

Tres estrategias para la definición del *Layout* de celdas de manufactura empleando algoritmos genéticos

Three strategies for the definition of manufacturing cells Layout using genetic algorithms

Trois stratégies pour la définition des dispositions de cellules de manufacture par l'emploi d'algorithmes génétiques

Três estratégias para a definição do Layout de células de manufatura empregando Algoritmos Genéticos

Camilo Mejía*
Rubén Dorado **
Luis Cobo ***

Fecha de recepción: 6 de octubre
Fecha de aprobación: 28 de octubre
Pp. 87-118

* Magíster en Ingeniería Mecánica, Universidad Nacional de Colombia. Pregrado en Ingeniería Mecánica, Universidad Nacional de Colombia.

** Magíster en Ciencias de la Computación, Universidad de Tokyo. Ingeniero de Sistemas, Universidad Nacional de Colombia.

*** Ph.D. en Ingeniería, Universidad de los Andes y en Génie Informatique, École Polytechnique de Montréal. Magister en Ingeniería de Sistemas y Computación, Universidad de los Andes

RESUMEN

Este artículo presenta tres estrategias para abordar la solución del problema de la definición de la distribución de planta o Layout de celdas de manufactura empleando algoritmos genéticos: la primera, corresponde a la solución independiente de la formación de las celdas y el Layout; la segunda, es la solución en su forma multi-objetivo empleando el algoritmo NSGA2; y finalmente, se presenta la solución simultánea en la cual se agrupa en una sola función objetivo la formación de las celdas y la definición del Layout que se soluciona con un algoritmo genético mono-objetivo, igual que se hizo en la primera estrategia. Los resultados demostraron que la segunda estrategia facilita el trabajo del diseñador porque da una visión amplia de las posibles soluciones, sin embargo, las otras dos estrategias presentan ventajas interesantes que pueden ser aplicadas en casos concretos que posean recursos limitados.

PALABRAS CLAVE

Celdas de manufactura, *Layout*, optimización, algoritmos genéticos, NSGA2.

ABSTRACT

This article presents three strategies to address the solution to the problem of definition of plant distribution or the layout of manufacturing cells using genetic algorithms: the first strategy is the independent solution to the formation of cells and layout; the second one is the solution in its multi-objective manner using the algorithm NSGA2; and finally the simultaneous solution in which the formation of cells and the definition of the layout that is solved with a single-objective genetic algorithm as was done in the first strategy are grouped into a single objective function. The results showed that the second strategy facilitates the designer's work because it gives a broad view of the possible solutions, however, the other two strategies have interesting advantages that can be applied in specific cases with limited resources.

KEYWORDS

Manufacturing cells, layout, optimization, genetic algorithms, NSGA2.

RÉSUMÉ

Cet article présente trois stratégies permettant d'aborder la résolution du problème pour la définition de la disposition des cellules de manufacture grâce à l'emploi d'algorithmes génétiques: la première solution rend compte de la formation des cellules et de leur disposition; la seconde correspond à la forme multi-objectif employant l'algorithme NSGA2; et la troisième stratégie présente une solution simultanée regroupant en une seule fonction la formation des cellules et la définition de la disposition qui se résout par un algorithme génétique mono-objectif identique à celui de la première stratégie. Les résultats ont montré que la seconde stratégie facilite la tâche du dessinateur car elle offre une vision élargie des solutions possibles. Les deux autres stratégies présentent certains avantages pouvant être appliqués dans des cas concrets de ressources limités.

MOTS CLEFS

Cellules de manufacture, disposition, optimisation, algorithmes génétiques, NSGA2.

RESUMO

Este artigo apresenta três estratégias para abordar a solução do problema da definição do layout de planta ou na implantação de células de manufatura utilizando algoritmos genéticos: a primeira corresponde a solução independente da formação de células e o layout; a segunda é a solução na sua forma multiobjetivo usando o algoritmo NSGA2; e, finalmente, a solução simultânea na qual se agrupa em uma única função objetivo a formação das células e definindo do layout que se soluciona com um algoritmo genético mono objetivo similar ao que se usou na primeira estratégia. Os resultados mostraram que a segunda estratégia facilita o trabalho do designer, porque dá uma visão ampla das soluções possíveis, no entanto, as outras duas estratégias têm vantagens interessantes que podem ser aplicadas em casos específicos dos que têm recursos limitados.

PALAVRAS-CHAVE

Células de manufatura, layout, otimização, algoritmos genéticos, NSGA2.

1. Introducción

Las celdas de manufactura es un concepto usado en la organización de plantas de producción modernas, cuyo principal objetivo son simplificar el gigantesco número de procesos interrelacionados de manera compleja que representa un sistema de producción, mediante la agrupación de procesos o máquinas a manera de mini fábricas, logrando un ordenamiento productivo que permite la fabricación de conjuntos de partes llamados <<familias de partes>>. Diseñar una planta de esta manera, usando el concepto de celdas, involucra agrupar los procesos que están más relacionados entre sí para minimizar los gastos que implica el transporte y manipulación de materiales y productos terminados. A este diseño se le conoce como el Layout de las celdas de manufactura y requiere de dos procesos: el primero, se trata del agrupamiento de las máquinas para formar la celdas y de las partes para formar las familias; y el segundo, es la definición del Layout que requiere establecer la posición física de las máquinas al interior de las celdas y la posición de las celdas entre sí.

La tarea de diseñar las celdas consiste en agrupar las partes en familias a partir de las similitudes que tienen con respecto a sus características de fabricación: materia prima, procesos de fabricación, tamaño, geometría, etc. En otras palabras, el conjunto de máquinas y procesos se agrupan en celdas siguiendo las similitudes en las partes que en ellas se fabrican.

