

Programación de operaciones de manufactura con restricciones de elegibilidad de trabajadores

Jaime Antero Arango Marin, PhD¹, Omar Danilo Castrillón, PhD¹, and Jaime Alberto Giraldo Garcia, PhD¹

¹ Universidad Nacional de Colombia, Facultad de Ingeniería y Arquitectura, Departamento de Ingeniería Industrial, Campus La Nubia Bloque Q piso 2, Manizales - Colombia. jaarangom@unal.edu.co, odcastrillong@unal.edu.co, jaiagiraldog@unal.edu.co

Resumen— Se considera el problema de programación de operaciones en una planta de producción manufacturera con elegibilidad de los trabajadores según su calificación para las diferentes tareas. Se formula un modelo matemático correspondiente a un problema de programación lineal entera con codificación binaria. Para resolverlo, se propone un algoritmo genético con población generada aleatoriamente, puntos de cruce, probabilidad de mutación y criterio de parada fijados por el usuario. El algoritmo se implementa en un aplicativo de software y se prueba con datos de una empresa fabricante de toneles de madera. Los resultados muestran atracción hacia el mejor valor de la función objetivo, y permiten detectar que los factores que más lo favorecen son el criterio de parada y la probabilidad de mutación.

Palabras clave—Programación de tareas, Heurísticas, Asignación.

I. INTRODUCCIÓN

Las empresas del sector manufacturero se enfrentan a situaciones donde es importante el mejor aprovechamiento de sus recursos humanos. En particular, es prioritario asignar las actividades buscando que sean realizadas por los operarios con las habilidades y el entrenamiento apropiado para cada tarea.

Este tipo de problemas se asimila a un conocido problema de programación de tareas conocido como Flow shop híbrido, que corresponde a una configuración común en la industria [1]. El problema ha sido estudiado considerando diferentes restricciones que permiten modelar situaciones de entornos reales de producción, como las mostradas en [2]. Una de esas es la restricción de elegibilidad o de conjuntos de procesamiento, y se refiere a la condición donde solo algunos de los recursos son aptos para procesar ciertos trabajos [3].

Se han propuesto algoritmos de solución para problemas de secuenciación con restricciones de elegibilidad usando diferentes alternativas. [4] mostraron los límites para un problema de elegibilidad por grado de servicio en un entorno de máquinas paralelas no relacionadas. [5] propuso un algoritmo basado en Programación Lineal para ese mismo tipo de elegibilidad.

Entre los trabajos más recientes, [6] proponen heurísticas para resolver el problema de programación lineal entera mixta sobre un entorno productivo de alta flexibilidad. [7] usan un algoritmo evolutivo mejorado para lograr un 8% de ahorro de energía un problema de secuenciación en células de manufactura con elegibilidad. [8] presenta un algoritmo

genético híbrido basado en embalaje de contenedores para programar grupos de máquinas paralelas no relacionadas. [9] analiza un problema de pintura de automotores, y propone soluciones que consideran restricciones de elegibilidad. [10] presenta algoritmos rápidos de relación constante para programación de lotes paralelos con capacidades no idénticas. [11] desarrollaron un algoritmo de dos aproximaciones para un problema con elegibilidad anidada.

En empresas manufactureras intensivas en mano de obra, la necesidad del trabajo en equipo para algunas operaciones es una cuestión para considerar en el proceso de programación. [12] analizaron los equipos humano-robot y proponen un algoritmo genético y un modelo de programación por restricciones para mejorar su eficiencia.

El presente trabajo presenta un modelo matemático y un algoritmo de solución para el problema de asignar operaciones de trabajo a un conjunto de trabajadores. Algunas tareas requieren ser realizadas por dos o más trabajadores en conjunto. Los trabajadores tienen diferentes grados de calificación, no todos son elegibles para todas las tareas.

La solución presentada se probó con un conjunto de datos basado en una empresa real de fabricación de toneles de madera, con uso intensivo de mano de obra.

A continuación, se hace la descripción completa del problema, luego se presenta el modelo matemático, el algoritmo de solución, los resultados y las conclusiones.

II. PLANTEAMIENTO DEL PROBLEMA

En una empresa manufacturera, el proceso de producción se desarrolla por una secuencia de operaciones que requieren diferentes competencias de parte de los operarios. Algunas operaciones implican el trabajo conjunto de varios operarios con diferentes habilidades, mientras otras pueden desarrollarse por un operario trabajando en forma individual.

