

Content-Based Conference Scheduling Optimization

F. Castaño, N. Velasco, and J. Carvajal-Beltrán

Abstract—A scientific conference is an activity organized by a community of researchers aimed at offering a platform for knowledge sharing and scientific exchange. The planning of a conference includes the creation of the program schedule where decisions about track composition, in terms of talks, and the temporal allocation of those tracks need to be considered. Designing such a schedule represents a difficult task in which aspects as content similarity and the overlapping of similar tracks should be taken into account. In this paper a two-phase solution approach is proposed for helping planners designing schedules. In first place, talks are grouped into tracks upon the basis of talks content. Then, a timetable for the tracks is created by group them over the basis of content similarity, avoiding those with similar content to be scheduled in parallel. Both problems demonstrate to require a considerably computational effort; consequently, advanced algorithmic approaches as column generation and GRASP are exploited with the purpose of finding near-optimal solutions in a short computational time.

Index Terms—Conference Scheduling, Maximum diversity grouping, Maximum diversity problem, Metaheuristics, Column generation.

I. INTRODUCCIÓN

Las conferencias científicas tienen el propósito de generar espacios de discusión entre investigadores donde se comparten los últimos avances en investigación. Dichos eventos constituyen una plataforma para el intercambio de conocimiento, la formación de redes y, de esta forma, el fortalecimiento de la ciencia [1]. La cantidad de conferencias y el número de asistentes parece crecer cada año, lo cual suele conllevar a labores titánicas de logística para su organización. Una conferencia científica típica combina algunos elementos como conferencias magistrales, sesiones paralelas, sesiones de póster, *workshops*, entre otros. Las conferencias magistrales, usualmente buscan invitar a todos los asistentes a una

presentación de un investigador relevante en los tópicos centrales, para la discusión de su investigación reciente, además de motivar a otros a investigar en dicho tópico. Durante las sesiones paralelas los investigadores asistentes a la conferencia comparten sus investigaciones recientes con otros miembros de la comunidad científica de la cual participan. Las sesiones de póster, por su lado, permiten la interacción directa entre autores e interesados mediante la presentación de un resumen corto del trabajo, que permita un acercamiento entre las partes y conduzca al fortalecimiento de las redes de investigación, y las investigaciones mediante la realimentación a las mismas.

El proceso de planificación de una conferencia incluye la creación de un cronograma de programa científico, es decir, la asignación local y temporal de todas las charlas que se presentarán durante el encuentro. Si bien globalmente, es fácil definir los momentos en los cuales sesiones plenarias, sesiones paralelas, presentaciones de poster y otros eventos toman lugar, la conformación de las sesiones paralelas (también conocidas como sesiones o *tracks*) y la asignación temporal de estas constituye una tarea compleja [2]. Entre las razones que podrían aducirse como causantes de tal dificultad se encuentra el hecho que se deben considerar aspectos relacionados con cuán “enriquecedora” esta es para los asistentes [3], [4], buscándose particularmente que estos puedan participar en aquellas que consideran más relevantes, además de algunas restricciones “duras”, que limitan el número de charlas que se pueden llevar a cabo durante ventanas de tiempo predefinido y/o simultáneamente.

Lo anterior es particularmente cierto si se considera que, en años recientes, se viene discutiendo en la comunidad científica la utilidad de las conferencias como espacio de interacción y los efectos colaterales de la participación de los investigadores en estas [5]. Como consecuencia, la necesidad de organizar programas que permitan a los asistentes participar en los eventos de su interés adquiere una mayor relevancia y, por consiguiente, la necesidad de tecnificar el proceso de agendamiento.

Si bien asumir que cada artículo viene clasificado en un tópico dado es común en la práctica, la delimitación de tales temáticas puede ser bastante difusa. Como consecuencia de esto es poco común que se considere específicamente el contenido de las presentaciones en la generación de las agendas y así tal preclasificación no necesariamente conlleva a generar agendas que agrupen las charlas en sesiones de forma satisfactoria. Para mitigar este problema, una posible práctica es agrupar los artículos basándose en su contenido generando sesiones sólo una vez el contenido de los artículos o resúmenes se conoce. En este artículo esta perspectiva se explora siguiendo un enfoque de dos fases que agrupa las charlas en sesiones de tamaño dado

F. Castaño, Departamento de Ingeniería Civil e Industrial, Pontificia Universidad Javeriana, Calle 18 No. 118-250, Cali, Colombia fabian.castano@javerianacali.edu.co; School of Management, Universidad de los Andes, Carrera 1 No. 19-27 Ed. LA, Oficina 106, Bogotá, Colombia.

N. Velasco, School of Management, Universidad de los Andes, Calle 21 No. 1-20, Bogotá, Colombia, nvelasco@uniandes.edu.co.

J. Carvajal, Departamento de Ingeniería Industrial, Universidad Nacional de Colombia, Km 9 vía al Aeropuerto La Nubia, Manizales, Colombia, jcarvajalb@unal.edu.co.

con base en su contenido y, posteriormente, agrupa las sesiones evitando que contenido similar sea dispuesto para ser presentado en paralelo. El método se evalúa en una conferencia de tamaño mediano y mediante la utilización de instancias de prueba.

II. ESTADO DEL ARTE Y TRABAJO RELACIONADO

La organización de conferencias puede ser una tarea fácil cuando el tamaño de esta no es significativamente grande, es decir no muchas charlas tomarán lugar durante el evento (típicamente menos de 50). Por el contrario, puede ser una labor muy difícil cuando el número de charlas crece y algunas características que se deben considerar durante la organización comienzan a jugar un papel relevante [6]. En una conferencia típica de hasta 100 charlas se organizan entre tres y cuatro bloques de sesiones paralelas por cada día, cada una contando con entre 3 y 4 charlas. Las charlas son agrupadas de acuerdo con criterios preestablecidos, usualmente la similitud entre estas y/o su afinidad con el tema de la sesión.

En la organización de conferencias es frecuente encontrar que el grado de tecnificación de actividades como la organización de la agenda es bajo [2]. Adicionalmente, pocos reportes sobre el proceso utilizado se encuentran disponibles, haciendo parecer que el agendamiento de conferencias es un tema de investigación fértil, en el cual muchos desarrollos son posibles en aras de hacer más eficiente el proceso y mejorar el resultado de dicha actividad.

De acuerdo con Thompson [3] y Vangerven *et al.* [4] se podrían clasificar los enfoques de programación de conferencias entre aquellos guiados por las preferencias de los participantes (Attendees-based perspective, ABP) y los que se guían por la disponibilidad de los autores (Presenters-Based-Perspective, PBP). Además de lo anterior, una revisión de literatura sugiere que la mayoría de los enfoques consideran los artículos como pertenecientes o no a un tópico y optimizan la agenda usando exclusivamente esta información. Esto suele ser impulsado por el hecho que los autores son obligados a clasificar sus presentaciones al momento de cargarlas en los sistemas de información dispuestos para tal fin.

La perspectiva PBP ha sido probablemente la que más consideración ha recibido por parte de la comunidad científica [7]. Para esta se han desarrollado diferentes enfoques que consideran aspectos relacionados con las preferencias de los autores y que se combinan con elementos como la distancia entre la localización de las sesiones [2], la similitud de contenidos en cada sesión [8], solicitudes especiales por parte de los autores, entre otros.