El proceso general de formación de celdas consiste en generar una matriz con un bloque diagonal compuesto de dígitos binarios (unos y ceros). Cada bloque es una sub-matriz de la matriz de incidencia máquinas-partes (Tabla

1) formada por la intersección de filas, que representan una celda de manufactura, y columnas, que representan una familia de partes. En este caso, un espacio vacante es un elemento cero "0" que aparece en el bloque diagonal y un elemento excepcional es un elemento uno "1" que aparece fuera del bloque diagonal (Figura 1). Estos últimos elementos generan movimientos de partes de una celda a otra para ser procesadas.

Tabla 1. Matriz de incidencia de 5 máquinas y 7 partes.

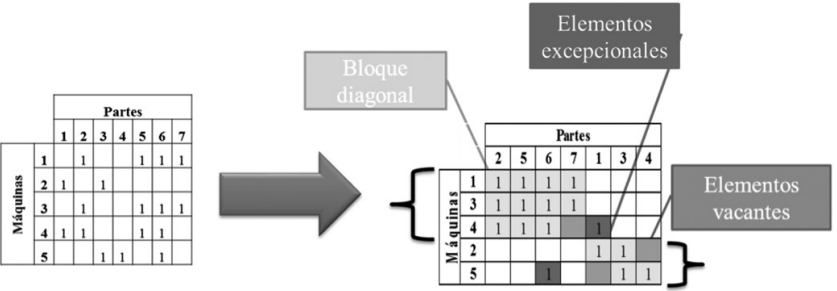
		Partes						
		1	2	3	4	5	6	7
Máquinas	1	0	1	0	0	1	1	1
	2	1	0	1	0	0	0	0
	3	0	1	0	0	1	1	1
	4	1	1	0	0	1	1	0
	5	0	0	1	1	0	1	0

Fuente. Vitanov, Tjahjono, & Marghalany, (2008).

Posteriormente a la formación de las celdas de manufactura es necesario resolver el problema del *Layout* intra e inter celdas de manufactura, que consiste en la asignación de las máquinas o procesos que conforman las celdas a una posición dentro de estas (*intra-layout*), y establecer la posición relativa de las celdas entre sí (*inter-layout*). Este problema del *Layout* se plantea de forma general minimizando el costo del transporte y manejo de materiales tal como lo describen Singh & Sharma (2006) y Drira, Pierreval, & Hajri-Gabouj, (2007), que está directamente relacionado con el sistema de manejo de materiales (Tavakkoli-Moghaddam, 2003), el cual es muy importante ya que de él depende la forma que puede tomar el *Layout* al interior de la celda. La posición de las máquinas dentro de la celda puede establecerse considerando un espacio continuo o discreto. En el primer

caso, el problema a solucionar es determinar las coordenadas de la posición de cada máquina, lo cual implica manejar variables continuas, mientras que al considerar el espacio discreto se debe asignar a cada máquina una posición de un conjunto de posibles posiciones previamente definidas.

Figura 1. Proceso de formación de celdas y familias de partes a partir de matrices de incidencia



Fuente.Elaboración propia de los autores.

El resto de este artículo se estructura de la siguiente manera. La sección 2, presenta los modelos matemáticos de cada una de las estrategias, así como los operadores y la codificación empleada en los diferentes algoritmos. La sección 3, muestra la experimentación, los resultados obtenidos; y la sección 4, finaliza con el análisis de los resultados y las conclusiones.

2. Solución propuesta con algoritmos genéticos

El presente trabajo compara tres estrategias para abordar la solución del problema del *Layout* de las celdas de manufactura con algoritmos genéticos mono-objetivo y multi-objetivo, empleando cómo funciones objetivo el costo de transporte y la Eficacia de Agrupamiento Γ . La primera estrategia plantea realizar de forma independiente el proceso de formación de las celdas de manufactura y la definición del *Layout* de estas, empleando algoritmos genéticos mono-objetivo; la segunda estrategia, resuelve los dos procesos en su forma multi-objetivo empleando el algoritmo NSGA2 (Deb, Pratab, Agarwal, & Meyarivan, 2002); y la tercera, es solucionar de forma simultánea los dos procesos integrándolos en una sola función objetivo. Para comparar las tres estrategias se emplearon tres problemas de prueba. A continuación se describen los algoritmos, sus operadores y la codificación empleada, y luego se presentan los modelos matemáticos de cada una de las estrategias.

2.1 Algoritmos Genéticos (- AG)

Los algoritmos genéticos, introducidos por (Holland, (1975), son algoritmos de búsqueda basados en la mecánica de la selección natural y la genética. Los AG se han aplicado en diversos campos como las matemáticas, ingeniería, biología y ciencias sociales (Goldberg, 1989). En los AG se combinan el concepto de supervivencia del más apto, con una estructura de intercambio de información al azar para formar un algoritmo de búsqueda. Se procede generando

una población inicial de posibles soluciones (individuos o cromosomas), posteriormente se evalúa la adaptación de cada uno de los individuos de la población y se seleccionan algunos de ellos para ser <<padres>> y reproducirse a través de un operador de cruce. Adicionalmente se emplea un operador de mutación que modifique al azar la composición genética de algunos individuos con el fin de mejorar la diversidad de la población. Los procesos de evaluación, selección, reproducción y mutación se repiten hasta que se alcanza el criterio de terminación.

2.2 Non-dominated sorting genetic algorithm

2 - NSGA2.

El algoritmo NSGA2 (Deb et al., 2002), es un algoritmo evolutivo multi-objetivo y elitista basado en dominancia de Pareto y *Crowding*, que emplea un enfoque de ordenamiento rápido de no dominancia para clasificar las soluciones en frentes de Pareto, y un operador de comparación de hacinamiento (*Crowded comparison*) para definir los individuos de la siguiente generación, considerando la clasificación en frentes de Pareto y la distancia *Crowding* de cada solución. Para una descripción más detallada de su estructura y operadores se recomienda Deb et al. (2002).

