

Extremal Optimization Applied to the Minimum Order Frequency Assignment Problem

Pedro Gomez-Meneses , Héctor Ayala

Abstract—Extremal Optimization (EO) is a metaheuristic used to solve complex combinatorial optimization problems. However, the behavior of this algorithm has not been studied for various interesting problems. Such as, it is the case of the Minimum Order Frequency Assignment Problem (MO-FAP). Specifically, this problem seeks to minimize the number of frequencies assigned to a group of transceiver, without violating any of the problem's constraints. In particular, the aim of this study was to use EO to solve the MO-FAP. Furthermore, it was found that EO was able to find the best reported value for most instances of the dataset used for experimentation. Additionally, it was possible to identify how the EO τ hyperparameter reported better results for much higher values than those reported in the literature. Finally, the difficulty of defining and configuring a fitness function for each element of the solution was evidenced. Therefore, this can affect the effectiveness of the algorithm according to the particular characteristics of the dataset to be solved. Consequently, this opens an opportunity to develop improvements in the form of defining the contribution of each element of the solution to the objective function.

Index Terms—Frequency assignment problem, Computational intelligence, Evolutionary computation, Extremal optimization, Meta-heuristic algorithms.

I. INTRODUCCIÓN

Los algoritmos metaheurísticos, en general, son recomendados para solucionar problemas combinatorios del tipo NP-Hard, ya que han demostrado funcionar mejor, en tiempo y uso de recursos computacionales, que los métodos exhaustivos o los voraces [1]–[3]. Dentro de las técnicas metaheurísticas más comunes y estudiadas se pueden encontrar los Algoritmos Genéticos [4], Optimización por Colonia de Hormigas [5], Optimización por Partículas de Enjambres [6], Recocido Simulado [7], Búsqueda Tabú [8], Programación Genética [9], Evolución Diferencial [10] entre otras. Por otro lado, el teorema de No Hay Almuerzo Gratis [11] postula que no existe un único algoritmo lo suficientemente bueno para resolver todos los tipos de problemas de optimización que existen. Por este motivo, se está constantemente proponiendo nuevos algoritmos o aplicando metaheurísticas ya existentes a otro tipos de problemas no antes explorados [12]–[15].

Entre las metaheurísticas no tan comúnmente estudiadas se puede encontrar a Extremal Optimization (EO), propuesta por Boettcher y Percus [16] y que está inspirada en el modelo de co-evolución de Bak-Sneppen [17]. En el último tiempo, EO ha surgido como una interesante alternativa para solucionar problemas complejos de optimización debido a su simplicidad

de implementación y configuración, como también, a su capacidad de poder escapar del estancamiento de algún óptimo local dentro del espacio de búsqueda. Por este motivo, se ha estado generando una diversidad de estudios para observar el comportamiento y aplicabilidad de EO a nuevos tipos de problemas. Así, EO ha sido utilizado principalmente para dar solución a problemas de optimización del tipo combinatorial. Entre los problemas más representativos se pueden nombrar algunos como, partición y coloreo de grafos [18]–[20], diseño de tubos de calor [21], detección de comunidades en redes complejas [22], problema de máxima satisfactibilidad (MAX-SAT) [23], problema del vendedor viajero [24], planificación de producción de acero [25], problema de compromiso de unidades para sistemas de potencia [26], asignación de recursos espaciales [27], diseño de ubicación de pozos [28], problema de vidrio de spín [29], entre otros.

La asignación de frecuencias es un problema de optimización combinatoria en el que un grupo de diferentes usuarios, que requieren establecer conexiones inalámbricas con cierta calidad predefinida, compiten por el uso del limitado espectro electromagnético disponible. La importancia de este tipo de problemas es su aplicabilidad en el mundo de las telecomunicaciones donde se deben realizar periódicamente asignaciones de frecuencias para el uso de sus transceptores. Por ejemplo, los dispositivos inalámbricos debe administrar el espectro electromagnético de modo de que no se produzcan interrupciones ni largos periodos de espera entre un número creciente de usuarios [30], [31]. En el contexto militar, se requiere realizar asignaciones dinámicas en tiempo real para la utilización de teléfonos de campaña que funcionen en medio de situaciones adversas [32]. En la comunicación satelital, los satélites son susceptibles a interferencia co-canal cuando se mueven en la misma órbita [33]. Por último, las redes Wireless LAN deben distribuir las frecuencias de las que disponen, coexistiendo con otras redes Wireless LAN, considerando obstáculos físicos y la demanda de conexión de diversos aparatos tecnológicos [34].

Un problema de asignación de frecuencias es un problema NP-Hard [35]–[37], el cual, ha sido abordado por diversas técnicas metaheurísticas, como también de métodos exactos. Entre los principales métodos que han sido utilizados podemos nombrar a los Algoritmos Genéticos [38], [39], el Templado Simulado [40], [41], la Búsqueda Tabú [42], [43], la Programación Lineal [44], Algoritmos Voraces [45], [46], Redes Neuronales [47], [48] entre otros. En esta investigación se va a trabajar con el modelo de asignación de frecuencias MO-FAP, el cuál busca minimizar el número de frecuencias asignadas a un grupo de transceptores. Para ello se va a utilizar

Pedro Gomez-Meneses, Universidad Católica de la Santísima Concepción (UCSC), Concepción, Chile. (e-mail: pgomez@ucsc.cl).

Héctor Ayala, Universidad Católica de la Santísima Concepción (UCSC), Concepción, Chile. (e-mail: hayala@ucsc.cl).

el conjunto de datos de pruebas CELAR y GRAPH que está disponible en el proyecto EUCLID CALMA [49]. Entre los trabajos que han propuesto metaheurísticas para dar solución a este conjunto de datos del MO-FAP se puede mencionar a los Algoritmos Genéticos [50], Búsqueda Evolutiva [51], Recocido Simulado [52] y Búsqueda Tabú [43], [52]–[55].

Este estudio tiene principalmente dos propósitos. En primer lugar, evaluar el desempeño de la versión canónica de EO en un nuevo dominio de aplicación. Así, en la literatura no se encontró que esta metaheurística haya sido usada para solucionar algún problema de asignación de frecuencias tal como el MO-FAP. En segundo lugar, enriquecer los trabajos relacionados con los problemas de asignación de frecuencias como MO-FAP al proponer un nuevo enfoque metaheurístico como es EO. Ambos propósitos contribuyen en ampliar la frontera del conocimiento en sus respectivas áreas.

El resto del documento está organizado de la siguiente forma. La Sección II aborda la definición de MO-FAP, la descripción de EO y se expone como se aplica EO para resolver el MO-FAP. Aquí se explica el modelamiento del MO-FAP como un ecosistema en EO, la definición de la función de aptitud y la especificación del mecanismo de selección y reemplazo de EO. En la Sección III se describe la sintonización del parámetro τ de EO y de las constantes de peso de la función de aptitud, como también, se muestran los experimentos realizados y se discuten los resultados obtenidos. Finalmente, en la Sección IV se presentan las conclusiones y discusión de los trabajos futuros que surgen de este estudio.

II. MÉTODOS

A. Minimum Order Frequency Assignment Problem

En un problema de asignación de frecuencias se tiene un conjunto de conexiones V que están conformadas por enlaces inalámbricos entre dos trancceptores y que operaran utilizando frecuencias de un conjunto de canales D . Para cada conexión $v \in V$ se requiere asignar una cantidad mínima de frecuencias m_v , conocida como *multiplicidad de v* , que deben estar comprendidas en el subconjunto de frecuencias válidas para dicha conexión $D_v \subseteq D$. Así, m_v es la cantidad mínima de frecuencias necesarias para el funcionamiento adecuado de la conexión y dependerá de cada problema en particular.