Una similitud entre lo enfoques que asumen la perspectiva ABP es la utilización de información solicitada a los participantes en la construcción de las agendas [3], [9]. Thompson [3] utiliza esta información en la construcción y resolución de un interesante problema de optimización combinatorio cuyo objetivo es maximizar la participación de los asistentes en las sesiones de su preferencia. En esta misma línea, Le Page [10] utiliza esta información en la generación de una matriz de conflicto que representa el número de participantes que desean asistir a una charla dada y puede utilizarse en la construcción de agendas que minimicen dichos conflictos. Vangerven *et al.* [4], proponen la elaboración de

métricas que, con base en la información proporcionada por los asistentes a la conferencia, permiten medir la conveniencia de programar dos charlas en paralelo y la utilizan para organizar sesiones paralelas que maximicen la utilidad de la conferencia percibida por los participantes.

Dadas las características del problema, la construcción de una agenda manual, sin uso de herramientas computacionales apropiadas, puede convertirse en una desafiante tarea. Por esta razón, es común encontrar que se adopten diversas y sofisticadas técnicas de optimización para resolver el problema de forma satisfactoria. El objetivo es la optimización de una función objetivo que representa las características (in)deseables de la agenda.

En consideración a las métricas y características que hacen especial el problema de generación de agendas, es común encontrar que se plantee su optimización mediante la descomposición en problemas de menor tamaño. Vangerven *et al.* [4], por ejemplo, dividen el problema de generación de agendas en tres fases y demuestran que cada una de estas corresponde a un problema *NP-HARD*. Una consecuencia de esto es que, casi en general, la búsqueda de soluciones al problema se haga de forma aproximada. Idea potencializada, además, por la dificultad que representa la definición clara de un único objetivo que represente las expectativas respecto a una agenda de calidad [11].

Es bastante común encontrar que herramientas como programación matemática sean elegidas a la hora de tratar de dar una solución al problema. En algunos casos, este enfoque se utiliza para la resolución de un problema monolítico que busca la generación de la agenda considerando todas las restricciones [12], [13]. No obstante, en la mayoría de los casos una serie de modelos, cada uno abordando una necesidad diferente, son utilizados como respuesta a la dificultad del problema [2][6].

Otros métodos aproximados consisten en la combinación de métodos exactos y heurísticos para hallar una solución al problema. En algunos casos este enfoque se combina con la solución a modelos simplificados de programación entera, ya sea monolíticos o resueltos en cascada, con el propósito de generar eficientemente una solución que respete las restricciones del problema [6][11].

Finalmente, y quizá con el propósito de encontrar una solución eficientemente al problema, es común encontrar que algunos autores adopten enfoques puramente heurísticos. Por un costado, se pueden encontrar estrategias simples consistentes en la utilización de heurísticas codiciosas para generar soluciones al problema sacrificando la calidad de la solución [14]. Bastante menos común es encontrar que se utilicen estrategias de búsqueda inteligentes basadas en metaheurísticas con el fin de conseguir una mejor exploración del espacio de solución y así soluciones apropiadas al problema enfrenteado [3].

Este artículo propone un enfoque multi-etapa para el diseño de agendas de conferencia. Un énfasis especial se hace en la utilización de la información sobre el artículo que proveen los autores para la generación de agendas que, por un lado, agrupen de la forma más apropiada las sesiones y, por el otro, eviten que sesiones de contenido similar se realicen de forma simultánea. Se considera también la exploración de la eficacia de diferentes estrategias de solución, con el propósito de seleccionar una

apropiada para hacer frente al problema en tiempos computacionales reducidos.

III. UN ENFOQUE DE DOS FASES PARA EL AGENDAMIENTO DE CONFERENCIAS

A continuación, se describe el enfoque de solución utilizado para la generación de la agenda en conferencias que se basa en la solución de diferentes problemas combinatorios. El enfoque propuesto consiste en dos fases. En la primera fase las charlas se agregan en sesiones usando como criterio la similitud entre estas. Una vez se tiene la composición de las sesiones, un segundo problema de optimización se resuelve buscando minimizar la similitud entre sesiones que se desarrollan de forma simultánea.

A. Descripción del Problema

El problema de generación de agendas para conferencias consiste en el agendamiento de charlas que serán dictadas por los autores participantes en sesiones y franjas de tiempo. En cada franja de tiempo se lleva a cabo más de una sesión y, en cada sesión, hay más de una charla. En la Figura 1, por ejemplo, vemos que se ejecuta un bloque con tres sesiones en paralelo y, en cada una, se desarrollarán exactamente 3 charlas. Estas tomarán lugar durante un intervalo de tiempo suficiente para que las 3 charlas puedan culminarse.

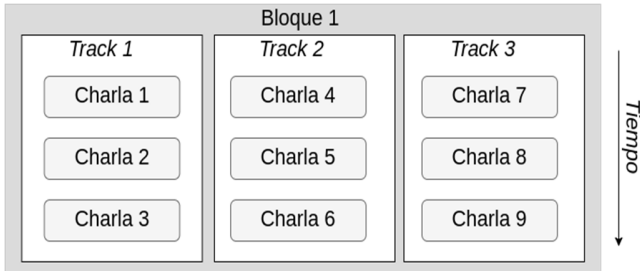


Fig. 1. Estructura de un bloque de sesiones paralelas.

Consideremos un conjunto de charlas C . Cada charla $i \in C$ debe ser asignada a una sesión $t \in T$ y, finalmente a una franja de tiempo $p \in P$. Para cada par de charlas es posible computar la similitud sobre la base del contenido [15], las palabras claves suministradas por los autores [8], los términos de clasificación usadas al momento de cargar el documento o combinaciones de los anteriores. Sea α_{ij} el beneficio obtenido por asignar las charlas $i \in C$ y $j \in C$ en la misma sesión. Para cada sesión $t \in T$, $L_t \subseteq C$ representa el subconjunto de charlas que han sido asignadas a dicha sesión, donde $|L_t| \leq m$, el máximo número de charlas que puede albergar una sesión.

La información básica recientemente introducida permite plantear una serie de modelos de optimización y/o algoritmos que determinan la agenda de la conferencia. La Fase I consiste en determinar la composición de las sesiones, agrupando hasta m charlas de tal forma que su contenido sea similar. Finalmente, en la Fase II, la asignación temporal de estas sesiones se define buscando minimizar los cruces de tópicos similares evitando que se programen en las mismas franjas de tiempo.

B. Composición de las Sesiones (Fase I)

Sea x_{it} una variable binaria que toma el valor de 1 para indicar que la sesión i se asigna a la sesión t . Esta variable representa la decisión básica necesaria para determinar la composición de las sesiones. Se introduce la variable binaria y_{ij}^t que toma el valor 1 cuando las conferencias i y j se asignan a la misma sesión t y el valor 0 de lo contrario. A continuación, se describe un modelo de optimización que representa la primera fase del problema de optimización abordado.

$$\max: \sum_{i \in C} \sum_{j \in C} \sum_{t \in T} y_{ij}^t \alpha_{ij} \quad (1)$$

Sujeto a :

$$\sum_{t \in T} x_{it} = 1 \quad \forall i \in C \quad (2)$$

$$\sum_{t \in T} x_{it} \leq m \quad \forall i \in C \quad (3)$$

$$\sum_{j \in C} y_{ij}^t \leq m x_{it} \quad \forall i \in C, t \in T \quad (4)$$

$$\sum_{i \in C} y_{ij}^t \leq m x_{jt} \quad \forall j \in C, t \in T \quad (5)$$

$$x_{it} \in \{0,1\} \quad \forall t \in T, i \in C \quad (6a)$$

$$y_{ij}^t \in \{0,1\} \quad \forall t \in T, i \in C, j \in C \quad (6b)$$

El objetivo del modelo (Ecuación (1)) es generar asignaciones de manera tal que las charlas con mayor similitud sean agrupadas en las mismas sesiones. Las Restricciones (2), garantizan que cada charla se asigna a una única sesión, mientras que las Restricciones (3) limitan el número de charlas que se asignan a cada sesión. De esta forma se garantiza que no se sobrepase el tiempo total disponible para presentar las charlas. Las Restricciones (4) y (5) se usan para vincular la variable y_{ij}^t con la programación de las sesiones. De esta forma, cada vez que dos charlas sean asignadas a la misma sesión t , y_{ij}^t podrá tomar el valor 1, contribuyendo así a la función objetivo el valor α_{ij} relacionado con la similitud de las charlas i y j .

