Inteligencia Artificial

Estado del Arte: Problema OSAP

Diego Esteban Rosales León

September 30, 2022

Evaluación

Resumen (5%):	___________
Introducción (5%):	___________
Definición del Problema (10%):	___________
Estado del Arte (35%):	___________
Modelo Matemático (20%):	___________
Conclusiones (20%):	___________
Bibliografía (5%):	___________
Nota Final (100%):	___________

Abstract

En las grandes organizaciones existe el desafío respecto a la distribución del espacio físico de las oficinas, cubículos, salas, entre otros, de la manera más eficiente posible. Este reto es llamado *Office Space Allocation Problem* (con sus siglas, OSAP), que consiste en asignar entidades (máquinas, personas, roles, etc.) a un grupo de habitaciones disponibles, rigiendóse por diferentes tipos de restricciones para así optimizar el uso del espacio. Este paper definirá el problema, a través de la revisión de diferentes modelos existentes que se usan para resolver este tipo de obstáculos, para finalmente formular un modelo matemático que permita analizar cómo se desenvuelven las variables del sistema, con el objetivo de minimizar todo espacio que esté mal distribuido, es decir, el espacio que no se esté ocupando y el que se esté sobre utilizando.

1 Introducción

En este paper se abordará un problema común y corriente en cualquier tipo de organización o empresa, relacionado a la distribución eficiente del espacio físico de todas sus instalaciones, conocido como OSAP. Su importancia radica en que la optimización del espacio impacta directamente en la productividad. El objetivo versa sobre entender detalladamente este problema en todas sus aristas, para buscar diferentes vías que puedan solucionarlo. Para lo anterior, se definirá este problema en específico, se hará una revisión exhaustiva del estado del arte hasta la fecha, se implementará un modelo matemático basal que guiará al hallazgo de la solución definitiva, para luego dar paso a las conclusiones obtenidas de este análisis.

2 Definición del Problema

OSAP se define como un problema de distribución para ciertos espacios finitos en base a diferentes tipos de parámetros (recursos) [1], con el objetivo principal de minimizar el espacio mal utilizado (no utilizado y sobre utilizado) y, a su vez cumplir con restricciones o límites designados. A saber, existen dos tipos de restricciones a considerar: duras o fuertes, que son limitantes que siempre se deben satisfacer y, las blandas, que son condiciones que pueden llegar a ser un óbstaculo inamovible si no se evaden, debido a que están directamente relacionadas con la calidad del problema, es decir, entre más soluciones blandas sean infringidas, peor será la solución encontrada. La solución más factible y óptima será la que cumpla con todas las restricciones duras y eluda cabalmente las restricciones blandas. Para lograr lo anterior, se utilizarán diferentes variables, tales como (1) total de entidades (personas, máquinas, etc.), (2) cantidad de pisos y habitaciones y (3) las restricciones anteriormente mencionadas. Además, es posible observar diferentes dificultades, como la obligación de que toda entidad sea ubicada en una habitación, hayan distancias específicas entre cada una o que se tengan que agrupar bajo ciertos criterios, entre otras. Un modelo clásico para resolver el OSAP, está relacionado al método de la búsqueda local [6], al que en versiones posteriores se le fueron introduciendo otras variables para mejorar o modificar su eficacia, como la mezcla de diferentes algoritmos o el uso Tabu Search [4]). Por otro lado, es necesario mencionar que surgieron opciones menos convencionales como la utilización de herramientas harmónicas para la búsqueda de soluciones [1] o adaptaciones del metódo Hill Climbing (ejemplo: Late Acceptance Hill Climbing) [3].

