb-Douglas function [13].

First, we analize the behavior of the different models of parallelism in the main algorithm will be studied. For this, the two available paradigms (MPI and OpenMP) will be used to perform the same function: divide the DMUs to be evaluated in equal groups, thus creating parallel executions of the algorithm, and therefore, of the metaheuristics and exact methods. In this evaluation, while comparing the parallel models, the different exact methods used in this paper are also compared: CPLEX and GUROBI. Therefore, there is going to be a comparison between how the MPI model works with CPLEX and GUROBI and how the OpenMP model also works with CPLEX and GUROBI.

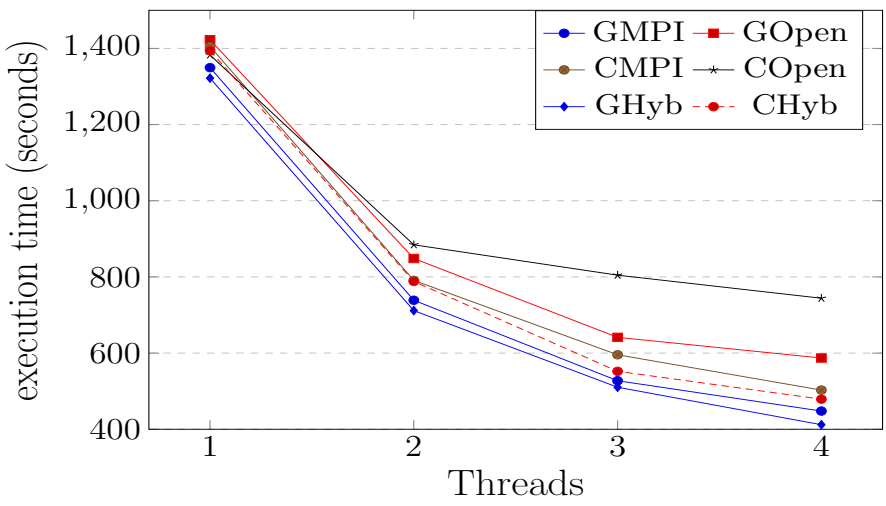

Fig. 2: Comparison of the execution time (in seconds) using different threads with the same problem, for each optimizer and different parallel algorithms: GUROBI with MPI (GMPI) and OpenMP (GOpen), and CPLEX with MPI (CMPI) and OpenMP (COpen).

Figure 2 shows how, as the number of CPUs increases, the computation time decreases. However, it can be seen how, when the number of CPUs grows, the improvement in the computational time decreases, becoming lower each time . This is due to the fact that, depending on the number of DMUs to be evaluated, from certain divisions, the resulting DMU groups have practically the same number of elements, therefore the computation time is smaller, but similar. In addition, this figure also shows that the exact methods chosen work well in parallel, being possible to execute multiple instances of them at the same time. It must be taken into account that the files in which the problems to be solved are written must have different names to avoid any confusion during the execution. This experiment shows that the MPI message-passing programming model works better than the OpenMP shared memory model in this first step of the algorithm. Therefore reaffirming what was stated in section 3.

Another important parameter to analyze, is the objective value achieved by each optimizer. For this, it is necessary to emphasize that the initial generation of solutions is done in a random way, so that the initial solutions generated by an optimizer are not the same as for the other, since the executions are independent of each other. To evaluate the results, 20 executions have been made for each optimizer, making 5 for each set of threads. The results shown in graph 3 show the average values of all the executions for each set of threads. Analyzing the obtained values, it can be affirmed that, after performing several experiments, the CPLEX optimizer obtains better fitness values in all the sets. As mentioned

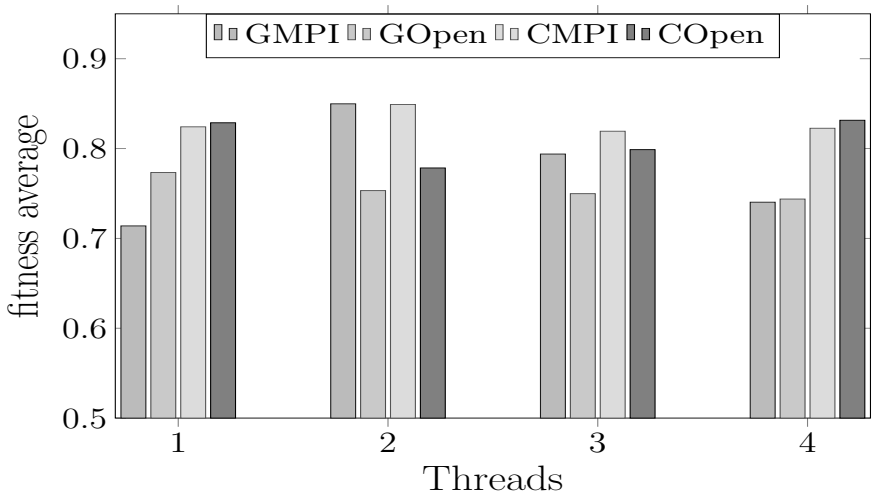

Fig. 3: Objective values obtained for each optimizer and different parallel algorithms: GUROBI with MPI (GMPI) and OpenMP (GOpen), and CPLEX with MPI (CMPI) and OpenMP (COpen). The values are the average of 5 executions.

before, the generation of initial solutions is randomly created and is independent for each optimizer. Even so, after a statistical study, CPLEX obtained better fitness values in this experiment.

5 Conclusions and Future Works

The application of the Principle of Least Action in DEA is a topic of relevance in recent DEA literature. However, it is well-known that from a computational point of view, this has usually been tackled with inadequate approaches, associated with combinatorial NP-hard problems.

Parallel algorithms are good solutions for solving these kind of problems. This is because a high number of independent problems must be solved, and those problems can be divided and solved by different threads. Furthermore, several optimized software packages can be used, but not all of them work well in parallel. To improve performance, is necessary to include optimizers that can be executed in different instances. For that, CPLEX and GUROBI are tools for these tasks. The parallel algorithm proposed in these paper works in an optimal way with both of these optimizers and with different parallel paradigms.

For future work, we propose the use of other free optimizers, and to check their performance compared with the most powerful optimizers in the market. In addition, it is also desirable to incorporate improvements in parallelism, so that the internal functions of the metaheuristic algorithm can be executed more quickly and accurately.

Acknowledgements

J. Aparicio and M. González thank the financial support from the Spanish 'Ministerio de Economía, Industria y Competitividad' (MINECO), the 'Agencia Estatal de Investigacion' and the 'Fondo Europeo de Desarrollo Regional' under grant MTM2016-79765-P (AEI/FEDER, UE).