Se requiere programar las diferentes actividades del proceso de producción, de tal manera que se logre cumplir con el conjunto de pedidos en el menor tiempo posible, sujeto a restricciones de disponibilidad de los perfiles de operarios.

Por lo tanto, es importante tanto la asignación de tareas a en forma individual o colectiva, como la programación de las tareas para cada trabajador y para cada trabajo.

III. MODELO MATEMÁTICO

Es un problema de optimización cuyo modelo matemático se puede plantear de la siguiente forma:

Las variables son las siguientes:

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).

DO NOT REMOVE

p_{ijk} = Tiempo que demora el trabajador/equipo de trabajadores j procesando el trabajo i en la operación k

x_{ijk} = Asignación en la operación k del trabajo i al trabajador/equipo de trabajadores j . 1 = Asignado, 0 = No asignado.

G_k = Tipo de trabajadores apto para la operación k .

i = Trabajo

j = Trabajador

k = Operación

m = Trabajadores elegibles.

n = Trabajos a programar

q = Total de operaciones del proceso.

La función objetivo totaliza los tiempos de procesamiento de las actividades a programar.

$$\text{Minimizar Tiempo máximo } z = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^q (p_{ijk}) x_{ijk} \quad (1)$$

Sujeto a las siguientes restricciones:

En cada operación cada trabajo solo se asigna a un trabajador/equipo de trabajadores:

$$\sum_{j=1}^m x_{ijk} \leq 1 \quad i = 1, 2, \dots, n \quad ; k = 1, 2, \dots, q \quad (2)$$

El trabajador debe ser elegible para la operación:

$$j \in G_k \quad j = 1, 2, \dots, m \quad (3)$$

El modelo es binario, la variable de decisión x_{ijk} vale 1 si el trabajo i se asigna al trabajador/equipo j y 0 en otro caso.

$$x_{ijk} = 0, 1 \quad i = 1, 2, \dots, n; j = 1, 2, \dots, m; k = 1, 2, \dots, q \quad (4)$$

Es un problema de programación entera con codificación binaria, cuya complejidad es NP-Hard. Por lo tanto, aunque podría solucionarse con una heurística constructiva como el *branch and bound*, es más eficiente utilizar una metaheurística que obtenga una solución de buena calidad en un tiempo razonable.

IV. MATERIAL Y MÉTODOS

Para la solución se adapta un algoritmo genético, con el siguiente diseño:

A. Diseño del cromosoma:

El cromosoma tendrá tantas posiciones como operaciones haya en el proceso productivo. Cada posición

representa la asignación de la operación a un trabajador/equipo de trabajadores.

B. Población inicial

Se genera en forma aleatoria un número de individuos según el tamaño de población fijado por el investigador.

C. Selección de padres

Aplicando una función fitness se sortean los padres de tal forma que la probabilidad de cada individuo de ser elegido sea proporcional a la calidad de la función objetivo evaluada en ese individuo.

D. Cruzamiento

En cada iteración se generan dos descendientes que alternan las características de los dos padres en cada punto de cruce. El número de puntos de cruce es un parámetro fijado por el investigador.

E. Mutación

Según la probabilidad de mutación parametrizada por el investigador, los descendientes pueden mutar cambiando al azar una de las asignaciones originales.

F. Actualización de la población

Se hace una comparación entre cada descendiente y el individuo con peor valor de la función objetivo. El mejor de ambos queda en la nueva población y el otro se desecha.

G. Criterio de parada

El proceso termina una vez se han completado un número de iteraciones prefijado por el investigador sin que se haya encontrado un mejor valor para la función objetivo.