3 Estado del Arte

La búsqueda de la distribución eficaz del espacio y los objetos en un lugar determinado, no tiene fecha de origen exacta, pero es posible observarla de manera global cuando las personas compran muebles, adornan su casa, designan habitaciones a diferentes integrantes de la familia, etc., puesto que cada centímetro disponible llega a ser decisivo. Esto se puede extrapolar a las empresas, organizaciones e instituciones existentes, por lo que surge de manera natural la necesidad de una herramienta tecnológica que resuelva este tema. Y por ende, se denominó este problema como OSAP en pos de aunar los esfuerzos académicos contra un enemigo común.
De lo anterior surgen diferentes métodos de búsqueda que analizan algoritmos y metodologías para buscar una solución, tales como:

1. **Late Acceptance Hill Climbing (LAHC):** Tiene un enfoque similar a Hill Climbing (HC) o a Simulated Annealing (SA). El primero obtiene una solución y la compara con otras alternativas hasta encontrar la más adecuada; el segundo utiliza el mismo proceso pero evita quedarse estático en un mínimo local. Ahora bien, LAHC se basa en lo anterior pero agrega más criterios de aceptación para la resolución del problema, lo que lo hace más completo. Sin embargo, se debe considerar que a mayor cantidad de criterios, mayor tiempo de búsqueda y comparación entre los mismos [3].

2. **Tabu Search:** Esta opción se enfoca en guardar los movimientos de búsqueda que ya fueron realizados, debido a que cualquier movimiento nuevo debe ignorar lo previamente calculado, es decir, se buscarán nuevas soluciones sin visitar las ya existentes. Si bien suena ventajoso, su mayor dificultad consiste en el ahorro de tiempo, debido a que puede lograr tener una cantidad de iteraciones.

3. **Hybrid Meta Heuristics (HMH):** Si bien la metaheurística es un método heurístico para resolver un problema, la HMH le suma la optimización del tiempo, permitiendo mejoras generales a gran escala. Su mayor óbstaculo se relaciona con lo dificultoso que es integrar ambas metodologías de manera óptima [2].

4. **Harmonic Search (HS):** Es una metaheurística de optimización que fue creada para observar el nivel de improvisación que tienen los músicos. Consiste en que hay una memoria harmónica y que guarda la solución creada, la misma esta basada en la consideración de la memoria, la improvisacion sobre los instrumentos y el tono producido [5]. Según Awadallah, Tajudín y Azmi [1], el uso de HS en OSAP, puede ser el equivalente a considerar los espacios físicos como una memoria, elegir al azar las habitaciones como improvisación y seleccionar la habitación con menos penalización de las restricciones como el tono. Si bien es una solución innovadora, su dificultad radica en la incertidumbre de su realización.

Por otro lado, dentro del estado del arte, existe un estudio, de Lopes y Girimonte [6], que hace una comparación entre estos cuatro tipos de metaheurísticas ya mencionadas: Hill Climbing (HC), Simulated Annealing (SA), Tabu Search (TS), Hybrid Meta heuristic (HMH), con el objetivo de resolver un OSAP basado en "Investigación Espacial Europea" (Europe Space Agency), donde evaluaron cada algoritmo obteniendo la siguiente tabla comparativa:

Largo	10000 iteraciones			
Algoritmo	HC	SA	TS	HMH
Espacio mal usado (f_1)	449.37	440.26	430.08	416.81
Restricciones blandas (f_2)	203.57	201.87	206.22	209.52
Total (f)	652.93	642.13	636.30	**626.33**
Largo	5000 iteraciones			
Algoritmo	HC	SA	TS	HMH
Espacio mal usado (f_1)	436.08	440.11	450.11	424.42
Restricciones blandas (f_2)	209.42	213.17	209.82	207.88
Total (f)	645.50	653.28	659.93	**632.30**
Largo	1050 iteraciones (10*n)			
Algoritmo	HC	SA	TS	HMH
Espacio mal usado (f_1)	442.36	451.93	438.87	432.68
Restricciones blandas (f_2)	242.51	231.40	219.54	244.19
Total (f)	684.87	682.32	**658.40**	676.87

Table 1: Resultados obtenidos de los cuatro métodos mencionados anteriormente para [6]

En los dos primeros casos de la tabla anterior (5000 y 10000 iteraciones) se puede apreciar que HMH tiene un rendimiento superior al resto de las metaheurísticas, debido al tamaño de las iteraciones (entre más iteraciones mejor), pero, cuando se tiene una menor cantidad de iteraciones, como en el tercer caso (1050 iteraciones), HMH tiene un menor rendimiento que TS, debido a que este último converge a una mayor rapidez por sobre cualquier otro método. Mientras que HC y SA, en cualquiera de los tres casos, siempre tienen valores similares. Sin embargo, es necesario mencionar que las diferencias de desempeño no son abruptamente lejanas, por lo que para grandes escenarios HMH sobresale como mejor opción y para pequeños escenarios predomina TS.

