Fecha de recepción: 12/11/2021
Fecha de aceptación: 17/04/2022
Pp. 116 – Pp. 141

An Extension of the Nested Partitions Method

Ebert Brea
ebrea@ucab.edu.ve
Centro de Investigación y Desarrollo de Ingeniería & Escuela de Ingeniería Industrial,
Universidad Católica Andrés Bello, Caracas, Venezuela

Abstract

This article addresses a new extension of the well known Nested Partitions (NP) method for globally solving mixed integer nonlinear optimization problems under bound constraints. The extension, called Mixed Integer Nested Partitions (MINP) method, is based on the same stages of the NP method at each iteration, i.e.: partitioning; random sampling; identifying of the promising region, which presumes to contain at least a global solution of the problem; and verifying of the stopping rule. Nevertheless, both a new scheme of partitioning and a stopping rule proposal are here presented as main contributions to mixed integer programming. The article has also included a theoretical study of the behavior of the MINP method from the point of view of the Markov chain. Numerical examples have made sure the correct functionality of the algorithmic method and its new stopping rule.

Key words: Nested Partitions method, mixed integer nonlinear programming, global optimization.

Una Extensión del Método de Particiones Anidadas

Resumen

Este artículo aborda una nueva extensión del conocido método de Particiones Anidadas (PA) para resolver globalmente problemas de optimización no lineales enteros mixtos bajo restricciones de bandas. La extensión, llamada método de Particiones Anidadas Enteros Mixtos (PAEM), se basa en las mismas etapas del método PA en cada iteración, es decir: partición; muestreo aleatorio; identificación de la región prometedora, que presume contener al menos una solución global del problema; y verificación de la regla de parada. Sin embargo, tanto un nuevo esquema de partición como una propuesta de regla de parada que se presentan aquí son las principales contribuciones a la programación entera mixta. El artículo también ha incluido un estudio teórico del comportamiento del método PAEM desde el punto de vista de la cadena de Markov. Ejemplos numéricos han asegurado la correcta funcionalidad del método algorítmico y su nueva regla de parada.

Palabras clave: método de Particiones Anidadas, programación no lineal entera mixta, optimización global.

Uma Extensão do Método de Partições Aninhadas

Resumo

Este artigo aborda uma nova extensão do conhecido método de Partições Aninhadas (PA) para resolver globalmente problemas de otimização não linear de inteiros mistos com restrições. A extensão, chamada método de Partições Aninhadas Inteiras Mistas (PAIM), é baseada nos mesmos estágios do método PA em cada iteração: particionamento; amostragem aleatória; identificação da região promissora, que presume conter pelo menos uma solução global do problema; e verificação do critério de parada. Não obstante, tanto um novo esquema de particionamento como uma proposta de critério de parada são aqui apresentados como principais contribuições à programação inteira mista. O artigo também incluiu um estudo teórico do comportamento do método PAIM do ponto de vista da cadeia de Markov. Exemplos numéricos garantem o correto funcionamento do método algorítmico e do novo critério de parada.

Palavras chave: Método de Partições Aninhadas, programação não linear inteira mista, otimização global.

i. INTRODUCTION

Consider the following bound constrained mixed integer nonlinear minimization problem:

Problem 1

$$\underset{z\in\mathbb{R}^n\times\mathbb{Z}^m}{\text{minimize}}\, f(z); \tag{1a}$$

subject to

$$l \preceq z \preceq u, \tag{1b}$$

where: z will henceforward denote any mixed integer vector, i.e., $z = (z^{(1)}, \dots, z^{(n)}; z^{(n+1)}, \dots, z^{(n+m)})^t$, in the (n+m)-multidimensional Euclidean space $\mathbb{R}^n \times \mathbb{Z}^m$; $f(z) : \mathbb{R}^n \times \mathbb{Z}^m \to \mathbb{R}$ is a nonlinear function, for which analytical and explicit mathematical expression cannot be obtained; and $l, u \in \mathbb{R}^n \times \mathbb{Z}^m$ respectively are both the lower and upper bounds of the mixed integer feasible region Θ.

Note that we have here denoted by the symbol $\preceq$ for indicating that a vector precedes to another vector, what will be defined later.

In this case, the objective function must be evaluated by an appropriate simulation model or by solving a nonlinear system equations, and besides the objective function has no gradient function, because the objective function domain is defined in the set mixed integer $\mathbb{R}^n \times \mathbb{Z}^m$.

Nowadays this kind of problems has had an important presence in the industry, due to fact the enormous challenges facing the industry, which must find answers for efficiently designing equipment and systems, and therefore friendly with the environment, and at the same time taking into account the economic sustainability.

Examples of these problems in the branch of chemical engineering are presented by Floudas **[1]**, who introduces an important number of mixed integer nonlinear optimization problems on: design, scheduling and planning of batch processes; heat exchanger network synthesis, etc. Grossmann and Kravanja also present an overview of the applications in many areas within the engineering process **[2, 3]**. Tawarmalani and Sahinidis also address the mixed integer nonlinear programming through a rigorous study presented in **[4, 5]**. For his part, Brea has found optimum solutions for the design of equipment making use of the mixed integer optimization viewpoint **[6, 7]**, and he also has developed a new metaheuristic based on a game framework of random pattern search algorithms, offering a new approach for globally solving mixed integer nonlinear problems **[8, 9, 10]**. Another example worthwhile of commenting, it is the recently published article by Kantor and coworkers, who propose a model for finding an integrated solution, which allows the industry to provide optimum integral solutions within plants and potential industrial symbiosis options, using a mixed integer linear programming approach **[11]**.

There exists a vast amount of algorithmic optimization methods via a wide variety metaheuristic approaches for globally solving constrained and unconstrained mixed integer nonlinear optimization problems, which could be grouped in two large classes, namely: the bio-inspired metaheuristic algorithms and the non bio-inspired metaheuristic algorithms. Among the bio-inspired metaheuristic algorithms, it can be included: Genetic Algorithms **[12, 13]**; Ant Colony **[14]**; Particle Swarm optimization, and Artificial Bee Colony **[15]**, whereas in the group of non bio-inspired metaheuristic algorithms, we have: Simulated Annealing **[16]**; Branch-and-Bound method **[17]**, Nested Partitions method **[18]**; Game of Patterns **[9]**, etc.

The best of our knowledge, there exists only one published article concerning an extension the Nested Partitions method for globally solving mixed integer optimization problems **[19]**, which is based on a hybrid of the Nested Partitions method **[18, 20]** and the well known CPLEX **[21]**. Nevertheless, we here propose a new extension of the Nested Partitions method for finding global solutions to bound constrained mixed integer nonlinear optimization problems based on the main idea of the Nested Partitions, namely: partition of the promising region; sampling of each subregion from partitioned promising region and the surrounding region to the promising region; and identification of the next promising region, which presumes to have a global solution of our problem. Nonetheless, the algorithmic extension, which has been called Mixed Integer Nested Partitions (MINP) method includes: both a novel partitions scheme and a new algorithm stopping rule represent the main contributions of this article.

Numerical examples have allowed us to assert that the MINP method has successfully identified promising regions, in a relative small amount of iterations. Nevertheless, the MINP method has resulted to be a very expensive method, from the viewpoint of times the objective function requires to be evaluated by the algorithm for globally solving this kind of problems.

Due to the fact that the MINP method has resulted to be an expensive algorithm from the viewpoint of the number of function evaluations for identifying at least a global solution, we have therefore considered to undertake furthermore research in the future, for hybridizing the MINP method with some local search method, e.g., the mixed integer randomized pattern search algorithm developed by Brea [**10**], and is thought that by this via, it could be improved its performance.

The rest of the article is organized as follows. Next section introduces basic concepts and main idea of the NP method **[18, 20]** through a very simple didactic example in $\mathbb{R}^2$. Section iii presents the theoretical main results of the MINP method, which are based on the NP method approach. A pseudocode of the MINP method is proposed in Section iv, whereby its procedures and functions have been included in appendices. Section v addresses the main properties of the MINP method, which has been studied from a Markov chain viewpoint. Section vi presents a software, which operates the MINP method for taking r independent samplings of performance measurements of the MINP method. Section vii shows a set of numerical experiments for statistically analyzing the performance of the MINP method. Finally, Section viii discusses advantages and disadvantages of the MINP method, and future research for improving performance of the MINP method. A list of mixed integer nonlinear problems for testing the MINP method has been included in Appendix A.

ii. THE NESTED PARTITIONS METHOD

With the aim of giving a didactic explanation of the NP method principles, consider the following two dimensional real optimization example.

Example 1

$$\underset{x\in\mathbb{R}^2}{\operatorname{minimize}} f(x) \tag{2a}$$

subject to

$$(1,1)^t \preceq x \preceq (9,9)^t, \tag{2b}$$

where $f(x):\mathbb{R}^2\rightarrow\mathbb{R}$ *is a nonlinear objective function.*

The feasible region given by (2b) is indubitably depicted by a square region with both width and height equal to 8 units, such as is shown in Figure 1 in green color.

Note that this region is considered the initial promising region $\sigma(0)$, because the entire feasible region at least contains, and without any discussion, a global solution of the problem.

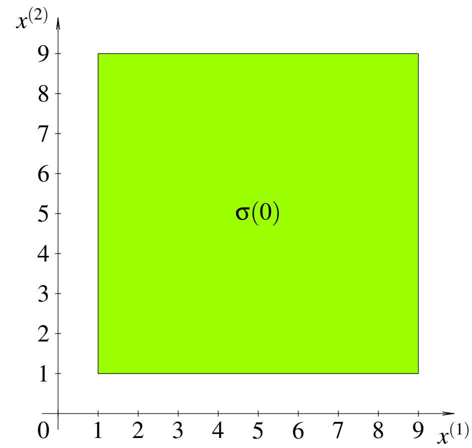

Figure 1. Entire feasible region, promising region $\sigma(0)$

As can be seen from Figure 2, the promising region $\sigma(0)$ is partitioned or divided into 4 subregions, yielding the set of subregions $\{\sigma_j(0)\}_{j=1}^{4}$, and at this iteration $k=0$, there exists no surrounding region to $\sigma(0)$, i.e., $S(\sigma(0))=\emptyset$.

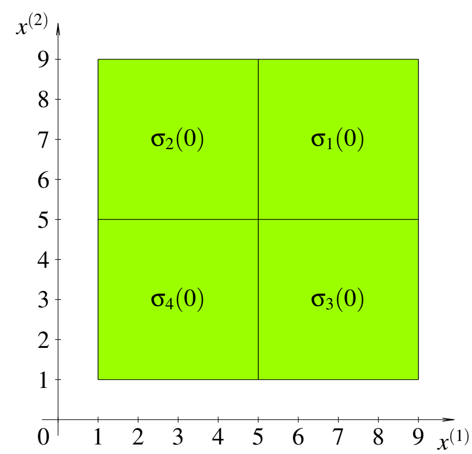

Figure 2. Partitions of promising region $\sigma(0)$

According to the NP method **[18]**, the algorithm takes random trial points from each jth subregion, denoted by $\sigma_j(0)$, for estimating the next promising region by finding an index $\hat{j}$, which indicates where the best function value has come from, as it can then be seen from Figure 3 by a red cross on $\sigma_3(0)$, i.e., $\hat{j}=3$, and hence, the next promising region will be given by the subregion $\sigma_3(0)$.

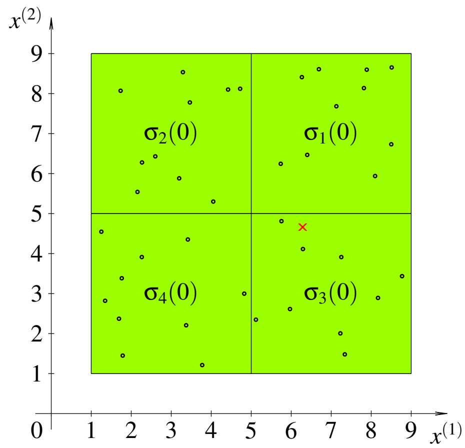

Figure 3. Sampling of each jth subregion $\{\sigma_j(0)\}_{j=1}^{4}$

We must say that if at this stage, the NP method identifies more than one subregion likewise promising, the NP method will arbitrarily break this draw, for choosing just one subregion, of course.

In the example shown from Figure 4, we have hence assumed that the promising region resulted to be $\sigma_3(0)$, which will therefore be the next promising region $\sigma(1)$, and so starts a next iteration of the NP method.

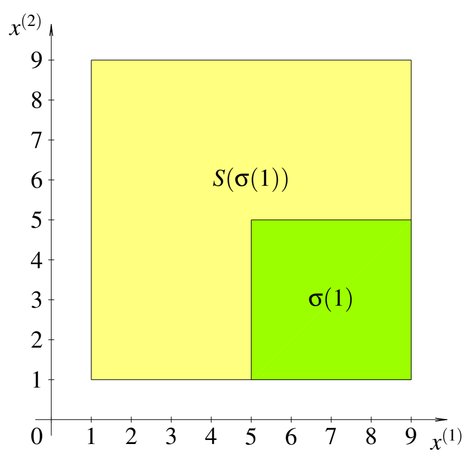

Figure 4. promising region $\sigma(1)$

The partitions of the promising region $\sigma(1)$, into a new set of four subregions $\{\sigma_j(1)\}_{j=1}^{4}$, and that had been denoted by $\sigma_3(0)$ at the 0th iteration, which is depicted in Figure 5 in green color, and also the surrounding region to $\sigma(1)$, that is here denoted by $S(\sigma(1))$, which is shown by the yellow area of the figure.

Figure 6 displays by points the set of random trial points that have been taken from each jth subregion $\sigma_j(1)$, and from the current surrounding region $S(\sigma(1))$. Here, we can say that we have 5 subregions, namely: $\{\sigma_j(1)\}_{j=1}^{4}$; and $\sigma_5(1) = S(\sigma(1))$, what allows the NP method to identify the best subregion, this means, if the best subregion results $\hat{i} \in \{1, 2, 3, 4\}$ the algorithm go toward to next iteration, where the promising region is therefore smaller that the current promising region, whilst if $\hat{i} = 5$, then the NP method backtracks to the initial promising region $\sigma(0)$.

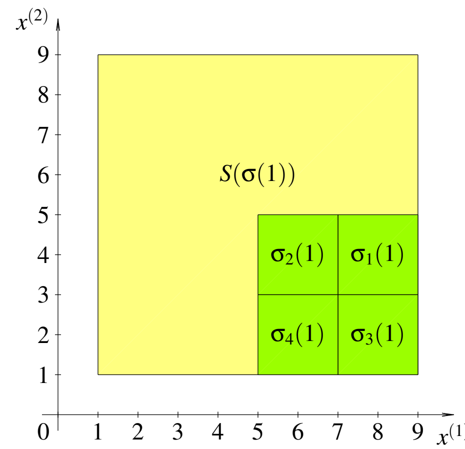

Figure 5. Partitions of promising region $\sigma(1)$

We have assumed that the sampling procedure yielded that the best function value belongs to the subregion $\sigma_2(1)$, which has thus been marked by a red cross on subregion $\sigma_2(1)$, and therefore $\hat{j} = 2$.