2.3 Codificación de las soluciones

La codificación que se emplea para el problema del *Layout* está basada en grupo u orden, como lo describen Gen, Lin, & Zhang (2009). Cada una de las soluciones se representa en un cromosoma, en el cual la primera parte corresponde a un vector que contiene la agrupación de las máquinas, a continuación la agrupación de las partes, luego la cantidad de celdas o familias de partes, y finaliza con los límites

en las posiciones de las permutaciones de las máquinas y partes para conformar las celdas y familias (Figura 2).

Figura 2. Ejemplo codificación basada en grupo para el problema del Layout

	Permutación Máquinas					Permutación Partes				Cantidad de celdas	Límites celdas			Límites familias					
Cromosoma	6	4	3	1	2	5	3	4	1	2	3	2	5	6	0	1	3	4	0

Posiciones	1	2	3	4	5	6	1	2	3	4		
Límites	2				5	6	1			3		4
	C1		C2		C3	F1		F2		F3		
	Celdas					Familias						

Nota: para el ejemplo de la figura 5-5 las celdas y familias de partes quedarían conformadas de la siguiente forma:

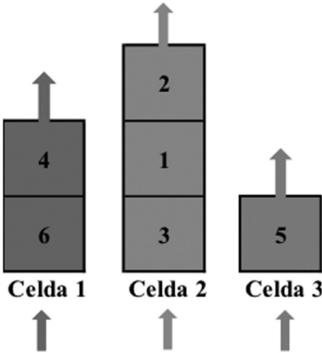
Celda 1= Máquinas 6 y 4 Familia 1= Parte 3

Celda 2= Máquinas 3, 1 y 2 Familia 2= Parte 4 y 1

Celda 3= Máquinas 5 Familia 3= Partes 2

Fuente.Elaboración propia de los autores.

Figura 3. Representación del Layout del ejemplo de la Figura 2



Fuente.Elaboración propia de los autores.

2.4 Generación de la población inicial

En este caso se generan las permutaciones para las máquinas y partes, ordenando de forma aleatoria un arreglo que va desde 1 hasta el número de máquinas o partes, la cantidad

de celdas se asigna con un numero aleatorio entre 1 y el máximo número de celdas, el cual define la cantidad de límites para formar las celdas y familias, asignando a estas las máquinas o partes que quedan en los intervalos que estos definen.

2.5 Selección

La selección de los padres para el cruzamiento se realiza por torneo binario (Coello, 2015). En este proceso se eligen de forma aleatoria dos individuos, los cuales compiten entre sí, aquel que tenga el mejor *fitness* es seleccionado como padre.

2.6 Cruzamiento

El cruzamiento se realiza separando el cromosoma en cada una de sus partes. A las dos primeras correspondientes a las permutaciones de las máquinas y las partes, se les generan de forma aleatoria dos puntos de cruce, el segmento que queda entre estos se intercambia entre un padre y el otro para generar los hijos, el resto de las permutaciones se completan empleando cruzamiento PMX (Partially Mapped Crossover) (Gen & Cheng, 2000); el resto del cromosoma se intercambia completamente para formar cada hijo (Figura 4).

Figura 4. Ejemplo de cruzamiento PMX para la codificación de las soluciones

	Permutación Máquinas	Permutación Partes	Cantidad de celdas	Límites celdas	Límites familias
Padre 1	6 4 3 1 2 5	4 1 3 2	3	2 5 6 0	1 3 4 0
Padre 2	3 5 1 6 4 2	2 3 1 4	2	3 6 0 0	2 4 0 0
PMX					
Hijo 1	6 5 1 4 2 3	3 1 1 4	2	3 6 0 0	2 3 0 0
Hijo 2	5 4 3 6 1 2	1 3 2	3	2 5 6 0	1 3 4 0

Fuente.Elaboración propia de los autores.

2.7 Mutación

En la mutación se tiene un tratamiento diferente para las permutaciones que para el resto del cromosoma; las primeras se mutan por inversión y el resto del cromosoma se muta generando un nuevo individuo (Figura 5). Para cada uno de los individuos se genera de forma aleatoria un número entre $[0,1]$, si el número es menor que la probabilidad de mutación se aplica el proceso de mutación, en caso contrario se continua con el siguiente individuo.

Figura 5. Ejemplo mutación individuo

Individuo seleccionado					Individuo mutado				
Permutación Máquinas	Permutación Partes	Cantidad de celdas	Límites celdas	Límites familias	Permutación Máquinas	Permutación Partes	Cantidad de celdas	Límites celdas	Límites familias
6 4 3 1 2 5	3 4 1 2	3	2 5 6 0	1 3 4 0	6 2 1 3 4 5	3 1 4 2	4	1 2 4 5	1 2 3 4

Fuente.Elaboración propia de los autores.