Durante el proceso de asignación, se debe evitar asignar frecuencias cercanas en el espectro electromagnético a conexiones que estén físicamente próximas. Estas limitaciones se pueden representar como un grafo de restricciones $G = (V, E)$. En este grafo cada vértice representa una conexión $v \in V$, y cada arista $\{v_i, v_j\} \in E$ representa una interferencia entre las frecuencias usadas en sus dos vértices.

Para cada par de frecuencias $f \in D_{v_i}$ y $g \in D_{v_j}$, existe una penalización p_{fv_i, gv_j} , que corresponde a una cuantificación del nivel de interferencia mutua en el que incurrirían dichas frecuencias en caso de ser asignadas a las conexiones v_i y v_j , respectivamente. Esta penalización tiene estrecha relación con la distancia mínima permitida entre las frecuencias escogidas [32], de la cual se obtiene la matriz de compatibilidad C , donde cada elemento C_{v_i, v_j} corresponde a la distancia mínima de separación entre f y g .

De esta manera, el objetivo del MO-FAP es lograr asignar el mínimo número de frecuencias comprendidas en D de tal manera de satisfacer todas las conexiones de V sin incurrir en interferencias inaceptables.

En MO-FA, una interferencia inaceptable se puede dar por cualquiera de los siguientes tipos de restricciones:

1. *Restricciones bidireccionales*: Para un par de conexiones v_i y v_j , las frecuencias utilizadas f y g respectivamente, deben estar a una distancia fija y pre-establecida de separación, es decir, $|f - g| = d_{v_i, v_j}$.
2. *Restricciones de interferencia*: Para un par de conexiones v_i y v_j , las frecuencias utilizadas f y g respectivamente, deben estar a una distancia mínima de separación, es decir, $|f - g| > d_{v_i, v_j}$.
3. *Restricciones de dominio*: Este tipo de restricciones corresponden a los subdominios de frecuencias posibles para cada conexión; si f es asignada a v_i , debe cumplirse que $f \in D_{v_i}$, $D_{v_i} \subseteq D$.
4. *Pre-asignaciones de frecuencias*: Algunas conexiones pueden contar con una frecuencia pre-asignada y su modificación podría no ser posible.

B. Extremal Optimization

EO es un algoritmo metaheurístico inspirado en la naturaleza, el cual, aplica la auto-organización crítica, una propiedad de los sistemas dinámicos, para modelar la co-evolución de las especies a través del tiempo [16]. La auto-organización crítica establece que ciertos sistemas presentan largos periodos de estabilidad, seguidos por cortos y abruptos periodos de inestabilidad [17]. Esto se traduce en que el algoritmo esté mejorando de forma constante una solución candidata hasta que ésta cambia de forma abrupta y repentina en su calidad. Este comportamiento permite explotar soluciones de mayor calidad como una búsqueda local. Como también le permite escapar de los posibles óptimos locales con un cambio radical de la solución que le facilita explorar el espacio de búsqueda de una nueva área de potenciales soluciones.

En EO varias especies conviven, cada una con un determinado valor de aptitud, el cual representa cuán apta es la especie dentro del ecosistema. En cada generación se elige estocásticamente una especie para ser reemplazada de acuerdo a la distribución de potencia mostrada en la ecuación 1.

$$P_K \propto K^{-\tau}, 1 \leq K \leq n \quad (1)$$

Donde K representa a una especie de un total de n especies ordenadas desde la peor a la mejor según su valor de aptitud. Esto propicia que especies con un valor de aptitud malo tengan una mayor probabilidad de ser seleccionados para ser reemplazados. Sin embargo, igual existe una baja probabilidad de que una especie bien evaluada sea elegida. La ocurrencia de este fenómeno permite emular a los cambios abruptos presentes en la auto-organización crítica.

El valor de τ es el único parámetro que requiere ser establecido previamente. Un valor de $\tau = 0$ hará que el algoritmo se comporte como una búsqueda randómica al otorgarle a cada especie la misma posibilidad de ser eliminada. Bajo un valor $\tau \rightarrow \infty$, el algoritmo se comportará como una búsqueda local

voraz que sólo eliminará la peor especie, perdiendo así la capacidad de explorar. El proceso de EO es mostrado en el Algoritmo 1, para más información de EO referirse a [56].

Algoritmo 1 EO canónico para un problema de minimización.

```

1:  $S \leftarrow \text{SolucionInicial}()$ 
2:  $S_{best} \leftarrow S$ 
3: for  $i = 1$  to  $\text{num\_iteraciones}$  do
4:   for  $j = 1$  to  $\text{longitud}(S)$  do
5:      $\text{aptitud\_elemento}_j \leftarrow \text{Fitness}(S_j)$ 
6:   end for
7:    $\text{OrdenarElementos}(S, \text{aptitud\_elemento})$ 
8:    $\text{elemento}_{weak} \leftarrow \text{SeleccionarConProbabilidad}P_K(S)$ 
9:    $\text{elemento}_{new} \leftarrow \text{SeleccionarRemplazo}(S)$ 
10:   $S \leftarrow \text{Remplazo}(S, \text{elemento}_{weak}, \text{elemento}_{new})$ 
11:  if  $\text{Costo}(S) \leq \text{Costo}(S_{best})$  then
12:     $S_{best} \leftarrow S$ 
13:  end if
14: end for
15: return  $S_{best}$ 

```

C. Implementación de EO para Solucionar el MO-FAP

En esta sección se describe cómo se modela el MO-FAP de acuerdo a la filosofía de funcionamiento de EO.

C1. Representación: En primer lugar, se procede a representar el MO-FAP como un ecosistema a través de una estructura de datos unidimensional. Así, el conjunto de conexiones V está representado como un vector, donde cada posición corresponde a una conexión v particular, que en el contexto de EO, representa a una especie. El valor almacenado en cada posición es la frecuencia asignada a la conexión v_i , siendo i la posición de vector (ver la Fig. 1).

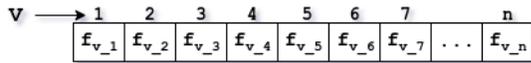


Fig 1. Arreglo de conexiones y frecuencias.

Cabe resaltar que EO trabaja sólo con una solución que va cambiando en cada iteración del proceso de búsqueda. De manera diferente de lo que ocurre con un Algoritmo Genético que trabaja en base a la interacción de una población de soluciones distintas.

En EO, cada especie participa en la *competencia* dentro del ecosistema por no ser eliminado. Para ello, es necesario calcular un valor de *aptitud* individual para cada especie o componente del vector solución. De esta manera, cada especie va actualizando en cada iteración sus respectivas cifras asociadas con respecto a su valor de aptitud, al número de restricciones violadas y a la cantidad de veces que ha sido utilizada la frecuencia que le fue asignada.