C. Agendamiento Temporal de las Sesiones (Fase I)

Una vez conocida la solución al problema de la Fase I, y así conformadas las sesiones L_t^* , es posible determinar la similitud total entre cada par de sesiones t y t' como:

$$\beta_{t,t'} = \sum_{i \in L_t^*} \sum_{j \in L_{t'}^*} \alpha_{ij}$$

Una solución apropiada a la asignación de las sesiones a los bloques $b \in P$ debería evitar que aquellas de contenido similar se programen en paralelo. Una modificación menor a los modelos previamente propuestos permite alcanzar dicho objetivo. Sea $z_{t't}^p$ una variable binaria que toma el valor 1 cuando las sesiones t y t' se asignan simultáneamente al bloque p y ρ_{tp} una variable binaria que toma el valor 1 cuando la sesión t se programa en el bloque p , de capacidad máxima Cap_p , el modelo de agendamiento de las sesiones en bloques paralelos puede representarse como:

$$\min: \sum_{t \in T} \sum_{t' \in C} \sum_{p \in P} z_{t't}^p \beta_{t't} \quad (7)$$

Sujeto a :

$$\sum_{p \in P} \rho_{tp} = 1 \quad \forall t \in T \quad (8)$$

$$\sum_{t \in T} \rho_{tp} \leq Cap_p \quad \forall p \in P \quad (9)$$

$$\rho_{tp} + \rho_{t'p} \leq 1 + z_{t't}^p \quad \forall t, t' \in T, p \in P \quad (10)$$

$$z_{t't}^p \in \{0,1\} \quad \forall t, t' \in T, p \in P \quad (11a)$$

$$\rho_{tp} \in \{0,1\} \quad \forall t \in T, p \in P \quad (11b)$$

En este caso, el objetivo, expresado mediante la Ecuación (7), consiste en conformar grupos de sesiones que minimicen el cruce de contenido similar que toma lugar en paralelo durante la conferencia. Las restricciones (8) y (9) buscan, respectivamente, que cada sesión se asigne a un único bloque paralelo y que esta asignación respete la capacidad de tales bloques., típicamente relacionado con el número de salas disponibles para llevar a cabo el evento. Finalmente, la restricción (10) cumple una función auxiliar, y básicamente ayuda a contar cada caso en el que dos sesiones son asignadas al mismo bloque de sesiones paralelas.

IV. ESTRATEGIAS DE SOLUCIÓN

A. Método Heurístico Basado en Generación de Columnas

Una forma extra de representar el Problema (1) - (6) consiste en asumir que se tiene conocimiento de todos los subconjuntos $L_t \subseteq \Omega$ de asignaciones que respetan las restricciones de capacidad expresadas por las Ecuaciones (2) y (3). Sea γ_{it} un parámetro binario usado para indicar que la charla $i \in C$ ha sido asignada al subconjunto L_t , y θ_t una variable binaria que asume el valor 1 cuando el subconjunto L_t forma parte de la asignación óptima. El costo total asociado a un subconjunto L_t puede ser computado como $c_t = \sum_{i \in L_t} \sum_{j \in L_t} \alpha_{ij}$. Así, es posible representar el problema utilizando un modelo cuyo propósito ya no es directamente asignar las charlas a una sesión, sino escoger aquellos conjuntos de asignaciones que maximizan la suma de las similitudes c_t para todas las agrupaciones.

$$\max: \sum_{t \in \Omega} c_t \theta_t \quad (12)$$

Sujeto a :

$$\sum_{t \in \Omega} \gamma_{it} \theta_t = 1 \quad \forall i \in C \quad (13)$$

$$\sum_{t \in \Omega} \theta_t \leq \left\lfloor \frac{|T|}{m} \right\rfloor \quad (14)$$

$$\theta_t \in \{0,1\} \quad \forall t \in T \quad (15)$$

Considerando que los subconjuntos $L_t \subseteq \Omega$ respetan las restricciones de capacidad, una solución al problema sólo debe garantizar que todas las charlas son asignadas a un único grupo (Restricción (13)) y que el número total de sesiones seleccionadas no sobrepasa la capacidad disponible para la instalación de la conferencia (Restricción (14)).

El Modelo (12) – (15), desde ahora denotado Problema Maestro (PM), puede ser demasiado grande para que su resolución en la práctica sea viable. De hecho, el número de subconjuntos $|\Omega|$ crece significativamente con el número de charlas a ser programadas. En ese caso, el modelo podría contener hasta $|\Omega| = \sum_{i=0}^m \binom{|C|}{i}$ variables binarias, resultando en ocasiones impráctica la optimización mediante la utilización directa de programación entera. Además de esto, se sabe que a lo sumo un total de $\left\lfloor \frac{|T|}{m} \right\rfloor$ variables tomarán un valor diferente a cero, haciendo innecesaria la enumeración de todas las variables para dar una solución al problema.

En la práctica, los modelos del tipo (12) – (15) pueden ser abordados mediante la técnica de generación de columnas [16]. Esta técnica divide el problema en dos etapas que se resuelven de forma iterativa y, de esta forma, evita la enumeración exhaustiva de todas las variables. El proceso inicia al considerar como punto de partida una solución inicial a la relajación lineal de este problema, conocido como problema maestro (PM), sobre un subconjunto de variables $\Omega' \subseteq \Omega$. Posteriormente, se añaden nuevas variables, correspondientes a columnas en el modelo, que son promisorias en el propósito de mejorar la función objetivo.

La inclusión iterativa de nuevas columnas se hace siguiendo el criterio del costo reducido. Este proceso se repite hasta que se puede probar que no existe ninguna nueva columna promisorias (costo reducido positivo), momento en el cual se obtiene la solución óptima a la relajación lineal del problema (12)-(15). A partir de este punto, se pueden adoptar dos caminos: *i)* Utilizar la solución actual y resolver problema usando las columnas disponibles (Ω') y considerando, además, las restricciones de integralidad en las variables θ_t , en cuyo caso la solución obtenida corresponde a una solución aproximada (heurística) al problema original; *ii)* Continuar el proceso de optimización aplicando una estrategia tipo *Branch & Price*, cuyo resultado garantiza que la solución hallada es óptima [17]. En este artículo, considerando los requerimientos prácticos de la aplicación, se adopta la primera opción. Como

se verá más adelante, dicha selección permite encontrar soluciones dentro de un margen de diferencia relativamente bajo en comparación con la mejor solución posible.

