[1]ALGUNOS ALGORITMO DE TECNOLOGÍA DE GRUPOS

Por: BONILLA Londoño Héctor Fabio.

1.0 Introducción

En la actualidad se han desarrollado algoritmos para la formación de grupos de máquinas y para la formación y asignación de familias de partes a grupos de máquinas. Es importante mencionar que los algoritmos aquí presentados son heurísticos y no garantizan soluciones óptimas para la formación de familias de partes y de máquinas.

El paso inicial para resolver cualquier problema de tecnología de grupos es crear la matriz de incidencia o matriz de componente – máquina. Dicha matriz, es aquella que relaciona las máquinas con los componentes a producir. Entonces la matriz de incidencia:

M, representa en su posición M_{ij} el uso o no de una máquina en la producción de un componente. Esto es, $M_{ij} = 1$ significa que la parte j es procesada por la máquina i, y $M_{ij} = 0$, lo contrario.

El proceso que siguen los algoritmos de agrupación consiste en manipular la matriz de incidencia de tal forma que se formen "clústeres" o zonas donde todas las posiciones sean 1s (unos). Las posiciones $M_{ij} = 1$ de un clúster representan familias de partes a ser Procesadas por una celda de manufactura (grupo de máquinas).Ver Figura No 1. Las partes están numeradas de P1 a P3 y las máquinas de M1 a M3. Después de una serie de manipulaciones en la matriz se logra la estructura de la figura 1b. Estas manipulaciones son generalmente intercambios de filas y de columnas. Las celdas resultantes son {M1,M3} y {M2} y las familias son {P1, P3} y {P2}. Claramente en casos reales no será posible formar familias de partes que sean asignadas a una sola celda.

	P1	P2	P3
M1	1	0	1
M2	0	1	0
M3	1	0	1

	P1	P2	P3
M1	1	1	1
M2	1	1	0
M3	1	0	1

Figura 1. Matriz de Incidencia y formación de clústeres

[1]*Información Extraída Documento Tecnología de Grupos* ***NT-2202-000-VP.*** *Departamento de Ingeniería Industrial. Área Fundamentos de Producción Universidad de Andes*

Ejemplo1: En la tabla 1 se describe las componentes con las respectivas máquinas que se utilizan para su elaboración

Componente con su secuencia de operaciones

Componente	Secuencia de Operaciones
1	A,C,G
2	D,E
3	A,C
4	A,B
5	A,C
6	F
7	F
8	A,C
9	A,C
10	D,E,F,H

Solución: A partir de la **tabla1** se construye la matriz de incidencia ***M,*** que se muestra en la figura No 2.

Máquina	**Componentes**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
A	1	0	1	1	0	0	0	1	1	0
B	0	0	0	1	0	0	0	0	0	0
C	1	0	1	0	1	0	0	1	1	0
D	0	1	0	0	0	0	0	0	0	0
E	0	1	0	0	0	0	0	0	0	1
F	0	0	0	0	0	1	1	0	0	1
G	1	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	0	0	1

Figura 2. Matriz de Incidencia para el ejemplo 1

1.1 Rank-Order Clustering Algorithm

También llamado algoritmo de King, en honor a su autor quien lo presentó en 1979[2]. El presente algoritmo es una manera sencilla de formar grupos y asignar familias de partes a esos grupos. Se tiene un problema con n partes y m máquinas. El algoritmo consiste en calcular valores para ponderar cada fila, que se llamarán *pesos,* ordenar filas, calcular pesos para cada columna y ordenar columnas hasta que no se pueda ordenar más. Al final se obtiene la matriz de incidencia ordenada, con patrones casi definidos para determinar la creación de los grupos. Algunas veces es fácil identificar los grupos con sólo observar la matriz, pero cuando no existe tal patrón, se debe acudir a alguna técnica que ayude a identificar los grupos. El algoritmo Rank Order se explica detalladamente a continuación[3]:

Inicio

Se llamará k al número de iteraciones. Al final k será el número total de iteraciones

Necesarias para definir las familias de partes.