We must point out that two backtracking rules have been proposed by Shi and Ólafsson, namely: the first one causes a backtracking process to the previa promising region; and the second one effectuates a backtracking process to the entire feasible region **[18]**.

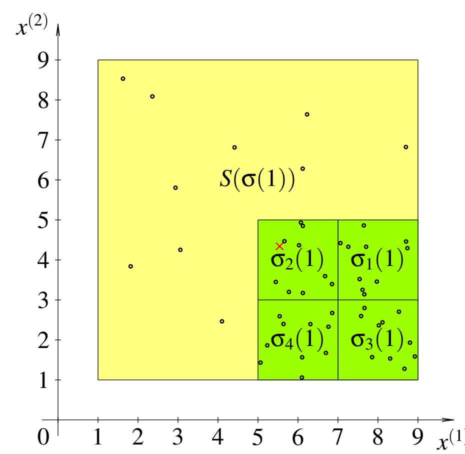

Figure 6. Sampling of each jth subregion $\{\sigma_j(1)\}_{j=1}^{4}$

As is shown in Figure 7, we have assumed that the best objective function value comes from $\sigma_2(1)$, what allows us to identify the subregion $\sigma_2(1)$ as the next promising region, namely, $\sigma(2) = \sigma_2(1)$.

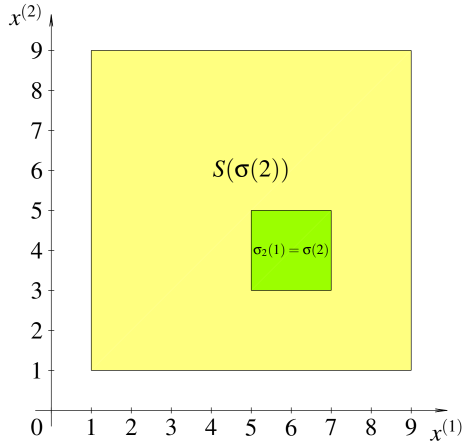

Figure 7. promising region $\sigma(2)$

This procedure is recurrently carried out until some stopping criterion is met.
Shi and Ólafsson have proposed two stopping rules for the NP method, namely: the first one is based on ordinal optimization concepts; and the second one has been developed applying concepts of order statistics for objective function values **[22]**.

iii. THE MIXED INTEGER NESTED PARTITIONS METHOD

The MINP method has been designed for globally solving mixed integer bound constrained nonlinear problems. For that, we have developed the MINP method based on similarity stages to the NP method, namely: i) partitioning of the promising region $\sigma(k)$ into M_σ subregion $\sigma_i(k)$ for all $i \in \{\ell\}_{\ell=1}^{M_\sigma}$; ii) sampling of the subregion $\sigma_i(k)$ by random mixed integer trial points from each subregion; iii) identifying of the next promising region $\sigma(k+1) = \sigma_{\hat{i}}(k)$, where $\hat{i}$ denotes the index where the best value of the objective function has come from; iv) verifying whether the MINP method will go next iteration or it will stop, and as a result of this verifying stage, i.e., the algorithm will accordingly test at each iteration, if the stopping rule is met.

a. PRELIMINARIES

This subsection deals with some definitions for familiarizing the reader with concepts and notations that have been included both in the explanation of the MINP method and its pseudocode.

Definition 1 (Precedent vector) *Let z_1 and z_2 be two mixed integer vectors, i.e., $z_1, z_2 \in \mathbb{R}^n \times \mathbb{Z}^m$. It is said that z_1 precedes z_2 in order, and will be denoted by $z_1 \preceq z_2$, if their respective components $z_1^{(i)} \leq z_2^{(i)}$ for each $i \in \{\ell\}_{\ell=1}^{n+m}$.*

Definition 2 (Subsequent vector) *Let z_1 and z_2 be two mixed integer vectors, i.e., $z_1, z_2 \in \mathbb{R}^n \times \mathbb{Z}^m$. It is said that z_1 succeeds z_2 in order, and will be denoted by $z_1 \succeq z_2$, if their respective components $z_1^{(i)} \geq z_2^{(i)}$ for each $i \in \{\ell\}_{\ell=1}^{n+m}$.*

The relationship between the vectors that have been defined above are respectively called strictly precedes or strictly succeeds, if the above inequalities are true as strict inequalities for all ith components of z_1 and z_2.

Definition 3 (Promising region) *Let $\sigma(k)$ be a nonempty promising region at the current iteration k of the MINP method. It is then said that $\sigma(k)$ is given by*

$$\sigma(k) = \{z \in \mathbb{R}^n \times \mathbb{Z}^m : l_{\sigma(k)} \preceq z \preceq u_{\sigma(k)}\}, \ \forall k \in \mathbb{N}, \quad (3)$$

where: k hereafter depicts the counter of iteration that has been carried out by the MINP method from the initial promising region $\sigma(0) = \Theta = \{z \in \mathbb{R}^n \times \mathbb{Z}^m : l \preceq z \preceq u\}$; $l_{\sigma(k)}, u_{\sigma(k)} \in \mathbb{R}^n \times \mathbb{Z}^m$ respectively are the lower and upper bound vector of the kth promising region $\sigma(k)$, which presumes to contain at least a global solution of Problem 1.

Note that according to Definition 3, each kth promising region must be a nonempty, because it presumes to contain at least a global solution of Problem 1. Moreover, the smallest promising region, in the number of element sense, that is contained on the feasible region of Problem 1, is a singleton set.

Definition 4 (Subregion) *Let $\sigma_i(k)$ denote the ith subregion of the current kth promising region. It is then said that $\sigma_i(k) \subset \sigma(k)$ for each $i \in \{\ell\}_{\ell=1}^{M_\sigma}$, and $\sigma_i(k) \cap \sigma_j(k) = \emptyset$ for all $i \neq j$ and $i, j \in \{\ell\}_{\ell=1}^{M_\sigma}$, where M_σ denotes the number of subregion.*

Definition 5 (Surrounding region) *Let $S(\sigma(k))$ be the surrounding region to the kth promising region $\sigma(k)$. It is then said that $S(\sigma(k)) = \sigma(0) \setminus \sigma(k)$.*

Definition 6 (Depth vector) *Let $d(k)$ denote the kth mixed integer depth vector of the current kth promising region. It is then said that $d(k) = u_\sigma - l_\sigma$, i.e.,*

$$d(k) = \left(u^{(1)}_{\sigma(k)} - l^{(1)}_{\sigma(k)}, \ldots, u^{(n+m)}_{\sigma(k)} - l^{(n+m)}_{\sigma(k)}\right)^t, \forall k \in \mathbb{N}. \quad (4)$$

Note that according to above definitions: i) $\sigma(0)$ is the feasible region Θ of Problem 1; ii) the kth promising region $\sigma(k)$ is a convex hull, which is bounded by $l_{\sigma(k)} \preceq z \preceq u_{\sigma(k)}$, and it will hereafter be simplified its notation by $l_\sigma \preceq z \preceq u_\sigma$; and iii) $\sigma(k) = \bigcup_{i=1}^{M_\sigma} \sigma_i(k)$. Finally, let $\kappa \in \mathbb{N}$ be the cumulative counter iterations that are executed by the MINP method, which always is greater than or equal to k.

b. ON THE PARTITIONS OF THE PROMISING REGION

We now turn our attention to the proposed procedure of partitioning of the kth promising region $\sigma(k)$, which is based on a novel method for getting the boundaries of each ith subregion $\sigma_i(k)$ in the current promising mixed integer region, as is shown in Figure 18 on page 135.

Proposition 1 *Let $\sigma^{(j)}(k) = \{x \in \mathbb{R} : l^{(j)}_\sigma \leq x \leq u^{(j)}_\sigma\}$ be the jth interval of real number or set of the jth real value component that defines the current promising mixed integer region $\sigma(k) \in \mathbb{Z}^m$. Then, an adequate partition of the jth component of $\sigma^{(j)}(k)$ into two disjoint subsets is given by*

$$\sigma^{(j)}_1(k) = \{x \in \mathbb{R} : l^{(j)}_\sigma \leq x \leq \delta^{(j)}_\sigma\}; \quad (5a)$$

$$\sigma^{(j)}_2(k) = \{y \in \mathbb{R} : \delta^{(j)}_\sigma < y \leq u^{(j)}_\sigma\}, \quad (5b)$$

where $\delta^{(j)}_\sigma = (l^{(j)}_\sigma + u^{(j)}_\sigma)/2$ is the midpoint of the rect line segment defined by both $l^{(j)}_\sigma$ and $u^{(j)}_\sigma$.

Proof. We know that a partition of an integer number set into two subsets must then yield two disjoint subsets and the union of both subsets must contain all elements of the original set.
As can be seen: first, $\sigma^{(j)}_1(k) \cap \sigma^{(j)}_2(k) = \emptyset$, and second, $\sigma^{(j)}(k) = \sigma^{(j)}_1(k) \cup \sigma^{(j)}_2(k)$, what has allowed us to partition the set $\sigma^{(j)}(k)$ properly. ∎

Proposition 2 *Let $\bar{\sigma}^{(j)}(k) = \{y \in \mathbb{Z} : \bar{l}^{(j)}_\sigma \leq y \leq \bar{u}^{(j)}_\sigma\}$ be the jth interval of integer number or set of the jth integer value component that defines the current promising mixed integer region $\bar{\sigma}(k) \in \mathbb{Z}^m$. Then, an appropriate partition of the jth component of $\bar{\sigma}^{(j)}(k)$ into two disjoint subsets is given by*

$$\begin{cases} [\bar{l}^{(j)}_\sigma, \delta^{(j)}] \cup [\delta^{(j)} + 1, \bar{u}^{(j)}_\sigma], & \text{if } \lfloor \delta^{(\ell)} \rfloor = \lceil \delta^{(\ell)} \rceil; \\ [\bar{l}^{(j)}_\sigma, \lfloor \delta^{(\ell)} \rfloor] \cup [\lceil \delta^{(\ell)} \rceil, \bar{u}^{(j)}_\sigma], & \text{if } \lfloor \delta^{(\ell)} \rfloor \neq \lceil \delta^{(\ell)} \rceil, \end{cases} \quad (6)$$

where $\delta^{(j)}_\sigma = (\bar{l}^{(j)}_\sigma + \bar{u}^{(j)}_\sigma)/2$ is the midpoint of the rect line segment defined by $\bar{l}^{(j)}_\sigma$ and $\bar{u}^{(j)}_\sigma$.

Proof. Because we have two cases, the first one, when $\delta^{(j)} = (\bar{l}^{(j)}_\sigma + \bar{u}^{(j)}_\sigma)/2$ is an integer number, and the second one, when $\delta^{(j)}$ is a no integer number, these cases will separately be proved.
Part i) if $\delta^{(j)}$ is an integer number, i.e., $\lfloor \delta^{(\ell)} \rfloor = \lceil \delta^{(\ell)} \rceil$ the set given by $\bar{\sigma}^{(j)}(k) = \{y \in \mathbb{Z} : \bar{l}^{(j)}_\sigma \leq y \leq \bar{u}^{(j)}_\sigma\}$ can be partitioned into two subsets, namely, $\bar{\sigma}^{(j)}_1(k) = \{y \in \mathbb{Z} : \bar{l}^{(j)}_\sigma \leq y \leq \delta^{(j)}\}$ and $\bar{\sigma}^{(j)}_2(k) = \{y \in \mathbb{Z} : \delta^{(j)} + 1 \leq y \leq \bar{u}^{(j)}_\sigma\}$, where both mentioned subsets indeed are disjoint, i.e., $\bar{\sigma}^{(j)}_1(k) \cap \bar{\sigma}^{(j)}_2(k) = \emptyset$, and $\bar{\sigma}^{(j)}(k) = \bar{\sigma}^{(j)}_1(k) \cup \bar{\sigma}^{(j)}_2(k)$.
Part ii) In the case when $\delta^{(j)}$ is a no integer number, then the subsets $\bar{\sigma}^{(j)}_1(k) = \{y \in \mathbb{Z} : \bar{l}^{(j)}_\sigma \leq y \leq \lfloor \delta^{(j)} \rfloor\}$ and $\bar{\sigma}^{(j)}_2(k) = \{y \in \mathbb{Z} : \lceil \delta^{(j)} \rceil \leq y \leq \bar{u}^{(j)}_\sigma\}$ are such that $\bar{\sigma}^{(j)}_1(k) \cap \bar{\sigma}^{(j)}_2(k) = \emptyset$, and $\bar{\sigma}^{(j)}(k) = \bar{\sigma}^{(j)}_1(k) \cup \bar{\sigma}^{(j)}_2(k)$. ∎
Based on Propositions 1 and 2, it was designed Partitioning procedure shown in Figure 18.

Remark 1 *Assume, without any loss of generality, that Partitions procedure in Figure 18 is conducted on any kth promising region $\sigma(k)$ for solving Problem 1, as a result of Propositions 1 and 2. Then Partitions procedure yields $M_\sigma = 2^{n+m}$ disjoint subregions from the current promising region $\sigma(k)$.*

Proof. Due to the fact that according to Partitioning procedure in Figure 18, at each kth iteration, both each of the n real components is divided by into 2 subsets and each of the m integer components is also divided into 2 subsets, yielding respectively at each iteration 2^n and 2^m subsets, thus, this fact allows us to obtain $M_\sigma = 2^{n+m}$. ∎
As can be seen from Partitions procedure in Figure 18, at each kth iteration a set of $\{\sigma_i(k)\}_{i=1}^{M_\sigma}$ subregions from the current promising region $\sigma(k)$ are mutually disjoint subsets, i.e., $\sigma_i(k) \cap \sigma_i(k) = \emptyset$ for all $i \neq j$ and $i, j \in \{\ell\}_{\ell=1}^{M_\sigma}$, and moreover $\sigma(k) = \bigcup_{i=1}^{M_\sigma} \sigma_i(k)$.

c. ON THE STOPPING RULE

As was mentioned in the introduction, one of the contributions of this research has been the inclusion of a new stopping rule, which is based on the depth vector $d(k)$ of the current promising region, without considering any objective function value, because our approach is only based on partitioning scheme, leaving for another research the study of stopping rule based on statistical viewpoints, such as statistical considerations on the feasible global minimum function value applying non-standard parametric inference. A good approach on non-standard parametric inference is introduced by Cheng **[23]**, which could be taken into account for using it together with the concepts of order statistics presented by De Haan **[24]**.