La Figura 2 ilustra la generación de columnas heurística (CG) propuesta. En cada iteración se requiere identificar una solución que sea promisoria, lo cual implica buscar una asignación de un subgrupo de charlas a una sesión. Esto corresponde a identificar una nueva variable de costos reducidos positivos que pueda entrar al proceso y ayudar a mejorar la función objetivo del PM.

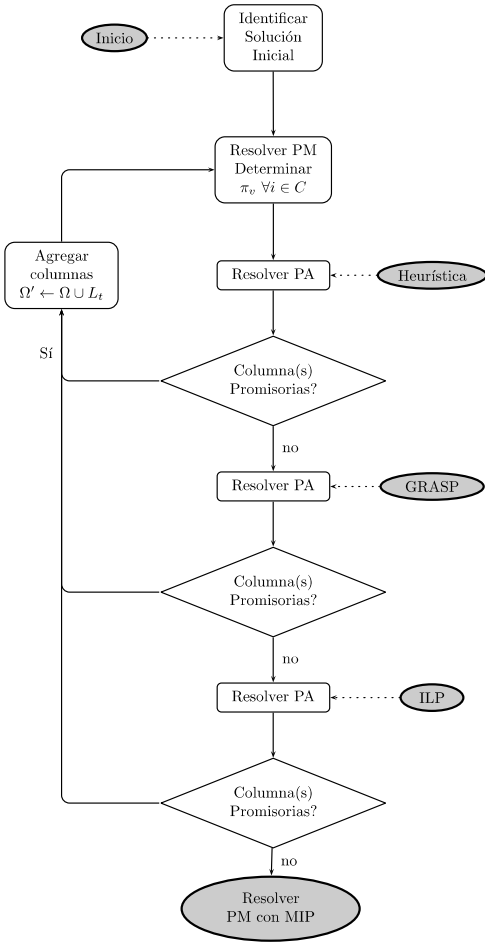


Fig. 2 Generación de columnas heurística.

Como se ilustra en la figura 2, este proceso puede complementarse mediante la utilización de métodos heurísticos, usados como primera aproximación a la solución, o mediante métodos exactos, resolviendo, por ejemplo, otro modelo de programación entera. El objetivo en ambos casos consiste en la conformación de una única columna, que represente las asignaciones de charlas a una sesión, un problema que podría verse como una variante del problema de la diversidad máxima [18]. En este caso, la función de costos depende de la interacción entre cada par de charlas en un subconjunto L_t y una contribución individual asociada a su participación en este. Sea π_i el valor de las variables duales asociadas a las restricciones (13), y Π la variable dual asociada

a la restricción (14), el problema auxiliar (PA) puede ser representado como:

$$\max_{L_t \in \Omega \setminus \Omega'} \sum_{i \in C} \sum_{j \in C} y_{ij}^t \alpha_{ij} - \sum_{i \in C} \pi_i * \gamma_{it} - \Pi \quad (16)$$

Sujeto a :

$$\sum_{j \in C} y_{ij}^t \leq m \gamma_{it} \quad \forall i \in C, t \in T \quad (17)$$

$$\sum_{i \in C} y_{ij}^t \leq m \gamma_{jt} \quad \forall j \in C, t \in T \quad (18)$$

$$\sum_{t \in T} \gamma_{it} \leq m \quad \forall i \in C \quad (19)$$

$$\gamma_{it} \in \{0,1\} \quad \forall i \in C \quad (20a)$$

$$y_{ij}^t \in \{0,1\} \quad \forall i \in C, j \in C \quad (20b)$$

En cada iteración, un nuevo problema t se resuelve y, con la ayuda de las variables duales, se determina una solución γ_{it}^* y y_{ij}^{t*} al problema (Ver Figura 2). Estos valores se transfieren al PM mediante una nueva variable y su correspondiente columna que, generada mediante el Modelo (16)-(20), respeta la restricción del máximo número de charlas que se pueden asignar a una sesión.

Con el fin de que el método propuesto se ejecute de forma eficiente, se propone extender su implementación básica mediante diferentes métodos conocidos por su eficacia en el mejoramiento de la convergencia del proceso de generación de columnas [19], [20]. En primer lugar, se considera la hibridación de la generación de columnas con enfoques (meta)heurísticos que permitan solucionar el PA de forma eficiente. Además, se decide retornar múltiples columnas en cada iteración, una estrategia que, en general, suele conducir a grandes reducciones en el número de iteraciones requeridas para alcanzar la solución óptima a la relajación lineal del PM [21].

Considerando que no se requiere explícitamente una solución óptima al PA para llevar a cabo una nueva iteración de CG, se propone utilizar procedimientos heurísticos para proveer soluciones de forma eficiente. Tal y como se ilustra mediante la Figura 2, en primer lugar, se adopta un simple procedimiento heurístico aleatorizado que construye una solución al problema usando como criterio de selección la contribución a la función objetivo del problema auxiliar (Ver Algoritmo 1). Este procedimiento consiste en: i) la construcción de una lista de charlas candidatas para ser añadidas a una sesión usando el criterio de la variable dual asociada a dicha charla (sobre las Restricciones (13)), ii) la selección aleatoria de una de estas, y iii) la inclusión iterativa de nuevas charlas a la sesión siendo considerada, evaluando el efecto combinado de su similitud a otras charlas y de su variable dual correspondiente en el cálculo de los costos reducidos, tal y como se expresa en la Ecuación (16).

Algoritmo 1 Heurística constructiva.

```

function HEURÍSTICA_CONSTRUCTIVA( $\vec{\pi}, C, \mu, \alpha$ )
   $L_t \leftarrow \emptyset$ 
   $\omega \leftarrow \emptyset$ 
   $RCL \leftarrow \{i \mid \frac{\pi_i - \min \vec{\pi}}{\max \vec{\pi} - \min \vec{\pi}} \geq 1 - \mu \quad \forall i \in C\}$ 
   $e \leftarrow \text{select\_random}(RCL)$ 
   $L_t \leftarrow L_t \cup e$ 
  while  $|L_t| < p$  do
     $\omega \leftarrow \{\sum_{j \in C \setminus L_t} a_{ij} - \pi_i \quad \forall i \in C \setminus L_t\}$ 
     $RCL \leftarrow \{i \mid \frac{\omega_i - \min \omega}{\max \omega - \min \omega} \geq 1 - \mu \quad \forall i \in C \setminus L_t\}$ 
     $e \leftarrow \text{selecc\_aleatoria}(RCL)$ 
     $L_t \leftarrow L_t \cup e$ 
  end while
   $f(L_t) \leftarrow \sum_{j \in L_t} \sum_{i \in L_t} a_{ij} - \sum_{i \in L_t} \pi_i$ 
  return  $L_t, f(L_t)$ 
end function

```

Posteriormente, en caso que una solución interesante no sea encontrada, el proceso continua con una segunda etapa en la cual el espacio se explora siguiendo una heurística multi-arranque GRASP (Algoritmo 2) [22]–[24]. Esta segunda fase emplea, además de la heurística constructiva, un proceso de búsqueda local cuyo objetivo es mejorar las soluciones iniciales halladas en esta primera etapa. Finalmente, sólo si es requerido, el método recurre a la utilización de programación entera (ILP) para encontrar una solución al PA. En el caso que una nueva solución sea encontrada, el proceso continúa de nuevo desde la utilización de la heurística constructiva e itera hasta encontrar la solución óptima a la relajación lineal. Para terminar, el modelo se resuelve sobre el conjunto de variables previamente encontradas, obligando al cumplimiento de la restricción de integralidad en las variables θ_t .