References

1. J. Aparicio, J. L. Ruiz and I. Sirvent. Closest targets and minimum distance to the Pareto-efficient frontier in DEA. *Journal of Productivity Analysis*, 28:209-218, 2007.
2. W. Briec and J. B. Lesourd. Metric Distance Function and Profit: Some Duality Results. *Journal of Optimization Theory and Applications*, 101(1):15-33, 1999.
3. J. T. Pastor and J. Aparicio. The relevance of DEA benchmarking information and the Least-Distance Measure: Comment. *Mathematical and Computer Modelling*, 52:397-399, 2010.
4. J. Aparicio and J. T. Pastor. A well-defined efficiency measure for dealing with closest targets in DEA. *Applied Mathematics and Computation*, 219:9142-9154, 2013.
5. J. Aparicio and J. T. Pastor. Closest targets and strong monotonicity on the strongly efficient frontier in DEA. *Omega*, 44:51-57, 2014.
6. J. Aparicio and J. T. Pastor. On how to properly calculate the Euclidean distance-based measure in DEA. *Optimization*, 63(3):421-432, 2014.
7. J. Aparicio, B. Mahlberg, J. T. Pastor and B. K. Sahoo. Decomposing technical inefficiency using the principle of least action. *European Journal of Operational Research*, 239:776-785, 2014.
8. C. Benavente, J. J. López-Espín, J. Aparicio, J. T. Pastor and D. Giménez. Closest targets, benchmarking and data envelopment analysis: a heuristic algorithm to obtain valid solutions for the shortest projection problem. In *11th International Conference on Applied Computing*, 2014.
9. J. J. López-Espín, J. Aparicio, D. Giménez and J. T. Pastor. Benchmarking and data envelopment analysis. An approach based on metaheuristics. In *Proceedings of the International Conference on Computational Science, ICCS 2014, Cairns, Queensland, Australia, 10-12 June, 2014*, 390-399, 2014.
10. M. Gónzalez, J. J. López-Espín, J. Aparicio, D. Giménez and J. T. Pastor. Using Genetic Algorithms for Maximizing Technical Efficiency in Data Envelopment Analysis. In *Proceedings of the International Conference on Computational Science, ICCS 2015, Reykjavík, Iceland, 01-03 June, 2015*, 51:374-383, 2015.
11. E.-G. Talbi. Hybrid metaheuristics. Studies in computational intelligence. *Springer Verlag, Berlin, Germany*, vol. 434, 2013.
12. M.González, J.J. López Espín, J. Aparicio, D. Giménez and E. Talbi. A Parameterized Scheme of Metaheuristics with Exact Methods for determining the Principle of Least Action in Data Envelopment Analysis. *Program of the 2017 IEEE Congress on Evolutionary Computation.*
13. C.W. Cobb, P.H. Douglas A theory of production. *The American Economic Review*, 18(1):139-165, 1928.

Separatas

- *González, M., López-Espín, J. J. & Aparicio (2020). A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers. Electronics, 9(9), 1541.*

electronics

MDPI

Article

A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers

Martín González *, Jose J. López-Espín and Juan Aparicio

Center of Operations Research, Univeristy Miguel Hernandez of Elche, 03202 Elche, Spain; jlopez@umh.es (J.J.L.-E.); j.aparicio@umh.es (J.A.)

* Correspondence: martin.gonzaleze@umh.es

Received: 5 August 2020; Accepted: 15 September 2020; Published: 21 September 2020

Abstract: Metaheuristic and exact methods are one of the most common tools to solve Mixed-Integer Optimization Problems (MIPs). Most of these problems are NP-hard problems, being intractable to obtain optimal solutions in a reasonable time when the size of the problem is huge. In this paper, a hybrid parallel optimization algorithm for matheuristics is studied. In this algorithm, exact and metaheuristic methods work together to solve a Mixed Integer Linear Programming (MILP) problem which is divided into two different subproblems, one of which is linear (and easier to solve by exact methods) and the other discrete (and is solved using metaheuristic methods). Even so, solving this problem has a high computational cost. The algorithm proposed follows an efficient decomposition which is based on the nature of the decision variables (continuous versus discrete). Because of the high cost of the algorithm, as this kind of problem belongs to NP-hard problems, parallelism techniques have been incorporated at different levels to reduce the computing cost. The matheuristic has been optimized both at the level of the problem division and internally. This configuration offers the opportunity to improve the computational time and the fitness function. The paper also focuses on the performance of different optimization software packages working in parallel. In particular, a comparison of two well-known optimization software packages (CPLEX and GUROBI) is performed when they work executing several simultaneous instances, solving various problems at the same time. Thus, this paper proposes and studies a two-level parallel algorithm based on message-passing (MPI) and shared memory (Open MP) schemes where the two subproblems are considered and where the linear problem is solved by using and studying optimization software packages (CPLEX and GUROBI). Experiments have also been carried out to ascertain the performance of the application using different programming paradigms (shared memory and distributed memory).

Keywords: parallel algorithm; exact methods; Mixed Integer Problems; MILP decomposition; matheuristics

1. Introduction

Mixed Integer Linear Programming (MILP) models deal with mathematical optimization linear problems involving two families of variables: discrete and continuous. The computation related with this problem is high when the number of variables increases.

Mixed Integer Linear Programming (MILP) problems are well-known in the literature of mathematical programming and they have been very useful for modelling classical problems in operations research (knapsack, inventory, production, location, allocation, scheduling, etc.). These classical problems, at the same time, are currently being used for making decisions in a wide variety of contexts (military, banking, business, etc.). Their great utility for modelling very popular problems is one of the reasons why many researchers have been interested in proposing different algorithms for solving MILP problems. Indeed, some of these problems, such as the famous Traveling

Salesman Problem (TSP), are combinatorial optimization problems classified as NP-hard and are therefore, difficult to solve when the size of the problem is large (see [1]).

In practice, two types of algorithms are used to solve MILP problems: exact methods and metaheuristic algorithms. On the one hand, the exact methods (for example, branch and bound, simplex, etc.) allow to determine the optimal solution of the problem being studied. In fact, these algorithms ensure optimality. However, they are not generally applicable for large problems and are known to be time-consuming for big or more complex databases. On the other hand, metaheuristics do not guarantee the optimality of the found solution but can be implemented when the instance to be solved becomes too large or difficult for exact methods. Two categories of metaheuristics are usually considered: single-solution algorithms (local search, tabu search, etc.) and population-based algorithms (evolutionary algorithms, swarm optimization, etc.) [2]. Nevertheless, a recent approach, called matheuristics that combines the two philosophies (Pradenas et al. [3], Li et al. [4]), i.e., exact methods and metaheuristics, has been proposed in the literature in an attempt to provide a more efficient solution method (see, for example [5,6]).

Matheuristic methodology has been introduced to find approximate solutions in a reasonable time for MILP problems (see [7]). In this paper, a parallel algorithm for a new matheuristic algorithm is proposed and studied taking into account an MILP-based decomposition [8], where the main problem is decomposed into two hierarchical subproblems where different families of optimization algorithms are used. This decomposition is based on the nature of the decision variables: continuous and discrete.

Overall, the matheuristic methods have been designed by investigating differing cooperation between metaheuristics and exact methods, to find the best combination to solve an MILP problem. A general classification of existing approaches combining exact methods and metaheuristics for MILP optimization is presented in [9].

The matheuristic algorithm studied in this paper uses an integrative combination, in which the metaheuristic provides information for the exact method, which solves the problem by providing new information to the metaheuristic. The main idea is to reduce the problem into much smaller subproblems which can be solved exactly by state-of-the-art mathematical programming algorithms. The variables and the constraints are partitioned into two sets, decomposing the main problem into two hierarchical subproblems: The metaheuristic fixes the decision variables in one set and the exact method optimizes the problem over the other set.

Some popular techniques found in the literature to solve these problems using decomposition approaches that exploit the problem structure [10] are studied, such as constraint decomposition approximation (cutting plane methods) [11] or inner approximation (Dantzig-Wolfe method) [12,13] and variable decomposition methods ([14,15]). Variable decomposition methods are considered in this paper.