Taking into account the main pseudocode of the MINP method in Figure 8 on page 124, we can notice that when $d(k+1) \preceq d(k)$ holds, and the $\hat{\hat{i}} \in \{\ell\}_{\ell=1}^{M_\sigma}$ the algorithm carries out a new partitioning on the new promising region, otherwise, as a result of that $\hat{\hat{i}} = M_\sigma + 1$, the algorithm backtracks to the entire feasible region, and it restarts the iteration counter by letting k be equal to 0. Note that the surrounding to the promising region $\sigma(k)$ has been here denoted by $\sigma_{M_\sigma+1}(k)$, i.e., $S(\sigma(k)) = \sigma_{M_\sigma+1}(k)$.

Since the MINP method estimates the current promising region at each kth, the current depth mixed integer vector $d(k)$ can be then easily calculated, for being afterward tested at the end of each kth iteration, and so verifying whether $\varepsilon \preceq d(k)$ or not, whereby in the case that $\varepsilon \preceq d(k)$ the MINP method will go to next iteration, otherwise, the MINP method will stop.

We shall now focus on calculating the maximum number of iteration, denoted by $\hat{k}$, that the MINP method needs for stopping the algorithm, considering that any backtracking operation has not been carried out during the identification of a global solution to Problem 1.

Due to the fact that we have still no stopping rule for the MINP method, it is reasonable to enunciate the following condition.

Condition 1 *Suppose that is executed the iterative loop of the MINP method, that will afterward be shown in Figure 8, without considering any stopping rule for globally solving Problem 1.*

Lemma 1 *Assume Condition 1 holds. Then the ith real component of the mixed integer depth $d(k)$ is given by*

$$d^{(i)}(k) = \frac{u^{(i)} - l^{(i)}}{2^k}, \quad \forall i \in \{\ell\}_{\ell=1}^{n}, k \in \mathbb{N}, \tag{7}$$

where $l, u \in \mathbb{R}^n \times \mathbb{Z}^m$ respectively are the lower and upper bound constrains given by (1b).

Proof. Due to the fact that the MINP method partitions each ith component of the promising region into two subsets, the ith real component is successively divided by 2 at each kth, resulting the geometric series given by (7), for each ith real component of the mixed integer depth vector. ■

Lemma 2 *Assume Condition 1 holds and the expected mixed integer depth vector ε is fixed to $(\epsilon, \ldots, \epsilon; 0, \ldots, 0)^t$. The MINP method hence stops when*

$$\frac{\max_{i \in \{1,\ldots,n\}} (u^{(i)} - l^{(i)})}{2^k} < \epsilon \tag{8}$$

is met.

Proof. During the execution of the algorithm, successive divisions by two are carried out, both to the real and integer components, what allows us to assert that any integer component $\bar{d}^{(i)}(k)$ for all $i \in \{\ell\}_{\ell=n+1}^{m}$ of the mixed integer depth vector becomes 0, faster than any real component $d^{(i)}(k)$ for all $i \in \{\ell\}_{\ell=1}^{n}$ of the mixed integer depth vector. Besides, all integer components of the mixed integer vector $\bar{d}^{(i)}(k)$ for all $i \in \{\ell\}_{\ell=n+1}^{m}$ become 0 in a finite number of iteration k, whereas that the real components $d^{(i)}(k)$ for all $i \in \{\ell\}_{\ell=1}^{n}$ approach to 0 when $k \to \infty$. Since Lemma 1 holds, the depth vector $d(k)$ will therefore be precede to the expected mixed integer depth vector ε, when (8) is met. ■

In what follows, we provide evidence that the proposed stopping rule successfully works, avoiding an over iterations of the algorithm, when the stopping rule had been met, as a result that exists a positive probability that is reached.

Theorem 1 *Let $\hat{k} = \lceil k \rceil$ be the closest integer greater than or equal to k, assume Condition 1 holds, and the expected mixed integer vector ε has been fixed to $(\epsilon, \ldots, \epsilon; 0, \ldots, 0)^t$. Then minimum number of required iterations by the MINP method for being stopped, i.e., when $d(\hat{k}) \prec \varepsilon$ has just been met, it is given by*

$$\hat{k} = \left\lceil \log_2 \left(\frac{\max_{i \in \{1,\ldots,n\}} (u^{(i)} - l^{(i)})}{\epsilon} \right) \right\rceil. \tag{9}$$

Proof. Getting k by solving (8) from Lemma 2, the MINP method stops at iteration $\hat{k}$ given by (9). ■

Now, we shall pay attention on the estimation of the probability of carrying out a backtracking operation to the entire region Θ at any kth iteration, such that $0 < k < \hat{k}$.

Remark 2 *Assume Condition 1 holds. Each kth promising region is therefore given by*

$$\sigma(k) = \{z \in \mathbb{R}^n \times \mathbb{Z}^m : l_\sigma \preceq z \preceq u_\sigma\}, \qquad (10)$$

where $l_\sigma = (l_\sigma^{(1)}, \ldots, l_\sigma^{(n)}; \bar{l}_\sigma^{(n+1)}, \ldots, \bar{l}_\sigma^{(n+m)})^t$ and $u_\sigma = (u_\sigma^{(1)}, \ldots, u_\sigma^{(n)}; \bar{u}_\sigma^{(n+1)}, \ldots, \bar{u}_\sigma^{(n+m)})^t$ respectively are the lower and upper mixed integer bounds of the kth promising region.

Proof. Since Condition 1 is always satisfied, at each kth iteration is rightly partitioned the current promising region $\sigma(k)$ due to Propositions 1 and 2, what allow us to describe exactly $\sigma(k)$ by (10). ■

Note that by Partitioning procedure in Figure 18 clearly defines each generated kth promising region $\sigma(k)$.

Theorem 2 *Suppose that there is no any stopping rule for the MINP method. If ν and N are both positive quantities, there will then exist a positive probability that the algorithm goes forward in depth, getting a new depth vector $d(k+1)$ that precedes to $d(k)$, at any kth iteration, i.e., $Pr\{d(k+1) \prec d(k)\} > 0$ for all $k \in \mathbb{N}$.*

Proof. Using Propositions 1 and 2, we clearly have that the event $\{d(k+1) \prec d(k)\}$ takes place with a positive probability. ■

The next result shows how Theorem 2 allows us to affirm that the MINP method does not over run for getting a global solution, namely, it finishes with a positive probability.

Corollary 1 *Suppose Condition 1 holds for globally solving Problem 1. Besides, assume that ν and N are both positive quantities; and the expected mixed integer vector ε is fixed to both real and integer no negative components. Then the MINP method will meet the stopping rule, i.e., $d(k) \prec \varepsilon$ in a limited number of successive iterations with a positive probability.*

Proof. Because of Theorem 2, there exists a positive probability that the MINP method reaches the stopping rule $d(k) \prec \varepsilon$, and then it does not carry out any more iterations, causing a stopping of the algorithm. ■

Further results on the behavior of the MINP method will afterward be presented in next section.

d. ON THE PERFORMANCE MEASUREMENT

We now consider the issue of measuring the performance of the MINP method, when it has been collected a set of solutions that has been yielded from r independent runs of the MINP method, under the same conditions.

For this reason, we shall here show the main results on the proposed performance measurement, which was introduced by Brea **[25]** for comparing the Game of Patterns **[9]** versus two implementations of Genetic Algorithms, when these last mentioned algorithms are used for globally solving constrained nonlinear problems in $\mathbb{R}^n$

Proposition 3 *Let p be a mixed integer bound constrained problem in $\mathbb{R}^n \times \mathbb{Z}^m$; let $\eta^{(\ell)}(p,n,m) > 0$ be the number of times that the objective function has been evaluated during ℓth run of the algorithm for searching of a solution to Problem p; and let $\lambda^{(\ell)}(p,n,m) \geq 0$ be the distance between the true point or solution of Problem p, and the best achieved solution by the MINP method at the ℓth replication. Then, for each $\ell \in \mathbb{N}_+$,*

$$q^{(\ell)}(p,n,m) = \frac{1}{1 + \eta^{(\ell)}(p,n,m) \cdot \lambda^{(\ell)}(p,n,m)}, \qquad (11)$$

is on $(0,1]$.

Proof. It suffices to verify the implication (11), due to the fact that $\eta^{(\ell)}(p,n,m) > 0$ and $\lambda^{(\ell)}(p,n,m) \geq 0$, what easily allows us to prove that for each ℓth algorithm run or also called replication is yielded a sampling $q^{(\ell)}(p,n,m) \in (0,1]$. ■

Of course, that any collection $\{q^{(\ell)}(p,n,m)\}_{\ell=1}^{r}$ of r independent executions of the algorithm would then yield an empirical distribution so far from being a normal distribution, whereby a parametric statistical analysis, under standard situations, of these data would be unsuitable for obtaining a right description of the here called random variable quality, which will hereafter be denoted by $Q(p,n,m)$, and given by

$$Q(p,n,m) = \frac{1}{1 + N(p,n,m)\, L(p,n,m)}, \qquad (12)$$

where $N(p,n,m)$ and $L(p,n,m)$ respectively are the random variables: number of function evaluation and distance to the true point, and whose distribution functions are unknown.

Algorithm. The Mixed Integer Nested Partitions Method

Initialization

Given:

the number of real components, n;
the number of integer components, m;
the problem

$$\underset{z\in\mathbb{R}^n\times\mathbb{Z}^m}{\text{minimize}}\, f(z),$$

$$\text{subject to} \quad l \preceq z \preceq u,$$

where $l,u\in\mathbb{R}^n\times\mathbb{Z}^m$, e.i.: $l=(\underbrace{l^{(1)}),\dots,l^{(n)})}_{n};\underbrace{\bar{l}^{(n+1)},\dots,\bar{l}^{(n+m)}}_{m})^t$ and $u=(\underbrace{u^{(1)}),\dots,u^{(n)})}_{n};\underbrace{\bar{u}^{(n+1)},\dots,\bar{u}^{(n+m)}}_{m})^t$;

the number $M_{\sigma(0)}$ of subregions of the current most promising region $\sigma(0)=\Theta$;
the number of sample points $N_j(k)$ from each jth subregion $\sigma_j(k)$ at each kth iteration;

Choose:

an $\varepsilon^{(i)}\in\mathbb{R}_+$ for each $i\in\{1,\dots,n\}$;
an $\bar{\varepsilon}^{(i)}\in\mathbb{N}$ for each $i\in\{n+1,\dots,n+m\}$, what allows us to define the expected maximum depth of the mixed integer vector, $\varepsilon=(\underbrace{\varepsilon^{(1)},\dots,\varepsilon^{(n)}}_{n};\underbrace{\bar{\varepsilon}^{(n+1)},\dots,\bar{\varepsilon}^{(n+m)}}_{m})^t\in\mathbb{R}_+^n\times\mathbb{N}^m$;

Declare:

the mixed integer depth vector $d(k)=(\underbrace{d^{(1)}(k),\dots,d^{(n)}(k)}_{n};\underbrace{\bar{d}^{(n+1)}(k),\dots,\bar{d}^{(n+m)}(k)}_{m})^t$;

Let $k=0$;

Calculate:

the initial mixed integer depth vector, e.i., $d(0)=\mathcal{D}(0,n,m,\sigma(0),\Theta)$, of the promising region $\sigma(0)$;

while $\varepsilon\preceq d(k)$ **do**

Partitioning
Partition the current promising region $\sigma(k)$ into $\{\sigma_j(k)\}_{j=1}^{M_{\sigma(k)}}$ subregion;
Aggregate the current surrounding region $\sigma_{M_{\sigma(k)+1}}(k)=\Theta\setminus\sigma(k)$, and denote it as $S(\sigma(k))$;

Random Sampling
Execute the Sampling Procedure for taking $N_j(k)$ random points from each jth subregion $\{\sigma_j(k)\}_{j=1}^{M_{\sigma(k)}}$, namely: for each jth subregion, get the set of sampled points $\{\theta_{j,s}\}_{s=1}^{N_j(k)}$;
Execute the Sampling Procedure for also taking $N_{\sigma(k)+1}(k)$ random points from surrounding region $\sigma_{M_{\sigma(k)+1}}(k)$, that is, for getting $\{\theta_{M_{\sigma(k)+1},s}\}_{s=1}^{N_{M_{\sigma(k)+1}}(k)}$;

Measuring
Calculate the performance of each got sampled points using the objective function $f(\theta)$, namely: $f(\theta)|_{\theta_{j,s}}$ for jth subregion at each sth random sampling, and also the sampled points from the surrounding region $\sigma_{M_{\sigma(k)+1}}(k)$;

Estimating
Estimate the promising region, that is, finding the index of the best performance, namely, first estimate

$$\hat{I}(\sigma_j(k))=\min_{s\in\{1,\dots,N_j\}} f(\theta_{j,s}),$$

and then

$$\hat{j}_k=\min_{j\in\{1,\dots,M_{\sigma(k)+1}\}} \hat{I}(\sigma_j(k)).$$

In the case that two or more regions are equally promising, the tie can be randomized broken.

Decision
if $\hat{j}_k\leq M_{\sigma(k)}$ **then**
  Let $\sigma(k+1)=\sigma_{\hat{j}_k}(k)$;
  Update by using $d(k+1)=\mathcal{D}(k+1,n,m,\sigma(k+1),\Theta)$;
  Let $k\leftarrow k+1$;
else
  Backtrack to the entire feasible region Θ;
  Let $k\leftarrow 0$;
  Let $\sigma(k)=\Theta$;
  Update by using $d(0)=\mathcal{D}(0,n,m,\sigma(k),\Theta)$;

Figure 8. The MINP method

iv. THE PSEUDOCODE OF THE MINP METHOD

Taking into account the theoretical main results above, it has been proposed the MINP method, which is shown in Figure 8 using an easy description of its source code. The figure shows: the preamble of the algorithm, in which given: i) the problem; ii) a chosen expected depth vector; and iii) a declared vector $d(k)$, the MINP method executes iterative loops for solving Problem 1, whilst the stopping rule has not been met.

v. ON THE MINP METHOD

In this section we study the main properties of the MINP method from a stochastic viewpoint. According to the MINP method, it is reasonable to figure up that the MINP method generates a Markov stochastic process on a countable and infinite discrete stochastic state space, i.e., a Markov chain that can be described by a set of $\{D(\kappa)\}_{\kappa=0}^{\infty}$, where $\kappa \in \mathbb{N}$ here depicts a cumulative iteration counter of the MINP method, what means in a nonmathematical language: *the future only depends on today, and not what have occurred in past time*

We shall therefore turn up our attention to the Markov chain state space that is generated by the MINP method, when it is run for globally solving any mixed integer problem given by Problem 1. However, before introducing our main results on the Markov chain that is generated by the MINP method (MINPMC), we must point out that each state of the MINPMC space will be measured by a depth vector of the kth baggy hull of the current promising region $\sigma(k)$, which will be called baggy hull depth vector, and we shall then be defined it as follows.