Algoritmo 2 Heurística GRASP usada para resolver PA.

```

function GRASP_PM( $\vec{\pi}, C, \mu, \alpha$ )
   $f(L_t) \leftarrow 0$ 
   $L_t \leftarrow \emptyset$ 
  while  $\text{time}() < \text{Time\_limit}$  do
     $L'_t, f'(L_t) \leftarrow \text{Heurística\_Constructiva}(\vec{\pi}, C, \mu, \alpha)$ 
    while True do
      for  $i \in L_t; j \in C \setminus L_t$  do
         $\Delta \leftarrow (\sum_{k \in L_t \setminus i} a_{jk} - \pi_j) - (\sum_{k \in L_t \setminus i} a_{ik} - \pi_i)$ 
        if  $\Delta > 0$  then
           $L'_t \leftarrow (L_t \setminus i) \cup j$ 
           $f'(L_t) \leftarrow f(L_t) + \Delta$ 
          break
        end if
      end for
      if  $\Delta = 0$  then
        break
      end if
    end while
    if  $f'(L_t) > f(L_t)$  then
       $L_t \leftarrow L'_t$ 
    end if
  end while
  return  $L_t$ 
end function

```

B. Heurística GRASP para la Asignación de Charlas a Sesiones

Una alternativa de solución diferente la constituye la utilización de procedimientos heurísticos con el fin de resolver el problema de forma global. Con este fin, se plantea la adaptación de los procedimientos presentados en los Algoritmos 1 y 2, y su incorporación dentro de una heurística GRASP. El procedimiento utilizado incorpora las ideas propuestas por Urosëvić [25]. En particular, se adoptan los

procedimientos de búsqueda descritos por el autor, consistentes en evaluar el efecto de intercambiar ya sea 2 o 3 charlas asignadas a sesiones diferentes. El Algoritmo 3 ilustra detalladamente las operaciones empleadas en la búsqueda de una solución apropiada a la agrupación de charlas en sesiones.

Algoritmo 3 Heurística GRASP usada para resolver el problema.

```

function GRASP_GLOBAL( $\vec{\pi}, C, \mu, \alpha$ )
   $f(Sol) \leftarrow 0$ 
   $Sol \leftarrow \emptyset$ 
   $\vec{\pi} \leftarrow 0$ 
  while  $\text{time}() < \text{Time\_limit}$  do
    while  $|C| > 0$  do
       $L_t \leftarrow \text{Heurística\_Constructiva}(\vec{\pi}, C, \mu, \alpha)$ 
       $C \leftarrow C \setminus L_t$ 
       $Sol' \leftarrow L_t$ 
    end while
     $f(Sol') \leftarrow \sum_{L_t \in Sol} \sum_{i, j \in L_t} a_{ij}$ 
    while True do
      for  $L_{t_1} \in Sol; L_{t_2} \in Sol$  do
        for  $i \in L_{t_1}; j \in L_{t_2}$  do
           $\Delta_1 \leftarrow (\sum_{k \in L_{t_1} \setminus i} a_{jk}) - (\sum_{k \in L_{t_1}} a_{ik})$ 
           $\Delta_2 \leftarrow (\sum_{k \in L_{t_2} \setminus j} a_{ik}) - (\sum_{k \in L_{t_2}} a_{jk})$ 
          if  $\Delta_1 + \Delta_2 > 0$  then
             $L_{t_1} \leftarrow (L_{t_1} \setminus i) \cup j$ 
             $L_{t_2} \leftarrow (L_{t_2} \setminus j) \cup i$ 
             $f(Sol') \leftarrow f(Sol') + \Delta_1 + \Delta_2$ 
            break
          end if
        end for
      end for
      if  $\Delta_1 + \Delta_2 \leq 0$  then
        break
      end if
    end while
    if  $f'(Sol') > f(Sol)$  then
       $Sol \leftarrow Sol'$ 
    end if
  end while
  return  $Sol$ 
end function

```

Inicialmente el algoritmo construye una solución inicial usando como criterio la contribución que este haría a la función objetivo. Con este fin, se genera una lista RCL de charlas candidatas a ser asignadas a un determinado grupo, conformada por aquellas que se encuentren por encima del $100(1 - \mu)\%$ de aquellos con más potencial para incrementar la función objetivo. Luego se selecciona entre estos uno de forma aleatoria y se añade a la sesión siendo considerada. Se repite este proceso, agregando charlas a las sesiones, una a una, hasta que estas últimas completen su capacidad o no haya más charlas requiriendo ser asignadas. Posteriormente, se aplican procedimientos de búsqueda local sobre las soluciones inicialmente generadas, consistentes en la exploración de un vecindario conformado por soluciones en las cuales las dos charlas han cambiado su posición, encontrándose asignadas a sesiones diferentes. En el método considerado, una solución se mueve hacia otra, cada vez que se encuentra que dicho movimiento mejora la función objetivo (*First improvement*).

V. RESULTADOS Y DISCUSIÓN

Con el fin de evaluar los resultados obtenidos a través del enfoque propuesto se plantea su utilización en la generación de la agenda para el evento *International Conference on Production Research 2018 (ICPR Americas 2018)*. Adicionalmente, y con el fin de evaluar rigurosamente la

aplicabilidad de este, se plantea utilizar instancias clásicas, aunque modificadas para representar las características realistas del problema, que permitan evaluar objetivamente su rendimiento. Las instancias utilizadas corresponden a un subconjunto de las propuestas por Fan *et al.* [26], seleccionando en particular aquellas con entre 120 y 240 elementos (en este caso charlas) que deben ser agrupadas en sesiones de tamaño 3 y 4, respectivamente (que difieren de los tamaños de grupo originalmente propuestos por los autores). Para cada par de charlas, se conoce el beneficio que trae pertenecer a un mismo grupo. Tal y como lo reportan las instancias seleccionadas dicho valor corresponde a un número entero asignado de acuerdo con una distribución uniforme $U(0,100)$.

Todos los métodos han sido implementados usando el lenguaje de programación Julia 0.6. Los modelos han sido implementados a través de JuMP y el software de optimización Gurobi 7.0 se ha utilizado en la resolución de estos. Un tiempo límite de 3600s se impone cuando el método de solución adoptado es programación lineal. En contraste, los métodos heurísticos, es decir el GRASP, se detienen cuando alcanzan un total de 10s y 30s (en instancias de 120 y 240 charlas respectivamente) o cuando no existen más columnas disponibles en el caso de CG.

A. Resultados en Instancias de Prueba

En consideración al hecho que los procedimientos heurísticos propuestos presentan una componente aleatoria, se plantea inicialmente explorar el efecto del parámetro μ sobre el valor alcanzado por la función objetivo. De esta forma, se espera seleccionar aquel valor que, en efecto, permita explotar el máximo potencial de las heurísticas propuestas. Con este fin, se ejecutan hasta 5 experimentos por cada instancia y por cada valor de $\mu \in \{0.1, 0.2, \dots, 1.0\}$ se almacena el mejor valor hallado de la función objetivo.