En la figura 1 se puede apreciar el pseudocódigo del algoritmo utilizado.

```

Inicio
Leer parámetros: TamañoPoblación, CondiciónParada, ProbabilidadMutación
Para Individuos = 1 hasta TamañoPoblación
    Generar nuevo individuo
    Evaluar Función objetivo en nuevo individuo
FinPara
ContarIteraciones = 0
Mientras ContarIteraciones < CondiciónParada
    SeleccionarPadres
    CruzarPadres
    Evaluar Función objetivo en descendientes
    Si (Aleatorio < ProbabilidadMutación)
        Mutar (Intercambiar dos posiciones elegidas al azar en descendiente)
    FinSi
    Si (Función Objetivo(Descendiente) < Función Objetivo(Peor de la población))
        Descendiente reemplaza a peor de la población
    FinSi
    Si (Función Objetivo(Descendiente) < Función Objetivo(Mejor de la población))
        ContarIteraciones = 0
    Sino
        ContarIteraciones = ContarIteraciones + 1
    FinSi
FinMientras
Mostrar resultado
    
```

FIGURA 1. PSEUDOCÓDIGO DEL ALGORITMO GENÉTICO UTILIZADO PARA SOLUCIONAR EL PROBLEMA

Fuente: Elaboración propia

V. RESULTADOS

El algoritmo se probó con un conjunto de datos basado en una empresa real que manufactura toneles de madera para el añejamiento de bebidas alcohólicas. Los grupos de trabajadores que se tienen en la empresa se muestran en la tabla 1.

TABLA 1. TIPOS DE GRUPOS DE TRABAJADORES

| GRUPO | DESCRIPCIÓN |
|-------|--|
| TM | TRABAJADOR MÁS CALIFICADO |
| TO | TRABAJADOR CALIFICADO |
| NT | TRABAJADOR NO CALIFICADO |
| TN | EQUIPO DE UN TRABAJADOR CALIFICADO (TO) Y UNO NO CALIFICADO (NT) |
| TT | EQUIPO DE DOS TRABAJADORES CALIFICADOS (TO) |

Fuente: Elaboración propia

La lista de operaciones con los tipos de trabajador que se requieren para cada operación, y los minutos de trabajo que demanda, se muestra en la tabla 2

TABLA 2. SECUENCIA DE OPERACIONES

| SECUENCIA | GRUPO | TIEMPO |
|-----------|-------|--------|
| 1 | TM | 480 |
| 2 | TN | 60 |
| 3 | TN | 30 |
| 4 | TN | 20 |
| 5 | NT | 60 |
| 6 | TM | 20 |
| 7 | TN | 20 |
| 8 | TM | 20 |
| 9 | TN | 20 |
| 10 | TN | 10 |
| 11 | TO | 60 |
| 12 | NT | 240 |
| 13 | TO | 300 |
| 14 | NT | 50 |
| 15 | TO | 600 |
| 16 | TO | 120 |
| 17 | NT | 600 |
| 18 | TO | 600 |
| 19 | TO | 600 |
| 20 | TO | 120 |
| 21 | TO | 60 |
| 22 | TT | 1200 |
| 23 | TO | 1200 |

| SECUENCIA | GRUPO | TIEMPO |
|-----------|-------|--------|
| 24 | NT | 90 |
| 25 | TO | 300 |
| 26 | TO | 600 |
| 27 | NT | 900 |
| 28 | NT | 600 |

Fuente: Elaboración propia

El personal disponible en la empresa y su grado de calificación se muestra en la Tabla 3.

TABLA 3. PERSONAL DISPONIBLE

| CÓDIGO | TIPO |
|--------|------|
| TMEX1 | TM |
| TMEX2 | TM |
| TNLR1 | TO |
| TNLR2 | TO |
| TNLR3 | TO |
| TNLR4 | TO |
| TNLR5 | TO |
| TNLR6 | TO |
| NTNL1 | NT |
| NTNL2 | NT |
| NTNL3 | NT |
| NTNL4 | NT |
| NTNL5 | NT |
| NTNL6 | NT |
| NTNL7 | NT |
| NTNL8 | NT |

Fuente: Elaboración propia

Se preparó un caso de estudio con un grupo de trabajos a ser programados, como se muestra en la tabla 4.