There are different methodologies to design metaheuristics. Some design parameters determine the characteristics of each metaheuristic. Those parameters are framed into different search components: initialization, improvement, selection, combination and stopping criteria. Several methodologies [16] have been generalized to matheuristics, in which exact optimization is combined with a set of metaheuristics. Nevertheless, this paper does not focus in obtaining the best metaheuristic algorithm for the problem, but assumes that the best metaheuristic for the problem is known and focuses on the parallelization of the mathehuristic algorithm.

The parallel algorithms studied in this paper are based on message-passing and shared-memory paradigms since they are the most extended scheme of parallel algorithms in the literature. In the experiments, message-passing (MPI) [17] and OpenMP [18] are the two APIs used to developed the algorithms as they are extended library routines over C and C++.

Nowadays, usually HPC systems integrate both shared and distributed memory architectures. For such hybrid architectures, one can perform hybrid parallel programming by combining different parallel programming models that are used at different architectural levels within a single code. This can offer the program a greater degree of parallelism as well as better performance.

The NP-hard problem proposed requires the evaluation of a large space of solutions, which requires too much time and computation. Therefore, one way to reduce the cost is to divide this space into different sections, and explore them in parallel. Furthermore, the problem proposed requires evaluating numerous MILP problems to obtain a final solution, which, being independent of each other, can be evaluated in parallel. Thus, we have achieved two ways of optimizing computation time using two levels of parallelism: the problems to be evaluated, and the search space for solutions by problem.

In this paper, the main contribution is related with the throughput of a parallel algorithm solving a combination of MILP problems, using several optimization packages in the literature (CPLEX [19] and GUROBI [20]). We evaluate two parallel paradigms (MPI and OpenMP) to obtain the best configuration of resources that combines both paradigms by resorting to the use of two optimization software packages (CPLEX and GUROBI), obtaining the lowest time possible to solve a combination of MILP problems.

The organization of the paper is as follows. In Section 2.1, the proposed decomposition of MILP problems is presented. Section 2.2 details the matheuristic strategy combining linear continuous programming and discrete metaheuristics. In Section 3, we will focus on the parallel algorithm. Finally, Section 4 gives some computational experiments.

2. Related Work

2.1. MILP-Based Decomposition

Let us consider the following linear problem (LP):

$$\max \{cx : Ax \leq b, x \geq 0, x \in \mathbb{R}^n\} \tag{1}$$

where A is a $m \times n$ matrix, c a n-dimensional row vector, b a m-dimensional column vector, and x a n-dimensional column vector of continuous variables. If we add the restriction that certain variables must take integer values, a Mixed Integer Linear Program (MILP) appears, which could be described as follows:

$$\begin{aligned} &\max cx + hy \\ &s.t. \\ &Ax + Gy \leq b \\ &x \geq 0, x \in \mathbb{R}^n \\ &y \geq 0, y \in \mathbb{Z}^p \end{aligned} \tag{2}$$

where A is a $m \times n$ matrix, G is $m \times p$ matrix, h is a p row-vector, and y is a p column-vector of integer variables.

An MILP problem is defined as a problem where discrete variables (y), which are restricted to integer values, and continuous variables (x), which can assume any value on a given continuous interval, are combined with integrality constraints.

Solving large-scale and complex instances using MILP techniques is not efficient in terms of search time. Indeed, large MILP problems are often difficult to solve by exact methods, due to the complexity of solving an optimization problem which includes integer and continuous variables. It is possible to solve large MILP problems by dividing them into smaller subproblems, and then solve them individually.

Figure 1 shows how a general MILP problem is decomposed into two hierarchical subproblems with different complexities:

- The subproblem (P1), which includes the discrete variables, is a computationally complex problem when it is large and an exact method is used for solving it. In this case, metaheuristic approaches could be more efficient than exact methods.

- The subproblem (P2), which includes the continuous variables, is a linear continuous problem that is easy to solve with exact methods.

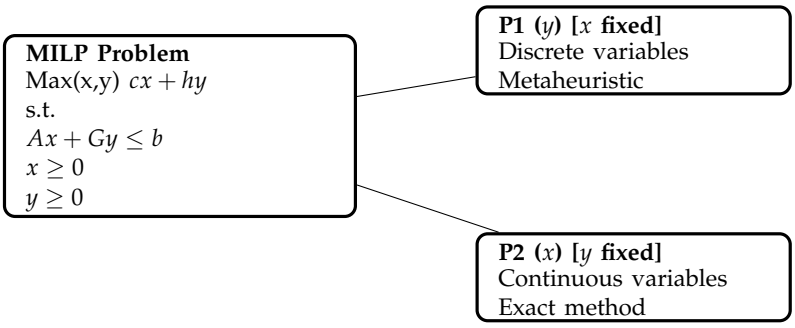

Figure 1. MILP problems decomposed into two subproblems.

2.2. Matheuristic Methodology

A matheuristic algorithm is defined as an algorithm which is made up of the combination of metaheuristics and exact methods. Thus, it is an optimization algorithm produced by the combination of metaheuristics and mathematical programming techniques. It can be the optimization of any mathematical problem which can be divided into different subproblems, where the main criteria are the nature of its variables.

In this work, a matheuristic algorithm is proposed to solve the problem shown in Figure 1, where the model is divided into two different subproblems, one of which is linear and the other discrete. The linear problem is easier to solve by exact methods, and the other subproblem can be studied using metaheuristic methods.

Whether or not the functions incorporated in the metaheuristic are executed (namely, initial population, improvement, selection, crossing and diversification), will depend on the selected metaheuristic. For example, an evolutionary algorithm (EA) does not use the improvement step, or a Greedy Randomized Adaptative Search Procedure (GRASP) does not use the crossover function. In this work, a parameterized scheme [21] that was previously studied by the authors in [16], is used to set the best metaheuristics.

Following the generation of the initial population by a metaheuristic, an exact method is involved to solve the subproblems generated. In this method, relaxation or decomposition techniques of the mathematical model are used. Relaxation methods consist of relaxing a strict requirement in the target optimization problem. This approach consists of ignoring the integrity constraints of an integer program and solving it using LP solvers. For that, the metaheuristic generates the decision variables and shares this information with the exact method.

Handling the constraints in the proposed decomposition methodology is a critical issue. The infeasible solutions generated by the exact method are evaluated and classified by assigning them a value based on certain parameters of the exact method. This parameter is related to the amount of restrictions that these solutions do not meet, and is modeled with a numeric value. This fitness penalty-based value is assigned to infeasible solutions. When this value is close to 0 it means that the solution is close to be a feasible solution. This implies that it requires fewer changes than other infeasible solutions to reach the feasible search.

A certain number of elements from both groups (feasible and infeasible solutions) are selected from the initial population and are used to generate new solutions through crossover and diversification functions. All these new generated solutions are also evaluated and improved in order to maximize the number of feasible solutions. When the algorithm uses the exact method, only the discrete variables in P1 can be used in the crossover function. The new variables generated are used to obtain a new solution solving P2 by the exact method. Those steps of the algorithm are repeated in a given number of iterations. Algorithm 1 shows the scheme of the main matheuristic algorithm. This algorithm

defines the parts involving the metaheuristic and the exact method. Every time that a model must be solved, the exact method is included, but if the model does not need to be solved, and only one solution is going to be evaluated, the metaheuristic is able to work alone.