La Figura 3 ilustra el valor promedio que se obtiene de la función objetivo para diferentes valores de la constante μ . Como se puede observar, es difícil concluir con claridad acerca del valor que, en efecto, produce los mejores resultados. Además, hay poca evidencia estadística que respalde la hipótesis de que hay un solo valor con mejor rendimiento que los demás. Por tal razón, para seleccionar solo uno que sea definitivamente sobresaliente respecto a los demás, se sigue como criterio el mejor resultado promedio sobre todos los experimentos, correspondiente a un valor de $\mu = 0.1$.

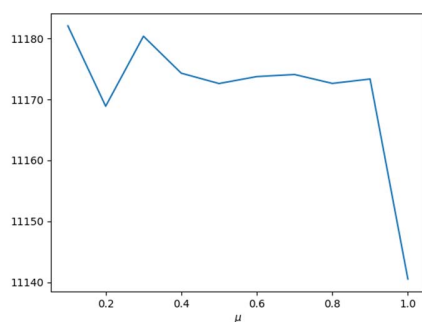


Fig. 3. Objetivo promedio obtenido para diferentes valores de μ .

La Tabla I presenta los resultados obtenidos para la fase de asignación de charlas a sesiones. En particular se analizan los

resultados correspondientes a la mejor solución hallada por cada uno de los métodos y modelos propuestos, de acuerdo con el criterio de parada establecido y enfocado exclusivamente en las instancias de hasta 120 charlas. Para cada uno de los métodos, la Tabla I presenta los resultados obtenidos en términos de la función objetivo (Obj), la brecha porcentual a la mejor solución posible (gap%) y el tiempo requerido para alcanzar dicha solución (Tiempo (s)). Posteriormente, estos resultados se comparan frente a la mejor solución obtenida para la relajación lineal del Modelo (12)-(14), que puede usarse como una cota superior de la función objetivo. Los resultados sugieren que existe una superioridad en términos de calidad en las soluciones halladas mediante la metaheurística GRASP y CG. Estos dos métodos obtuvieron consistentemente resultados más cercanos a la cota superior (CG Bound) que los obtenidos mediante la aplicación de programación entera (ILP). Los resultados indican que el gap frente a la solución óptima que presentan las soluciones heurísticas es bajo, típicamente inferior al 3%, y que corresponde a una solución aceptable dentro del contexto del problema estudiado.

La Tabla II muestra los resultados de la Fase I y la Fase II para la metaheurística GRASP propuesta y CG sobre las instancias de mayor tamaño con hasta 240 charlas que se deben agrupar en sesiones de tamaño 4. Los resultados sugieren que en la medida que las instancias se hacen más grandes la eficiencia del GRASP sobrepasa los resultados alcanzados mediante la generación de columnas heurística propuesta. De hecho, no solo la calidad de las soluciones es inferior, sino que también los tiempos de cómputo crecen significativamente, siendo necesario detener el proceso tras 3600s de ejecución. La heurística GRASP, en contraparte, produce resultados que son consistentemente superiores en calidad y en tiempos de ejecución de tan sólo 30 segundos, usados como criterio de parada en este grupo de instancias. Un análisis detallado de las causas del ralentizamiento de CG sugiere que hay una mayor tendencia a requerir el uso de métodos exactos para resolver el PA. Una posible estrategia para solucionar este inconveniente es la utilización de métodos más sofisticados en la resolución de PA, evitando que se recurra a métodos que afecten de manera notoria la convergencia a una solución óptima de la relajación lineal.

En cuanto a los resultados de la fase final, se explora exclusivamente la utilización de programación entera como estrategia de solución y se compara contra una modificación de la heurística GRASP propuesta, en la cual los criterios de selección son adaptados para resolver un problema de minimización. Se utilizan los resultados de las agrupaciones obtenidos mediante la heurística GRASP adoptada para la Fase I. Un tiempo límite de ejecución de 30s es impuesto a la heurística GRASP mientras que 3600s se imponen como tiempo límite de ejecución de la CG.

Los resultados en la Tabla II, sugieren que aun cuando en la fase II el problema es significativamente más pequeño, todavía se requiere un considerable esfuerzo computacional para hallar una solución óptima. De hecho, en ningún caso el software de optimización utilizado pudo hallar soluciones que se encontrarán cercanas en calidad a las obtenidas mediante el enfoque GRASP propuesto. Por desgracia, la calidad de la cota obtenida mediante programación entera fue muy débil

TABLA I
RESULTADOS EXPERIMENTALES PARA LA FASE I

Instance	CG Bound		LP			GRASP			CG		
	Obj	Obj	gap (%)	Tiempo (s)	Obj	gap (%)	Tiempo (s)	Obj	gap (%)	Tiempo (s)	
RanInt_n120_ss3_01	11404.7	9333	22.0%	TL	11204	1.8%	10	11040	3.3%	10.346	
RanInt_n120_ss3_02	11492.9	9705	18.0%	TL	11265	2.0%	10	11229	2.4%	10.023	
RanInt_n120_ss3_03	11480.4	9189	25.0%	TL	11239	2.1%	10	11181	2.7%	7.632	
RanInt_n120_ss3_04	11459.8	9057	27.0%	TL	11338	1.1%	10	11150	2.8%	29.984	
RanInt_n120_ss3_05	11426.0	9472	21.0%	TL	11212	1.9%	10	11374	0.5%	8.742	
RanInt_n120_ss3_06	11425.3	9118	25.0%	TL	11171	2.3%	10	11077	3.1%	9.963	
RanInt_n120_ss3_07	11454.4	9060	26.0%	TL	11162	2.6%	10	11363	0.8%	8.855	
RanInt_n120_ss3_08	11475.6	9120	26.0%	TL	11274	1.8%	10	11403	0.6%	5.073	
RanInt_n120_ss3_09	11468.9	9061	27.0%	TL	11232	2.1%	10	11169	2.7%	6.718	
RanInt_n120_ss3_10	11489.8	9313	23.0%	TL	11245	2.2%	10	11482	0.1%	5.394	

TABLA II
RESULTADOS FASE I Y FASE II USANDO GRASP Y MÉTODOS EXACTOS

Instancia	FASE I			FASE II		
	GRASP	CG	Δ	GRASP	IP solver	Δ
RanInt_n120_ss3_01	11204	11040	1,5%	20115	25680	-27,7%
RanInt_n120_ss3_02	11265	11229	0,3%	20091	26388	-31,3%
RanInt_n120_ss3_03	11239	11181	0,5%	20436	25630	-25,4%
RanInt_n120_ss3_04	11188	11150	0,3%	20862	26109	-25,2%
RanInt_n120_ss3_05	11212	11374	-1,4%	20587	25827	-25,5%
RanInt_n120_ss3_06	11171	11077	0,8%	20215	25912	-28,2%
RanInt_n120_ss3_07	11162	11363	-1,8%	20196	25531	-26,4%
RanInt_n120_ss3_08	11274	11403	-1,1%	20471	25380	-24,0%
RanInt_n120_ss3_09	11232	11169	0,6%	20648	26508	-28,4%
RanInt_n120_ss3_10	11245	11482	-2,1%	20706	26322	-27,1%
RanInt_n240_ss4_01	32871	32823	-1,4%	100974	108718	-7,12%
RanInt_n240_ss4_02	32265	32260	0,0%	99598	108003	-8,4%
RanInt_n240_ss4_03	33084	32290	2,4%	101493	108528	-6,9%
RanInt_n240_ss4_04	32802	32239	1,7%	101241	108839	-7,5%
RanInt_n240_ss4_05	32945	32378	1,7%	101981	108074	-6,0%
RanInt_n240_ss4_06	32753	32377	1,1%	101522	108872	-7,2%
RanInt_n240_ss4_07	32878	32437	1,3%	100453	106821	-6,3%
RanInt_n240_ss4_08	32792	31575	3,7%	100997	108684	-7,6%
RanInt_n240_ss4_09	32729	32439	0,9%	100396	108635	-8,2%
RanInt_n240_ss4_10	32891	31268	4,9%	101863	109311	-7,3%

limitando así la posibilidad de cuantificar la calidad de la solución obtenida por el método propuesto. La diferencia porcentual entre la mejor solución obtenida por la heurística GRASP es comparada contra la mejor obtenida mediante IP. Los resultados bajo esta métrica siempre se muestran favorables hacia el uso de la metaheurística, indicando que, en efecto, es una alternativa apropiada para el diseño de las agendas. *Caso de Estudio ICPR Americas 2018*