C2. Función de Aptitud: La particularidad y dificultad de la función de aptitud en EO está en el hecho de que ésta se debe calcular para elemento de la solución y no para la solución completa, cómo ocurre en la mayoría de los métodos meta-heurísticos. Por este motivo, se tomó como base una función de aptitud usada para un Algoritmo Genético en [50], la cual se modificó y simplificó para adaptarla a la evaluación individual de EO, quedando así, definida en la ecuación 2.

$$f(v_i) = w_1 U(f_{v_i}) + w_2 B(v_i) + w_3 I(v_i), \quad \forall v_i \in V \quad (2)$$

El primer término en la función $U(f_{v_i})$, calcula el número de veces que ha sido usada la frecuencia f_{v_i} en la solución. El término está normalizado, por lo que solo puede tomar valores comprendidos desde cero, si la misma frecuencia es usada en todas las conexiones, hasta uno en el caso de que la frecuencia no sea usada en ninguna otra conexión.

El segundo y tercer término, $B(v_i)$ e $I(v_i)$, corresponden a la cantidad de restricciones bidireccionales y de interferencia violadas respectivamente. Ambos términos pueden tomar valores comprendidos entre cero, en caso de que no ocurran violaciones, y n en caso de que se violen todas las restricciones, siendo n el número total de restricciones bidireccionales o de interferencia presentes en el problema.

Como lo primero que se busca es entrar en un espacio de soluciones factibles, la función de aptitud valorizará peor a aquellas especies que presenten una alta cantidad de restricciones violadas, en desmedro del número de veces que han sido usadas las frecuencias que tienen asignadas. Esto será así en cuanto el número de violaciones permanezca por sobre un umbral de uno. Cuando se cae por debajo de dicho valor es cuando el primer término adquiere la misma relevancia numérica (o peso) en relación al segundo y tercero, aumentando su relevancia para así buscar bajar el número de frecuencias usadas en la asignación. Las frecuencias que menos se repitan propiciarán que cada especie que haga uso de ellas sea peor valorada, y por lo tanto, que tenga mayor probabilidad de ser remplazada en la siguiente iteración.

Los pesos w_1, w_2, w_3 son constantes reales y toman valores entre cero y uno. Son tal que $w_1 + w_2 + w_3 = 1$ ya que representan una proporción de la importancia de cada término en el valor final, pudiendo ser ajustados para dar mayor predominio a cualquiera de ellos.

C3. Selección, Reemplazo y Condición de Término: La elección de la especie más débil se realiza utilizando el método de selección de la ruleta, según la distribución de probabilidades vista en la ecuación 1. Con el valor de aptitud individual ya calculado para cada elemento $v_i \in V$ utilizando la ecuación 2; se procede a ordenar las especies desde la peor a la mejor evaluada. Así, mientras mayor sea el valor numérico de la función de aptitud, peor es la evaluación de la especie y por ende tiene una mayor probabilidad de ser eliminada.

Una vez seleccionada la especie que va a ser eliminada de la solución, esta es reemplazada por una nueva especie de forma aleatoria dentro de las frecuencias pertenecientes al dominio de la variable; es decir, el reemplazo se realiza cuidando de no violar una restricción de dominio.

La condición de término del algoritmo puede ser definida como un número máximo de iteraciones, un tiempo máximo de ejecución o un periodo máximo en que no se obtenga mejoras. Para esta investigación se utilizará como criterio de término un número de iteraciones máximo. Inicialmente, se utilizará un valor de 200.000 iteraciones para las instancias en que la propuesta de EO sea más eficaz, para después ampliarlo a 2.000.000 para las instancias más complejas. Esto, con el objetivo de estudiar el comportamiento del algoritmo para este tipo de instancias con el objetivos que generar futuras mejoras.

C4. Conjunto de Datos de Prueba: El conjunto de datos utilizado para probar el algoritmo propuesto son CELAR y

GRAPH, utilizados en proyecto EUCLID CALM. Estos se obtiene del sitio web *FAP web - A website about Frequency Assignment Problems* que está alojado en el Zuse Institute Berlin [49]. A partir de este conjunto de datos se seleccionan 10 instancias, con distintas características y complejidad, con el objetivo de observar el comportamiento del algoritmo propuesto en esta investigación. Las instancias elegidas y sus características se pueden observar en la Tabla I.

TABLA I
DESCRIPCIÓN DE LAS INSTANCIAS DE CELAR Y GRAPH.

Instancia	N°C	N°RB	N°RI	N°RD	N°FPA	N°TR	N°FSO
CELAR 01	916	458	5.090	916	0	6464	16
CELAR 02	200	100	1.135	200	0	1435	14
CELAR 03	400	200	2.560	400	0	3160	14
CELAR 04	680	340	3.627	400	280	4647	46
CELAR 11	680	340	3.763	680	0	4783	22
GRAPH 01	200	100	1.034	200	0	1334	18
GRAPH 02	400	200	2.045	400	0	2645	14
GRAPH 08	680	340	3.417	680	0	4437	18
GRAPH 09	916	458	4.788	916	0	6162	18
GRAPH 14	916	458	4.180	916	0	5554	08

N°C: Número de Conexiones.

N°RB: Número de Restricciones Bidireccionales.

N°RI: Número de Restricciones de Interferencias.

N°RD: Número de Restricciones de Dominio.

N°FPA: Número de Frecuencias Pre-Asignadas.

N°TR: Número Total de Restricciones.

N°FSO: Número de Frecuencias de la Solución Óptima.

Cada instancia se ejecuta con 30 valores de semillas diferentes, para posteriormente tabular y graficar su comportamiento.

III. EXPERIMENTOS Y RESULTADOS

Las pruebas son realizadas en un servidor DELL Poweredge R815 con 4 procesadores AMD Opteron 6278 de 8 núcleos, corriendo a 2400 Mhz y con 192 GB en memoria RAM. El lenguaje y compilador utilizado fue GNU FORTRAN 2008 sobre un sistema operativo Ubuntu Server 20.04.

En la Fig. 2 se muestra un esquema gráfico que permite visualizar y entender el proceso llevado a cabo en para la sintonización de parámetros y ejecución de pruebas para la obtención de los resultados de análisis. Así, en las siguientes secciones se procede a describir en detalle dicho proceso.

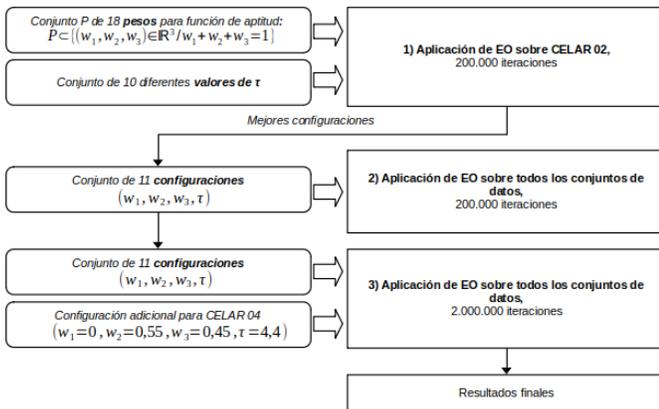


Fig 2. Diagrama del proceso de sintonización de parámetros y pruebas.

A. Sintonización de Parámetros EO y Función de Aptitud

Los primeros experimentos tienen relación con la configuración del parámetro τ de EO y los pesos w_1, w_2, w_3 de la función de aptitud. Para el parámetro τ se probaron valores entre 1,4 a 6,8 variando en 0,6. Los pesos de la función objetivo fueron seleccionados a partir de datos experimentales con la instancia CELAR 02. Para ello se definieron 18 diferentes permutaciones de valores (que por simplicidad, han sido denominadas con una letra del alfabeto), que en conjunto a la variación de τ , conformaron 180 diferentes configuraciones.