Algorithm 1: Parallel matheuristic algorithm

input : $Problems(x, y)$
output: Best solution in each of the problems

Fix the metaheuristic parameters;
for $k = 1$ **to** $TotalProblems$ **do in parallel**
  //Create S_k set of solutions for problem k-th;
  for $j = 1$ **to** $Population_k$ **do in parallel**
    Fix discrete variables vd_j of problem P1;
    Obtain continuous variables vc_j solving P2 through exact method;
    $S_k \leftarrow [Solution_j := (vd_j, vc_j)]$;
    if $Solution_j$ *is not feasible* **then**
      Improve $Solution_j$ using the best neighbourhood algorithm;
    end
  end
  not *EndCondition //Select SS_k subset of S_k such as $|SS_k| > 1$*;
  for $w = 1$ **to** $Combination_k$ **do in parallel**
    Select $s_1, s_2 \in S_k$ randomly ;
    Combine vd_1 and vd_2 the discrete variables of s_1 and s_2 saving as the discrete variables of a new solution s_w;
    Obtain vc_w continuous variables of s_w solving P2;
    if $Fitness(w) > Fitness(r_1)$ & $Fitness(w) > Fitness(r_2)$ **then**
      $SS_k \leftarrow s_w$;
    end
  end
  //Improve SS_k subset of S_k;
  for $w = 1$ **to** $Improve_k$ **do in parallel**
    Select $s_w \in SS_k$ randomly ;
    REPEAT:;
    Modify vd_w using the best neighbourhood algorithm and obtain vc_w solving P2 through exact method;
    UNTIL $Fitness(s_w)$ *increase achieve EndConditions*;
  end
  //Diversify SS_k subset of S_k;
  for $w = 1$ **to** $Diversification_k$ **do in parallel**
    Select $s_w \in SS_k$ randomly ;
    Modify randomly vd_w of s_w;
    Obtain vc_w solving P2 through exact method;
  end
  Include SS_k in S_k;
  BestSolution$_k \leftarrow s \in S_k$ such as $Fitness(s) \geq Fitness(w) \forall w \in S_k$;
end

3. Parallel Algorithm

The parallel algorithm proposed is schemed in two levels. The first one considers the solutions of different independent MILP problems at the same time and the second one, the execution of each

solution of the mathehuristic algorithm. Thus, this scheme has the advantage of distributing the processors depending on the number of independent problems which must be solved and the number of solutions proposed for the matheuristic algorithm. Figure 2 shows the proposed scheme.

A set of solutions is created in each problem. Each solution contains discrete and continuous variables solving the two problems described in Figure 1. The discrete variables are set and the continuous variables are obtained using the discrete variables and lineal programming. When the initial set of solutions S_k is created, a number of valid and invalid solutions, selected randomly from the reference set, are improved. A valid solution is considered when the continuous variables are obtained and the problem is feasible. Other solutions (infeasible ones) are considered invalid. The algorithm works with valid and invalid solutions improving the fitness value from the first one, and converts invalid solutions into valid ones.

The improvement function is developed following the variable neighbourhood search. It explores distant neighbourhood solutions of the currently selected solution. If an improvement is made, the old solution is replaced by this new one. The local search method is applied repeatedly to improve the selected solutions in the neighbourhood, obtaining local optima. At this point, a certain number of the best valid and invalid solutions are selected from the reference set S_k, creating a new subset SS_k. A combination function is applied to these selected solutions. It combines pairs of solutions randomly chosen from those previously selected. The combination is performed mixing the discrete variables using a binary mask, and then, the continuous variables are obtained through linear programming. Using this combination, new solutions are generated, but only the valid solutions are included in the reference set SSk.

In the implementation of the parallel algorithm, both shared memory (OpenMP) and message-passing (MPI) schemes have been proposed according to the following ideas: MPI is used in the first level of parallelization where the different independent MILP problems are considered. Therefore, the total number of MILP problems are distributed between the processor and the resources assigned to MPI. On the other hand, OpenMP is used in the internal algorithms. Then, the different parts of the matheuristic are parallelized using the processors and the resources assigned to OpenMP.

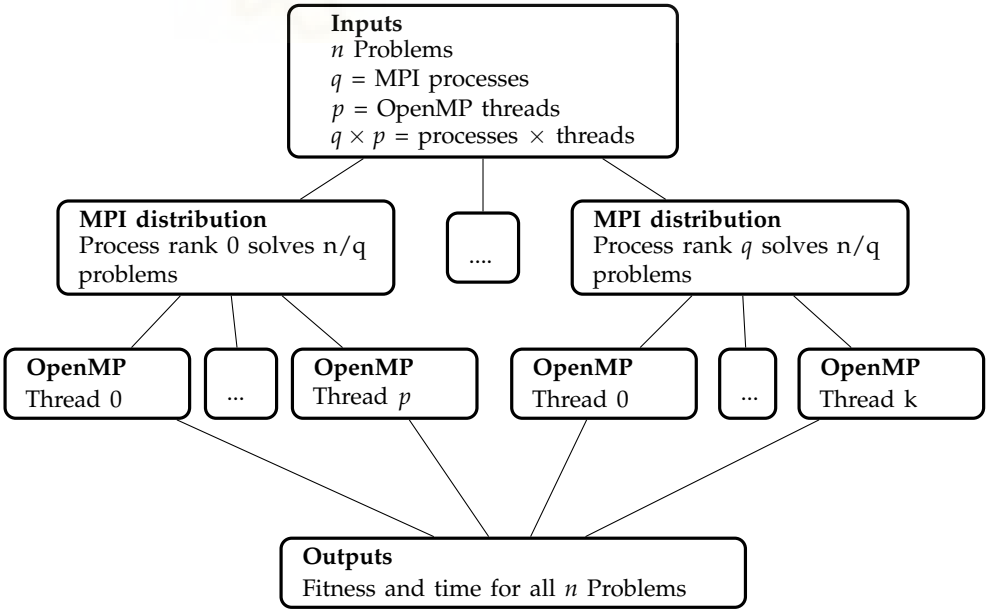

Figure 2. The parallel algorithm scheme based on resource decomposition in two levels.

In the literature, there are several mathematical methods that can solve both mixed integer linear programming and linear programming problems. In the algorithm developed in this paper, an MILP-based decomposition is used to divide the main problem, which is difficult to solve, into smaller LP-type problems that are easier to solve. In this regard, an exact method able of

optimally solving numerous LP problems is needed. For this task, two well-known optimization software packages, CPLEX and GUROBI, are evaluated to measure their performance in combination with our parallel algorithm.

4. Experimental Results

In this section, a computational experiment is carried out by applying the Algorithm 1 on an MILP problem associated with a modern Data Envelopment Analysis (DEA) technique. This is a non-parametric technique whose objective is to determine the technical efficiency of a set of n firms (in general, a set of Decision Making Units—DMUs), which use m inputs to produce s outputs (see [22])

In particular, we focus our attention on the MILP problem proposed in [23] in the context of DEA. A formalization of the problem is described as follows. Let us assume that data on m inputs and s outputs for n homogeneous DMUs are observed. For the j-th DMU, these are represented by $z_{ij} \geq 0, i = 1, \ldots, m$ and $q_{rj} \geq 0, r = 1, \ldots, s$. The DEA model that should be solved for evaluating the performance of DMU k is as follows:

$$\max \left\{ \beta_k - \frac{1}{m} \sum_{i=1}^{m} \frac{t_{ik}^-}{z_{ik}} \right\}$$

s.t.