Definition 7 (Baggy hull of a promising region) *Let $\sigma(k)$ be a kth nonempty promising region given by*

$$\sigma(k) = \{z \in \mathbb{R}^n \times \mathbb{Z}^m : l_{\sigma(k)} \preceq\ z \preceq\ u_{\sigma(k)}\}, \qquad (13)$$

with mixed integer depth vector $d(k) = u_{\sigma(k)} - l_{\sigma(k)}$. It says to be a baggy hull of $\sigma(k)$, denoted by $\tilde{\sigma}(k)$, is the smallest oversized hull $\sigma(k)$ such that $\tilde{\sigma}(k) \supseteq \sigma(k)$.

Basing on this last definition, we thus have the following result.

Proposition 4 *Let $\tilde{d}(k)$ be the kth baggy hull depth vector of the current promising region $\sigma(k)$, which is given by*

$$\tilde{d}(k) = \left(\frac{u^{(1)} - l^{(1)}}{2^k}, \ldots, \frac{u^{(n)} - l^{(n)}}{2^k}; \left\lceil \frac{\bar{u}^{(n+1)} - \bar{l}^{(n+1)}}{2^k} \right\rceil, \ldots, \left\lceil \frac{\bar{u}^{(n+m)} - \bar{l}^{(n+m)}}{2^k} \right\rceil \right)^t \qquad (14)$$

Then $\tilde{d}(k)$ depicts the measurements of the smallest baggy hull in $\mathbb{R}^n \times \mathbb{Z}^m$ and thus $\tilde{d}(k) \succeq\ d(k)$ for each $k \in \mathbb{N}$.

Proof. For proving this statement, we first pay attention to the real components, and later to the integer components of the promising region $\sigma(k)$.

Because of Proposition 1, each jth real component of the current promising region $\sigma(k)$ has successively been divided by 2, yielding as a result consecutive partitions into two subsets per each kth iteration, we therefore have by (5a) and (5b) that

$$\delta_{\sigma(k)}^{(j)} - l_{\sigma(k)}^{(j)} \leq \frac{u^{(j)} - l^{(j)}}{2^k}; \qquad (15\text{a})$$

$$u_{\sigma(k)}^{(j)} - \delta_{\sigma(k)}^{(j)} \leq \frac{u^{(j)} - l^{(j)}}{2^k}, \qquad (15\text{b})$$

where $\delta_{\sigma(k)}^{(j)} = (l_{\sigma(k)}^{(j)} + u_{\sigma(k)}^{(j)})/2$.

Secondly, due to Proposition 2, each jth integer component of the current promising region $\sigma(k)$ has successively been partitioned into a pair of subsets, consecutively generating two subsets not necessarily the same size per each kth iteration, hence, by (6) we know that if $\delta_{\sigma(k)}^{(j)} = (\bar{l}_{\sigma(k)}^{(j)} + \bar{u}_{\sigma(k)}^{(j)})/2$ is an integer number, i.e., $\lfloor \delta_{\sigma(k)}^{(j)} \rfloor = \lceil \delta_{\sigma(k)}^{(j)} \rceil$, then

$$\delta_{\sigma(k)}^{(j)} - \bar{l}_{\sigma(k)}^{(j)} \leq \left\lceil \frac{\bar{u}^{(j)} - \bar{l}^{(j)}}{2^k} \right\rceil; \qquad (16\text{a})$$

$$\bar{u}_{\sigma(k)}^{(j)} - \delta_{\sigma(k)}^{(j)} - 1 \leq \left\lceil \frac{\bar{u}^{(j)} - \bar{l}^{(j)}}{2^k} \right\rceil. \qquad (16\text{b})$$

Whereas, if $\delta_{\sigma(k)}^{(j)}$ is a non integer number, i.e., $\lfloor \delta_{\sigma(k)}^{(j)} \rfloor \neq \lceil \delta_{\sigma(k)}^{(j)} \rceil$, then

$$\lfloor \delta_{\sigma(k)}^{(j)} \rfloor - \bar{l}_{\sigma(k)}^{(j)} \leq \left\lceil \frac{\bar{u}^{(j)} - \bar{l}^{(j)}}{2^k} \right\rceil; \qquad (17\text{a})$$

$$\bar{u}^{(j)}_{\sigma(k)} - \lceil \delta^{(j)}_{\sigma(k)} \rceil \leq \left\lceil \frac{\bar{u}^{(j)} - \bar{l}^{(j)}}{2^k} \right\rceil . \tag{17b}$$

■

Note that the measurements of any kth promising region $\sigma(k)$ are given by all the components of $d(k)$, and similarly denoting $\tilde{\sigma}(k)$ as the baggy hull of the correspondent promising region $\sigma(k)$, and whose measurements are given by all the components of $\tilde{d}(k) \in \mathbb{N}$, we can hence say that $\sigma(k) \subseteq \tilde{\sigma}(k)$.

We are now interested in describing the Markov chain that is generated by the MINP method, we can hence say that due to Proposition 4, the Markov chain from the MINP method (MINPMC) can rigorously be explained using (14), yielding either, under Condition 1 an unlimited countable set of discrete stochastic states $\{D(\kappa) = \tilde{d}(k)\}_{k=0}^{\infty}$ for all $k \in \mathbb{N}$; or under Condition 2 a limited countable set of discrete stochastic states $\{D(\kappa) = \tilde{d}(k)\}_{k=0}^{\hat{k}}$, where $\hat{k}$ is given by (9).

Observe the difference that has been made between the counters k and κ of the algorithm. The first one means the number of iterations that are executed by the algorithm from initial state $D(\kappa) = \tilde{d}(0)$; whilst the second one depicts the number of cumulative iterations that has been carried out by the MINP method from its starting.

Let

$$p^{(k,\ell)}_{\kappa} = Pr\{D(\kappa+1) = \tilde{d}(\ell) \mid D(\kappa) = \tilde{d}(k)\} \tag{18}$$

denote the single-step transition probability that the MINPMC will visit state $D(\kappa+1) = \tilde{d}(\ell)$ at the cumulative iteration $\kappa+1$, if the MINPMC is at state $D(\kappa) = \tilde{d}(k)$ at the κth cumulative iteration counter, for all $k, \kappa \in \mathbb{N}$ and $\ell \in \{0, k+1\}$. Nevertheless, it is reasonable to assume that (18) dependents on k, whereby the MINPMC is said to be an nonhomogeneous Markov chain, what will hereafter be denoted by

$$\begin{aligned} p^{(k,\ell)}_{k} =& Pr\{D(k+1) = \tilde{d}(\ell) \mid D(k) = \tilde{d}(k)\}, \\ & \forall\, k, \kappa \in \mathbb{N}, \ell \in \{0, k+1\}. \end{aligned} \tag{19}$$

Figure 9 illustrates the MINPMC, wherein each state is symbolized by denoted rounded node $\tilde{d}(k)$, and each transition probability $p^{(k,\ell)}_{k}$ by directed arcs, which represent the probability that the MINPMC changing from $\tilde{d}(k)$ state to either $\tilde{d}(\ell)$ state for each $\ell \in \{0, k+1\}$, at kth iteration counter.

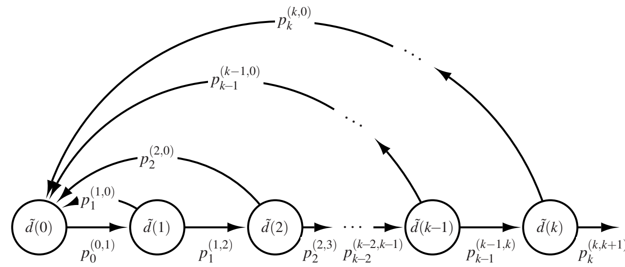

Figure 9. The MINP Markov chain

Condition 2 *Suppose that the MINP method has successfully been run for globally solving Problem 1 applying proposed the algorithm in Figure 8.*

Condition 3 *Assume that the MINP method uniformly takes random samplings by randomized located trial points from both each subregion and the current surrounding region, using both independent and identically distributed (i.d.d.) continuous uniform random variables and i.d.d. discrete uniform random variables.*

We shall now introduce a measurement of mixed integer mixed region size, which will allows us to estimate the probability of the MINPMC events.

Definition 8 (Hypervolume of a region) *Let* $\mathcal{V}[\sigma(k)]$ *denote the hypervolume of* $\sigma(k)$*, which is defined by*

$$\begin{aligned} \mathcal{V}[\sigma(k)] =& \prod_{\ell=1}^{n} \left(u^{(\ell)}_{\sigma(k)} - l^{(\ell)}_{\sigma(k)} \right) + \prod_{\ell=1}^{m} \left(\bar{u}^{(\ell)}_{\sigma(k)} - \bar{l}^{(\ell)}_{\sigma(k)} \right), \\ & \forall k \in \mathbb{N}. \end{aligned} \tag{20}$$

Note that Definition 8 can also be used for calculating the baggy hull hypervolume of any promising region, which is

$$\begin{aligned} \mathcal{V}[\tilde{\sigma}(k)] =& \prod_{\ell=1}^{n} \left(\frac{u^{(\ell)} - l^{(\ell)}}{2^k} \right) + \prod_{\ell=1}^{m} \left\lceil \frac{\bar{u}^{(\ell)} - \bar{l}^{(\ell)}}{2^k} \right\rceil, \\ & \forall k \in \mathbb{N}. \end{aligned} \tag{21}$$

Based on (21), we shall now consider the following proposition.

We now consider as a successful event $\mathcal{F}$, which has occurred as a consequence of that the MINP method has discovered the existence of the best value of the objective function comes from any subregion, among all sampled trial points at a kth iteration, i.e., at a $\tilde{d}(k)$ state, and by contrast, let $\mathcal{B}$ denote the event that a backtracking operation is carried out by the algorithm as a result that the MINP method identified the best function value comes from the kth surrounding region at the same kth counter iteration, therefore, a geometric distribution is taken place in the estimation of the probability distribution of the MINPMC states.

Proposition 5 *Assume, without loss of generality (w.l.o.g.), Condition 1 and 3 are met, and let* $N_i = \nu(>0)$ *for all* $i \in \{\ell\}_{\ell=1}^{M_\sigma}$ *be the number of random samplings that are taken from each* i*th subregion* $\sigma_i(k)$ *at the* κ*th iteration, and let* $N(>0)$ *be the number of random trial points that are taken from the surrounding region* $S(\sigma(k))$*. Then*

$$p_k^{(k,k+1)} = \begin{cases} 1, & \text{if } k=0; \\ \varphi\, M_\sigma\, \gamma_k\, (1-\gamma_k)^{M_\sigma-1}, & \text{if } k \in \mathbb{N}_+, \end{cases} \tag{22}$$

where:

$$\varphi = \frac{M_\sigma \nu}{M_\sigma \nu + N}; \tag{23a}$$

$$\gamma_k = \frac{\prod_{\ell=1}^{n}\left(\frac{u^{(\ell)}-l^{(\ell)}}{2^{k+1}}\right) + \prod_{\ell=1}^{m}\left\lceil\frac{\bar{u}^{(\ell)}-\bar{l}^{(\ell)}}{2^{k+1}}\right\rceil}{\prod_{\ell=1}^{n}\left(\frac{u^{(\ell)}-l^{(\ell)}}{2^{k}}\right) + \prod_{\ell=1}^{m}\left\lceil\frac{\bar{u}^{(\ell)}-\bar{l}^{(\ell)}}{2^{k}}\right\rceil}, \ \forall\, k \in \mathbb{N}_+. \tag{23b}$$

Proof. Let $\mathcal{F}$ denote the event that the MINP method has identified that the best value of $f(z)$ has come from the current subregion $\sigma(k)$, let Σ_i denote the event that the best value of $f(z)$ has come from any ith subregion $\sigma_i(k)$ at the kth iteration, then,

$$p_k^{(k,k+1)} = Pr\{\Sigma_i, \mathcal{F}\}, \ \forall k \in \mathbb{N}_+. \tag{24}$$

Using conditional probability, we have

$$p_k^{(k,k+1)} = Pr\{\Sigma_i \mid \mathcal{F}\}\, Pr\{\mathcal{F}\}, \ \forall k \in \mathbb{N}_+. \tag{25}$$

Under Condition 3, we can say that letting $Pr\{\mathcal{F}\} = \varphi$

$$\varphi = \frac{M_\sigma \nu}{M_\sigma \nu + N}. \tag{26}$$

On the other hand, due to Conditions 2 and 3, and Sampling procedure in Figures 20 and 21, and since sampling trial points are taken from uniform random distributions for each subregion and the current surrounding region $S(\sigma(k))$, we can then say that (22) is easily got from a binomial distribution with M_σ subregions, because a set $\{\sigma_i(k)\}_{i=1}^{M_\sigma}$ is generated by Partition procedure in Figure 18 at each kth iteration, and parameter γ_k expressed by

$$\gamma_k = \frac{\mathcal{V}[\tilde{\sigma}_i(k)]}{\mathcal{V}[\tilde{\sigma}(k)]} = \frac{\prod_{\ell=1}^{n}\left(\frac{u^{(\ell)}-l^{(\ell)}}{2^{k+1}}\right) + \prod_{\ell=1}^{m}\left\lceil\frac{\bar{u}^{(\ell)}-\bar{l}^{(\ell)}}{2^{k+1}}\right\rceil}{\prod_{\ell=1}^{n}\left(\frac{u^{(\ell)}-l^{(\ell)}}{2^{k}}\right) + \prod_{\ell=1}^{m}\left\lceil\frac{\bar{u}^{(\ell)}-\bar{l}^{(\ell)}}{2^{k}}\right\rceil}, \tag{27}$$

for all $k \in \mathbb{N}_+$, because $\mathcal{V}[\tilde{\sigma_1}(k)] =, \cdots, = \mathcal{V}[\tilde{\sigma_{M_\sigma}}(k)]$, and hence $\gamma_k = \mathcal{V}[\tilde{\sigma_i}(k)]/\mathcal{V}[\tilde{\sigma}(k)]$ for any $i \in \{1,\ldots,M_\sigma\}$, what yields

$$Pr\{\Sigma_i \mid \mathcal{F}\} = \binom{M_\sigma}{1}\gamma_k(1-\gamma_k)^{M_\sigma-1}, \ \forall k \in \mathbb{N}_+. \tag{28}$$

By substituting (26) and (28) in (25), we obtain

$$p_k^{(k,k+1)} = \varphi\, M_\sigma \gamma_k\, (1-\gamma_k)^{M_\sigma-1}, \ \forall k \in \mathbb{N}_+. \tag{29}$$

Besides, the event that the MINPMC visits the $\{D=1\}$ state after it has been at the $\{D=0\}$ state always occurs, leaving no doubt, because $S(\sigma(0)) = \emptyset$, whereby $p_0^{(0,1)} = 1$. ■

Therefore, we shall hereafter denote, without causing a loss of its meaning, $Pr\{D(k) = \tilde{d}(k)\}$ as $Pr\{D(k) = k\} = \pi^{(k)}$.