Los resultados de estos experimentos permitieron definir 11 configuraciones, para los valores del parámetro τ y de los pesos de la función de aptitud, que alcanzaron el número de frecuencias de la solución óptima con una desviación estándar de cero. Estas configuraciones se muestran en la Tabla II, la cuales, posteriormente fueron usadas para extrapolar los experimentos al resto de los conjuntos de prueba.

TABLA II
MEJORES VALORES DE τ Y CONFIGURACIÓN DE PESOS EN CELAR 02.

τ	config.	w_1	w_2	w_3	frecuencias		iteración	
					(\bar{x})	(σ)	(\bar{x})	(σ)
5,0	D	0,6	0,2	0,2	14	0	41.027	28.438
3,8	D	0,6	0,2	0,2	14	0	41.131	23.141
4,4	D	0,6	0,2	0,2	14	0	45.100	30.215
3,8	N	0,2	0,4	0,4	14	0	45.333	33.920
4,4	N	0,2	0,4	0,4	14	0	48.622	33.820
5,0	G	0,6	0,1	0,3	14	0	55.332	40.946
4,4	G	0,6	0,1	0,3	14	0	71.200	35.253
4,4	M	0,4	0,2	0,4	14	0	71.320	48.566
5,0	N	0,2	0,4	0,4	14	0	72.321	43.965
3,8	M	0,4	0,2	0,4	14	0	74.206	50.854
5,0	R	0,2	0,2	0,6	14	0	81.065	43.998

De este trabajo de definición de parámetros es posible desprender las siguientes observaciones.

1. Los pesos w_2 y w_3 tienen igual valor para más de la mitad de las permutaciones con mejores resultados, teniendo en la otra mitad, a excepción de la configuración R, una diferencia pequeña de solo 0,2 (ver Tabla II). De esto se puede deducir que una diferenciación en el tipo de violación no acarrea alguna ventaja significativa dentro de la función de aptitud utilizada.
2. Se advierte un mejor desempeño con el uso de valores de τ comprendidos entre 3,8 y 5,0.

La segunda observación se relaciona con el comportamiento de la búsqueda bajo diferentes valores de τ . En la Fig. 3 se ilustra la variación de estos resultados, utilizando la configuración de pesos D como ejemplo. Los valores de τ bajo 2,6 no fueron capaces de dar soluciones factibles, ilustrándose esto en la Fig. 3a. En la Fig. 3b se muestra en qué iteración se logró el mejor resultado. Descartando los valores bajos de τ , que en el gráfico anterior mostraron inviabilidad, se observa una convergencia más veloz bordeando valores entre 3,8 y 5,0. La Fig. 3c muestra el valor de la función objetivo contra τ . A medida que se aumenta el valor de τ , se observa una disminución continua en las frecuencias usadas por la asignación. Esta pendiente negativa se hace presente hasta el valor de 3,8 aproximadamente, donde se mantiene estable en el óptimo hasta alcanzar el valor de 5,0.

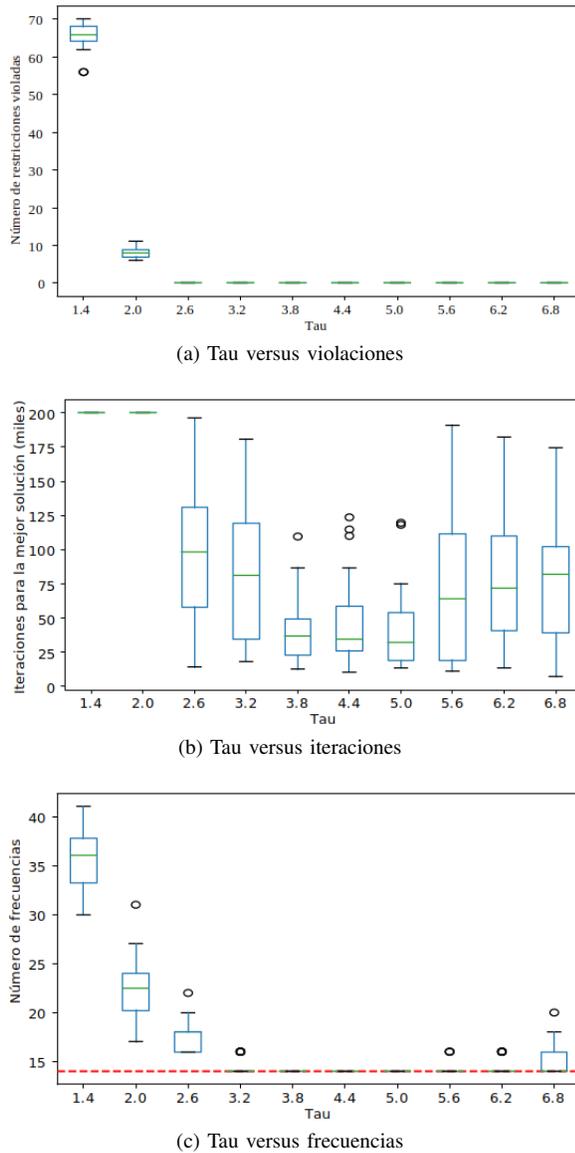


Fig 3. Efecto de τ en la configuración D sobre CELAR 02.

B. Resultados

Se aplican las configuraciones de la Tabla II a todas las instancias con una condición de término de 200.000 iteraciones.

Para las instancias CELAR 02, GRAPH 01, GRAPH 02 y GRAPH 08 se logra obtener el mejor resultado conocido en la literatura. Para las tres primeras, en todas las pruebas se alcanza con facilidad la zona de soluciones factibles y en una gran mayoría de las veces se iguala al mejor resultado conocido. Sin embargo, en la instancia GRAPH 08, a pesar de que es capaz de encontrar al mejor resultado conocido, esto ocurrió sólo para cuatro de las configuraciones utilizadas.

Para las instancias CELAR 01, CELAR 03, GRAPH 09 y GRAPH 14 se logra alcanzar con relativa facilidad la zona de soluciones factibles, sin embargo no se pudo obtener el mejor resultado conocido. En el mejor de los casos, sólo faltó disminuir en dos unidades el número de frecuencias asignadas. Ahora, en el caso de la instancia GRAPH 09, ésta se comportó igual a GRAPH 08, en el sentido de que se tiene una mayor

dificultad en alcanzar la zona de soluciones factibles.

Para las instancias CELAR 04 y CELAR 11, no se logra encontrar una solución factible en un número significativo de casos. Si bien hay una disminución continua de las violaciones y de las frecuencias usadas al avanza en la búsqueda, la zona de factibilidad fue tocada en experimentos aislados. Cabe destacar el caso CELAR 04, donde la solución inicial usualmente contaba con una menor cantidad de frecuencias que la solución óptima. Este comportamiento deriva en una dificultad para entrar a una zona de soluciones factible, ya que la función de aptitud propicia la disminución de las frecuencias usadas, haciendo difícil la adopción de nuevas frecuencias para cumplir con todas las restricciones. Una idea para arreglar este comportamiento es la utilización de pesos diferentes que no den importancia al primer término de la función de aptitud.