3. Estrategias implementadas

Dada la naturaleza multi-objetivo del problema del Layout de las celdas de manufactura, es posible abordar su solución de diversas formas. En este trabajo se presentan tres estrategias de solución para resolver el problema de la formación de las celdas de manufactura y la definición del Layout de estas. Para la formación de las celdas de manufactura se empleó la Eficacia de Agrupamiento Γ que fue propuesta por Suresh Kumar & Chandrasekharan (1990), la cual es una métrica ampliamente usada como función objetivo, que permite agrupar las máquinas en celdas y las partes en familias, mediante maximización del agrupamiento logrado por cada solución, los valores que puede tomar entre 0 y 1, siendo los mejores aquellos cercanos o iguales a uno. En su forma detallada se define de la siguiente forma:

$$\Gamma = \frac{\sum_{k=1}^{mc} \sum_{j=1}^n \sum_{i=1}^m Y_{jk} X_{ik} a_{ij}}{\sum_{j=1}^n \sum_{i=1}^m a_{ij} + \sum_{k=1}^{mc} [(\sum_{j=1}^n Y_{jk}) (\sum_{i=1}^m X_{ik})] - \sum_{k=1}^{mc} \sum_{j=1}^n \sum_{i=1}^m Y_{jk} X_{ik} a_{ij}} \quad (1)$$

Donde:

i =índice de las máquinas; $i=1, \dots, m$ número de máquinas

j =índice de las partes; $j=1, \dots, n$ número de partes

k =índice de las celdas; $k=1, \dots, mc$ máximo número de celdas

$X_{ik}=1$, si la máquina i pertenece a la celda k , en caso contrario

$X_{ik}=0$

$Y_{jk}=1$, si la parte j pertenece a la celda k , en caso contrario

$Y_{jk}=0$

$a_{ij}=1$, si la parte j es procesada por la máquina i , en otro caso

$a_{ij}=0$ (a_{ij} es la matriz de incidencia)

Dado que los algoritmos genéticos que se emplean se definieron para realizar minimización, y considerando

que Γ se maximiza y toma valores entre $[0,1]$, se genera una función objetivo con la forma $1.1 - \Gamma$, con el ánimo de lograr la maximización de Γ y minimizar la componente del costo de transporte en la solución simultánea, el valor de 1,1 garantiza que cuando $\Gamma = 1$ no se anule la solución simultánea:

$$\text{fitness}_{\text{agrupamiento}} = 1.1 - \Gamma \quad (2)$$

La definición del *Layout* de las celdas de manufactura se realiza minimizando el costo de transporte, la cual es una forma generalizada de evaluación del desempeño del layout (Drira et al., 2007; Singh & Sharma, 2006). Para definir el costo de transporte como función objetivo se tuvieron en consideración las siguientes suposiciones:

El espacio se considera discreto y en este se ordenan las celdas como líneas rectas paralelas (Figura 3).

Todos los movimientos que realizan las partes intra e inter celdas siguen la secuencia de operaciones definida en la ruta de proceso de cada una.

Se establece que el flujo de las partes al interior de las celdas y entre estas, debe ejecutarse sin realizar retrocesos o movimientos en contraflujo. En el caso de ser necesario que una parte deba pasar por una máquina que está en una posición anterior a la máquina que la precede al interior de la celda, esta debe salir de la celda y volver a ingresar, por lo tanto la distancia que se calcula para cada movimiento contempla esta regla.

Se establecen valores de costo diferentes para cada uno de los posibles movimientos que puede realizar una parte al interior de las celdas, entre estas y al ingresar o salir de ellas.

Para cada uno de los movimientos se consideran los volúmenes de producción y el tamaño del lote de cada parte que se puede transportar en cada uno de los movimientos que se realiza.

A continuación se define el costo de transporte empleado en este trabajo:

$$TC = \sum_{j=1}^n \sum_{l=1}^{NO_j} C_{j,l} d_{j,l} \frac{VP_j}{B_j} \quad (3)$$

Donde:

i =índice de las máquinas; $i=1, \dots, m$ número de máquinas

j =índice de las partes; $j=1, \dots, n$ número de partes

k =índice de las celdas; $k=1, \dots, mc$ máximo número de celdas

l =índice de operaciones; $l=1, \dots, NO_j$ número de operaciones de la parte j

NO_j =máximo número de operaciones de la parte j

$X_{ik}=1$, si la máquina i pertenece a la celda k ,
en caso contrario $X_{ik}=0$

$Y_{jk}=1$, si la parte j pertenece a la celda k , en caso contrario
 $Y_{jk}=0$

$Z_{ipk}=1$, si la máquina i ocupa la posición p de la celda k , en
caso contrario $Z_{ipk}=0$

$d_{j,l}$ =distancia que recorre la parte j entre la operación
 $l-1$ y l

$C_{j,l}$ =Costo del movimiento de la parte j entre la operación
 $l-1$ y l

$C_{j,l\text{externo}}$ = Costo de un movimiento externo, si $l=1$ o $l=NO_j$
(antes de la primera operación y después de la última
operación de la parte)

$C_{j,l\text{normal}}$ = Costo de un movimiento dentro de la celda
siguiendo el flujo normal de esta, si $2 \leq l \leq NO_j$ y no es
necesario un retroceso

$C_{j,l}^{\text{retroceso}}$ = Costo de un movimiento que requiere realizar un retroceso, si $2 \leq l \leq NO_j$

$C_{j,l}^{\text{celdas}}$ = Costo de un movimiento entre celdas, si la operación l se debe realizar en una celda diferente a la de la operación $l-1$.

VP_j = volumen de producción de la parte j

B_j = tamaño del lote para cada movimiento de la parte j

Cada una de las tres estrategias está definida por un modelo matemático que se presenta a continuación.

♦ Primera estrategia – solución individual

La primera estrategia busca resolver, de forma independiente, los problemas de formación de las celdas de manufactura y la definición del *Layout* de las celdas de manufactura, para obtener las mejores soluciones de cada uno de forma independiente, empleando algoritmos genéticos mono-objetivo.