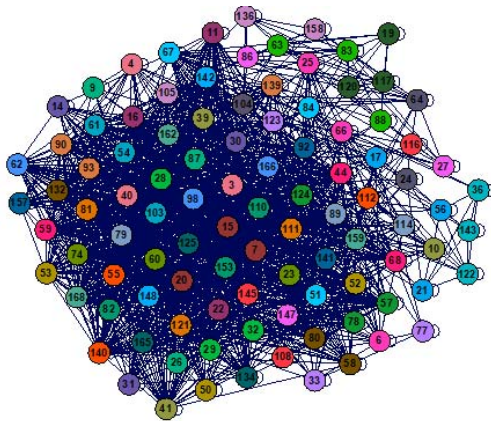
El paso final de la evaluación del método propuesto consiste en su evaluación para generar la agenda de presentaciones de un evento internacional de talla mediana en el que múltiples idiomas de presentación son permitidos. El evento, especializado en tópicos de producción y logística, recibe para su octava edición alrededor de 170 artículos de los cuales 101 son aceptados para su presentación en las sesiones paralelas. Por decisión de los organizadores, se deben organizar las charlas en 9 bloques de sesiones paralelas tal que hasta 4 presentaciones se pueden llevar a cabo de forma simultáneas. En cada sesión serán asignadas hasta 3 presentaciones, para las cuales es deseable que correspondan a tópicos similares. Para el cálculo de la similitud entre cada par de charlas se utiliza específicamente los términos de

clasificación brindadas por los autores, los *keywords* y algunos otros sugeridos por los evaluadores en los casos en los que poca información, fuera del artículo en sí mismo, estaban disponibles. Concretamente, se asignan un punto a los valores α_{ij} por cada término que aparece simultáneamente en un par de artículos asociados, respectivamente, a las charlas i y j . Aunque la situación más deseable sería la utilización del contenido del texto, el multilingüismo propio de la conferencia hacía difícil la labor.

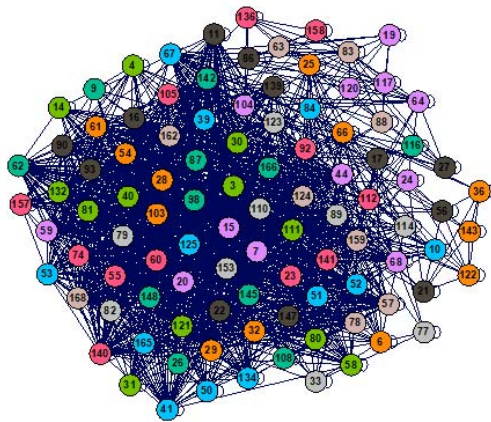
La Figura 4 representa, mediante un grafo, los artículos (nodos) presentados a la conferencia y que aparecen en el cronograma final. En la gráfica, la existencia de un arco entre dos charlas significa que comparte contenido en común; cada arco tiene un peso, representando el grado de similitud entre ambas charlas. El grafo ha sido generado tomando en cuenta el peso de los arcos, así pues, se espera que los elementos que se encuentran lejanos en el grafo contengan temáticas disímiles.

Tras la ejecución de la Fase I los artículos son agrupados en sesiones con base en su similitud. En la Figura 4a los colores de los nodos representa la sesión a la cual cada charla

ha sido asignada. Como puede observarse, los artículos de contenido similar (ceranos en el grafo) tienden a ser asignados a la misma sesión, confirmando la efectividad del enfoque propuesto. Un resultado similar se presenta en la Figura 4b. En este último caso, a diferencia del anterior, el color de los nodos representa la franja horaria en la que las charlas han sido asignadas. Se puede identificar en los resultados que los grupos de charlas cuyo contenido es similar tienden a ser asignados a franjas horarias diferentes, facilitando así la asistencia de los participantes a las charlas en los tópicos alineados con sus intereses.



a. Agrupamiento de charlas en sesiones (colores)



b. Agrupamiento de sesiones en bloques (colores)

Fig. 4. Representación gráfica de la agenda de la conferencia ICPR Americas 2018.

VI. CONCLUSIONES

En este trabajo se explora el diseño de métodos computacionales para facilitar el proceso de generación de agendas para conferencias. Aunque es un problema bastante común dentro de la comunidad científica, se observan pocos reportes manifestando la utilización de estrategias similares en la práctica. Esta primera aproximación permite acercarse a las características del problema para identificar criterios que puedan utilizarse a futuro con el propósito de adaptar los métodos propuestos a las expectativas de quienes participan u organizan este tipo de eventos.

En particular se identifican, por un lado, métodos adecuados para abordar el problema de forma eficiente y posibles criterios para extenderlos a casos, no necesariamente más complejos, en los que la perspectiva de quienes participan también pueda ser tomada en cuenta. En el futuro, el presente trabajo podría extenderse para manejar otros aspectos como conflictos de horarios o solicitudes especiales de los participantes que constituyen un desafío y que, en la práctica, constituyen la norma y no la excepción en este tipo de eventos. Similarmente, podrían considerarse aspectos como la selección de los salones para las sesiones considerando aspectos como su ubicación y/o distancia entre estos o evitar el agendamiento simultáneo de charlas populares.

Algunas lecciones quedan tras la adopción del enfoque propuesto en una aplicación real. Un primer aspecto lo constituye la calidad de la clasificación suministrada por los autores, siendo un punto que en efecto debilita la programación e indica la relevancia de considerar el contenido en la generación de la agenda. En segundo lugar encontramos que existe una fuerte subjetividad en lo que se considera una agenda apropiada para el problema. Lo anterior indica la necesidad de considerar el problema en un contexto multiobjetivo en el que los organizadores puedan determinar, dentro de una base de soluciones posibles al problema, aquellas que representen un mejor *trade-off* entre las expectativas en diferentes aspectos, e.g. cruces horarios, preferencias de los asistentes, similitud de las sesiones, etc., para la selección de un programa que satisfaga estos múltiples criterios.

AGRADECIMIENTOS

Esta investigación ha sido financiada parcialmente por el Departamento Administrativo de Ciencia, Tecnología e Innovación (COLCIENCIAS) mediante la Beca 784-2018 de apoyo a estancias postdoctorales.