We shall now pay especial attention to the probability distribution estimating of the MINPMC states, which may be expressed by a $(\hat{k}+1)$ dimensional vector $\pi = (\pi^{(0)},\ldots,\pi^{(\hat{k})})^t$, which depicts

$$\pi(k) = \sum_{\ell=0}^{\hat{k}} \pi^{(\ell)}\, \delta[k-\ell], \ \forall k \in \mathbb{Z}, \tag{30}$$

where $\delta[k] : \mathbb{Z} \to \{0,1\}$ is here called discrete impulse function, which is a special case of a 2-dimensional Kronecker delta function δ_{ij}, and a generalized ℓ-shifted mathematical expression of $\delta[k-\ell]$ is given by

$$\delta[k-\ell] = \begin{cases} 1, & \text{if } k=\ell; \\ 0, & \text{another case of } k \in \mathbb{Z}. \end{cases} \tag{31}$$

Note that we have used in (30) $\pi(\cdot)$ for denoting the probability distribution function, and this is not to be confused with the superscript argument notation $\pi^{(\cdot)}$ conventionally used for denoting the component of a vector, in this case, π vector.

Proposition 6 *Suppose that Conditions 2 and 3 hold, and let* $\pi^{(0)} = Pr\{D(0) = 0\}$ *be the probability that the MINPMC visits the state* $D(0) = \tilde{d}(0)$*, which of course always occurs by the starting of the MINP method; or by backtracking operations, which can be eventually carried out by the algorithm. Then*

$$\pi^{(\ell)} = \begin{cases} \pi^{(0)}, & \text{if } \ell \in \{0,1\}; \\ \pi^{(0)} \prod_{j=1}^{\ell-1} p_j^{(j,j+1)}, & \text{if } \ell \in \{2,\ldots,\hat{k}\}, \end{cases} \tag{32}$$

where:

$$\pi^{(0)} = \frac{1}{2 + \sum_{\ell=2}^{\hat{k}} \prod_{j=1}^{\ell-1} p_j^{(j,j+1)}}; \tag{33}$$

and $p_j^{(j,j+1)}$ *is given by* (22)*.*

Proof. By induction, we have

$$\pi^{(\ell)} = \pi^{(0)} \prod_{j=0}^{\ell-1} p_j^{(j,j+1)}, \forall \ell \in \mathbb{N}_+. \tag{34}$$

We besides know,

$$\sum_{\ell=0}^{\hat{k}} \pi^{(\ell)} = 1. \tag{35}$$

Substituting (34) in (35), and knowing that $p_0^{(0,1)} = 1$ because of (22), we therefore deduce

$$2\,\pi^{(0)} + \sum_{\ell=2}^{\hat{k}} \pi^{(0)} \prod_{j=1}^{\ell-1} p_j^{(j,j+1)} = 1. \tag{36}$$

Solving (36) for $\pi^{(0)}$, we effortlessly obtain

$$\pi^{(0)} = \frac{1}{2 + \sum_{\ell=2}^{\hat{k}} \prod_{j=1}^{\ell-1} p_j^{(j,j+1)}}. \tag{37}$$

Knowing that $p_j^{(j,j+1)}$ is estimated by Proposition 5, and using both (34) and (37) the proof is completed. ■

For almost ending this analysis, we apply the above results and (30) together for getting our mathematical expression of the probability distribution function of the MINPMC states, yielding

$$\pi(k) = \sum_{\ell=0}^{1} \pi^{(0)}\,\delta[k-\ell] + \sum_{\ell=2}^{\hat{k}} \pi^{(0)} \prod_{j=1}^{\ell-1} p_j^{(j,j+1)} \delta[k-\ell], \quad \forall k \in \mathbb{Z}. \tag{38}$$

The results heretofore obtained allow us to describe the behavior of the MINP method, which is a nonhomogeneous Markov chain with a conditional geometric distribution of states given by (38).

Finally, we shall make the follows statements for justifying our approach: By Proposition 4 and taking into account that $\sigma(k) \subseteq \tilde{\sigma}(k)$, we can assert that for each $k \in \mathbb{N}$, $\mathcal{V}[\tilde{\sigma}(k)] \geq \mathcal{V}[\sigma(k)]$, and besides, by (21) we have that $\mathcal{V}[\tilde{\sigma}(k)] \geq \mathcal{V}[\tilde{\sigma}(k+1)]$, what would allow us to say as a reasonable conjecture, that the MINP method concentrates its finding better promising regions during the progress of its iterative process, due to the fact of the continuous process of reduction of the baggy hull of the identified promising region.

vi. OPERATING THE MINP METHOD

In this section, we shall show a main software that will operate the MINP method for taking the collection of the performance measurements: $\eta^{(\ell)}(p,n,m)$; $\lambda^{(\ell)}(p,n,m)$; and $q^{(\ell)}(p,n,m)$ from each ℓth independent replication of the MINP method is depicted in Figure 10.

As is shown in Figure 10 on page 129, the MINP method is called for being r-times executed with different random seed each, thus achieving r independent runs for taking i.d.d. unknown function empirical distribution of the performance measurements, when it is used for globally solving Problem 1.

Main software of the Mixed Integer Nested Partitions

Given:

a pth bound constrained mixed integer nonlinear problem $\mathcal{P}_p$

$$\underset{z\in\mathbb{R}^n\times\mathbb{Z}^m}{\text{minimize}}\ f(z),$$

subject to $l \preceq z \preceq u$,

where $l, u \in \mathbb{R}^n \times \mathbb{Z}^m$;

the global minimum or considered true point $\hat{z}_p$ of the bounded constrained mixed integer nonlinear problem $\mathcal{P}_p$;

a maximum number of replication, r;

a set of S random generator, namely, $\Omega = \{u(s)\}_{s=\mathcal{I}(1)}^{\mathcal{I}(S)}$, which depends on the sth random seed $\mathcal{I}(s)$;

the $\mathrm{MINP}(s,p,n,m)$ algorithmic method;

Declare:

a counter replication, $\ell \in \{1,\ldots,r\}$;

an ℓth $\check{z}_\ell \in \mathbb{R}^n \times \mathbb{Z}^m$ point, which will be used for saving the best identified point by the ℓth running of the $\mathrm{MINP}(s,p,n,m)$ method;

an ℓth performance measure sample $q^{(\ell)}(p,n,m)$ for the $(n+m)$ multidimensional pth problem, which is given by

$$q^{(\ell)}(p,n,m) = \frac{1}{1+\eta^{(\ell)}(p,n,m)\cdot\lambda^{(\ell)}(p,n,m)}, \quad \forall \ell \in \mathbb{N}_+,$$

where: $\eta^{(\ell)}(p,n,m) \in \mathbb{N}$ is the number of times that has been evaluated the objective function during the ℓth running of the $\mathrm{MINP}(s,p,n,m)$ method; and $\lambda^{(\ell)}(p,n,m) = ||\hat{z} - \check{z}|| \in \mathbb{R}$ is the distance or norm between the true point or global optimum point $\hat{z}$ and the best point $\check{z}$ identified by the $\mathrm{MINP}(s,p,n,m)$;

for $\ell \leftarrow 1$ **to** r **do**

Choose a not used sth random seed for getting an i.i.d. random number generator for each ℓ replication;

Run the $\mathrm{MINP}(s,p,n,m)$ method for solving the $(n+m)$ multidimensional pth problem;

Compute the ℓth $q^{(\ell)}(p,n,m)$;

Save:

the ℓth of $\eta^{(\ell)}(p,n,m)$, $\lambda^{(\ell)}(p,n,m)$ and $q^{(\ell)}(p,n,m)$;

Estimate:

minimum, mode, mean, maximum, range, and deviation of $N(p,n,m)$, using the set of sampled $\{\eta^{(\ell)}(p,n,m)\}_{\ell=1}^{r}$;

minimum, mode, mean, maximum, range, and deviation of $L(p,n,m)$, using the set of sampled $\{\lambda^{(\ell)}(p,n,m)\}_{\ell=1}^{r}$;

minimum, mode, mean, maximum, range, and deviation of $Q(p,n,m)$, using the set of sampled $\{q^{(\ell)}(p,n,m)\}_{\ell=1}^{r}$;

Figure 10. Main software of the MINP for taking sampling

vii. NUMERICAL EXPERIMENTS

In this section, we shall summarize from a set of three numerical examples, which are described in Appendix A. The performance of the MINP method, and whose analysis will then be discussed latter for illustrating the eventual usefulness of the MINP method.

One of main noteworthy features of each one of following problems is the existence of 2^{n+m} local minima within its correspondent feasible region, and only one of them is a global minimum, whereby could result a challenge, because these problems are relatively difficult to identify their respective global minimum.

The experiments were conducted for: a number of random trial points per subregion $N_{\sigma_j(k)} = 6$; number of random trial points per surrounding region $N_{\sigma_{M+1}(k)} = 96$; and an expected maximum depth vector $\varepsilon = (\epsilon,\ldots,\epsilon;\bar{\epsilon},\ldots,\bar{\epsilon})^t$, where $\epsilon = 0.1$ and $\bar{\epsilon} = 0$.

a. GOLDSTEIN-PRICE PROBLEM

In this first example, we have taken into account an extension of Goldstein-Price function to the mixed integer Euclidean field $\mathbb{R}^2 \times \mathbb{Z}^2$, which has explicitly been defined in Appendix A. For this numerical experiment, we run $r = 100$ independent replications of the MINP method, using the software of Figure 10.

Table I. Summary of the MINP method, in which just 1 sample reached 53 cumulative iterations

	$f(z)$	$N(1,2,2)$	$L(1,2,2)$	$Q(1,2,2)$
Mean	108.65	2566.08	6.103	1.03e-2
SSD	107.49	1597.31	6.744	3.81e-2
Min	6.00	1056	0.001	9.29e-6
Q1	27.45	1056	1.058	4.44e-5
Q2	91.17	2064	3.185	1.26e-4
Q3	166.06	3528	9.107	5.34e-4
Max	696.09	8736	24.41	2.67e-1

As is shown in Table I, in a few replications the MINP method achieved to identify the global solution of the problem. In fact, according to the reported summary from the table, the $25\ \%$ of the replications reach to identify solutions to a distance less than 1.058. Moreover, as can be seen in Figure 11, the algorithm globally solved the problem as much as $16\ \%$ of the samplings. Besides, from Table I it may be concluded that the MINP method required, in average 2566 objective function evaluations for solving the problem, what could be considered as a good enough algorithmic method. Nevertheless, its quality performance measurement resulted to be very low, yielding a maximum value equal to 2.67×10^{-1}.

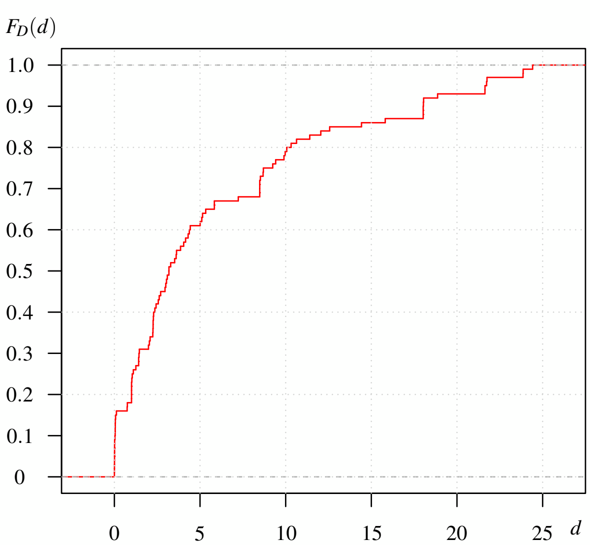

Figure 11. Empirical CDF of distance to the true point (DTP) for the Goldstein-Price problem.

As shown in Figure 11, it is depicted the empirical cumulative distribution function (CDF) of the random variable distance to the true point (DTP) or global solution point as a function of d. As can be seen from the figure, approximately a $25\ \%$ of the replications of the MINP method yielded good enough results, i.e., results less than 1.

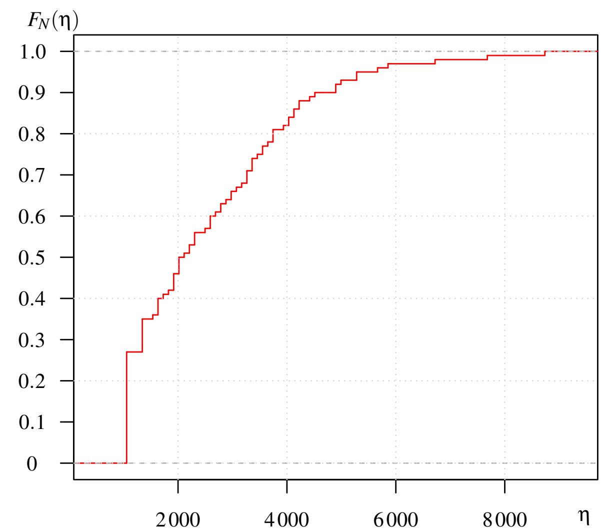

Figure 12. Empirical CDF of number of function evaluations (NE) for the Goldstein-Price problem.

Apart from that, Figure 12 shows the empirical CDF of the number of function evaluation (NE) random variable for the Goldstein-Price problem, which allows us to see the performance of the algorithm from the viewpoint of the NE.

b. W PROBLEM

Our second problem is a minimization problem of a objective function, which has been proposed by the author as a challenge in the mixed integer programming, and whose mathematical expression is given in Appendix A.

Table II presents our main performance measurements for the W problem, namely: $N(2,2,2)$, $L(2,2,2)$ and $Q(2,2,2)$. As is shown in the table, the MINP method spent in average 5512.32 function evaluations for stopping the iterative process, and the $25\ \%$ of replications achieved solutions to a distance to global minimum less than 20.84, in fact Figure 14 illustrates the value range of the NE. The table also shows the low values of quality performance $Q(2,2,2)$, that has been reached by the algorithm during the computational experimentation for this problem.

Table II. Summary of the MINP, in which just 1 sample reached 163 cumulative iterations

	$f(z)$	$N(2,2,2)$	$L(2,2,2)$	$Q(2,2,2)$
Mean	-95.30	5512.32	25.18	1.69e-5
SSD	524.23	4191.62	9.22	2.99e-5
Min	-453.59	2016	1.45	1.25e-6
Q1	-306.94	2808	20.84	5.49e-6
Q2	-215.29	4272	25.05	9.25e-6
Q3	-108.82	6672	32.23	1.45e-5
Max	4265.34	28032	44.54	1.99e-4

From Figure 13 it may be inferred that less than $10\ \%$ of the reported replications identified solutions to a distance to the true point less than 10.