TABLA 4. TRABAJOS A PROGRAMAR

| PEDIDO | CANTIDAD |
|--------|----------|
| 1 | 50 |
| 2 | 75 |
| 3 | 35 |
| 4 | 30 |
| 5 | 35 |
| 6 | 30 |
| 7 | 30 |
| 8 | 65 |
| 9 | 50 |
| 10 | 15 |

Fuente: Elaboración propia

Para probar el algoritmo se diseñó un experimento según un diseño factorial 2^4 , variando el tamaño de la población, la probabilidad de mutación, los puntos de cruce y el criterio de parada según se muestra en la tabla 5

TABLA 5. DISEÑO DE EXPERIMENTO

| COMBINACIÓN | POBLACIÓN | PUNTOS DE CRUCE | PROBABILIDAD MUTACIÓN | CRITERIO DE PARADA |
|-------------|-----------|-----------------|-----------------------|--------------------|
| (1) | 10 | 3 | 0,05 | 3000 |
| d | 10 | 3 | 0,05 | 5000 |
| c | 10 | 3 | 0,1 | 3000 |
| cd | 10 | 3 | 0,1 | 5000 |
| b | 10 | 5 | 0,05 | 3000 |
| bd | 10 | 5 | 0,05 | 5000 |
| bc | 10 | 5 | 0,1 | 3000 |
| bcd | 10 | 5 | 0,1 | 5000 |
| a | 15 | 3 | 0,05 | 3000 |
| ad | 15 | 3 | 0,05 | 5000 |
| ac | 15 | 3 | 0,1 | 3000 |
| acd | 15 | 3 | 0,1 | 5000 |
| ab | 15 | 5 | 0,05 | 3000 |
| abd | 15 | 5 | 0,05 | 5000 |
| abc | 15 | 5 | 0,1 | 3000 |
| abcd | 15 | 5 | 0,1 | 5000 |

Se corren 30 réplicas de cada combinación en un programa de computador desarrollado por los autores en el que los parámetros del algoritmo pueden ser seleccionados por el usuario.

La tabla 6 resume los resultados obtenidos en las 30 réplicas de cada una de las combinaciones del diseño experimental.

TABLA 6. RESUMEN DE RESULTADOS, VALOR DE FUNCIÓN OBJETIVO

| COMBINACIÓN | PROMEDIO | DESV ESTÁNDAR | MEJOR VALOR |
|-------------|-------------|---------------|-------------|
| (1) | 87,94266667 | 0,028031181 | 87,92 |
| a | 87,94333333 | 0,029400133 | 87,92 |
| ab | 87,94133333 | 0,021291626 | 87,92 |
| abc | 87,93233333 | 0,021444931 | 87,92 |
| abcd | 87,92666667 | 0,015387709 | 87,92 |
| abd | 87,93733333 | 0,022427405 | 87,92 |
| ac | 87,92633333 | 0,013514573 | 87,92 |

| COMBINACIÓN | PROMEDIO | DESV ESTÁNDAR | MEJOR VALOR |
|-------------------------|-----------------|--------------------|--------------|
| acd | 87,92366667 | 0,009278575 | 87,92 |
| ad | 87,93166667 | 0,016626388 | 87,92 |
| b | 87,95066667 | 0,041434063 | 87,92 |
| bc | 87,93333333 | 0,019710549 | 87,92 |
| bcd | 87,93 | 0,02421171 | 87,92 |
| bd | 87,93666667 | 0,035363466 | 87,92 |
| c | 87,93566667 | 0,027251489 | 87,92 |
| cd | 87,92666667 | 0,016470105 | 87,92 |
| d | 87,93766667 | 0,024869776 | 87,92 |
| Promedio general | 87,93475 | 0,024932899 | 87,92 |

Fuente: elaboración propia

La tabla 6 muestra que, en todas las combinaciones se llegó al menos en una iteración al mejor valor conocido de la función objetivo, y que aunque todas las combinaciones llegaron a valores promedio cercanos a ese valor, las combinaciones abcd, ac, acd y cd fueron las que más cerca estuvieron del mínimo. Igualmente son las combinaciones con mejor desviación estándar lo que indica la cercanía entre sí de los valores obtenidos. La tabla 7 muestra el análisis de varianza del experimento, con respecto al valor de la función objetivo.