$$\begin{array}{rll}
\beta_k + \frac{1}{s} \sum_{r=1}^{s} \frac{t_{rk}^+}{q_{rk}} \leq 1 & & (c.1) \\
-\beta_k - \frac{1}{s} \sum_{r=1}^{s} \frac{t_{rk}^+}{q_{rk}} \leq -1 & & (c.2) \\
-\beta_k z_{ik} + \sum_{j=1}^{n} \alpha_{jk} x_{ij} + t_{ik}^- \leq 0 & \forall i & (c.3) \\
\beta_k z_{ik} - \sum_{j=1}^{n} \alpha_{jk} x_{ij} - t_{ik}^- \leq 0 & \forall i & (c.4) \\
-\beta_k q_{rk} + \sum_{j=1}^{n} \alpha_{jk} y_{rj} - t_{rk}^+ \leq 0 & \forall r & (c.5) \\
\beta_k q_{rk} - \sum_{j=1}^{n} \alpha_{jk} y_{rj} + t_{rk}^+ \leq 0 & \forall r & (c.6) \\
-\sum_{i=1}^{m} \nu_{ik} z_{ij} + \sum_{r=1}^{s} \mu_{rk} q_{rj} + d_{jk} \leq 0 & \forall j & (c.7) \\
\sum_{i=1}^{m} \nu_{ik} z_{ij} - \sum_{r=1}^{s} \mu_{rk} q_{rj} - d_{jk} \leq 0 & \forall j & (c.8) \\
-\nu_{ik} \leq -1 & \forall i & (c.5) \\
-\mu_{rk} \leq -1 & \forall r & (c.6) \\
-d_{jk} \leq -M b_{jk} & \forall j & (c.7) \\
\alpha_{jk} \leq M(1 - b_{jk}) & \forall j & (c.8) \\
b_{jk} = 0, 1 & \forall j & (c.9) \\
-\beta_k \leq 0 & & (c.10) \\
-t_{ik}^- \leq 0 & \forall i & (c.11) \\
-t_{rk}^+ \leq 0 & \forall r & (c.12) \\
-d_{jk} \leq 0 & \forall j & (c.13) \\
-\alpha_{jk} \leq 0 & \forall j & (c.14)
\end{array} \tag{3}$$

where M is a positive big number. For this particular MILP problem, the vector of continuous variables x consists of (β_k, t_{ik}^-, t_{rk}^+, d_{jk} and α_{jk}), while the vector of integer variables consists exclusively of b_{jk}.

In DEA, each DMU has an MILP problem to be solved. Regarding the data, in our simulations, the m inputs and s outputs of each of the n DMUs are generated randomly but take into account that the production function that governs the production situation is the Cobb–Douglas function [24], which is well-known in microeconomics.

For all the experiments, the IBM ILOG CPLEX Optimization Studio (CPLEX) and the GUROBI Optimizer v.8.1 are used with a Free Academic License. The experiments are executed in a DELL PowerEdge R730 node with 2 Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30 GHz (Santa Clara, USA), with 20 cores (40 threads) at 2.4 GHz and 25 MB SmartCache memory. A comparative analysis of both optimization solvers in terms of capabilities showing different features of their architectures is shown in [25].

Experiments were performed to analyze the performance of the proposed algorithm using different parallelism tools. In addition, the performance of the different optimization packages when they are executed in parallel is analyzed. A fixed configuration of parameters for each metaheuristic function (population, combination, improvement, diversification and EndCondition in Algorithm 1) is used for conducting all the experiments. Parameters have been established based on previous analysis and experiences of the authors. Those parameters are: population = 310, combination = 49, improvement = 17, diversification = 10, EndCondition = 10 iterations (or five without improving).

In this paper, several DEA problems (model (3)) have been generated randomly with different sizes, so that the analyses are not dependent on the size of the problem. The problem sizes generated are the following:

- Size 1: $m = 3/n = 50/s = 1$
- Size 2: $m = 4/n = 50/s = 2$
- Size 3: $m = 5/n = 250/s = 3$

In the first experiment, the performance of the different optimization packages are analyzed. At this point, these optimization packages are evaluated solving several problems simultaneously; conflicts and problems of the optimization packages when they are executed simultaneously are also studied, using multiple instances at the same time through several threads.

Figure 3 shows the time that each optimization package takes for solving a single problem (in milliseconds) when, at the same time, there are different instances doing the same job when varying the number of processors. The time is always measured in processor number 0, for the execution of the same problem. At this point, just a single problem is executed (DMU 1 of n). It can be seen that, as more executions are performed simultaneously, the more the computing time increases. This is because the number of instances that can be simultaneously executed with these optimization packages are not unlimited. After a certain number of executions, the software starts to suffer delays. Analyzing the graph, it can be seen that from 10 simultaneous instances until 40, the computation time increases. This means that the improvement in computing time, when the number of resources increases, starts to decrease when using more than a number of processors which depends on the problem size.

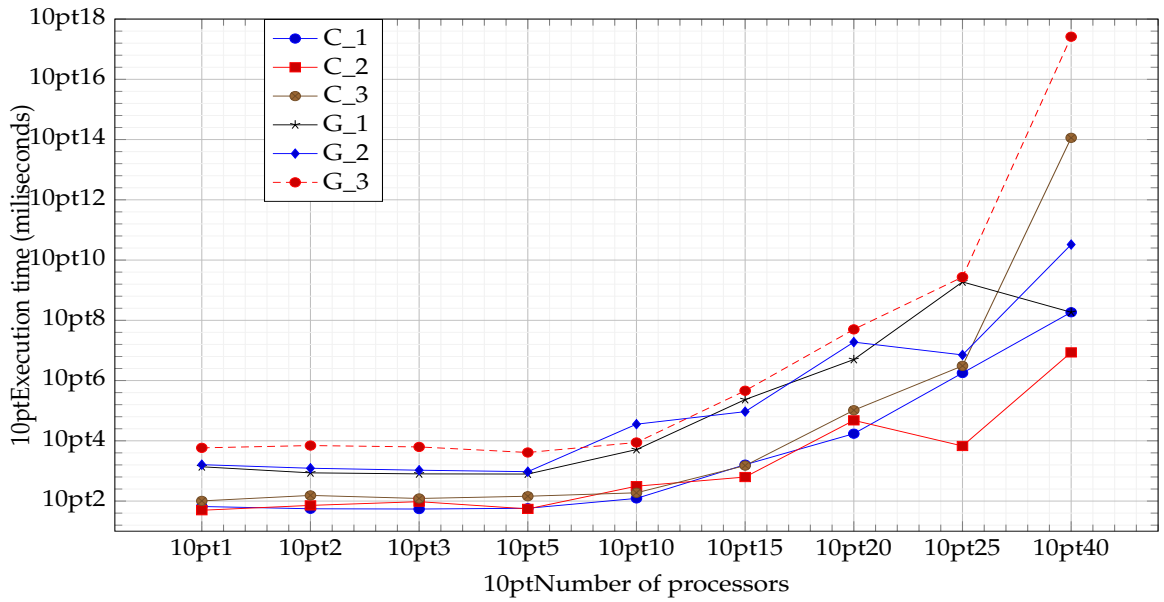

Figure 3. Computational time solving a unique LP problem in multiple processors using CPLEX and GUROBI solvers. The evaluated problems are: Size 1 with CPLEX (C_1), Size 2 with CPLEX (C_2), Size 2 with CPLEX (C_3), Size 1 with GUROBI (G_1), Size 2 with GUROBI (G_2) and Size 3 with GUROBI (G_3).