De estos resultados, para 200.000 iteraciones, se pueden desprender las siguientes observaciones:

1. La elección de CELAR 02 para probar las configuraciones de pesos resulta insuficiente y no necesariamente representativa para las demás instancias. Para las instancias GRAPH 08 y GRAPH 09, los mejores resultados se logran con configuraciones donde peso sobre las restricciones bidireccionales es menor al peso de las de interferencia. Este resultado se contrapone a lo apuntado en la primera observación en la Sección A.
2. Continuando la idea del primer punto y tomando el caso de CELAR 04, dependiendo de la naturaleza del problema, se puede hacer necesario utilizar una distribución de pesos *a medida* para ciertas instancias.
3. En todas las instancias se observa una mejora continua sobre la solución inicial sin quedarse estancado en un óptimo local, lo que sugiere que una mayor exploración pueda obtener mejores resultados.

Partiendo de la tercera observación, con el subconjunto de configuraciones que presentaron los mejores resultados, se decide probar el incrementar el límite de la condición de término, a intervalo de 200.000 iteraciones, para analizar la progresión del mejor resultado encontrado por el algoritmo propuesto. La mejora continua sobre la solución inicial se mantiene hasta la condición de término de manera constante, como se evidencia en la Fig 4. Por un tema de tiempo de procesamiento se fija el límite superior de la condición de término en 2.000.000 de iteraciones.

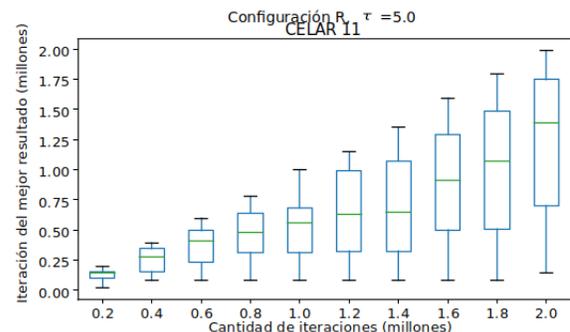


Fig 4. CELAR 11, iteración donde se encontró el mejor resultado versus criterio de término en intervalos de 200.000 iteraciones.

TABLA III
MEJORES RESULTADOS DE EO.

Instancia	τ	Conf.	Solución Óptima	200.000 Iteraciones						2M Iteraciones					
				frecuencia			violaciones			frecuencia			violaciones		
				min	med	max	min	med	max	min	med	max	min	med	max
CELAR 01	3,8	D	16	18	20	22	0	0	0	16	18	20	0	0	0
CELAR 02	5	D	14	14	14	14	0	0	0	14	14	14	0	0	0
CELAR 03	4,4	D	14	16	18	20	0	0	0	14	16	16	0	0	0
CELAR 04	4,4	G	46	44	46	46	46	181	477	44	44	46	0	25,5	260
CELAR 11	3,8	M	22	32	38	46	2	2	4	28	32	42	0	2	2
GRAPH 01	3,8	D	18	18	18	20	0	0	0	18	18	18	0	0	0
GRAPH 02	4,4	M	14	14	14	16	0	0	0	14	14	14	0	0	0
GRAPH 08	3,8	M	18	18	22	26	0	0	0	18	18	20	0	0	0
GRAPH 09	5	R	18	20	24	32	0	0	0	18	18	20	0	0	0
GRAPH 14	5	G	8	10	10	18	0	0	0	8	10	10	0	0	0

TABLA IV
CONFIGURACIÓN ALTERNATIVA CON MEJOR RESULTADO PARA CELAR 04.

Instancia	τ	Conf.	Mejor Resultado Conocido	200.000 Iteraciones						2M Iteraciones					
				frecuencia			violaciones			frecuencia			violaciones		
				min	med	max	min	med	max	min	med	max	min	med	max
CELAR 04	4,4	Z	46	44	46	46	0	329	588	45	46	46	0	1	148

Como se puede apreciar en la Tabla III, tras el aumento en el número de iteraciones se observa una mejora en el desempeño global. Con respecto a las restricciones, se logra alcanzar la zona factible en todas las instancias a excepción de CELAR 04 y CELAR 11, donde solo ocurrió con algunos experimentos. Ahora, en relación a minimizar en número de frecuencias utilizadas, también se logra una mejora global significativa, obteniendo el mejor resultado conocido para 9 de las 10 instancias del conjunto de datos de prueba.

En las pruebas con la condición de término en 200.000 iteraciones, las instancias GRAPH 08 y GRAPH 09 no presentaron un comportamiento homogéneo para las distintas configuraciones de la Tabla II. Debido a esto, se configuró un nuevo experimento tomando como instancia de prueba a GRAPH 08. Aquí, se procedió a comparar dos configuraciones; la mejor configuración (M, $\tau = 3,8$) contra una de las peores (D, $\tau = 3,8$), obtenidas de las pruebas con 200.000 iteraciones. Ahora, ambas configuraciones se testearon aumentando la condición de termino en 200.000 iteraciones hasta alcanzar el tope de 2.000.000 de iteraciones. Los resultados obtenidos con las dos configuraciones muestran que se logran mejoras significativas a medida que se avanzaba en la búsqueda. Sin embargo, con la configuración D se logró alcanzar el óptimo una mayor cantidad de veces (ver la Fig. 5). El análisis de estos resultados dejan ver que algunas configuraciones permiten converger rápidamente a la zona de soluciones factibles, sin embargo son menos eficaces en ajustar la solución actual para alcanzar los mejores resultados. En cambio, configuraciones que se pudieran catalogar como *lentas* permiten realizar una mejor exploración del espacio de búsqueda. Esto se traduce en que el comportamiento del algoritmo sea más efectivo en cada ejecución, independiente del valor de semilla utilizada para su ejecución.

Otra instancia interesante para un análisis adicional es CELAR 04. Con CELAR 04 se realizó una prueba adicional, para mitigar el problema que suscita el primer término de la función de aptitud. Esto, debido a que la creación aleatoria de la solución inicial genera configuraciones con una cantidad

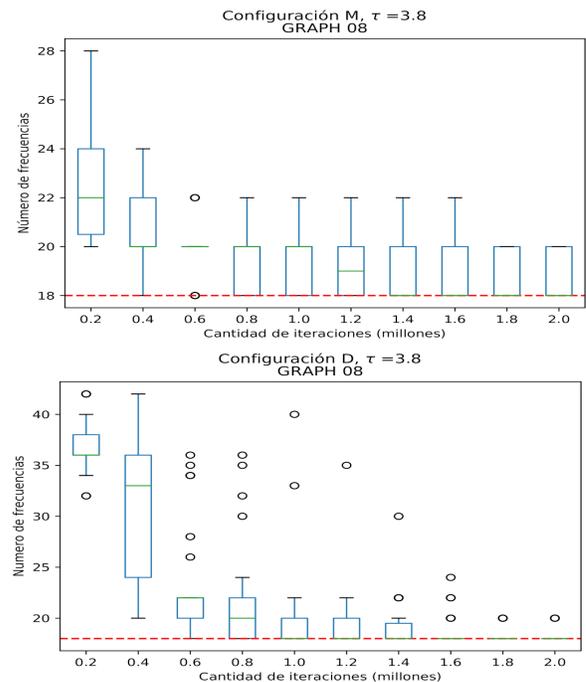


Fig 5. Comparación de dos configuraciones para GRAPH 08.

menor de frecuencias que la solución óptima. Así, se trabajó con la configuración de pesos Z, $w_1 = 0; w_2 = 0,55$ y $w_3 = 0,45$, que deposita una mayor importancia sobre el número de violaciones en vez del número de frecuencias. Los experimentos mostraron que la configuración Z obtuvo mejores resultados a los obtenidos con la configuración G de la Tabla I, hecho que se puede constatar en la Tabla IV. Este resultado enfatiza la cuestión de que se hace necesario realizar una correcta sintonización de los parámetros de peso de acuerdo a las características particulares de la instancia a solucionar.