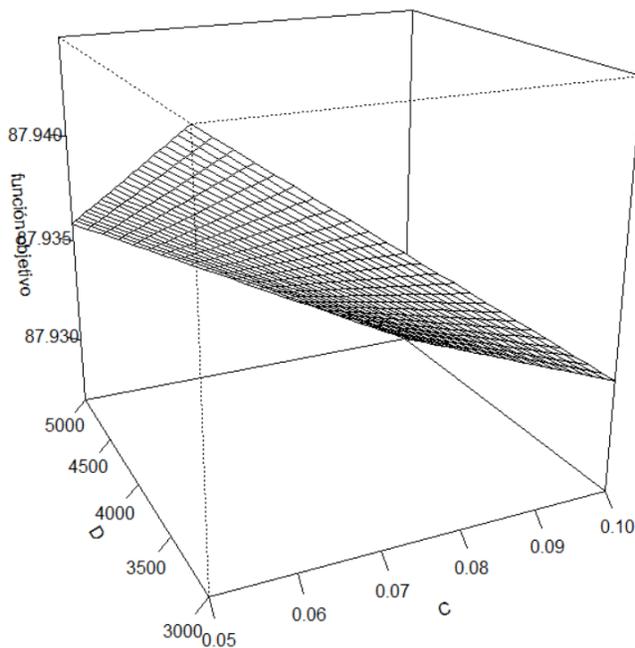
TABLA 7. ANÁLISIS DE VARIANZA. FUNCIÓN OBJETIVO

| Factor | Suma cuad | Gr Lib | Cuadrado medio | F | Prob (>F) | Sig n |
|--------|-----------|--------|----------------|--------|-----------|-------|
| A | 0,001763 | 1 | 0,0017633 | 2,994 | 0,084230 | . |
| B | 0,000801 | 1 | 0,0008008 | 1,360 | 0,244166 | |
| C | 0,014083 | 1 | 0,0140833 | 23,914 | 0,000001 | *** |
| D | 0,005741 | 1 | 0,0057408 | 9,748 | 0,001907 | ** |
| AB | 0,000041 | 1 | 0,0000408 | 0,069 | 0,792423 | |
| AC | 0,000013 | 1 | 0,0000133 | 0,023 | 0,880462 | |
| AD | 0,000101 | 1 | 0,0001008 | 0,171 | 0,679224 | |
| BC | 0,000001 | 1 | 0,0000008 | 0,001 | 0,970009 | |
| BD | 0,000003 | 1 | 0,0000033 | 0,006 | 0,940061 | |
| CD | 0,000367 | 1 | 0,0003675 | 0,624 | 0,429960 | |
| ABC | 0,000241 | 1 | 0,0002408 | 0,409 | 0,522824 | |
| ABD | 0,000120 | 1 | 0,0001200 | 0,204 | 0,651912 | |
| ACD | 0,000001 | 1 | 0,0000008 | 0,001 | 0,970009 | |
| BCD | 0,000030 | 1 | 0,0000300 | 0,051 | 0,821534 | |
| ABCD | 0,001203 | 1 | 0,0012033 | 2,043 | 0,153553 | |
| Error | 0,273260 | 464 | 0,0005889 | | | |
| Total | 0,297770 | 479 | | | | |

Significancia: 0 ‘***’ 0,001 ‘**’ 0,01 ‘*’ 0,05 ‘.’ 0,1 ‘ ’ 1
Fuente: Elaboración propia

Se encuentra que en el factor C se hallan variaciones incluso a un nivel de significancia de 0,001, en el factor D hay variación a un nivel de significancia de 0,01 y en el factor A a un nivel de significancia de 0.1. Esos factores están presentes en las combinaciones en las que se encontraron mejores promedios de la función objetivo.

Para examinar mejor el efecto de los dos factores en los que se encontró variación con mayor nivel de confianza se graficaron ambos factores frente al valor de la función objetivo, como se ve en la figura 2.



Slice at A = 12.5, B = 4

FIGURA 2. GRÁFICO EN 3D DE FACTORES C Y D VS FUNCIÓN OBJETIVO.

Fuente: Elaboración propia

En la figura 2 se aprecia que el mejor valor de la función objetivo se obtiene cuando tanto el factor C como el factor D están en su nivel más alto (0,10 de probabilidad de mutación y 5000 iteraciones de criterio de parada). En ese punto se llega un valor de la función objetivo cercano al mejor valor conocido.

VI. CONCLUSIONES

El problema de programar operaciones en una línea de manufactura intensiva en mano de obra con restricciones de elegibilidad de los recursos es un problema de programación lineal entera con codificación binaria, clasificado como NP-Hard.