Minimizar: $f_{\text{fitness-agrupamiento}} = 1.1 - \Gamma$ (4)

Sujeto a:

$$\sum_{k=1}^{mc} X_{ik} = 1, \quad i = 1, \dots, m \text{ número de máquinas} \quad (5)$$

$$\sum_{k=1}^{mc} Y_{jk} = 1, \quad j = 1, \dots, n \text{ número de partes} \quad (6)$$

$$\sum_{k=1}^{mc} \sum_{p=1}^{mp_k} Z_{ipk} = 1, \quad p = 1, \dots, mp_k \text{ máximo número de posiciones de la celda } k, \\ k = 1, \dots, mc \text{ máximo número de celdas} \quad (7)$$

Minimizar: $f_{\text{fitness-layout}} = TC$ (8)

Sujeto a:

$$\sum_{k=1}^{mc} X_{ik} = 1, \quad i = 1, \dots, m \text{ número de máquinas} \quad (9)$$

$$\sum_{k=1}^{mc} Y_{jk} = 1, \quad j = 1, \dots, n \text{ número de partes} \quad (10)$$

$$\sum_{k=1}^{mc} \sum_{p=1}^{mp_k} Z_{ipk} = 1, \quad p = 1, \dots, mp_k \text{ máximo número de posiciones de la celda } k, \\ k = 1, \dots, mc \text{ máximo número de celdas} \quad (11)$$

♦ **Segunda estrategia – solución multi-objetivo**

La segunda estrategia resuelve los dos procesos en su forma multi-objetivo empleando el algoritmo NSGA2.

$$\text{Minimizar: } f_{\text{fitness-agrupamiento}} = 1.1 - \Gamma \quad (13)$$

$$f_{\text{fitness-layout}} = TC \quad (14)$$

Sujeto a:

$$\sum_{k=1}^{mc} X_{ik} = 1, \quad i = 1, \dots, m \text{ número de máquinas} \quad (15)$$

$$\sum_{k=1}^{mc} Y_{jk} = 1, \quad j = 1, \dots, n \text{ número de partes} \quad (16)$$

$$\sum_{k=1}^{mc} \sum_{p=1}^{mp_k} Z_{ipk} = 1, \quad p = 1, \dots, mp_k \text{ máximo número de posiciones de la celda } k, \\ k = 1, \dots, mc \text{ máximo número de celdas} \quad (17)$$

♦ **Tercer estrategia – solución simultánea**

La tercera estrategia resuelve los dos procesos de forma simultánea, integrando en una sola función objetivo la Eficacia de Agrupamiento Γ y el costo de transporte TC . Empleando un algoritmo genético mono-objetivo.

$$\text{Minimizar: } f_{\text{fitness-simultánea}} = TC(1.1 - \Gamma) \quad (18)$$

Sujeto a:

$$\sum_{k=1}^{mc} X_{ik} = 1, \quad i = 1, \dots, m \text{ número de máquinas} \quad (19)$$

$$\sum_{k=1}^{mc} Y_{jk} = 1, \quad j = 1, \dots, n \text{ número de partes} \quad (20)$$

$$\sum_{k=1}^{mc} \sum_{p=1}^{mp_k} Z_{ipk} = 1, \quad p = 1, \dots, mp_k \text{ máximo número de posiciones de la celda } k, \\ k = 1, \dots, mc \text{ máximo número de celdas} \quad (21)$$

En los tres casos se manejan restricciones de unicidad que garantizan que las máquinas y partes se asignen únicamente a una celda y que las máquinas se asignen a una sola posición dentro de la celda a la que pertenece, garantizando que dos máquinas no ocupen la misma posición.

4. Metodología

El proceso realizado en este trabajo consistió en la evaluación de las tres estrategias antes descritas mediante su implementación en lenguaje Fortran, lo cual permitió obtener los resultados que se presentan más adelante. En primer lugar, se realizó un proceso de ajuste de parámetros de los algoritmos genéticos implementados, para luego proceder a obtener los resultados en la mejor configuración de estos. El criterio de terminación de los algoritmos genéticos fue el número de llamados a la función objetivo, el cual tuvo un valor de 10.000.000 para el caso mono-objetivo, y 50.000.000 para el caso multi-objetivo. En ambos casos se realizaron múltiples pruebas: 100 para cada solución mono-objetivo y 50 para las multi-objetivo, de las cuales se presentan únicamente los mejores resultados. Los demás parámetros se describen a continuación (Tabla 2).

Tabla 2. Parámetros Algoritmos Genéticos

	Algoritmo Genético Mono-objetivo	NSGA2
Tamaño de la población	100	250
Probabilidad de cruce	95%	90%
Probabilidad de mutación	30%	35%

Fuente. Elaboración propia de los autores.

La evaluación de desempeño se realizó con tres problemas de prueba: el primero poseía 5 máquinas y 7 partes, tal como fue presentado por Vitanov et al. (2008) (Tabla 1); el segundo poseía 7 máquinas y 11 partes, tal como fue presentado por Seifoddini & Djassemi (1996) (Tabla 3); y el último, poseía 14 máquinas y 24 partes, tal como fue presentado por King (1980) (Tabla 4).

Tabla 3. Matriz de incidencia de 7 máquinas y 11 partes

	Partes											
		1	2	3	4	5	6	7	8	9	10	11
Máquinas	1	0	0	1	0	0	0	1	0	0	0	1
	2	1	1	0	0	0	1	0	0	1	0	0
	3	1	1	0	0	0	1	0	0	1	0	0
	4	0	0	0	1	1	0	0	1	0	1	0
	5	0	0	1	0	0	0	1	0	0	0	1
	6	0	0	1	0	0	0	1	0	0	0	1
	7	0	0	0	1	1	0	0	1	0	1	0

Fuente. Seifoddini & Djassemi, (1996)

Tabla 4. Matriz de incidencia de 14 máquinas y 24 partes

	Partes																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
4	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	1	0
5	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0
6	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	1	0	0
7	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	1	0	0	0	1	1	0	1	0	1	1	1	0	0	0	0	0	1	0	0
9	0	0	0	0	1	0	0	0	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
11	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
12	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
14	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0

Fuente. King, (1980)