Paso 1: Calcular los pesos w_i de las n filas y ordenarlas de forma ascendente con base en los pesos. Continuar con el paso 2. Si las n filas ya están ordenadas, seguir al paso 2. El peso se calcula de la siguiente manera:

$$w_i = \sum_{j}^{m} 2^j M_{ij} \qquad (1)$$

Paso 2: Calcular los pesos w_j de las m columnas y ordenarlas de forma ascendente. Continuar con el paso 3. Si las m columnas, ya están ordenadas, seguir al paso 3. El peso se calcula de la siguiente manera:

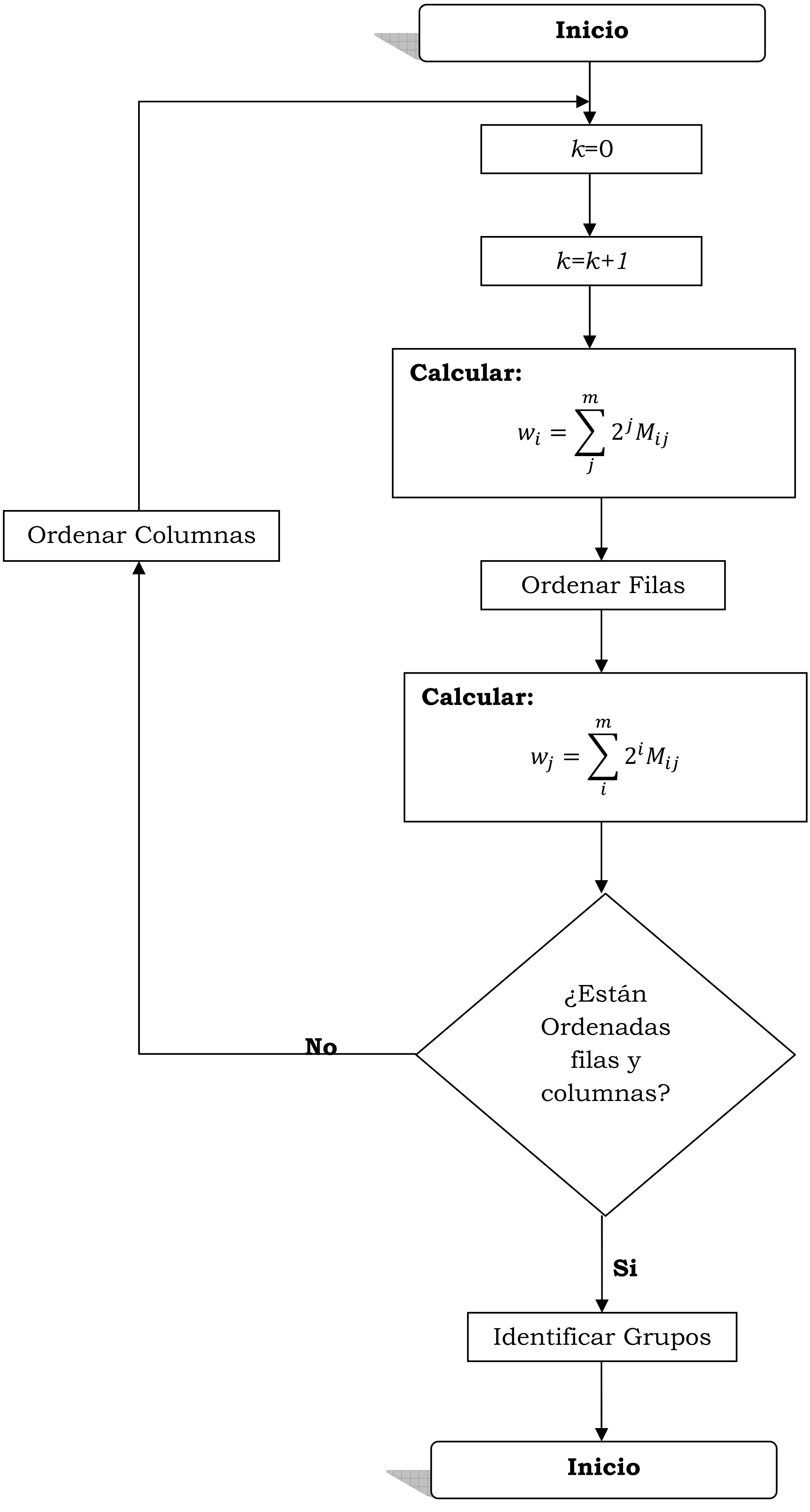
$$w_j = \sum_{i}^{m} 2^i M_{ij} \qquad (2)$$