Another objective of the experiments is to compare how the proposed exact methods behave (CPLEX and GUROBI) according to the parallelization strategy. At this point, only the problem with size 1 (m = 3, n = 50 and s = 1) is analyzed in the following experiments. The results obtained are similar to the other sizes. Figure 4 shows the execution cost obtained when using the optimization packages and solving the problem shown in expression (3) with different paradigms of parallelism, solving all the DMUs included in the main problem. In this experiment, the parallelization levels (see Figure 2) have been executed separately. In the first instance, all the processors have been allocated to the highest level of parallelization (MPI), where the different problems to be solved among the available cores are divided. In the second instance, the processors have been allocated to the second level of parallelization (OpenMP). A previous work [26] shows that using OpenMP in the high level of parallelization is worse than using MPI and thus, this experiment is not considered in this paper. As a final result, a complete comparison of the levels of parallelism is presented, with the different proposed optimization packages in each level. It can be observed how, in terms of performance, a greater improvement in computing time is obtained using the second level of parallelism (OpenMP), and, in addition, when using CPLEX, the computational time is lower than when using GUROBI. However, as already mentioned, a convergence can be seen when more than 10 cores are used. This is because, despite the improvement due to the division of problems and the optimization of internal functionalities, by increasing the number of simultaneous instances, the optimization packages interfere with each other.

Once the effectiveness of the parallelism has been studied, the question is raised as to how to distribute the resources between the different parallel levels. For this, a third experiment has been developed to find the best configuration of available resources by dividing them between both parallel levels. The aim is find the optimal values for the variables m and k in Figure 2).

In this experiment, different possible configurations have been tested, obtaining, in each case, the time taken by the algorithm to find the optimal solution. Therefore, after some tests, we get the configuration that improves the best time shown in Figure 4. The time has been compared with that obtained using all the resources in both OpenMP and MPI, where all the available cores are focused on only one level of parallelism. These times are:

- OpenMP[CPLEX] = 43.1825 s
- OpenMP[GUROBI] = 74.3996 s
- MPI[CPLEX] = 119.4421 s
- MPI[GUROBI] = 159.6177 s

Table 1 shows the average of the time of 10 executions for each resources configuration. It is shown that computation time is minimized when resources are divided between different levels, compared to that obtained when all resources are provided at only one level. This proves that dividing the parallelism into different levels, and establishing the resources in an optimal way, the performance of the application improves. The experiments with lower times than those obtained by using MPI or OpenMP separately (shown above) have been highlighted in bold. In this way, it can be seen that in most of the cases, the cost using a mix parallel scheme is better than the cost of using just one of them. It can be observed that the configurations that allocate more resources to the shared memory level (OpenMP), such as combination 3-13 obtain better results than those that allocate more resources to the division of problems in distributed memory, such as configuration 13-3.

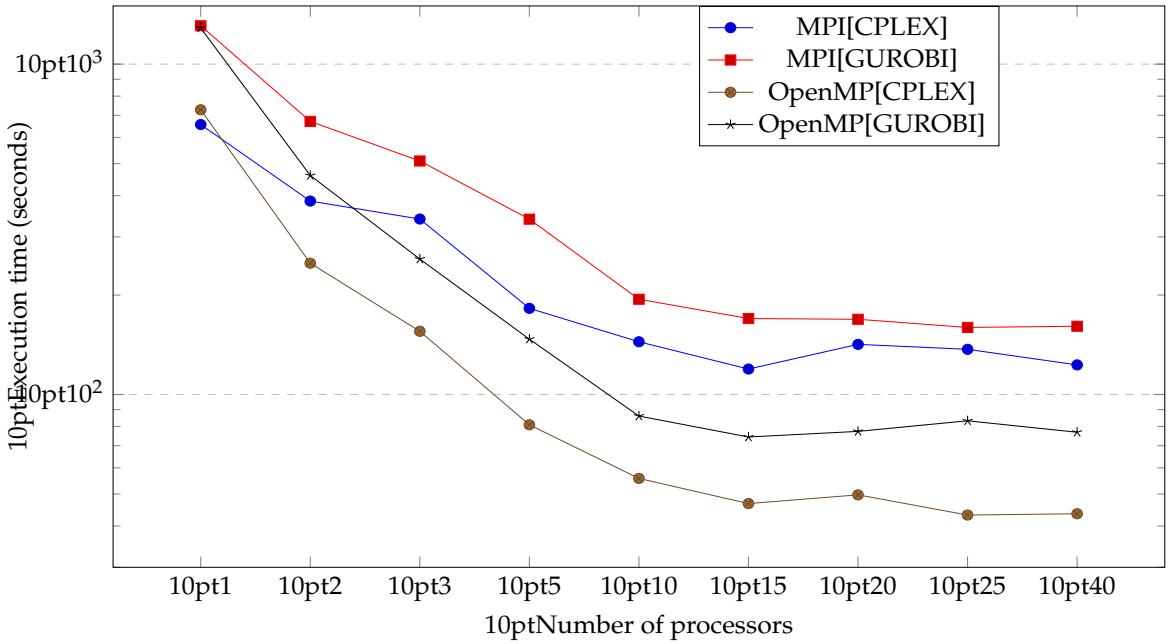

Figure 4. Performance of each level of parallelization. Comparison using all the resources with message-passing (MPI) or OpenMP with several optimization packages (CPLEX and GUROBI).

Table 1. Comparison between several configurations with MPI and OpenMP in a hybrid mode. Time for each solver is expressed in seconds. The number of processors used in each parallel level is also shown.

Hybrid Parallel Configuration							
GUROBI	**CPLEX**	**MPI**	**OpenMP**	**GUROBI**	**CPLEX**	**MPI**	**OpenMP**
$223.7383_{16.4439}$	$137.3334_{3.3673}$	2	2	$104.2315_{8.9342}$	$66.4368_{5.7361}$	3	3
$81.3666_{5.8771}$	$59.2839_{1.9190}$	4	3	$\mathbf{59.6567}_{2.1628}$	$45.1889_{2.8821}$	4	5
$\mathbf{68.5390}_{3.6255}$	$51.4602_{3.0931}$	5	4	$\mathbf{56.6961}_{2.7679}$	$\mathbf{40.6429}_{1.5087}$	3	8
$\mathbf{53.2853}_{2.2711}$	$\mathbf{42.5253}_{1.9582}$	5	8	$\mathbf{70.2143}_{6.3089}$	$54.2284_{2.3153}$	6	4
$\mathbf{52.8286}_{2.9948}$	$\mathbf{40.0809}_{2.4266}$	2	20	$\mathbf{71.9496}_{4.2597}$	$75.3947_{4.6624}$	20	2
$\mathbf{49.8065}_{1.9282}$	$\mathbf{37.3163}_{2.4220}$	3	13	$75.7056_{4.4348}$	$62.6644_{2.2626}$	13	3
$\mathbf{50.8675}_{2.3173}$	$\mathbf{39.1378}_{1.8670}$	4	10	$\mathbf{70.8639}_{4.0900}$	$56.7864_{2.6158}$	10	4

5. Conclusions and Future Works

In this paper, a parallel matheuristic algorithm for solving a set of Mixed Integer Linear Programming (MILP) problems is presented. The algorithm put forward follows a decomposition strategy proposed for large-scale MILP optimization problems. This decomposition is based on the nature of the decision variables (continuous versus discrete). In the proposed algorithm, an incomplete encoding representing only discrete decision variables is explored by metaheuristics. The encoding of solutions is completed for the continuous decision variables by solving a linear problem.

In the implementation of the parallel algorithm, both shared memory (OpenMP) and message-passing (MPI) schemes have been proposed in combination, in accordance with the following ideas: MPI is used in the first level of parallelization where the different independent MILP problems are considered and OpenMP is used for internal algorithms.