REFERENCIAS

- [1] F. L. L. de Leon and B. McQuillin, "The Role of Conferences on the Pathway to Academic Impact: Evidence from a Natural Experiment," *SSRN Electron. J.*, 2014.
- [2] T. Stidsen, D. Pisinger, and D. Vigo, "Scheduling EURO-k conferences," *Eur. J. Oper. Res.*, vol. 0, pp. 1–10, 2017.
- [3] G. M. Thompson, "Improving conferences through session scheduling," *Cornell Hotel Restaur. Adm. Q.*, vol. 43, no. 3, pp. 71–76, 2002.
- [4] B. Vangerven, A. M. C. Ficker, D. R. Goossens, W. Passchyn, F. C. R. Spieksma, and G. J. Woeginger, "Conference scheduling - A personalized approach," *Omega (United Kingdom)*, vol. 0, pp. 1–10, 2017.
- [5] G. Pierce, "The dilemma of attending (or not) scientific conferences.," *Can. J. Physiol. {&} Pharmacol.*, vol. 92, no. 1, pp. v-- v, 2014.
- [6] A. Tesch, "Optimization of Large-Scale Conference Schedules," Technische Universität Berlin, 2013.
- [7] R. Eglese and G. Rand, "Conference seminar timetabling," *J. Oper. Res. Soc.*, vol. 38, no. 7, pp. 591–598, 1987.
- [8] M. Tanaka, Y. Mori, and A. Bargiela, "Granulation of Keywords into Sessions for Timetabling Conferences," *Proc. Soft Comput. Intell. Syst. (SCIS 2002)*, pp. 1–5, 2002.
- [9] B. Vangerven, A. M. C. Ficker, D. R. Goossens, W. Passchyn, F. C. R. Spieksma, and G. J. Woeginger, "Conference scheduling - A personalized approach," *Omega (United Kingdom)*, 2017.

- [10] Y. Le Page, "Optimized Schedule for Large Crystallography Meetings," in *Journal of Applied Crystallography*, 1996, vol. 29, no. 3, pp. 291–295.
- [11] M. G. Nicholls, "A small-to-medium-sized conference scheduling heuristic incorporating presenter and limited attendee preferences," *J. Oper. Res. Soc.*, vol. 58, no. 3, pp. 301–308, 2007.
- [12] E. B. Edis and R. S. Edis, "An Integer Programming Model for the Conference Timetabling Problem," *C.B.U. J. Sci.*, vol. 9, no. 2, pp. 55–62, 2013.
- [13] R. F. Potthoff and M. C. . Munger, "Use of Integer Programming to Optimize the Scheduling of Panels at Annual Meetings of the Public Choice Society," *Public Choice*, vol. 117, no. 1, pp. 163–175, 2003.
- [14] M. Gulati and A. Sengupta, "TRACS - tractable conference scheduling," in *Proceedings of the decision sciences institute annual meeting (DSI 2004)*, 2004, pp. 3161–3166.
- [15] A. Huang, "Similarity measures for text document clustering," *Proc. Sixth New Zeal.*, no. April, pp. 49–56, 2008.
- [16] P. C. Gilmore and R. E. Gomory, "A Linear Programming Approach to the Cutting-Stock Problem," *Oper. Res.*, vol. 9, no. 6, pp. 849–859, 1961.
- [17] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, "Branch-and-Price: Column Generation for Solving Huge In teger Programs," *Oper. Res.*, vol. 46, no. 9, pp. 316–329, 1998.
- [18] R. Martí, M. Gallego, A. Duarte, and E. G. Pardo, "Heuristics and metaheuristics for the maximum diversity problem," *Journal of Heuristics*, vol. 19, no. 4, pp. 591–615, 2013.
- [19] F. Castaño, E. Bourreau, A. Rossi, M. Sevaux, and N. Velasco, "Partial target coverage to extend the lifetime in wireless multi-role sensor networks," *Networks*, 2016.
- [20] F. Castaño and M. Sevaux, "Combining Metaheuristics with Column Generation : Successful Approaches to Enhance Column Generation Algorithms Performance.", *Proceedings of the Sixth International Workshop on Model-based Metaheuristic (Mathheuristics 2016)*
- [21] F. Castaño, E. Bourreau, N. Velasco, A. Rossi, and M. Sevaux, "Exact approaches for lifetime maximization in connectivity constrained wireless multi-role sensor networks," *Eur. J. Oper. Res.*, 2015.
- [22] T. A. Feo and M. G. C. Resende, "Greedy Randomized Adaptive Search Procedures," *J. Glob. Optim.*, vol. 6, no. 2, pp. 109–133, 1995.
- [23] C. Rodrigues Costa, V. Rosset, and M. Cristina Vasconcelos Nascimento, "A GRASP for the Steiner tree problem in graphs to support multicast routing," *IEEE Lat. Am. Trans.*, vol. 13, no. 12, pp. 3873–3878, 2015.
- [24] I. Cartes Rubilar and R. Medina Duran, "A GRASP algorithm for the elective surgeries scheduling problem in a Chilean public hospital," *IEEE Lat. Am. Trans.*, vol. 14, no. 5, pp. 2333–2338, 2016.
- [25] D. Urošević, "Variable neighborhood search for maximum diverse grouping problem," *Yugosl. J. Oper. Res.*, vol. 24, no. 1, pp. 21–33, 2014.
- [26] Z. P. Fan, Y. Chen, J. Ma, and S. Zeng, "A hybrid genetic algorithmic approach to the maximally diverse grouping problem," *J. Oper. Res. Soc.*, vol. 62, no. 1, pp. 92–99, 2011.



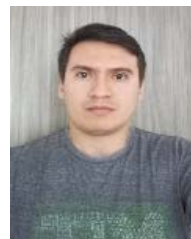
Fabián Castaño doctor en Ciencia y Tecnología de la Información y la Comunicación otorgado por la Université de Bretagne-Sud (Francia) e Ingeniería, otorgado por la Universidad de los Andes (Colombia). Cuenta con amplia experiencia en el área de optimización con foco en aplicaciones diversas que van desde el diseño de sistemas logísticos, pasando por logística hospitalaria y de transporte, hasta la

optimización de la operación de sistemas electrónicos. Su investigación busca la aplicación de herramientas de modelamiento matemático, simulación y algoritmos avanzados de optimización para dar solución o mejorar la operación de sistemas diversos. Actualmente es profesor asistente en el Departamento de Ingeniería Civil e Industrial en la Pontificia Universidad Javeriana sede Cali.



Nubia Velasco posee un doctorado en Automática e Informática Aplicadas de la Universidad de Nantes (Francia), actualmente es profesora asociada de la Facultad de Administración de la Universidad de los Andes y previamente del Departamento de Ingeniería Industrial de la misma institución. Ha liderado múltiples proyectos de investigación en

las áreas de logística y optimización al igual que dirigido y co-dirigido estudiantes doctorales mediante programas de financiación nacionales e internacionales a través de programas de doble titulación. Sus principales intereses son los algoritmos evolutivos, las metaheurísticas y la optimización, aplicados a los problemas logísticos en particular a los problemas de ruteo de vehículos, de la cadena de abastecimiento y de los sistemas de salud.



Jimmy Carvajal-Beltrán recibió su pregrado en ingeniería industrial de la universidad de Ibagué en el año 2009. En el año 2011 recibió grado de maestría en ingeniería industrial de la universidad de los Andes, en el énfasis de investigación de operaciones y estadística. Ha sido profesor e investigador de ingeniería industrial de varias universidades en Colombia, ha participado en diferentes congresos internacionales. Actualmente se encuentra en su segundo año del doctorado en ingeniería –Industria y organizaciones- de la universidad Nacional de Colombia sede Manizales.