Paso 3: Si tanto filas, como columnas ya están ordenadas en forma ascendente, finaliza el algoritmo, si no, se debe volver al paso 1 y hacer k=k+1.El criterio de terminación del algoritmo es que tanto filas como columnas estén ordenadas. Esto se explica en la figura 2.

---

[2] Chang, Wysk & Wang, 1998, p. 499.

[3] Ibid. págs. 499-500.



Extraído Por: Héctor Fabio Bonilla Londoño. Estudiante Ingeniería Industrial y Matemáticas Aplicadas PUJ-Cali. Noviembre 2009

Ejemplo 2: Sea un problema de 8 máquinas (A, B, C,..., H) y 10 partes (1, 2, 3,... ,10), como se muestra en la figura 3.

Máquina	**Componentes**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
A	1	0	1	1	0	0	0	1	1	0
B	0	0	0	1	0	0	0	0	0	0
C	1	0	1	0	1	0	0	1	1	0
D	0	1	0	0	0	0	0	0	0	0
E	0	1	0	0	0	0	0	0	0	1
F	0	0	0	0	0	1	1	0	0	1
G	1	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	0	0	1

Solución: en la siguiente figura (Figura 8) se muestra el desarrollo del procedimiento:
Se hace $k = 0$

Paso 1: Se calculan los pesos (W_i) de las filas

Paso 1:Cálculo de peso por cada fila											
n	**4**	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**	
2^n	**16**	**8**	**32**	**256**	**512**	**2**	**4**	**64**	**128**	**1024**	
Máquina	**Componentes**										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**w_i**
A	1	0	1	1	1	0	0	1	1	0	826
B	0	0	0	1	0	0	0	0	0	0	16
C	1	0	1	0	1	0	0	1	1	0	810
D	0	1	0	0	0	0	0	0	0	1	1028
E	0	1	0	0	0	0	0	0	0	0	4
F	0	0	0	0	0	1	1	0	0	1	1216
G	1	0	0	0	0	0	0	0	0	0	2
H	0	0	0	0	0	0	0	0	0	1	1024

Se ordenan las filas en forma ascendente, de acuerdo con su respectivo peso (W_i).

Paso 1.1 :Ordenamiento de Cada Peso Wi en forma Ascendente											
	Componentes										
Máquina	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**w_i**
G	1	0	0	0	0	0	0	0	0	0	2
E	0	1	0	0	0	0	0	0	0	0	4
B	0	0	0	1	0	0	0	0	0	0	16
C	1	0	1	0	1	0	0	1	1	0	810
A	1	0	1	1	1	0	0	1	1	0	826
H	0	0	0	0	0	0	0	0	0	1	1024
D	0	1	0	0	0	0	0	0	0	1	1028
F	0	0	0	0	0	1	1	0	0	1	1216