Experiment results are shown comparing cost and efficiency when processors and problem size vary, obtaining satisfactory results in terms of solution quality and execution time. The parallel study also focuses on the comparison of the CPLEX and GUROBI software packages since both are two of the most used packages when solving LP problems.

Finally, future research efforts could be aimed at determining the best metaheuristic, instead of assuming that the best parameters have already been found.

Author Contributions: M.G. and J.J.L.-E. designed the parallel model and the computational framework and analysed the data. J.A. designed the mathematical model. M.G. carried out the implementation and performed the calculations. M.G., J.J.L.-E. and J.A. wrote the manuscript with input from all authors. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: The authors are grateful for the financial support from the Santander Chair of Efficiency and Productivity of the University Miguel Hernandez of Elche (Spain).

Conflicts of Interest: The authors declare no confict of interest.

References

1. Winston, W.L.; Goldberg, J.B. *Operations Research: Applications and Algorithms*; Thomson/Brooks/Cole: Belmont, CA, USA, 2004; Volume 3.
2. Talbi, E.G. *Metaheuristics: From Design to Implementation*; John Wiley & Sons: Hoboken, NJ, USA, 2009; Volume 74.
3. Pradenas, L.; Garcés, J.; Parada, V.; Ferland, J. Genotype–phenotype heuristic approaches for a cutting stock problem with circular patterns. *Eng. Appl. Artif. Intell.* **2013**, *26*, 2349–2355. [CrossRef]
4. Li, X.; Yalaoui, F.; Amodeo, L.; Chehade, H. Metaheuristics and exact methods to solve a multiobjective parallel machines scheduling problem. *J. Intell. Manuf.* **2012**, *23*, 1179–1194. [CrossRef]
5. Talbi, E.G. Combining metaheuristics with mathematical programming, constraint programming and machine learning. *Ann. Oper. Res.* **2016**, *240*, 171–215. [CrossRef]
6. Jourdan, L.; Basseur, M.; Talbi, E.G. Hybridizing exact methods and metaheuristics: A taxonomy. *Eur. J. Oper. Res.* **2009**, *199*, 620–629. [CrossRef]
7. Archetti, C.; Speranza, M.G. A survey on matheuristics for routing problems. *EURO J. Comput. Optim.* **2014**, *2*, 223–246. [CrossRef]
8. Ralphs, T.K.; Galati, M.V. Decomposition methods for integer programming. *Wiley Encycl. Oper. Res. Manag. Sci.* **2010**. [CrossRef]
9. Puchinger, J.; Raidl, G.R. Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. In Proceedings of the International Work-Conference on the Interplay between Natural and Artificial Computation, Canary Islands, Spain, 15–18 June 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 41–53.
10. Vanderbeck, F.; Wolsey, L.A. Reformulation and decomposition of integer programs. In *50 Years of Integer Programming 1958–2008*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 431–502.
11. Kelley, J.E., Jr. The cutting-plane method for solving convex programs. *J. Soc. Ind. Appl. Math.* **1960**, *8*, 703–712. [CrossRef]
12. Vanderbeck, F. On Dantzig-Wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. *Oper. Res.* **2000**, *48*, 111–128. [CrossRef]
13. Ruszczyński, A. An augmented Lagrangian decomposition method for block diagonal linear programming problems. *Oper. Res. Lett.* **1989**, *8*, 287–294. [CrossRef]
14. Benders, J.F. Partitioning procedures for solving mixed-variables programming problems. *Numer. Math.* **1962**, *4*, 238–252. [CrossRef]
15. Rahmaniani, R.; Crainic, T.G.; Gendreau, M.; Rei, W. The Benders decomposition algorithm: A literature review. *Eur. J. Oper. Res.* **2017**, *259*, 801–817. [CrossRef]
16. González, M.; López-Espín, J.J.; Aparicio, J.; Giménez, D.; Talbi, E.G. A parameterized scheme of metaheuristics with exact methods for determining the principle of least action in data envelopment analysis. In Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), San Sebastian, Spain, 5–8 June 2017; pp. 588–595.
17. William Gropp, E.L.; Skjellum, A. *Using MPI: Portable Parallel Programming with the Message-Passing Interface*; MIT Press: Cambridge, MA, USA, 1999; Volume 1.
18. Dagum, L.; Menon, R. Openmp: An industry standard api for shared-memory programming. *IEEE Comput. Sci. Eng.* **1998**, *5*, 46–55. [CrossRef]
19. Cplex, I.I. V12. 1: User's Manual for CPLEX. *Int. Bus. Mach. Corp.* **2009**, *46*, 157.

20. Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*; Gurobi Optimization, LLC: Houston, TX, USA, 2014.
21. Almeida, F.; Giménez, D.; López-Espín, J.J.; Pérez-Pérez, M. Parameterized schemes of metaheuristics: Basic ideas and applications with genetic algorithms, scatter search, and GRASP. *IEEE Trans. Syst. Man Cybern. Syst.* **2013**, *43*, 570–586. [CrossRef]
22. Emrouznejad, A.; Yang, G.L. A survey and analysis of the first 40 years of scholarly literature in DEA: 1978–2016. *Socio-Econ. Plan. Sci.* **2018**, *61*, 4–8. [CrossRef]
23. Aparicio, J.; Ruiz, J.L.; Sirvent, I. Closest targets and minimum distance to the Pareto-efficient frontier in DEA. *J. Product. Anal.* **2007**, *28*, 209–218. [CrossRef]
24. Coma, C.W.; Douglas, P.H. A theory of production. In *Proceedings of the Fortieth Annual Meeting of the American Economic Association*; American Economic Association: Nashville, TN, USA, 1928; Volume 139, p. 165.
25. Anand, R.; Aggarwal, D.; Kumar, V. A comparative analysis of optimization solvers. *J. Stat. Manag. Syst.* **2017**, *20*, 623–635. [CrossRef]
26. González, M.; López-Espín, J.J.; Aparicio, J.; Giménez, D. A Parallel Application of Matheuristics in Data Envelopment Analysis. In Proceedings of the International Symposium on Distributed Computing and Artificial Intelligence, Toledo, Spain, 20–22 June 2018; Springer: Cham, Switzerland, 2018; pp. 172–179.

Bibliografía

Afriat, S. N. (1974). Production duality and the von Neumann theory of growth and interest. In Production Theory (pp. 97-99). Springer, Berlin, Heidelberg.

Almeida, F., Giménez, D., López-Espín, J. J., & Pérez-Pérez, M. (2013). Parameterized schemes of metaheuristics: Basic ideas and applications with genetic algorithms, scatter search, and GRASP. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 43(3), 570-586.

Aparicio, J., Ruiz, J. L., & Sirvent, I. (2007). Closest targets and minimum distance to the Pareto-efficient frontier in DEA. Journal of Productivity Analysis, 28(3), 209-218.

Aparicio, J. (2007). Una introducción al análisis envolvente de datos. Boletín de Estadística E Investigación Operativa. BEIO, 23(1), 12-17.

Apt, K. R. (2000). Principles of Constraint Programming.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., & Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. Operations research, 46(3), 316-329.

Banker, R.D., Charnes, A. y Cooper, W.W. (1984), "Some Models for Estimating Technical and Scale Inefficiencies in Data Envelopment Analysis", Management Science, 30: 1078-1092

Blesa, M. J., Blum, C., Gaspero, L., Di Gaspero, L., Roli, A., Sampels, M., & Schaerf, A. (Eds.). (2009). Hybrid Metaheuristics: 6th International Workshop, HM 2009 Udine, Italy, October 16-17, 2009 Proceedings (Vol. 5818). Springer Science & Business Media.