Un aspecto a considerar es que se observó que en la

mayoría de los experimentos, independiente de los pesos, las restricciones de interferencia violadas eran las primeras en ser corregidas hasta valores de un dígito o directamente cero. Por otro lado, todos los experimentos que no lograron llegar a cero violaciones fueron impedidos por restricciones de interferencia no cumplidas.

Por último, los mejores resultados de la propuesta basada en EO se pueden comparar con los mejores resultados de las otras propuestas metaheurísticas presentes en la literatura revisada en la Sección I y que se muestran en la Tabla V. Aquí, se puede observar que EO logra los mejores resultados comparados con las otras metaheurísticas presente hasta la fecha en la literatura. De hecho, al igual que la propuesta de Alrajhi [43] TS3 basado en un algoritmo de Búsqueda Tabú Híbrido, nuestra propuesta basado en un algoritmo canónico de EO encuentra el valor óptimo para todas las instancias, excepto para CELAR 11. Sin embargo, el valor encontrado por EO para CELAR 11 es mejor que el encontrado por TS3. Finalmente, EO demostró ser una metaheurística competitiva comparada con las otras metaheurísticas presentes en la literatura.

TABLA V
RESULTADOS DE EO Y OTRAS METAHEURÍSTICAS EN LA LITERATURA.

Instancia	Sol. Ópt.	GA [51]	ES [57]	SA [52]	TS1 [52]	TS2 [55]	TS3 [43]	EO
CELAR 01	16	20	-	16	16	18	16	16
CELAR 02	14	14	14	14	14	14	14	14
CELAR 03	14	16	14	14	14	14	14	14
CELAR 04	46	46	-	46	46	46	46	46
CELAR 11	22	32	-	24	22	24	38	22
GRAPH 01	18	20	18	-	18	18	18	18
GRAPH 02	14	16	14	-	14	16	14	14
GRAPH 08	18	-	-	-	20	24	18	18
GRAPH 09	18	28	-	-	22	22	18	18
GRAPH 14	8	14	-	-	10	12	8	8

IV. CONCLUSIONES

En esta investigación se evaluó el desempeño de la versión canónica de EO en un nuevo dominio de aplicación como son los problemas del tipo MO-FAP. Como también, se propuso un nuevo enfoque metaheurístico para enfrentar problemas de asignación de frecuencias como el MO-FAP a través de EO.

Los resultados obtenidos de esta investigación muestran que el algoritmo propuesto de la versión canónica de EO es capaz de solucionar el MO-FAP para una gran mayoría de su variantes de complejidad. En este estudio se logró encontrar los resultados del estados del arte para 9 de las 10 instancias de prueba, es decir en un 90%. Según la comparación de resultados mostrado en la en la Tabla V, el algoritmo propuesto de EO superó a los otros algoritmos en la literatura. Una ventaja comparativa de la propuesta canónica de EO es que no realiza búsquedas exhaustivas en las vecindades de las soluciones por lo que, en comparación con otros algoritmos que si lo hacen, como TS3 [43], es posible aumentar el número de iteraciones sin que el tiempo de cómputo se vea severamente afectado. Cuando se utilizan vecindades, se genera y evalúa toda la vecindad antes de tomar una decisión. Esto desencadena una búsqueda exhaustiva que puede sobrecargar y consumir una cantidad de recursos y tiempo considerable para cada iteración

del algoritmo principal, el cuál, podría estar basado en alguna metaheurística, como por ejemplo, la Búsqueda Tabú.

Los experimentos realizados mostraron, en primer lugar, que a medida que se aumentaba el número máximo de iteraciones, se conseguía una mejora continua en su desempeño general sin presentar un estancamiento permanente significativo. El ajuste de este parámetro responde a una relación lineal con el tiempo de ejecución. Esto permitiría programar la duración del algoritmo de acuerdo a los requerimientos del problema que se está resolviendo. Por ejemplo, tiempo disponible para encontrar una solución versus la calidad del resultado.

Con respecto al hiper-parametro τ , el valor de éste mantuvo una correlación directa con la velocidad de convergencia hacia una zona factible. Así, para valores más altos de τ se llega más rápidamente a la zona de soluciones factibles necesitando una menor cantidad de iteraciones para ello. Sin embargo, la capacidad de EO para explorar nuevas zonas del espacio de búsqueda se ven mermadas. Esto conlleva a una mayor probabilidad de que el algoritmo se quede estancado en zonas de óptimos locales. Este hecho fue observado de forma más notoria en las instancias CELAR 01, CELAR 03 y GRAPH 14, las cuales, obtuvieron resultados factibles con un número de frecuencias razonablemente bajo, pero con dificultad para alcanzar el mejor valor encontrado para la instancia en particular. Un punto relevante de destacar es que el valor de τ utilizado para los mejores resultados no coincide con la fórmula propuesta para obtener el mejor valor de τ en [58] o a los valores probados comúnmente en otros problemas de optimización [18]–[26], [59]. En este trabajo, los mejores resultados se obtuvieron para valores de τ entre 3.8 y 5.0, el cual, es más alto al rango comúnmente utilizado que está entre 1.2 a 2.7. Una posible explicación para estos valores de τ se puede dar por la gran cantidad de restricciones, tanto bidireccionales como de interferencia, que tienen cada problema. Este aspecto puede incidir en la necesidad de elegir con una mayor frecuencia el elemento o especie peor evaluado para poder llegar a una zona de soluciones factibles.

Ahora, en relación a los pesos de la función de aptitud, éstos deben ser ajustados dependiendo de la naturaleza del problema, ya que resulta contraproducente intentar generalizarlos para trabajar con todas las instancias. Esto se pudo observar con la instancia CELAR 04, donde las configuraciones estándares definidas no dieron los resultados esperados, debido al fenómeno analizado en la Sección B, cuando el número de frecuencias asignadas de la solución inicial en menor al resultado esperado. Otro caso interesante se observó con la instancia GRAPH 08, donde ciertas configuraciones entregan al algoritmo la capacidad de encontrar soluciones factibles con una menor cantidad de iteraciones pero menos precisas; mientras que otras requieren una mayor cantidad de pasos en favor de una solución más cercana a la óptima. Por consiguiente, las configuraciones generadas a partir de una instancia en particular, como fue el caso de CELAR 02 para esta investigación, no necesariamente se ajusta de la misma maneras para algunas otras instancias.