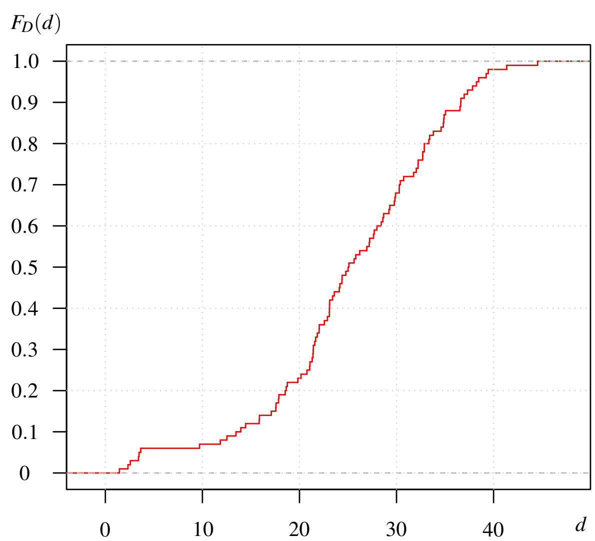

Figure 13. Empirical CDF of the DTP for the W function problem

Figure 14 shows cumulative distribution function of the NE when the MINP method is used for globally solving the W function problem. Note that the MINP at least required, in this case, about 28,000 objective function evaluations.

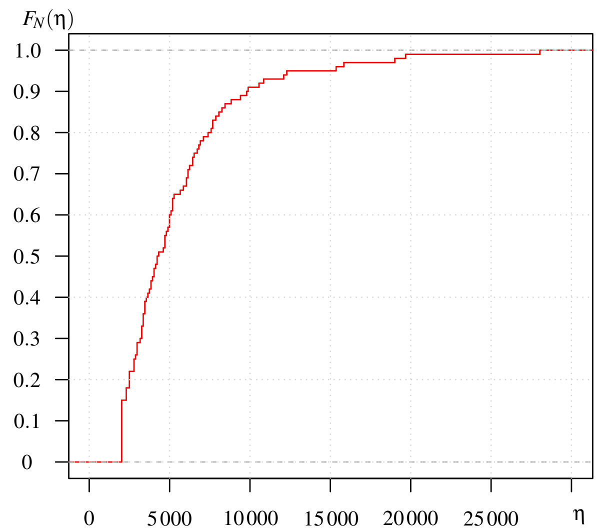

Figure 14. Empirical CDF of the NE for the W function problem

c. ICEBERG PROBLEM

The third problem of optimization is based on an unpublished objective function, whose mathematical expression is given in Appendix A.

Table III. Summary of the MINP, in which just 1 sample reached 101 cumulative iterations

	$f(z)$	$N(3,2,2)$	$L(3,2,2)$	$Q(3,2,2)$
Mean	-2649.00	4441.92	7.448	2.41e-4
SSD	583.91	3256.11	2.987	7.41e-4
Min	-3773.35	1440	0.005	6.07e-6
Q1	-3041.65	1920	6.135	2.32e-5
Q2	-2598.08	3504	8.368	4.07e-5
Q3	-2188.85	5736	9.472	7.99e-5
Max	-1326.02	16608	13.03	5.02e-3

Table III shows a statistical summary of the random variables: $N(3,2,2)$, $L(3,2,2)$ and $Q(3,2,2)$. As can be seen from the table, the MINP method reported good enough solutions, because the $25\ \%$ of replications identified solutions to the true point less than 6.135, and that can be verified from Figure 15, which illustrates the empirical CDF of the DTP for our problem.

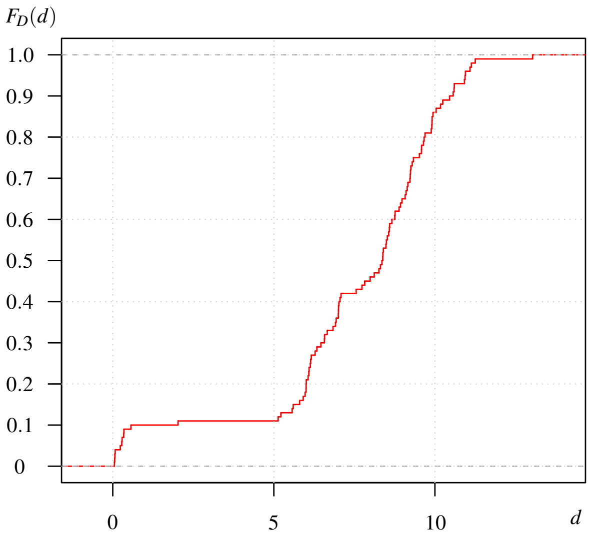

Figure 15. Empirical CDF of the DTP for the Iceberg problem

Figure 15 shows the empirical CDF of the DTP for the Iceberg problem. As can be seen from the figure, the MINP required less than 5000 objective function evaluations in the $70\,\%$ of replications for identifying at least a solution, which can be either a local or global solution.

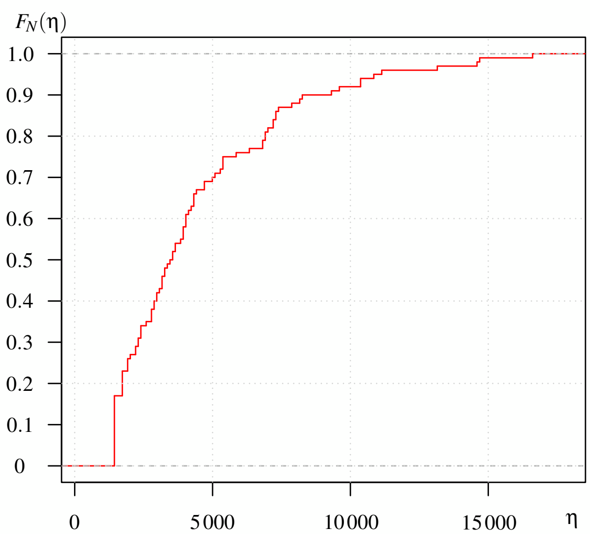

Figure 16. Empirical CDF of the NE for the Iceberg problem

Finally, Figure 16 allows us to infer the cost in function evaluations of the objective function, because as can be seen from the figure, the MINP method needed a high number of function evaluations for globally solving the Iceberg problem.

It is worthwhile pointing out that the MINP method was tested without having been tuned for this group of problems. However, some setting of its parameters were empirically fitted for improvement the performance of the algorithm, before running the numerical experiments.

viii. DISCUSSION AND FUTURE RESEARCH

The aim of this article has been to propose a new approach for globally solving bound constrained mixed integer nonlinear problems using, for reaching this target, the principles of the NP method viewpoint, namely: i) partitioning into subregions of the current promising region; ii) sampling scheme for obtaining random trial points from both each subregions and surrounding region to the current promising region; iii) locating of where has came from the best sampled trial point among all sampled trial points; and iv) testing of a stopping rule for making decision either executing a new iteration or finishing the iterative process of solving of the minimization problem. Nevertheless, heretofore this approach does not seem to have been effective enough, despite the theoretical foundations that have been developed in this research to reach our goal, if it is taken into account the results reported by Brea **[25]**, who carried out a comparative study among two implementations of Genetic Algorithms and the Game of Patterns in the n-dimensional real field.

Although, the MINP method has shown to be a powerful viewpoint for identifying promising regions, what would become a useful algorithmic procedure, and it could hence be hybridized with some local search algorithm, e.g., randomized pattern search algorithm **[10]**; pattern search algorithm **[26]**, because, the MINP method has experimentally shown to be effective enough for identifying promising regions, and hence with information of the promising region could be globally solved Problem 1. Besides, the approach that has been used in the MINP method could be easily parallelizable for encoding it in a parallel computer, what would be effective enough for finding global solutions to very large dimension mixed integer optimization problems.

This research has also raised several issues during the development of the MINP method. Among them, one can remark: i) the MINP method parameter tuning for improving it, for this target, one could hence use the viewpoint of Adenso-Díaz and Laguna **[27]**; ii) statistical analysis for the quality performance measurement applying non-standard parametric statistic, e.g., Cheng approach **[23]**; iii) the optimum quantity of random trial points from both each subregion and surrounding region as a function of the depth measurement, what could be solved by the approach of sampling budget, proposed by Chen and coworkers **[28]**, and including others approaches for improving the MINP method stopping rule using, e.g., Berkhout viewpoint **[29]**, who, applying the results of Chen, et al., **[28]**, presents a new and interesting approach for accelerating the NP method stopping rule.

On the other hand, we believe that the incorporation of criteria based on artificial intelligence (AI) for making decisions on sampling quantity for taking from each subregion and its surrounding region, it could be a large advance in the mixed integer programming.

Finally, as a future work we also propose a comparative study between the MINP method and the Game of Patterns algorithmic method, when they are applied for globally solving bound constrained mixed integer optimization problems.

A LIST OF PROBLEMS

We here present the objective functions of the test problems used in our numerical experiments taking into account the formulation of Problem 1. Besides, both lower and upper bound vectors have also been specified, and their respective global solutions.

a. EXTENDED GOLDSTEIN-PRICE PROBLEM

OBJECTIVE FUNCTION. Let $f(z):\mathbb{R}^{2n}\times\mathbb{Z}^{2m}\to\mathbb{R}$ be the Extended Goldstein-Price function, so called by the author, which is given by $f(z)=f(x)+f(y)$, where

$$\begin{aligned}f(x)=\sum_{i=0}^{2n-1}&\Big(1+(x^{(2i+1)}+x^{(2i+2)}+1)^2\cdot(19-14x^{(2i+1)}\\&+3x^{(2i+1)^2}-14x^{(2i+2)}+6\,x^{(2i+1)}x^{(2i+2)}+3\,x^{(2i+2)^2})\Big)\\&\cdot\Big(30+(2\,x^{(2i+1)}-3\,x^{(2i+2)})^2\Big(18-32\,x^{(2i+1)}+12\,x^{(2i+1)^2}\\&+48\,x^{(2i+2)}-36\,x^{(2i+1)}x^{(2i+2)}+27\,x^{(2i+2)^2}\Big)\Big),\end{aligned}\tag{39a}$$

and

$$\begin{aligned}f(y)=\sum_{i=0}^{2m-1}&\left(1+\left(\frac{y^{(2i+1)}}{10}+\frac{y^{(2i+2)}}{10}+1\right)^2\cdot\left(19-14\,\frac{y^{(2i+1)}}{10}\right.\right.\\&+3\left(\frac{y^{(2i+1)}}{10}\right)^2-14\,\frac{y^{(2i+2)}}{10}+6\,\frac{y^{(2i+1)}}{10}\,\frac{y^{(2i+2)}}{10}\\&\left.\left.+3\left(\frac{y^{(2i+2)}}{10}\right)^2\right)\right)\cdot\left(30+\left(2\,\frac{y^{(2i+1)}}{10}-3\,\frac{y^{(2i+2)}}{10}\right)^2\right.\\&\cdot\left(18-32\,\frac{y^{(2i+1)}}{10}+12\left(\frac{y^{(2i+1)}}{10}\right)^2+48\,\frac{y^{(2i+2)}}{10}\right.\\&\left.\left.-36\,\frac{y^{(2i+1)}}{10}\,\frac{y^{(2i+2)}}{10}+27\left(\frac{y^{(2i+2)}}{10}\right)^2\right)\right).\end{aligned}\tag{39b}$$

BOUND CONSTRAINTS. Let $l,u\in\mathbb{R}^{2n}\times\mathbb{Z}^{2m}$ be the bound constraints given by

$$l=(-2.5,\ldots,-2.5;-25,\ldots,-25)^t;\tag{40a}$$
$$u=(2.0,\ldots,2.0;20,\ldots,20)^t.\tag{40b}$$

OPTIMUM SOLUTION. The unique global minimum is located at

$$z^t=(\underbrace{0,-1,\ldots,0,-1}_{2n};\underbrace{0,-10,\ldots,0,-10}_{2m}),\tag{41}$$

and

$$f(\hat{z})=3(n+m)\tag{42}$$

b. W PROBLEM

OBJECTIVE FUNCTION. Let $f(z):\mathbb{R}^{n}\times\mathbb{Z}^{m}\to\mathbb{R}$ be the W function, so called by the author, which is given by

$$f(z)\sum_{i=1}^{n}\left(\frac{x^{(i)}}{4}\right)^4-\left(x^{(i)}-2\right)^2+\sum_{i=1}^{m}\left(\frac{y^{(i)}}{4}\right)^4-\left(y^{(i)}-2\right)^2\tag{43}$$

BOUND CONSTRAINTS. Let $l,u\in\mathbb{R}^{n}\times\mathbb{Z}^{m}$ be the bound constraints given by

$$l=(-100,\ldots,-100)^t;\tag{44a}$$
$$u=(100,\ldots,100)^t.\tag{44b}$$

SOLUTION. The unique global minimum is located at $\hat{z} = (\hat{x}^t; \hat{y}^t)^t$, where

$$\hat{x}^t = (\underbrace{-12.2054969669241, \ldots, -12.2054969669241}_{n}); \tag{45a}$$

$$\hat{y}^t = (\underbrace{-12, \ldots, -12}_{m}), \tag{45b}$$

and

$$f(\hat{z}) = -115.1035690056\, n - 115\, m. \tag{46}$$

c. ICEBERG PROBLEM

OBJECTIVE FUNCTION. Let $f(z) : \mathbb{R}^n \times \mathbb{Z}^m \to \mathbb{R}$ be the Iceberg function, so called by the author, which is given by

$$f(z) = \sum_{i=1}^{n} \left(x^{(i)^4} - \alpha \sin(x^{(i)}) \right) + \sum_{i=1}^{m} \left(y^{(i)^4} - \beta \sin(y^{(i)}) \right), \tag{47}$$

where $\alpha = \beta = 1000$.

BOUND CONSTRAINTS. Let $l, u \in \mathbb{R}^n \times \mathbb{Z}^m$ be the bound constraints given by

$$l = (-10, \ldots, -10)^t; \tag{48a}$$

$$u = (10, \ldots, 10)^t. \tag{48b}$$

SOLUTION. The unique global minimum is located at $\hat{z}^t = (\underbrace{1.55573432449541; \ldots; 1.55573432449541}_{n}; \underbrace{2, \ldots, 2}_{m})$, and $f(\hat{z}) = -994.028673136238\, n - 893.297426825682\, m$.