Finalmente, en el estado del arte no queda claro el futuro de los estudios e investigaciones respecto a una metodología de búsqueda que resuelva el OSAP, debido a la variedad de opciones y la cantidad de restricciones y factores a contemplar, por lo que sigue en vigencia el desarrollo de una herramienta de solución única.

4 Modelo Matemático

A continuación, se presentará un modelo matemático basado en la explicación de Francisco Castillo, María Cristina Riff y Elizabeth Montero [4], siendo necesario considerar que tiene un enfoque similar al de Ülker y Landa Silva [7]. Además existen alternativas de solución con métodos diferentes [1], [3] y [6] que no serán considerados por su falta de atingencia a las etapas posteriores de este paper.

4.1 Parámetros

Primero, se definen todas las variables respectivas a utilizar:

$R \rightarrow$ Conjunto de Habitaciones del edificio.
$E \rightarrow$ Conjunto de Entidades.
$J \rightarrow$ Conjunto de todas las restricciones, $J = \{as, asp, mh, dh, hnc, ady, crc, ljn, cap\}$.
$S_e \rightarrow$ Capacidad requerida de la entidad $e, \forall e \in E$.
$S_r \rightarrow$ Capacidad de la habitación $r, \forall r \in R$.
$A_{dr} \rightarrow$ Lista de las habitaciones adyacentes a $r, \forall r \in R$.
$C_{cr} \rightarrow$ Lista de las habitaciones cercanas a $r, \forall r \in R$.

Luego, se definen los parámetros para las restricciones duras y blandas:

$HC^j \rightarrow$ Conjunto de las restricciones duras $j, \forall j \in J$.
$SC^j \rightarrow$ Conjunto de las restricciones blandas $j, \forall j \in J$.

4.2 Variables

Las variables a utilizar son de dos tipos: una para la asignación de variables y la otra para medir si las restricciones blandas fueron infringidas o no.

$$x_{er} = \begin{cases} 1 & \text{si la entidad } e \text{ es asignada a la habitación } r, \ \forall e \in E, r \in R \\ 0 & \text{caso contrario.} \end{cases}$$

$$y_i^j = \begin{cases} 1 & \text{si la restricción } i \text{ del tipo } j \text{ fue infringida}, \ \forall i \in |SC^j|, j \in J \\ 0 & \text{caso contrario.} \end{cases}$$

4.3 Restricciones

En [4] se realizaron generalizaciones para nueve tipos de restricciones, las que pueden ser consideradas duras o blandas, (deben ser satisfechas y es deseable que sean satisfechas, respectivamente), siendo importante minimizar esta última que es un objetivo del problema, debido a que mejorará la solución a encontrar.

4.3.1 Restricciones Duras

1. Esta restricción se tiene que cumplir y no puede ser considerada como una restricción blanda. Toda entidad e tiene que tener asignada una habitacion r:

$$\sum_{r \in R} x_{er} = 1, \forall e \in E \tag{1}$$

2. Asignación: La entidad e tiene que estar asignada a una habitación r

$$x_{er} = 1 \tag{2}$$

3. Asignación Prohibida: La entidad e no tiene que ser asignada a una habitación r.

$$x_{er} = 0 \tag{3}$$

4. Misma Habitación: Dos entidades e_1 y e_2 tienen que estar en la misma habitación.

$$x_{e_1r} = 1, x_{e_2r} = 1,\ \forall r \in R$$

$$x_{e_1r} - x_{e_2r} = 0,\ \forall r \in R \tag{4}$$

5. Distinta Habitación: Dos entidades e_1 y e_2 no tienen que estar en la misma habitación.

$$x_{e_1r} = 1 \leftarrow x_{e_2r} = 0,\ \forall r \in R$$

$$x_{e_1r} + x_{e_2r} \leq 1,\ \forall r \in R \tag{5}$$

6. Habitación no compartida: La entidad e no puede compartir habitación con ninguna otra entidad

$$\sum_{f \in\ E-e} x_{fr} \leq (|E|-1) - (|E|-1)x_{er},\ \forall r \in R \tag{6}$$