Boles, J.N. (1966), "Efficiency Squared: Efficient Computation of Efficiency Indexes", Proceedings of the Thirty Ninth Annual Meeting of the Western Farm Economics Association. Pag. 137-142.

Boschetti, M. A., Maniezzo, V., Roffilli, M., & Röhler, A. B. (2009, October). Matheuristics: Optimization, simulation and control. In International Workshop on Hybrid Metaheuristics (pp. 171-177). Springer, Berlin, Heidelberg.

Bressler, R.G. (1966), "The Measurement of Productive Efficiency", Proceedings of the Thirty Ninth Annual Meeting of the Western Farm Economics Association. Pag. 129-136.

Charnes, A., Cooper, W.W. y Rhodes, E. (1978), "Measuring the Efficiency of Decision Making Units", European Journal of Operational Research, 2: 429-444.

Charnes, A., Cooper, W.W., Golany, B., Seiford, L. y Stutz, J. (1985), "Foundations of Data Envelopment Analysis for Pareto-Koopmans

Efficient Empirical Production Functions", Journal of Econometrics, 30: 91-107.

Charnes, A. y Cooper, W.W. (1962), "Programming With Linear Fractional Functionals", Naval Research Logistics Quarterly, 9: 181-186.

Chen, Y. (2003), "A non-radial Malmquist productivity index with an illustrative application to Chinese major industries", International Journal of Production Economics, 83: 27-35.

Colorni, A., Dorigo, M., & Maniezzo, V. (1992). A genetic algorithm to solve the timetable problem. Politecnico di Milano, Milan, Italy TR, 90-060.

Cooper, W.W., Park, K.S. y Pastor, J.T. (1999), "RAM: A Range Adjusted Measure of Inefficiency for Use with Additive Models, and Relations to Others Models and Measures in DEA", Journal of Productivity Analysis, 11: 5-42.

Cplex, I. I. (2009). V12. 1: User's Manual for CPLEX. International Business Machines Corporation, 46(53), 157.

Dantzig, G. B. (1947). U.S. Patent No. 2,421,745. Washington, DC: U.S. Patent and Trademark Office.

Debreu, G. (1951), "The Coefficient of Resource Utilization", Econometrica, 19: 273-292.

Drosinos, N., & Koziris, N. (2004, April). Performance comparison of pure MPI vs hybrid MPI-OpenMP parallelization models on SMP clusters. In 18th International Parallel and Distributed Processing Symposium, 2004. Proceedings. (p. 15). IEEE.

Edmonds, J. (1971). Matroids and the greedy algorithm. Mathematical programming, 1(1), 127-136.

Eiben, A. E., Michalewicz, Z., Schoenauer, M., & Smith, J. E. (2007). Parameter control in evolutionary algorithms. In Parameter setting in evolutionary algorithms (pp. 19-46). Springer, Berlin, Heidelberg.

Glover, F. (1986). Future paths for integer programming and links to ar tifi cial intelli g en ce. Computers operations research, 13(5), 533-549.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. Decision sciences, 8(1), 156-166.

Goldberg, D. E. (1989). Genetic algorithms in search. Optimization, and MachineLearning.

Gomory, R. E. (2010). Outline of an algorithm for integer solutions to linear programs and an algorithm for the mixed integer problem. In 50 Years of Integer Programming 1958-2008 (pp. 77-103). Springer, Berlin, Heidelberg.

González, M., López-Espín, J. J., Aparicio, J., & Giménez, D. (2018, June). A Parallel Application of Matheuristics in Data Envelopment Analysis.

In International Symposium on Distributed Computing and Artificial Intelligence (pp. 172-179). Springer, Cham.

González, M., López-Espín, J. J. & Aparicio (2020) A Parallel Algorithm for Matheuristics: A Comparison of Optimization Solvers. Electronics (High-Performance Computer Architectures and Applications)

González, M., López-Espín, J. J., Aparicio, J., Giménez, D., & Talbi, E. G. (2017, June). A parameterized scheme of metaheuristics with exact methods for determining the principle of least action in Data Envelopment Analysis. In 2017 IEEE Congress on Evolutionary Computation (CEC) (pp. 588-595). IEEE.

Gurobi Optimization, I. (2014). Gurobi optimizer reference manual, version 6.0. URL: http://www. gurobi. com. Retrieved.

Hansen, P., & Mladenović, N. (1999). An introduction to variable neighborhood search. In Meta-heuristics (pp. 433-458). Springer, Boston, MA.

Holland, J. H. (1992). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press.

Holland, J. H. (1962). Outline for a logical theory of adaptive systems. Journal of the ACM (JACM), 9(3), 297-314.

Katouda, M., & Nakajima, T. (2013). MPI/OpenMP hybrid parallel algorithm of resolution of identity second-order Møller–Plesset perturbation calculation for massively parallel multicore supercomputers. Journal of chemical theory and computation, 9(12), 5373-5380.

Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In Proceedings of ICNN'95-International Conference on Neural Networks (Vol. 4, pp. 1942-1948). IEEE.

Koopmans, T.C. (1951), "Analysis of Production as an Efficient Combination of Activities", en T.C. Koopmans (Ed.), Activity Analysis of Production and Allocation, New York, John Wiley.

Lawler, E. L., & Wood, D. E. (1966). Branch-and-bound methods: A survey. Operations research, 14(4), 699-719.

Pastor, J.T., Ruiz, J.L. y Sirvent, I. (1999), "An Enhanced DEA Russell Graph Efficiency Measure", European Journal of Operational Research, 115: 596-607.

Pinheiro, R. G. S., Martins, I. C., Protti, F., & Ochi, L. S. (2018). A matheuristic for the cell formation problem. Optimization Letters, 12(2), 335-346.

Pólya, G., & Conway, J. H. (1957). How to solve it: A new aspect of mathematical method. Princeton, NJ: Princeton University Press.

Rechenberg, I. (1989). Evolution strategy: Nature's way of optimization. In Optimization: Methods and applications, possibilities and limitations (pp. 106-126). Springer, Berlin, Heidelberg.

Seitz, W.D. (1966), "Efficiency Measures for Steam-Electric Generating Plants", Proceedings of the Thirty Ninth Annual Meeting of the Western Farm Economics Association. Pag. 143-151.

Shephard, G. C. (1953). Unitary groups generated by reflections. Canadian Journal of Mathematics, 5, 364-383.

Singh, A., & Rossi, A. (2013). A genetic algorithm based exact approach for lifetime maximization of directional sensor networks. Ad Hoc Networks, 11(3), 1006-1021.

Singh, A., Rossi, A., & Sevaux, M. (2013). Matheuristic approaches for Q-coverage problem versions in wireless sensor networks. Engineering Optimization, 45(5), 609-626.

Talbi, E. G. (2009). Metaheuristics: from design to implementation (Vol. 74). John Wiley & Sons.

Tone, K. (2001), "A slacks-based measure of efficiency in data envelopment analysis", European Journal of Operational Research, 130: 498-509.

Wang, X., & Jandhyala, V. (2008, August). Enhanced hybrid MPI-OpenMP parallel electromagnetic simulations based on low-rank

compressions. In 2008 IEEE International Symposium on Electromagnetic Compatibility (pp. 1-5). IEEE.