B PSEUDOCODE OF PROCEDURES

Procedure Surrounding(n, m, Θ, σ)

Given:

The number of real components, n;
The number of integer components, m;
The bounded feasible region Θ, namely:

$$l^{(\ell)} \le x^{(\ell)} \le u^{(\ell)}, \quad \forall \ell \in \{1, \ldots, n\},$$
$$\bar{l}^{(\ell)} \le y^{(\ell)} \le \bar{u}^{(\ell)}, \quad \forall \ell \in \{1, \ldots, m\};$$

A promising region $\sigma \subseteq \Theta$, defined by

$$l_\sigma^{(\ell)} \le x^{(\ell)} \le u_\sigma^{(\ell)}, \quad \forall \ell \in \{1, \ldots, n\},$$
$$\bar{l}_\sigma^{(\ell)} \le y^{(\ell)} \le \bar{u}_\sigma^{(\ell)}, \quad \forall \ell \in \{1, \ldots, m\};$$

Declare:

A matrix $\check{X} = [\check{x}^{(ij)}]_{i=1,j=1}^{4,n} \in \mathbb{R}^{4\times n}$ for saving real boundary components of the set $S(\sigma(k))$;
A matrix $\check{Y} = [\check{y}^{(ij)}]_{i=1,j=1}^{4,m} \in \mathbb{Z}^{4\times m}$ for saving integer boundary components of the set $S(\sigma(k))$;

for $\ell \leftarrow 1$ **to** n **do**
 $\check{x}^{(1,\ell)} \leftarrow l^{(\ell)}$;
 $\check{x}^{(2,\ell)} \leftarrow l_\sigma^{(\ell)}$;
 $\check{x}^{(3,\ell)} \leftarrow u_\sigma^{(\ell)}$;
 $\check{x}^{(4,\ell)} \leftarrow u^{(\ell)}$;

for $\ell \leftarrow n+1$ **to** $n+m$ **do**
 $\check{y}^{(1,\ell-n)} \leftarrow \bar{l}^{(\ell)}$;
 $\check{y}^{(2,\ell-n)} \leftarrow \bar{l}_\sigma^{(\ell)}$;
 $\check{y}^{(3,\ell-n)} \leftarrow \bar{u}_\sigma^{(\ell)}$;
 $\check{y}^{(4,\ell-n)} \leftarrow \bar{u}^{(\ell)}$;

Figure 17. Surrounding procedure

Procedure. Partitioning$(n, m, \Theta, \sigma, d(k))$

Given:

the number of real components, n;
the number of integer components, m;
the bounded feasible region Θ, namely:

$$l^{(i)} \leq x^{(i)} \leq u^{(i)}, \quad \forall i \in \{1, \ldots, n\},$$
$$\bar{l}^{(i)} \leq y^{(i)} \leq \bar{u}^{(i)}, \quad \forall i \in \{1, \ldots, m\};$$

the promising region $\sigma \subseteq \Theta$ to be partitioned, which is defined by

$$l_\sigma^{(i)} \leq x^{(i)} \leq u_\sigma^{(i)}, \quad \forall i \in \{1, \ldots, n\},$$
$$\bar{l}_\sigma^{(i)} \leq y^{(i)} \leq \bar{u}_\sigma^{(i)}, \quad \forall i \in \{1, \ldots, m\};$$

the mixed integer depth vector of the current promising region $\sigma(k)$ to be partitioned
$d^t(k) = (\underbrace{d^{(1)}(k), \ldots, d^{(n)}(k)}_{n}; \underbrace{\bar{d}^{(n+1)}(k), \ldots, \bar{d}^{(n+m)}(k)}_{m});$

Let $M_\sigma = 2^{n+m}$ be the number of subregions to be denoted by $\{\sigma_j\}_{j=1}^{M_\sigma}$;

Declare:

a matrix $X = [x^{(ij)}]_{i=1,j=1}^{2M_\sigma,n} \in \mathbb{R}^{2M_\sigma \times n}$ for saving real boundary components of the set $\{\sigma_j\}_{j=1}^{M_\sigma}$;
a matrix $Y = [y^{(ij)}]_{i=1,j=1}^{2M_\sigma,m} \in \mathbb{Z}^{2M_\sigma \times m}$ for saving integer boundary components of the set $\{\sigma_j\}_{j=1}^{M_\sigma}$;

Let $q = 1$ be the counter of subregion σ_j;

Initialization

Let $i = 1$;

for $q \leftarrow 0$ **to** $M_\sigma - 1$ **do**
  $(b^{(n+m-1)}, b^{(n+m-2)}, \ldots, b^{(0)})_2 \leftarrow c(q)$, where $c(q)$ is a convertor function, which transforms decimal numbers to binary numbers, namely, $(b^{(n+m-1)}, b^{(n+m-2)}, \ldots, b^{(0)})_2 \in \{0,1\}^{n+m}$;
  for $\ell \leftarrow 1$ **to** n **do**
    if $u_\sigma^{(\ell)} - l_\sigma^{(\ell)} > \varepsilon^{(\ell)}$ **then**
      $\delta^{(\ell)} = (l_\sigma^{(\ell)} + u_\sigma^{(\ell)})/2;$
      if $b^{(\ell-1)} = 0$ **then**
        $x^{(i,\ell)} \leftarrow l_\sigma^{(\ell)};$
        $x^{(i+1,\ell)} \leftarrow \delta^{(\ell)};$
      else
        $x^{(i,\ell)} \leftarrow \delta^{(\ell)};$
        $x^{(i+1,\ell)} \leftarrow u_\sigma^{(\ell)};$
  for $\ell \leftarrow n+1$ **to** $n+m$ **do**
    if $\bar{u}_\sigma^{(\ell)} - \bar{l}_\sigma^{(\ell)} > \bar{\varepsilon}^{(\ell)}$ **then**
      $\delta^{(\ell)} = (\bar{l}_\sigma^{(\ell)} + \bar{u}_\sigma^{(\ell)})/2;$
      if $(b^{(\ell-1)} = 0) \wedge (\lfloor \delta^{(\ell)} \rfloor = \lceil \delta^{(\ell)} \rceil)$ **then**
        $y^{(i,\ell-n)} \leftarrow \bar{l}_\sigma^{(\ell)};$
        $y^{(i+1,\ell-n)} \leftarrow \delta^{(\ell)};$
      if $(b^{(\ell-1)} = 1) \wedge (\lfloor \delta^{(\ell)} \rfloor = \lceil \delta^{(\ell)} \rceil)$ **then**
        $y^{(i,\ell-n)} \leftarrow \delta^{(\ell)} + 1;$
        $y^{(i+1,\ell-n)} \leftarrow \bar{u}_\sigma^{(\ell)};$
      if $(b^{(\ell-1)} = 0) \wedge (\lfloor \delta^{(\ell)} \rfloor \neq \lceil \delta^{(\ell)} \rceil)$ **then**
        $y^{(i,\ell-n)} \leftarrow \bar{l}_\sigma^{(\ell)};$
        $y^{(i+1,\ell-n)} \leftarrow \lfloor \delta^{(\ell)} \rfloor;$
      if $(b^{(\ell-1)} = 1) \wedge (\lfloor \delta^{(\ell)} \rfloor \neq \lceil \delta^{(\ell)} \rceil)$ **then**
        $y^{(i,\ell-n)} \leftarrow \lceil \delta^{(\ell)} \rceil;$
        $y^{(i+1,\ell-n)} \leftarrow \bar{u}_\sigma^{(\ell)};$
  Let $i \leftarrow i + 2$

Figure 18. Partitioning procedure

Preamble of Sampling and the Measuring of the Objective Function

Given:

The number of real components, n;
The number of integer components, m;

Let $M_\sigma \leftarrow 2^{n+m}$;

Given:

the number of random sample for being taken from each jth subregion $\sigma_j(k)$, N_j;

the number of random sample for being taken from the surrounding region $S(\sigma(k)) = \sigma_{M_\sigma+1}(k)$, N;

the real boundaries of each jth subregion $\sigma_j(k)$ for each kth iteration, which must be read from the real matrix $X = [x^{(j\ell)}]_{j=1,\ell=1}^{2M_\sigma,n}$;

the integer boundaries of each jth subregion $\sigma_j(k)$ for each kth iteration, which must be read from the integer matrix $Y = [y^{(j\ell)}]_{j=1,\ell=1}^{2M_\sigma,m}$;

the real boundaries of the surrounding region $S(\sigma(k))$ for each kth iteration, which must be read from the real matrix $\breve{X} = [\breve{x}^{(ij)}]_{i=1,j=1}^{4,n} \in \mathbb{R}^{4\times n}$;

the integer boundaries of the surrounding region $S(\sigma(k))$ for each kth iteration, which must be read from the integer matrix $\breve{Y} = [\breve{y}^{(ij)}]_{i=1,j=1}^{4,m} \in \mathbb{Z}^{4\times m}$;

a convertor function $c(q)$, which converts any $q \in \mathbb{N}$ number, which is given by its decimal representation, to its binary representation, namely, $q = (b^{(\lceil \lg_2(q)\rceil - 1)}, \ldots, b^{(0)})_2 \in \{0,1\}^{\lceil \lg_2(q)\rceil}$;

Declare:

the best current point $\hat{z}^t = (\hat{x}^{(1)}, \ldots, \hat{x}^{(n)}; \hat{y}^{(n+1)}, \ldots, \hat{y}^{(n+m)}) \in \mathbb{R}^n \times \mathbb{Z}^m$;

the index of the best performance of the objective function, given by

$$\hat{I}(\sigma_j(k)) = \min_{s\in\{1,\ldots,N_j\}} f(z_{s,j}),$$

where $z_{s,j}$ denotes the sth mixed integer sample point, which has been taken from the subregion $\sigma_j(k)$;

Choose:

an nth index seed, namely, $n \in \mathcal{N}$, which depends on the sth random seed $\mathcal{I}(s)$;
an $N_s \in \mathbb{N}_+$ number of sampling per sector of surrounding region $\sigma(k)$ at the kth iteration;

if $k = 0$ **then Assign** the current objective function value $\hat{f}(\hat{z})$, the largest possible value that can be represented in an x-bit computer;

Figure 19. Preamble of the sampling procedure

Procedure Sampling of Surrounding and Promising Region and the Measuring of the Objective Function (Part i)

Let $\hat{i} \leftarrow 0$;
if $k \geq 1$ **then**
  for $\ell \leftarrow 0$ **to** $2^n - 2$ **do**
    for $\bar{\ell} \leftarrow 0$ **to** $2^m - 2$ **do**
      $(b^{(n-1)}, \ldots, b^{(0)})_2 \leftarrow c(\ell)$;
      $(\bar{b}^{(m-1)}, \ldots, \bar{b}^{(0)})_2 \leftarrow c(\bar{\ell})$;
      for $r \leftarrow 1$ **to** N_s **do**
        for $j \leftarrow 1$ **to** n **do**
          if $b^{(j-1)} = 0$ **then**
            $x^{(j)} \leftarrow \mathrm{u}(\check{x}^{(2,j)}, \check{x}^{(3,j)}, s)$
          else
            $x^{(j)} \leftarrow \mathrm{w}(\check{x}^{(1,j)}, \check{x}^{(2,j)}, \check{x}^{(3,j)}, \check{x}^{(4,j)}, s)$
        for $j \leftarrow 1$ **to** m **do**
          if $\bar{b}^{(j-1)} = 0$ **then**
            $y^{(j)} \leftarrow \bar{\mathrm{u}}(\check{y}^{(2,j)}, \check{y}^{(3,j)}, s)$
          else
            $y^{(j)} \leftarrow \overline{\mathrm{w}}(\check{y}^{(1,j)}, \check{y}^{(2,j)}, \check{y}^{(3,j)}, \check{y}^{(4,j)}, s)$
        Let $z_s^t \leftarrow (\underbrace{x^{(1)}, \ldots, x^{(n)}}_{n}; \underbrace{y^{(n+1)}, \ldots, y^{(n+m)}}_{m})$
        Measure the objective function by $f(z)|_{z_s}$
        if $f(z_s) < \hat{f}(\hat{z})$ **then**
          Let $\hat{f}(\hat{z}) \leftarrow f(z_s)$;
          Let $\hat{z} \leftarrow z_s$;
          Let $\hat{i} \leftarrow M_\sigma + 1$;
  for $d \leftarrow 1$ **to** 2 **do**
    switch d **do**
      case d*=1* **do**
        Let $\ell = 2^n - 1$;
        Let $\bar{\ell} = 2^m - 2$;
      case d*=2* **do**
        Let $\ell = 2^n - 2$;
        Let $\bar{\ell} = 2^m - 1$;
    $(b^{(n-1)}, \ldots, b^{(0)})_2 \leftarrow c(\ell)$;
    $(\bar{b}^{(m-1)}, \ldots, \bar{b}^{(0)})_2 \leftarrow c(\bar{\ell})$;
    for $r \leftarrow 1$ **to** N_s **do**
      for $j \leftarrow 1$ **to** n **do**
        if $b^{(j-1)} = 0$ **then**
          $x^{(j)} \leftarrow \mathrm{u}(\check{x}^{(2,j)}, \check{x}^{(3,j)}, s)$
        else
          $x^{(j)} \leftarrow \mathrm{w}(\check{x}^{(1,j)}, \check{x}^{(2,j)}, \check{x}^{(3,j)}, \check{x}^{(4,j)}, s)$
      for $j \leftarrow 1$ **to** m **do**
        if $\bar{b}^{(j-1)} = 0$ **then**
          $y^{(j)} \leftarrow \bar{\mathrm{u}}(\check{y}^{(2,j)}, \check{y}^{(3,j)}, s)$
        else
          $y^{(j)} \leftarrow \overline{\mathrm{w}}(\check{y}^{(1,j)}, \check{y}^{(2,j)}, \check{y}^{(3,j)}, \check{y}^{(4,j)}, s)$
      Let $z_s^t \leftarrow (\underbrace{x^{(1)}, \ldots, x^{(n)}}_{n}; \underbrace{y^{(n+1)}, \ldots, y^{(n+m)}}_{m})$
      Measure the objective function by $f(z)|_{z_s}$
      if $f(z_s) < \hat{f}(\hat{z})$ **then**
        Let $\hat{f}(\hat{z}) \leftarrow f(z_s)$;
        Let $\hat{z} \leftarrow z_s$;
        Let $\hat{i} \leftarrow M_\sigma + 1$;

Figure 20. Sampling procedure, part i

Procedure Sampling of Surrounding and Promising Region and the Measuring of the Objective Function (Part ii)

for $i \leftarrow 0$ **to** $M_\sigma - 1$ **do**
- **for** $r \leftarrow 1$ **to** N_s **do**
  - **for** $j \leftarrow 1$ **to** n **do**
    - **if** $d^{(j)} \leq \varepsilon^{(j)}$ **then**
      - $x^{(j)} \leftarrow \hat{x}^{(j)}$
    - **else**
      - $x^{(j)} \leftarrow \mathrm{u}(x^{(2i+1,j)}, x^{(2i+2,j)}, s)$
  - **for** $j \leftarrow 1$ **to** m **do**
    - **if** $\bar{d}^{(j)} \leq \bar{\varepsilon}^{(j)}$ **then**
      - $y^{(j)} \leftarrow \hat{y}^{(j)}$
    - **else**
      - $y^{(j)} \leftarrow \bar{\mathrm{u}}(y^{(2i+1,j)}, y^{(2i+2,j)}, s)$
  - **Let** $z_s^t \leftarrow (\underbrace{x^{(1)}, \ldots, x^{(n)}}_{n}; \underbrace{y^{(n+1)}, \ldots, y^{(n+m)}}_{m})$
  - **Measure** the objective function by $f(z)|_{z_s}$
  - **if** $f(z_s) < \hat{f}(\hat{z})$ **then**
    - **Let** $\hat{f}(\hat{z}) \leftarrow f(z_s)$;
    - **Let** $\hat{z} \leftarrow z_s$;
    - **Let** $\hat{i} \leftarrow i$;
- **Let** $i \leftarrow i + 1$;

switch $\hat{i}$ **do**
- **case** $\hat{i} = 0$ **do**
  - **Backtrack** to entire feasible region Θ;
  - **Update** the depth vector d by using $d(0) = \mathcal{D}(0, n, m, \sigma(k+1), \Theta)$;
  - **Let** $k \leftarrow 0$;
- **other wise do**
  - **Let** $\sigma(k+1) = \sigma_{\hat{i}}(k)$;
  - **Update** by using $d(k+1) = \mathcal{D}(k+1, n, m, \sigma(k+1), \Theta)$;
  - **Let** $k \leftarrow k + 1$;