7. Adyacencia: Las entidades e_1 y e_2 deben asignarse en habitaciones adyacentes.

$$x_{e_1r} \leq \sum_{s \in A_{dr}} x_{e_2s} \leq 1,\ \forall r \in R \tag{7}$$

8. Cercanía: Las entidades e_1 y e_2 deben asignarse en habitaciones cercanas.

$$x_{er} \leq \sum_{s \in C_{cr}} x_{fs} \leq 1,\ \forall r \in R \tag{8}$$

9. Lejanía: Las entidades e_1 y e_2 deben asignarse lejos la una de la otra.

$$0 \leq \sum_{s \in C_{cr}} x_{e_2s} \leq 1 - x_{e_1r},\ \forall r \in R \tag{9}$$

10. Capacidad: La habitación r no debe superar cierta capacidad respecto a las entidades asignadas en ella. (Sobre uso)

$$\sum_{e \in E} S_e x_{er} \leq S_r \tag{10}$$

4.3.2 Restricciones Blandas

1. Asignación:

$$y_i^{as} = 1 - x_{er} \tag{11}$$

2. Asignación Prohibida:

$$y_i^{asp} = x_{er} \tag{12}$$

3. Misma Habitación:

$$y_{ir}^{mh} - 1 \leq x_{e_1r} - x_{e_2r} \leq 1 - \epsilon + \epsilon y_{ir}^{mh},\ \forall r \in R. \tag{13}$$

$$y_i^{mh} = \sum_{r \in R} y_{ir}^{mh} \tag{14}$$

4. Distinta Habitación:

$$(1+\epsilon)-(1+\epsilon)y_{ir}^{dh} \leq x_{e_1r}-x_{e_2r} \leq 2-y_{ir}^{dh},\ \forall r \in R \tag{15}$$

$$y_i^{dh} = \sum_{r\in R}(1-y_{ir}^{dh}) \tag{16}$$

5. Habitación no compartida:

$$(|E|-1)(2-x_{er}-y_{ir}^{hnc}) \geq \sum_{f\in\ E-e} x_{fr},\ \forall r \in R \tag{17}$$

$$\sum_{f\in\ E-e} x_{fr} \geq (|E|-1)(1-x_{er})+\epsilon-(|E|-1+\epsilon)y_{ir}^{hnc},\ \forall r \in R \tag{18}$$

$$y_i^{hnc} = \sum_{r\in R}(1-y_{ir}^{hnc}) \tag{19}$$

6. Adyacencia:

$$y_{ir}^{ady}+x_{e_1r}-1 \leq \sum_{s\in A_{dr}} x_{e_2s} \leq x_{e_1r}-\epsilon+(1+\epsilon)y_{ir}^{ady},\ \forall r \in R \tag{20}$$

$$y_i^{ady} = \sum_{r\in R}(1-y_{ir}^{ady}) \tag{21}$$

7. Cercanía:

$$y_{ir}^{crc}+x_{er}-1 \leq \sum_{s\in C_{cr}} x_{fs} \leq x_{er}-\epsilon+(1+\epsilon)y_{ir}^{crc},\ \forall r \in R \tag{22}$$

$$y_i^{ljn} = \sum_{r\in R}(1-y_{ir}^{ljn}) \tag{23}$$

8. Lejanía:

$$1-x_{e_1r}+\epsilon-(1+\epsilon)y_{ir}^{ljn} \leq \sum_{s\in C_{cr}} x_{e_2s} \leq 2-x_{e_1r}-y_{ir}^{ljn},\ \forall r \in R \tag{24}$$

$$y_i^{ljn} = \sum_{r\in R}(1-y_{ir}^{ljn}) \tag{25}$$

9. Capacidad:

$$\sum_{e\in E} S_e x_{er} + (S_r+\epsilon)(1-y_i^{cap}) \geq S_r+\epsilon \tag{26}$$

$$\sum_{e\in E} S_e x_{er} + \sum_{e\in E}(S_e-S_r)(1-y_i^{cap}) \leq \sum_{e\in E} S_e \tag{27}$$