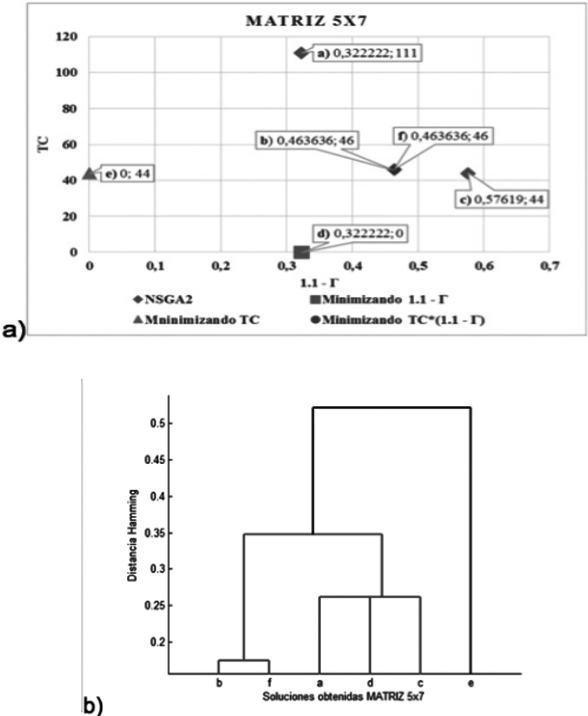
5. Resultados y discusión

Como se describió anteriormente, luego de la implementación de los algoritmos se procedió a obtener los mejores resultados de las diversas ejecuciones de cada estrategia. Este proceso se realizó de igual forma para cada uno de los tres problemas de prueba. A continuación se presentan estos resultados a través de una gráfica de las soluciones obtenidas para cada problema de prueba al ejecutar la primera estrategia, que corresponden a la minimización de TC y minimización de $1.1-\Gamma$ (Figura 6); la segunda estrategia, que corresponde a los resultados obtenidos con el algoritmo NSGA2 (Figura 8), y la tercera estrategia, que corresponde a la solución simultánea que minimiza $TC^*(1.1-\Gamma)$ (Figura 10).

A continuación se presentan los resultados obtenidos y los esquemas de los Layouts de cada una de las soluciones obtenidas (Figuras 7, 9 y 11), en las cuales se puede apreciar un comportamiento similar entre los resultados del problema con 5 máquinas y 7 partes, con el problema de 14 máquinas y 24 partes (Figuras 6 y 10). En los dos se presenta una distribución de las soluciones en forma de abanico que inicia en el extremo superior izquierdo con la solución que obtiene el menor valor de $1.1-\Gamma$, pero el mayor de TC, y se extiende hasta el extremo inferior derecho con un resultado inverso, el menor valor de TC con el mayor valor de $1.1-\Gamma$ (Figuras 6 y 10). Estos dos resultados definen los extremos del frente de Pareto presentado y coinciden con las soluciones obtenidas al implementar la primera estrategia. En la zona central se puede encontrar la solución simultánea que minimiza el producto $TC^*(1.1-\Gamma)$, la cual, aunque está en una zona intermedia, presenta un sesgo hacia la solución que minimiza TC debido a que este factor pesa más en la función objetivo que $1.1-\Gamma$.

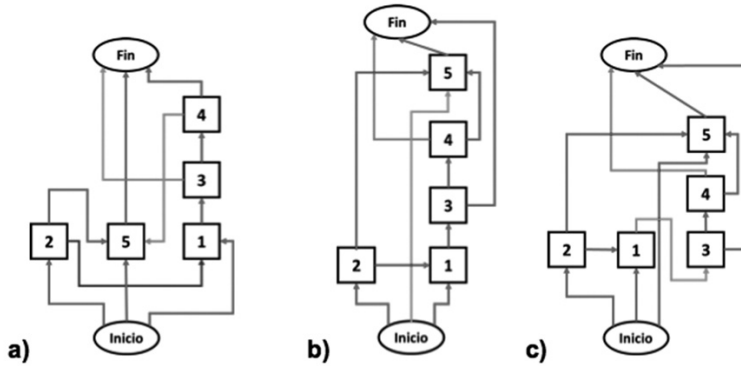
En términos generales se observa un frente de Pareto que demuestra que la formación de las celdas y la definición del Layout, son objetivos que pueden estar en conflicto en ciertos casos, ya que al observar lo correspondiente a la solución del problema de 7 máquinas y 11 partes (Figura 8) todas las soluciones coinciden a una sola; en este caso las tres estrategias convergen a la misma solución, en la cual se forman celdas con familias de partes que no requieren movimientos entre estas (Tabla 5), donde se puede apreciar un bloque diagonal compacto. Este resultado permite afirmar que al eliminar la necesidad de movimientos entre celdas por medio de duplicación de máquinas, el problema podría llegar a tener una única solución óptima.

Figura 6. Resultados problema de prueba de 5 máquinas y 7 partes, a) soluciones en el espacio de funciones, b) dendrograma de las soluciones



Fuente.Elaboración propia de los autores.

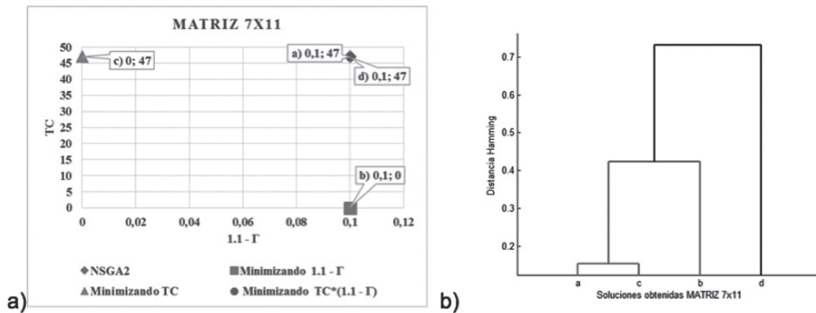
Figura 7. Esquemas de los layouts correspondientes a los resultados del problema de prueba de 5 máquinas y 7 partes, a) Solución con el menor valor de $1.1-\Gamma$, b) Solución simultánea, c) Solución con el menor valor de TC



Fuente.Elaboración propia de los autores.