El algoritmo genético es una opción eficiente para la solución del problema de programación de operaciones, dado que produce soluciones de buena calidad, en tiempos computacionales de menos de dos minutos en la mayoría de las configuraciones y de tres minutos y medio en promedio, lo que lo hace apropiado para su uso en la programación habitual de este tipo de proceso de manufactura.

Las mejores parametrizaciones del algoritmo genético son las que implican un valor alto de la probabilidad de mutación y del número de iteraciones sin mejora para la salida del algoritmo, ambos parámetros contribuyen a la diversidad de la población que permite evitar óptimos locales.

RECONOCIMIENTOS

Los autores desean expresar su gratitud a la Facultad de Ingeniería y Arquitectura de la Universidad Nacional de Colombia - Sede Manizales por su apoyo a esta investigación

REFERENCIAS

- [1] R. Ruiz and J. A. Vázquez-Rodríguez, "The hybrid flow shop scheduling problem," *Eur. J. Oper. Res.*, vol. 205, no. 1, pp. 1–18, Aug. 2010, doi: 10.1016/J.EJOR.2009.09.024.
- [2] A. Allahverdi, C. T. Ng, T. C. E. Cheng, and M. Y. Kovalyov, "A survey of scheduling problems with setup times or costs," *Eur. J. Oper. Res.*, vol. 187, no. 3, pp. 985–1032, Jun. 2008, doi: 10.1016/J.EJOR.2006.06.060.
- [3] J. Y. T. Leung and C. L. Li, "Scheduling with processing set restrictions: A survey," *Int. J. Prod. Econ.*, vol. 116, no. 2, pp. 251–262, Dec. 2008, doi: 10.1016/J.IJPE.2008.09.003.
- [4] K. Lim, K. Lee, and S. Y. Chang, "Improved bounds for online scheduling with eligibility constraints," *Theor. Comput. Sci.*, vol. 412, no. 39, pp. 5211–5224, Sep. 2011, doi: 10.1016/J.TCS.2011.05.029.
- [5] A. Zhang, "On the optimality of the LP-based algorithm for online scheduling with GoS eligibility constraints," *Oper. Res. Lett.*, vol. 43, no. 5, pp. 522–525, Sep. 2015, doi: 10.1016/J.ORL.2015.07.008.
- [6] H. Li, W. T. Huh, M. C. Sampaio, and N. Keng, "Planning Production and Equipment Qualification under High Process Flexibility," *Prod. Oper. Manag.*, vol. 30, no. 10, pp. 3369–3390, Oct. 2021, doi: 10.1111/POMS.13439.
- [7] Z. Hong, Z. Zeng, and L. Gao, "Energy-efficiency scheduling of multi-cell manufacturing system considering total handling distance and eligibility constraints," *Comput. Ind. Eng.*, vol. 151, p. 106998, Jan. 2021, doi: 10.1016/J.CIE.2020.106998.
- [8] B. Su, N. Xie, and Y. Yang, "Hybrid genetic algorithm based on bin packing strategy for the unrelated parallel workgroup scheduling problem," *J. Intell. Manuf.* 2020 324, vol. 32, no. 4, pp. 957–969, May 2020, doi: 10.1007/S10845-020-01597-8.
- [9] I. K. Singgih, O. Yu, B. I. Kim, J. Koo, and S. Lee, "Production scheduling problem in a factory of automobile component primer painting," *J. Intell. Manuf.* 2020 316, vol. 31, no. 6, pp. 1483–1496, Jan. 2020, doi: 10.1007/S10845-019-01524-6.
- [10] S. Li, "Parallel batch scheduling with inclusive processing set restrictions and non-identical capacities to minimize makespan," *Eur. J. Oper. Res.*, vol. 260, no. 1, pp. 12–20, Jul. 2017, doi: 10.1016/J.EJOR.2016.11.044.
- [11] Y. Z. Zhang and S. G. Li, "Scheduling Jobs with Release and Delivery Times Subject to Nested Eligibility Constraints," *J. Oper. Res. Soc. China* 2019 91, vol. 9, no. 1, pp. 63–77, Oct. 2019, doi: 10.1007/S40305-019-00268-7.
- [12] C. Ferreira, G. Figueira, and P. Amorim, "Scheduling Human-Robot Teams in collaborative working cells," *Int. J. Prod. Econ.*, vol. 235, p. 108094, May 2021, doi: 10.1016/J.IJPE.2021.108094.