Paso 2: Se calculan los pesos (W_j) de las columnas

Paso 2.0: Cálculo de peso por cada columna.												
	Componentes											
Máquina	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**2^m**	**m**
G	1	0	0	0	0	0	0	0	0	0	**2**	**1**
E	0	1	0	0	0	0	0	0	0	0	**4**	**2**
B	0	0	0	1	0	0	0	0	0	0	**8**	**3**
C	1	0	1	0	1	0	0	1	1	0	**16**	**4**
A	1	0	1	1	1	0	0	1	1	0	**32**	**5**
H	0	0	0	0	0	0	0	0	0	1	**64**	**6**
D	0	1	0	0	0	0	0	0	0	1	**128**	**7**
F	0	0	0	0	0	1	1	0	0	1	**256**	**8**
Wj	**50**	**132**	**48**	**40**	**48**	**256**	**256**	**48**	**48**	**448**		

Se verifica si los pesos de las columnas se encuentran ordenados. Como no lo están, se procede a realizar la ordenación. Se ordenan las columnas en forma ascendente, de acuerdo con su respectivo peso (W_j)

Paso 2.1 :Ordenamiento de cada Peso Wj en forma Ascendente										
Máquina	**Componentes**									
	4	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**
G	0	0	0	0	0	1	0	0	0	0
E	0	0	0	0	0	0	1	0	0	0
B	1	0	0	0	0	0	0	0	0	0
C	0	1	1	1	1	1	0	0	0	0
A	1	1	1	1	1	1	0	0	0	0
H	0	0	0	0	0	0	0	0	0	1
D	0	0	0	0	0	0	1	0	0	1
F	0	0	0	0	0	0	0	1	1	1
Wj	**40**	**48**	**48**	**48**	**48**	**50**	**132**	**256**	**256**	**448**

Ahora se vuelve al paso 1. De aquí en adelante, solo se mostrarán los valores de los pesos. Se actualiza $k = k+1$.

Paso 1.0 Se calculan los pesos (Wi) de las filas

n	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
2^n	**2**	**4**	**8**	**16**	**32**	**64**	**128**	**256**	**512**	**1024**	
Máquina	**Componentes**										
	4	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**	**w_i**
G	0	0	0	0	0	1	0	0	0	0	64
E	0	0	0	0	0	0	1	0	0	0	128
B	1	0	0	0	0	0	0	0	0	0	2
C	0	1	1	1	1	1	0	0	0	0	124
A	1	1	1	1	1	1	0	0	0	0	126
H	0	0	0	0	0	0	0	0	0	1	1024
D	0	0	0	0	0	0	1	0	0	1	1152
F	0	0	0	0	0	0	0	1	1	1	1792

Se ordenan las filas de acuerdo a sus respectivos pesos.

Máquina	**Componentes**										
	4	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**	**w_i**
B	1	0	0	0	0	0	0	0	0	0	2
G	0	0	0	0	0	1	0	0	0	0	64
C	0	1	1	1	1	1	0	0	0	0	124
A	1	1	1	1	1	1	0	0	0	0	126
E	0	0	0	0	0	0	1	0	0	0	128
H	0	0	0	0	0	0	0	0	0	1	1024
D	0	0	0	0	0	0	1	0	0	1	1152
F	0	0	0	0	0	0	0	1	1	1	1792

Paso 2.0 Se calculan los pesos (Wj) de las columnas

Máquina	**Componentes**											
	4	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**	**2^m**	**m**
B	1	0	0	0	0	0	0	0	0	0	**2**	**1**
G	0	0	0	0	0	1	0	0	0	0	**4**	**2**
C	0	1	1	1	1	1	0	0	0	0	**8**	**3**
A	1	1	1	1	1	1	0	0	0	0	**16**	**4**
E	0	0	0	0	0	0	1	0	0	0	**32**	**5**
H	0	0	0	0	0	0	0	0	0	1	**64**	**6**
D	0	0	0	0	0	0	1	0	0	1	**128**	**7**
F	0	0	0	0	0	0	0	1	1	1	**256**	**8**
Wj	**18**	**24**	**24**	**24**	**24**	**28**	**160**	**256**	**256**	**448**		