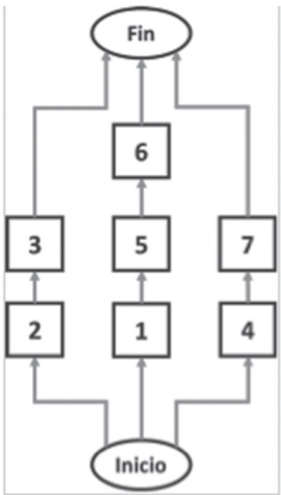
Se pueden ver las similitudes entre las respuestas, resaltando que en la medida que se mejora el desempeño de TC (Figura 7, c) se reducen o desaparecen los retrocesos, mientras que al mejorar el agrupamiento, estos se incrementan (Figura 7, a); además, claramente se reconocen patrones de secuencias que se repiten y se demuestra la importancia de las reglas de flujo establecidas al definir el costo de transporte, así también las similitudes entre las soluciones se pueden apreciar en el dendrograma (Figura 6, b), el cual permite apreciar que las soluciones comparten fragmentos de los cromosomas que las conforman, y al mismo tiempo demuestra que soluciones que tienen valores iguales de función objetivo, no necesariamente son iguales, presentándose isómeros. Este fenómeno se da de igual forma en los otros dos problemas (Figuras 8, b y 10, b). Por último, la solución simultánea (Figura 7, b) logra un buen agrupamiento y un flujo lineal de las partes al ser procesadas.

Figura 8. Resultados problema de prueba de 7 máquinas y 11 partes, a) soluciones en el espacio de funciones, b) dendrograma de las soluciones



Fuente.Elaboración propia de los autores.

Figura 9. Esquema del *Layout* correspondiente a los resultados del problema de prueba de 7 máquinas y 11 partes



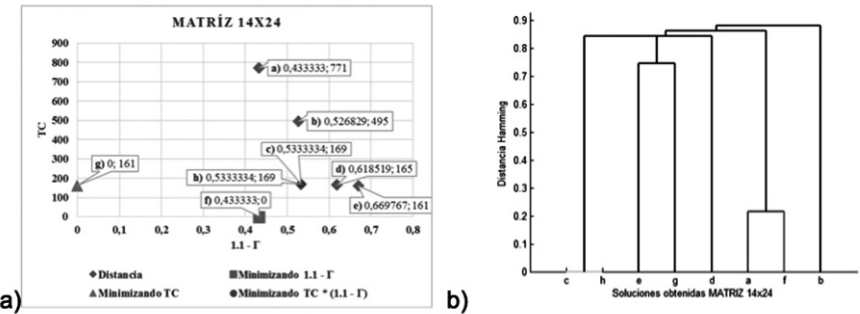
Fuente.Elaboración propia de los autores.

Tabla 5. Matriz de incidencia ordenada del problema de prueba de 7 máquinas y 11 partes

		Partes										
		1	2	6	9	3	7	11	4	5	8	10
Máquinas	2	1	1	1	1	0	0	0	0	0	0	0
	3	1	1	1	1	0	0	0	0	0	0	0
	1	0	0	0	0	1	1	1	0	0	0	0
	5	0	0	0	0	1	1	1	0	0	0	0
	6	0	0	0	0	1	1	1	0	0	0	0
	4	0	0	0	0	0	0	0	1	1	1	1
	7	0	0	0	0	0	0	0	1	1	1	1

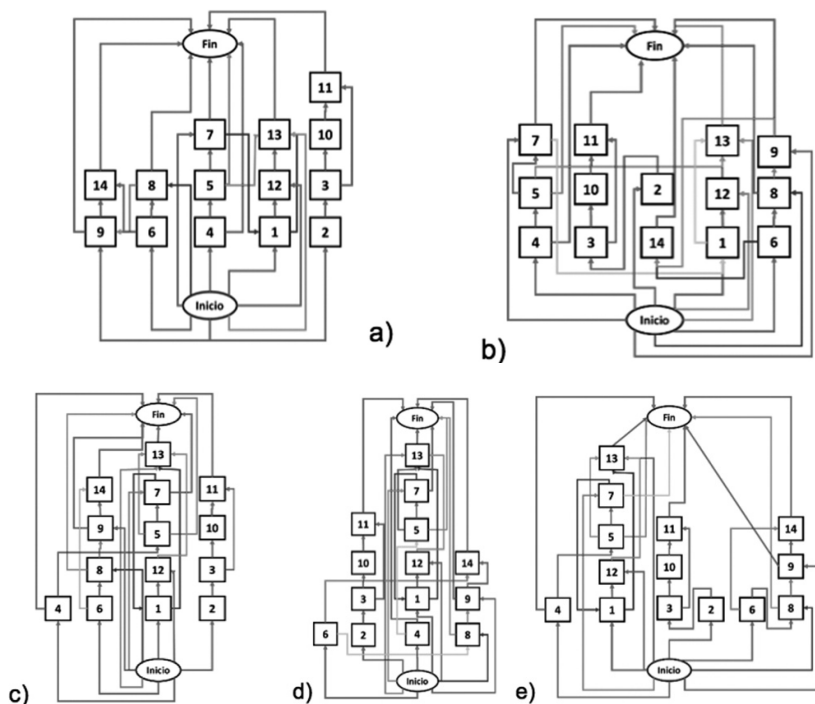
El resultado obtenido al solucionar el problema de 7 máquinas y 11 partes, representa el caso ideal, donde se obtiene una solución que satisface todas las expectativas con flujos lineales, sin retrocesos, ni movimientos entre celdas; sin embargo, es importante resaltar que los casos reales raramente presentan este comportamiento.