Figure 21. Sampling procedure, part ii

C PSEUDOCODE OF FUNCTIONS

Function Real Double Uniform(a_1, b_1, a_2, b_2, n)

Given:
- $a_1 \in \mathbb{R}$: a lower real bound of the first interval uniform random distribution;
- $b_1 \in \mathbb{R}$: an upper real bound of the first interval uniform random distribution;
- $a_2 \in \mathbb{R}$: a lower real bound of the second interval uniform random distribution;
- $b_2 \in \mathbb{R}$: an upper real bound of the second interval uniform random distribution;
- $n \in \mathcal{N}$: a nth seed from an available pseudorandom number generator set;
- $\mathrm{u}(n) \in (0, 1)$: a uniformly distributed random number between 0 and 1 from the nth index seed, namely, $\{\mathrm{u}(s)\}_{s=\mathcal{I}(1)}^{\mathcal{I}(S)}$, which depends on the sth random seed $\mathcal{I}(s)$;

Output: *a uniformly distributed random real number belonging to two disjunct intervals, namely:* (a_1, b_1) *or* (a_2, b_2) *;*

Function $\mathrm{w}(a_1, b_1, a_2, b_2, n)$**:**
- **Calculate** $\theta \in \mathbb{R}$ and $x \in \mathbb{R}$, namely:

  $$\theta = \frac{b_1 - a_1}{b_1 + b_2 - a_1 - a_2};$$

  $$x = a_1 + (b_1 + b_2 - a_1 - a_2)\,\mathrm{u}(n);$$
- **if** $\theta \leq \mathrm{u}(n) < 1$ **then**
  - **Let** $x \leftarrow x + a_2 - b_1$;
- **return** x;

Figure 22. Double real uniform distribution function

Function Integer Double Uniform($\bar{a}_1, \bar{b}_1, \bar{a}_2, \bar{b}_2, n$)

Given:

$\bar{a}_1 \in \mathbb{Z}$: a lower integer bound of the first interval uniform random distribution;
$\bar{b}_1 \in \mathbb{Z}$: an upper integer bound of the first interval uniform random distribution;
$\bar{a}_2 \in \mathbb{Z}$: a lower integer bound of the second interval uniform random distribution;
$\bar{b}_2 \in \mathbb{Z}$: an upper integer bound of the second interval uniform random distribution;
$n \in \mathcal{N}$: a nth seed from an available pseudorandom number generator set;
$\mathrm{u}(n) \in (0,1)$: a uniformly distributed random number between 0 and 1 from the nth index seed, namely, $\{\mathrm{u}(s)\}_{s=\mathcal{I}(1)}^{\mathcal{I}(S)}$, which depends on the sth random seed $\mathcal{I}(s)$;

Output: *a uniformly distributed random integer number belonging to two disjunct intervals, namely:* $[\bar{a}_1, \bar{b}_1]$ *or* $[\bar{a}_2, \bar{b}_2]$ *;*

Function $\overline{\mathrm{w}}(\bar{a}_1, \bar{b}_1, \bar{a}_2, \bar{b}_2, n)$**:**

Calculate: $\theta \in \mathbb{R}$ and $x \in \mathbb{R}$, namely:

$$\theta = \frac{1 + \bar{b}_1 - \bar{a}_1}{2 + \bar{b}_1 + \bar{b}_2 - \bar{a}_1 - \bar{a}_2};$$

$$x = \bar{a}_1 + (2 + \bar{b}_1 + \bar{b}_2 - \bar{a}_1 - \bar{a}_2)\,\mathrm{u}(n);$$

if $\theta \leq u(n) < 1$ **then**
  Let $x \leftarrow x + \bar{a}_2 - \bar{b}_1 - 1$;
return $\lfloor x \rfloor$;

Figure 23. Double integer uniform distribution function

Function depth $\mathcal{D}(k, n, m, \sigma(k), \Theta)$

Given:

The iteration counter, k;
The number of real components, n;
The number of integer components, m;
The kth promising region $\sigma(k) \subseteq \Theta$, which is then defined by

$$l_\sigma^{(i)} \leq x^{(i)} \leq u_\sigma^{(i)}, \quad \forall i \in \{1, \dots, n\},$$
$$\bar{l}_\sigma^{(i)} \leq y^{(i)} \leq \bar{u}_\sigma^{(i)}, \quad \forall i \in \{1, \dots, m\};$$

the bounded feasible region Θ, namely:

$$l^{(i)} \leq x^{(i)} \leq u^{(i)}, \quad \forall i \in \{1, \dots, n\},$$
$$\bar{l}^{(i)} \leq y^{(i)} \leq \bar{u}^{(i)}, \quad \forall i \in \{1, \dots, m\};$$

Declare:

$d = (\underbrace{d^{(1)}, \dots, d^{(n)}}_{n}; \underbrace{\bar{d}^{(n+1)}, \dots, \bar{d}^{(n+m)}}_{m})^t \in \mathbb{R}^n \times \mathbb{Z}^m$;

Output: *an updated depth vector* d*;*

Function $\mathcal{D}(k, n, m, \sigma(k), \Theta)$**:**

switch k **do**
  case $k = 0$ **do**
    for $j \leftarrow 1$ **to** n **do**
      $d^{(j)} \leftarrow u^{(j)} - l^{(j)}$;
    for $j \leftarrow n+1$ **to** $n+m$ **do**
      $\bar{d}^{(j)} \leftarrow \bar{u}^{(j)} - \bar{l}^{(j)}$;
  other wise do
    for $j \leftarrow 1$ **to** n **do**
      $d^{(j)} \leftarrow u_\sigma^{(j)} - l_\sigma^{(j)}$;
    for $j \leftarrow n+1$ **to** $n+m$ **do**
      $\bar{d}^{(j)} \leftarrow \bar{u}_\sigma^{(j)} - \bar{l}_\sigma^{(j)}$;
return d

Figure 24. Depth function

ACKNOWLEDGMENTS

This work was supported in part by the Centro de Investigación y Desarrollo de Ingeniería(CIDI) of the Universidad Católica Andrés Bello. Special thanks are due to Professor María I. López, Director of the CIDI for her efforts in obtaining the resources for this research. He also thanks Mr. Rolando Andrade for coding the algorithm under C++, and the two anonymous reviewers for very helpful comments and suggestions for clarifying the presentation of this article.

REFERENCES

[1] Floudas, C. A. Nonlinear and mixed-integer optimization: fundamentals and applications. New York, Oxford University Press, 1995.

[2] Grossmann, I. E. and Z. Kravanja,"Mixed-Integer Nonlinear Programming: A Survey of Algorithms and Applications", pp. 73–100. New York, NY: Springer New York, June 1997. https://doi.org/10.1007/978-1-4612-1960-6_5

[3] Grossmann, I. E. "Review of Nonlinear Mixed-Integer and Disjunctive Programming Techniques", Optimization and Engineering, vol. 3, no. 3, pp. 227–252, 2002. https://doi.org/10.1023/A:1021039126272

[4] Tawarmalani, M. and N. V. Sahinidis, Convexification and global optimization in continuous and mixed-integer nonlinear programming: theory, algorithms, software, and applications. Nonconvex optimization and its applications; v. 65, Dordrecht Boston: Kluwer Academic Publishers, 2002.

[5] Tawarmalani, M. and N. V. Sahinidis, "Global optimization of mixed-integer nonlinear programs: A theoretical and computational study", Mathematical Programming, vol. 99, no. 3, pp. 563–591, 2004. https://doi.org/10.1007/s10107-003-0467-6

[6] Brea, E. Extensiones del método de Nelder Mead a problemas de variables enteras y enteras mixtas, tech. rep., Universidad Central de Venezuela, Septiembre 2009. Trabajo de Ascenso para categoría de Profesor Titular.

[7] Brea, E. "Una extensión del método de Nelder Mead a problemas de optimización no lineales enteros mixtos", Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería, vol. 29, pp. 163–174, July 2013. https://doi.org/10.1016/j.rimni.2013.06.005

[8] Brea, E. (2016 July) On the Performance of the Mixed Integer Randomized Pattern Search Algorithm, in The 13th International Congress on Numerical Methods in Engineering and Applied Sciences (Y. González, E. Dávila, V. Duarte, M. Candal, O. Pelliccioni, J. Darías, and M. Cerrolaza, eds.), vol. 1, (Caracas, Venezuela), pp. Op 61–Op 72.

[9] Brea, E. (2017 April). Game of Patterns: An approach for solving mixed integer nonlinear optimization problems, in International Congress on Industrial Engineering and Operations Management (IEOM 2017), Bogota, Colombia. http://ieomsociety.org/bogota2017/proceedings

[10] Brea, E. "On the mixed integer randomized pattern search algorithm", Revista de la Unión de Matemáticos Argentinos, vol. 60, pp. 485-503, September 2019. https://doi.org/10.33044/revuma.v60n2a14

[11] Kantor, I., J.-L. Robineau, H. Bütün, and F. Maréchal, "A mixed-integer linear programming formulation for optimizing multi-scale material and energy integration", Frontiers in Energy Research, vol. 8, pp. 49.1–49.20, April 2020. https://doi.org/10.3389/fenrg.2020.00049

[12] Jalota, H. and M. Thakur, "Genetic Algorithm Designed for Solving Linear or Nonlinear Mixed-Integer Constrained Optimization Problems", International Proceedings on Advances in Soft Computing, Intelligent Systems and Applications, (Singapore), pp. 277–290, Springer Singapore, December 2018. https://doi.org/10.1007/978-981-10-5272-9_27

[13] Deep, K., K. P. Singh, M. Kansal, and C. Mohan, "A real coded genetic algorithm for solving integer and mixed integer optimization problems", Applied Mathematics and Computation, vol. 212, pp. 505–518, June 2009. https://doi.org/10.1016/j.amc.2009.02.044

[14] Schlüter, M., J. A. Egea, and J. R. Banga, "Extended ant colony optimization for non-convex mixed integer nonlinear programming", Computers & Operations Research, vol. 36, pp. 2217–2229, July 2009. https://doi.org/10.1016/j.cor.2008.08.015

[15] Mohammadi, M., S. N. Musa, and A. Bahreininejad, "Optimization of mixed integer nonlinear economic lot scheduling problem with multiple setups and shelf life using metaheuristic algorithms", Advances in Engineering Software, vol. 78, pp. 41–51, December 2014. https://doi.org/10.1016/j.advengsoft.2014.08.004

[16] Cardoso, M. F., R. L. Salcedo, S. F. de Azevedo, and D. Barbosa, "A simulated annealing approach to the solution of MINLP problems", Computers & Chemical Engineering, vol. 21, pp. 1349–1364, April 1997. https://doi.org/10.1016/S0098-1354(97)00015-X

[17] Gupta, O. K. and A. Ravindran, "Branch and bound experiments in convex nonlinear integer programming" Management Science, vol. 31, pp. 1533–1546, December 1985. http://www.jstor.org/stable/2631793

[18] Shi, L. and S. Ólafsson, "Nested Partitions method for global optimization", Operations Research, vol. 48, pp. 390–407, May 2000. https://doi.org/10.1287/opre.48.3.390.12436

[19] Xia, L., Y. Zhao, M. Xie, J. Shao, and J. Dong, "Mixed integer programming based nested partition algorithm for facility location optimization problems", in 2008 IEEE International Conference on Service Operations and Logistics, and Informatics, vol. 2, pp. 2375–2381, October 2008.

[20] Shi, L. and S. Ólafsson, Nested Partitions Method, Theory and Applications. No. 978-0-387-71909-2 in International Series in Operations Research and Management Science, Springer, 1st ed., December 2009.
https://doi.org/10.1007/978-0-387-71909-2

[21] IBM, ILOG® CPLEX® Optimization Studio interfaces, August 2021. https://www.ibm.com/analytics/

[22] Shi, L. and S. Ólafsson, "Stopping rules for the stochastic Nested Partitions method", Methodology and Computing in Applied Probability, vol. 2, pp. 37-58, April 2000.
https://doi.org/10.1023/A:1010055101140

[23] Cheng, R. Non-standard parametric statistical inference. Oxford, England: Oxford University Press, first ed., 2017.

[24] de Haan, L. "Estimation of the Minimum of a Function Using Order Statistics", Journal of the American Statistical Association, vol. 76, pp. 467–469, June 1981. http://www.jstor.org/stable/2287851

[25] Brea, E. (2019 January). Game of Patterns and Genetic Algorithms under a comparative study, in Hybrid Metaheuristics, HM2019 (M. B. Aguilera, C. Blum, H. G. Santos, P. Pinacho-Davidson, and J. G. del Campo, eds.), vol. 11299, (Concepcion, Chile), pp. 93–107, Springer, Cham.
https://doi.org/10.1007/978-3-030-05983-5_7

[26] Audet, C. and J. E. Dennis Jr., "Pattern Search Algorithms for Mixed Variable Programming", SIAM Journal on Optimization, vol. 11, pp. 573–594, July 2001.
https://doi.org/10.1137/S1052623499352024

[27] Adenso-Díaz, B. and M. Laguna, "Fine-Tuning of Algorithms Using Fractional Experimental Designs and Local Search", Operations Research, vol. 54, pp. 99–114, February 2006.
https://doi.org/10.1287/opre.1050.0243

[28] Chen, C.-H., D. He, M. Fu, and L. H. Lee, "Efficient Simulation Budget Allocation for Selecting an Optimal Subset", INFORMS Journal on Computing, vol. 20, pp. 579–595, May 2008.
https://doi.org/10.1287/ijoc.1080.0268

[29] Berkhout, J. "An accelerated stopping rule for the Nested Partition Hybrid Algorithm for discrete stochastic optimization", Discrete Event Dynamic Systems, vol. 25, pp. 441–452, September 2015.
https://doi.org/10.1007/s10626-014-0191-9