Paso 2. Se ordenan los pesos (Wj) de las columnas

Máquina	**Componentes**									
	4	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**
B	1	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	1	0	0	0	0
C	0	1	1	1	1	1	0	0	0	0
A	1	1	1	1	1	1	0	0	0	0
E	0	0	0	0	0	0	1	0	0	0
H	0	0	0	0	0	0	0	0	0	1
D	0	0	0	0	0	0	1	0	0	1
F	0	0	0	0	0	0	0	1	1	1

Wj	18	24	24	24	24	28	160	256	256	448

Se verifica si tanto las filas como las columnas están ordenadas. En este caso sí lo están:
Entonces el algoritmo se detiene. De manera intuitiva, si se observa la matriz final, se
pueden identificar los grupos.

Familias de partes para el ejemplo

Máquina	Componentes									
	4	**3**	**5**	**8**	**9**	**1**	**2**	**6**	**7**	**10**
B	1	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	1	0	0	0	0
C	0	1	1	1	1	1	0	0	0	0
A	1	1	1	1	1	1	0	0	0	0
E	0	0	0	0	0	0	1	0	0	0
H	0	0	0	0	0	0	0	0	0	1
D	0	0	0	0	0	0	1	0	0	1
F	0	0	0	0	0	0	0	1	1	1

1.2 Direct Clustering Technique

En la sección anterior se presentó el Rank Order clustering algorithm que funciona bien con pocas máquinas y pocos componentes. Este algoritmo no es conveniente en los casos con gran variedad de componentes y máquinas, pues se hace ineficiente en el momento de calcular los pesos correspondientes a la posición en la que está cada máquina o componente. Esto se observa al tener que hallar potencias muy altas (Ver fórmulas para el cálculo de los pesos de filas y columnas en el algoritmo anterior). Para sortear esta dificultad, nace el Direct Clustering Technique que se basa en la idea de mover bloques y moverlos conservando las relaciones entre componentes y máquinas. Fue desarrollado por Can & Milner en 1982 y por King & Nakornchai en 1982 de forma independiente.

El algoritmo tiene los siguientes pasos[4]:

Paso 1: Para cada fila i, calcular el peso Wi, así:

$$w_i = \sum_{j}^{m} M_{ij} \qquad (3)$$

Se ordenan las filas de acuerdo con su peso (Wi), en orden descendente.

Paso 2: Para cada columna j, calcular el peso W_j, así:

$$w_j = \sum_{i}^{m} M_{ij} \qquad (4)$$

Se ordenan las columnas de acuerdo con su peso (W_j), en orden ascendente.

Paso 3: Desde $i = 1$ hasta n, mover todas las columnas j, donde $M_{ij} = 1$, a la derecha, manteniendo el orden previo de las filas.

Paso 4: Desde $j = m$ hasta 1, mover todas las filas i, donde $M_{ij} = 1$, hacia arriba, manteniendo el orden previo de las columnas.

Paso 5: Si la matriz actual es igual a la anterior, deténgase, de lo contrario vuelva al paso 3.

[4] Ibid. Pág. 503.

Referencias Bibliográficas

- [1] Mariana Cascante, Jairo Coronado. TECNOLOGÍA DE GRUPOS NT-2202-000-VP.Fundamentos de Producción. Departamento de Ingeniería Industrial. Universidad de los Andes.(2007)

- [2] Askin, R & Standridge, C (1993). Modeling and analysis of manufacturing systems.John Wiley & sons.

- [3] Chang, T, Wysk, R & Wang H. (1998). Computer-Aided Manufacturing. PrenticeHall.

- [4] Niebel, B.(2004). Ingeniería industrial: métodos, estándares y diseño del trabajo.Alfaomega.

- [5] Sule, Dileep (2003). Instalaciones de manufactura. Thompson Learning.