Figura 10. Resultados problema de prueba de 14 máquinas y 24 partes, a) soluciones en el espacio de funciones, b) dendrograma de las soluciones



Fuente.Elaboración propia de los autores.

Figura 11. Esquemas de los *Layouts* correspondientes a los resultados del problema de prueba de 14 máquinas y 24 partes, a) Solución con el menor valor de $1.1-\Gamma$, b) Solución intermedia, c) Solución simultánea, d) Solución intermedia, e) Solución con el menor valor de TC



Fuente. Elaboración propia de los autores.

El último problema de prueba con 14 máquinas y 24 partes permite tener una mayor variedad de soluciones con un comportamiento análogo al de 5 máquinas y 7 partes, pero en este caso es más fácil observar soluciones más compactas asociadas a menores valores de $1.1-\Gamma$ (Figura 11, a), que reducirían el espacio requerido pero incrementarían los movimientos entre celdas, en el sentido opuesto en la medida que mejora el desempeño de TC (Figura 11, e), las soluciones obtenidas presentan menor número de movimientos entre celdas, pero las máquinas están más distribuidas lo cual puede representar un inconveniente

por el incremento del espacio necesario. Por último, la solución simultánea (Figura 11, c) ofrece una alternativa atractiva para un diseñador que desee una planta más compacta con flujos lineales con un número reducido de movimientos entre celdas.

6. Conclusiones

A partir de los resultados presentados, es claro que la estrategia más apropiada para solucionar el problema del *Layout* de las celdas de manufactura es la segunda, ya que emplear algoritmos genéticos multi-objetivo basados en dominancia de Pareto le brinda el horizonte de soluciones que viene con el frente de Pareto, donde se incluyen las otras dos estrategias. Sin embargo, si los recursos no lo permiten, una alternativa muy buena puede ser implementar la solución independiente de la formación de las celdas y el *Layout* -primera estrategia-, y la solución simultánea -tercera estrategia- lo cual le permitiría tener las soluciones extremas y una central, con las cuales puede diseñar de una forma más adecuada su sistema de manufactura celular.

Los resultados evidenciaron que lograr un buen agrupamiento no garantiza un buen flujo de las partes entre y al interior de las celdas, además un bajo costo de transporte no garantiza un adecuado agrupamiento de las partes; sin embargo, la solución simultánea permite lograr una configuración que tiene los dos atributos y es importante considerarla como base para el diseño del sistema de manufactura celular.

Por último es importante resaltar que al eliminar los movimientos entre celdas y los contraflujos o retrocesos mediante múltiples máquinas -en la medida que los recursos lo permitan-, el desempeño del agrupamiento y el costo de transporte mejorarían y podrían converger a una solución única que correspondería a la ideal, la cual se asemejaría a la obtenida al solucionar el problema de 7 máquinas y 11 partes.

7. Referencias bibliográficas

- Coello, C. A. C. (2015). *Introducción a la Computación Evolutiva* (Notas de Curso).
- Deb, K., Pratab, S., Agarwal, S., & Meyarivan, T. (2002). A Fast and Elitist Multiobjective Genetic Algorithm: NGSA-II. *IEEE Transactions on Evolutionary Computing*, 6(2), 182–197.
- Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2007). Facility layout problems: A survey. *Annual Reviews in Control*, 31(2), 255–267. Recuperado de: <http://doi.org/10.1016/j.arcontrol.2007.04.001>
- Gen, M., & Cheng, R. (2000). *Genetic Algorithms and Engineering Optimization*. John Wiley & Sons. Recuperado de: <https://books.google.com/books?id=U7MuV1q6P1oC&pgis=1>
- Gen, M., Lin, L., & Zhang, H. (2009). Evolutionary techniques for optimization problems in integrated manufacturing system: State-of-the-art-survey. *Computers and Industrial Engineering*, 56(3), 779–808. Recuperado de: <http://doi.org/10.1016/j.cie.2008.09.034>
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Recuperado de: <http://dl.acm.org/citation.cfm?id=534133>
- Holland, J. (1975). *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor: University of Michigan Press.

- King, R. (1980). *Machine-Component Group Formation in Group Technology*, 8(2), 2–8.
- Seifoddini, H., & Djassemi, M. (1996). A new grouping measure for evaluation of machine-component matrices. *International Journal of Production Research*, 34(5), 1179–1193. Recuperado de: <http://doi.org/10.1080/00207549608904960>
- Singh, S. P., & Sharma, R. R. K. (2006). A review of different approaches to the facility layout problems. *International Journal of Advanced Manufacturing Technology*, 30(5-6), 425–433. Recuperado de: <http://doi.org/10.1007/s00170-005-0087-9>
- Suresh Kumar, C., & Chandrasekharan, M. P. (1990). Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *International Journal of Production Research*, 28(2), 233–243. Recuperado de: <http://doi.org/10.1080/00207549008942706>
- Tavakkoli-Moghaddam, R. (2003). A continuous plane model to machine layout problems considering pick-up and drop-off points: an evolutionary algorithm. *International Journal of Engineering - Transactions B: Applications*, 16(1), 59. Recuperado de: <http://www.ije.ir/abstract/%7BVolume:16-Transactions:B-Number:1%7D=42>
- Vitanov, V., Tjahjono, B., & Marghalany, I. (2008). Heuristic rules-based logic cell formation algorithm. *International Journal of Production Research*, 46(2), 321–344. Recuperado de: <http://doi.org/10.1080/00207540601138494>