4.4 Función Objetivo

La función objetivo tiene por finalidad minimizar el mal uso del espacio (desuso y sobre uso) y la cantidad de veces que se infringen las restricciones blandas. Se debe notar que el sobre uso del espacio físico se penaliza como el doble del espacio en desuso. Para simplificar visualmente la función objetivo [4], se considerará w^j como la penalización de cada restricción blanda j, con $j \in J$. Finalmente queda:

$$Min\ z = \sum_{r\in R} max(S_r - \sum_{e\in E} x_{er}S_e,\ 2\sum_{e\in E} x_{er}S_e - S_r) + \sum_{j\in J} w^j \sum_{i=1}^{|SC^j|} y_i^j \tag{28}$$

5 Conclusiones

A modo de cierre, se concluye que el OSAP es un problema que surge desde la necesidad humana de sacar el máximo provecho al espacio físico que habita, tomando en cuenta diferentes factores. En este sentido, la existencia del OSAP llevó a la creación y construcción de diferentes tipos de métodos para encontrar una solución a este problema, coincidiendo primordialmente en su objetivo de utilizar eficientemente los espacios (disminuyendo su mal uso), tomando en cuenta restricciones basales (duras y blandas) y utilizando los métodos de búsqueda como principal herramienta. A su vez, sus diferencias radican en la forma y el enfoque que utilizan para resolver el problema, por lo que, si bien resuelven diferentes aristas del mismo, aún no logran que sea a nivel único y global.

A la luz de lo anterior, se vuelve importante destacar tanto los métodos de Tabu Search como los de Meta heurística híbrida, ya que ambos se acercan a una adecuada solución dependiendo de si se trata de un bajo nivel o un alto nivel de iteraciones, respectivamente. Por lo que, gracias a sus resultados positivos, es esencial poder considerarlos como base para futuros procedimientos e investigaciones.

Por otro lado, es importante tener en cuenta que en la mayoría de los estudios revisados, se utilizaron computadores con 4 GB de RAM o con procesadores que no pertenecen a la última generación, lo que puede significar una clara limitante para cada método desarrollado, considerando cómo afecta la velocidad y eficiencia del hardware en los mismos.

Es innegable que el OSAP es un problema interesante de resolver por el gran impacto que su resolución tendría tanto a nivel mundial (industrias y negocios) como a nivel individual (en la vida de las personas). Y aquí es relevante considerar algoritmos no revisados, aplicaciones computacionales a la vanguardia, equipos físicos con mayor capacidad o una mezcla de metodologías diferentes, que abran puertas y posibilidades a obtener una respuesta definitiva para el OSAP o que sienten bases y orientaciones diferentes para futuros investigadores.

6 Bibliografía

References

[1] Mohammed A Awadallah, Ahamad Tajudin Khader, Mohammed Azmi Al-Betar, and Phuah Chea Woon. Office-space-allocation problem using harmony search algorithm. In *International Conference on Neural Information Processing*, pages 365–374. Springer, 2012.

[2] Christian Blum and Andrea Roli. Hybrid metaheuristics: an introduction. In *Hybrid metaheuristics*, pages 1–30. Springer, 2008.

[3] Asaju La'aro Bolaji, Ikechi Michael, and Peter Bamidele Shola. Adaptation of late acceptance hill climbing algorithm for optimizing the office-space allocation problem. In *International Workshop on Hybrid Metaheuristics*, pages 180–190. Springer, 2019.

[4] Elizabeth Montero Francisco Castillo, María-Cristina Riff. New bounds for office space allocation using tabu search. *GECCO '16: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 869–876, 2016.

[5] X. Z. Gao, V. Govindasamy, H. Xu, X. Wang, and K. Zenger. Harmony search method: Theory and applications. *Computational Intelligence and Neuroscience*, 2015:258491, Apr 2015.

[6] Rui Lopes and Daniela Girimonte. The office-space-allocation problem in strongly hierarchized organizations. In *European Conference on Evolutionary Computation in Combinatorial Optimization*, pages 143–153. Springer, 2010.

[7] Özgür Ülker and Dario Landa-Silva. A 0/1 integer programming model for the office space allocation problem. *Electronic Notes in Discrete Mathematics*, 36:575–582, 2010.