Aún cuando se pudo encontrar los resultados del estado del arte para la mayoría de las instancias, para alguno de ellas, este logró no se cumplió de forma constante para todas las pruebas

con valor de semilla diferente. Una explicación para esta disparidad se podría deber a que, a diferencia de otras técnicas como los Algoritmos Genéticos, en EO se tiene que calcular la función de aptitud para cada elemento de la solución y no para la solución completa. Así, el cálculo de la función de aptitud se dificulta si no se tiene suficiente información para medir la contribución de cada elemento de la solución. Para el caso de MO-FAP, se definió una función de aptitud sencilla, cuyos términos tienen una relación directa con el problema (número de frecuencias y número de violaciones). Sin embargo, los desafíos que se encontraron con la función de aptitud sugieren que se podrían mejorar al elaborar una función que considere variables adicionales para poder desempatar el valor numérico de conexiones que tengan el mismo número de restricciones violadas y cuyas frecuencias utilizadas tengan igual cantidad de usos. Otra explicación puede estar dada por la gran cantidad de restricciones de dominio, de interferencia y bidireccionales de las instancias más complejas, principalmente en el caso de CELAR 11, donde la literatura reporta la dificultad de alcanzar la zona de soluciones factibles. En este aspecto, se sugiere trabajar en encontrar un criterio de reemplazo diferente al aleatorio que permita encontrar la correcta relación entre las frecuencias disponibles para cada variable con sus respectivas restricciones. Esto con el objetivo de mejorar la convergencia y permanencia en la zona de soluciones factibles.

Las futuras investigaciones deberían centrarse en las siguientes ideas. Implementar un método avanzado para la selección de los valores del parámetro τ y de los pesos de la función objetivo. Plantear otras formas de modelar el problema para agilizar y mejorar el cálculo de la función de aptitud y verificar el cumplimiento de las restricciones. Formular nuevas funciones de aptitud que permitan explorar y explotar de manera más eficaz el espacio de búsqueda. Implementar un operador de reemplazo de la especie seleccionada que permita mejorar la convergencia hacia el valor óptimo. Explorar la utilización de algún método de búsqueda local de apoyo para EO. Buscar una forma de implementar una versión concurrente del algoritmo, con hilos de ejecución que trabajen diferentes valores de τ , que permita crear una co-evolución de éstos para lograr resultados más diversos y efectivos. Analizar el comportamiento del algoritmo propuesto en otras variantes de problemas de asignación de frecuencias como el *minimum interference frequency assignment (MI-FAP)*, *minimum blocking Frequency assignment (MB-FAP)* y *minimum span frequency assignment (MS-FAP)* [32], [37].

AGRADECIMIENTOS

Los autores agradecen el apoyo a la Dirección de Investigación de la UCSC a través del proyecto DIN 13/2014.

REFERENCES

- [1] D. S. Hochba, "Approximation algorithms for np-hard problems," *ACM Sigact News*, vol. 28, no. 2, pp. 40–52, 1997.
- [2] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Information Sciences*, vol. 237, pp. 82–117, 2013. Prediction, Control and Diagnosis using Advanced Neural Computations.
- [3] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, "A survey on new generation metaheuristic algorithms," *Computers & Industrial Engineering*, vol. 137, p. 106040, 2019.
- [4] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989.
- [5] M. Dorigo and G. Di Caro, "The Ant Colony Optimization Meta-Heuristic," in *New Ideas in Optimization* (D. Corne, M. Dorigo, and F. Glover, eds.), Advanced Topics In Computer Science, ch. 2, pp. 11–32, London: McGraw-Hill, 1999.
- [6] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE International Conference on Neural Networks*, vol. 4, (Perth), pp. 1942–1948, 1995.
- [7] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, pp. 671–680, may 1983.
- [8] F. Glover and M. Laguna., *Tabu Search*. Boston: Kluwer Academic Publishers, 1997.
- [9] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems)*. The MIT Press, 1 ed., 1992.
- [10] R. Storn and K. Price, "Differential Evolution: A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [11] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, vol. 1, no. 1, pp. 67–82, 1997.
- [12] J. Henriques and F. Caldeira, "A model for planning telco work-field activities enabled by genetic and ant colony algorithms," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 7, no. 6, pp. 24–30, 2022.
- [13] C.-J. Juan, C.-S. Wang, B.-Y. Lee, S.-Y. Chiang, C.-C. Yeh, D.-Y. Cho, and W.-C. Shen, "Integration of genetic programming and tabu search mechanism for automatic detection of magnetic resonance imaging in cervical spondylosis," *International Journal of Interactive Multimedia and Artificial Intelligence ...*, vol. 6, no. 7, pp. 109–116, 2021.
- [14] M. S. B. M. Kasihmuddin, M. A. B. Mansor, S. Abdulhabib Alzaeemi, and S. Sathasivam, "Satisfiability logic analysis via radial basis function neural network with artificial bee colony algorithm," *International Journal of Interactive Multimedia and Artificial Intelligence ...*, vol. 6, no. 6, pp. 164–173, 2021.
- [15] R. Hans and H. Kaur, "Binary multi-verse optimization (bmvo) approaches for feature selection," *International Journal of Interactive Multimedia and Artificial Intelligence ...*, vol. 6, no. 1, pp. 91–106, 2020.
- [16] S. Boettcher and A. G. Percus, "Extremal Optimization: Methods derived from Co-Evolution," apr 1999.
- [17] P. Bak and K. Sneppen, "Punctuated equilibrium and criticality in a simple model of evolution," *Physical Review Letters*, vol. 71, pp. 4083–4086, dec 1993.
- [18] S. Boettcher and A. G. Percus, "Extremal Optimization for Graph Partitioning," *Physical Review E*, vol. 64, p. 26114, 2001.
- [19] S. Boettcher and A. G. Percus, "Extremal Optimization: an Evolutionary Local-Search Algorithm," *CoRR*, vol. cs.NE/0209, 2002.
- [20] S. Boettcher and A. G. Percus, "Extremal optimization at the phase transition of the three-coloring problem," *Phys. Rev. E*, vol. 69, p. 66703, 6 2004.
- [21] F. L. D. Sousa, V. Vlassov, and F. M. Ramos, "Heat Pipe Design Through Generalized Extremal Optimization," *Heat Transfer Engineering*, vol. 25, no. 7, pp. 35–45, 2004.
- [22] J. Duch and A. Arenas, "Community detection in complex networks using Extremal Optimization," *Physical Review E*, vol. 72, p. 27104, 2005.
- [23] M. E. Menaï and M. Batouche, "An effective heuristic algorithm for the maximum satisfiability problem," *Applied Intelligence*, vol. 24, no. 3, pp. 227–239, 2006.
- [24] Y.-W. Chen, Y.-Z. Lu, and P. Chen, "Optimization with extremal dynamics for the traveling salesman problem," *Physica A: Statistical Mechanics and its Applications*, vol. 385, no. 1, pp. 115–123, 2007.
- [25] Y.-W. Chen, Y.-Z. Lu, and G.-K. Yang, "Hybrid evolutionary algorithm with marriage of genetic algorithm and extremal optimization for production scheduling," *The International Journal of Advanced Manufacturing Technology*, vol. 36, no. 9, pp. 959–968, 2008.
- [26] J. Ding, Y. Lu, and J. Chu, "Extremal optimization for unit commitment problem for power systems," in *2012 IEEE Power and Energy Society General Meeting*, pp. 1–8, 2012.
- [27] E. Sidiropoulos, "Spatial resource allocation via extremal optimization enhanced by cell-based local search," *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 6, p. 1550020, 2 2015. doi: 10.1142/S1793962315500208.

- [28] F. Redolozza and L. Li, "A novel method for well placement design in groundwater management: Extremal optimization," *Advances in Water Resources*, vol. 132, 2019.
- [29] S. Boettcher, "Ground state properties of the diluted sherrington-kirkpatrick spin glass," *Physical Review Letters*, vol. 124, p. 177202, 2020.
- [30] T. Insights, "Iot connected devices by vertical 2030 | statista." Web Page, 2020.
- [31] Statista, "Internet users in the world 2025 | statista." Web Page, 2021.
- [32] K. I. Aardal, S. P. M. van Hoesel, A. M. C. A. Koster, C. Mannino, and A. Sassano, "Models and solution techniques for frequency assignment problems," *Annals of Operations Research*, vol. 153, pp. 79–129, 9 2007.
- [33] J. Wang and Y. Cai, "Multiobjective evolutionary algorithm for frequency assignment problem in satellite communications," *Soft Comput.*, vol. 19, p. 1229–1253, may 2015.
- [34] A. Stojanova, D. Bikov, G. Kobeaga, M. Kocaleva, T. Koca, T. Ashley, and T. Balabanov, "Self-organized networks," *Proceedings of the 131st European study group with industry*, pp. 18–25, 2017.
- [35] W. Hale, "Frequency assignment: Theory and applications," *Proceedings of the IEEE*, vol. 68, no. 12, pp. 1497–1514, 1980.
- [36] R. Borndörfer, A. Eisenblätter, M. Grötschel, and A. Martin, "Frequency assignment in cellular phone networks," *Annals of Operations Research*, vol. 76, pp. 73–93, 1998.
- [37] A. M. C. A. Koster, *Frequency Assignment - Models and Algorithms*. PhD thesis, Maastricht University, 1999.
- [38] K. Smith, "A genetic algorithm for the channel assignment problem," in *IEEE GLOBECOM 1998 (Cat. NO. 98CH36250)*, vol. 4, pp. 2013–2018 vol.4, 1998.
- [39] G. Colombo, "A Genetic Algorithm for Frequency Assignment with Problem Decomposition," *Int. J. Mob. Netw. Des. Innov.*, vol. 1, no. 2, pp. 102–112, 2006.
- [40] A. Quellmalz, A. Knalman, and B. Muller, "Efficient frequency assignment with simulated annealing," in *1995 Ninth International Conference on Antennas and Propagation, ICAP '95 (Conf. Publ. No. 407)*, vol. 2, pp. 301–304 vol.2, 1995.
- [41] Lu Liwei and Fan Rongshuang, "Simulated annealing algorithm in solving frequency assignment problem," in *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, vol. 1, pp. 1–361, IEEE, 8 2010.
- [42] D. J. Castellino, S. Hurley, and N. M. Stephens, "A tabu search algorithm for frequency assignment," *Annals of Operations Research*, vol. 63, no. 2, pp. 301–319, 1996.
- [43] K. Alrajhi, J. Thompson, and W. Padungwech, "Tabu Search Hybridized with Multiple Neighborhood Structures for the Frequency Assignment Problem," in *Hybrid Metaheuristics* (M. J. Blesa, C. Blum, A. Cangelosi, V. Cutello, A. DI NUOVO, M. Pavone, and E.-G. Talbi, eds.), (Cham), pp. 157–170, Springer International Publishing, 2016.
- [44] A. Eisenblätter, H. Geerdes, and I. Siomina, "Integrated Access Point Placement and Channel Assignment for Wireless LANs in an Indoor Office Environment," in *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pp. 1–10, 2007.
- [45] A. E. Koller and S. D. Noble, "Domination analysis of greedy heuristics for the frequency assignment problem," *Discrete Mathematics*, vol. 275, no. 1, pp. 331–338, 2004.
- [46] B. Koo, H. B. Yilmaz, C. Chae, H. Park, J. Ham, and S. Park, "Heuristics for frequency assignment problem with realistic interference constraints," in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, 5 2016.
- [47] K. Smith and M. Palaniswami, "Static and dynamic channel assignment using neural networks," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 238–249, 2 1997.
- [48] H. Sandalidis, P. Stavroulakis, and J. Rodriguez-Tellez, "Borrowing channel assignment strategies based on heuristic techniques for cellular systems," *IEEE Transactions on Neural Networks*, vol. 10, no. 1, pp. 176–181, 1999.
- [49] Z. I. Berlin, "CALMA Frequency Assignment Problems," 2010.
- [50] A. Kapsalis, P. Chardaire, V. J. Rayward-Smith, and G. D. Smith, "The radio link frequency assignment problem: A case study using genetic algorithms," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 993, pp. 117–131, 10 1995.
- [51] C. Crisan and H. Mühlenbein, "The frequency assignment problem: A look at the performance of evolutionary search," in *Artificial Evolution* (J.-K. Hao, E. Lutton, E. Ronald, M. Schoenauer, and D. Snyers, eds.), (Berlin, Heidelberg), pp. 263–273, Springer Berlin Heidelberg, 1998.
- [52] S. R. Tiourine, C. A. J. Hurkens, and J. K. Lenstra, "Local search algorithms for the radio link frequency assignment problem," *Telecommunication Systems*, vol. 13, no. 2, pp. 293–314, 2000.
- [53] A. Bouju, J. F. Boyce, C. Dimitropoulos, and J. Taylor, "Tabu search for the radio links frequency assignment problem," in *Applied Decision Technologies (ADT-95)*, 1995.
- [54] J.-K. Hao, R. Dorne, and P. Galinier, "Tabu Search for Frequency Assignment in Mobile Radio Networks," *Journal of Heuristics*, vol. 4, no. 1, pp. 47–62, 1998.
- [55] A. Bouju, J. Boyce, C. Dimitropoulos, G. Vom Scheidt, J. Taylor, A. Likas, G. Papageorgiou, and A. Stafylopatis, "Intelligent search for the radio link frequency assignment problem," in *Proceedings of the International Conference on Digital Signal Processing, Cyprus*, pp. 267–303, Citeseer, 1995.
- [56] Y.-Z. Lu, Y.-W. Chen, M. Chen, P. Chen, and G.-Q. Zeng, *Extremal Optimization: Fundamental, Algorithms and Applications*. CRC Press and Chemical Industrial Press, 10 2016.
- [57] M. Parsapoor and U. Bilstrup, "Ant colony optimization for channel assignment problem in a clustered mobile ad hoc network," in *Advances in Swarm Intelligence* (Y. Tan, Y. Shi, and H. Mo, eds.), (Berlin, Heidelberg), pp. 314–322, Springer Berlin Heidelberg, 2013.
- [58] S. Boettcher and A. G. Percus, "Extremal optimization: an evolutionary local-search algorithm," in *Computational Modeling and Problem Solving in the Networked World*, pp. 61–77, Springer, 2003.
- [59] S. Boettcher and A. G. Percus, "Optimization with extremal dynamics," *Physical Review Letters*, vol. 86, pp. 5211–5214, 6 2001.



Pedro Gomez-Meneses received his B. Sc. degree in Engineering, B. Eng. in Informatic Civil Engineering, PG. Dip. in Industrial Engineering from Universidad de Concepción, Chile. He received his Ph.D. degree in Information Technology from Bond University, Australia. He is currently assistant professor in the Department of Informatic Engineering at the Universidad Católica de la Santísima Concepción (UCSC), Chile. He has been Head of the Informatic Civil Engineering Career and Head of the Department of Informatic Engineering at the UCSC.

His research interests are in the area of nature-inspired computation, meta-heuristic search methods, combinatorial optimization and machine learning.



Héctor Ayala received his B.Sc. degree in Engineering, and Informatic Civil Engineer from the Universidad Católica de la Santísima Concepción, Concepción, Chile. He has experience in database and machine learning. His research interests are machine learning, combinatorial optimization and metaheuristic methods.