

---

## 4.- METODOLOGIA DE DISEÑO INTEGRADO

Ahora se presenta un esquema integrado para el diseño de talleres de trabajo de células fractales basados en las ideas desarrolladas arriba.

### 4.1.- ESQUEMA GLOBAL

Se usó un enfoque de descomposición para realizar las tareas de asignación y distribución. Tal estrategia es necesaria debido a la complejidad de la tarea de optimización. La fig. 6 muestra el enfoque del diseño integrado. El proceso se inicia con el análisis de capacidad y la localización de las estaciones de trabajo (por células) y posteriormente de realizar esto, nos dedicamos a la tarea del proceso de diseño.

Dentro de la fig. 6 existe un block dedicado a la tarea de asignación que se discutió en la sección previa. Esta se refiere a la asignación de productos hacia las estaciones de trabajo con las distancias tomadas con base en la distribución de la iteración previa.

Las restricciones básicas son aquellas que reflejan las capacidades de las máquinas. La salida de este módulo también proporciona el volumen de flujo entre réplicas de las estaciones de trabajo.

El flujo puede modificarse para reflejar y capturar las realidades día a día de los talleres de trabajo. Por ejemplo, Droler (1989) reporta simulación de sistemas de manufactura virtual (SMV) los productos que casi siempre siguen una de las cinco mejores rutas disponibles en los talleres de trabajo (respecto a la distancia) reflejada en su secuencia de procesamiento. La probabilidad de que un de los producto siga cierta ruta puede estimarse mediante un análisis de

simulación. Habiendo hecho esto, la matriz de flujo puede corregirse para tener esto en cuenta.

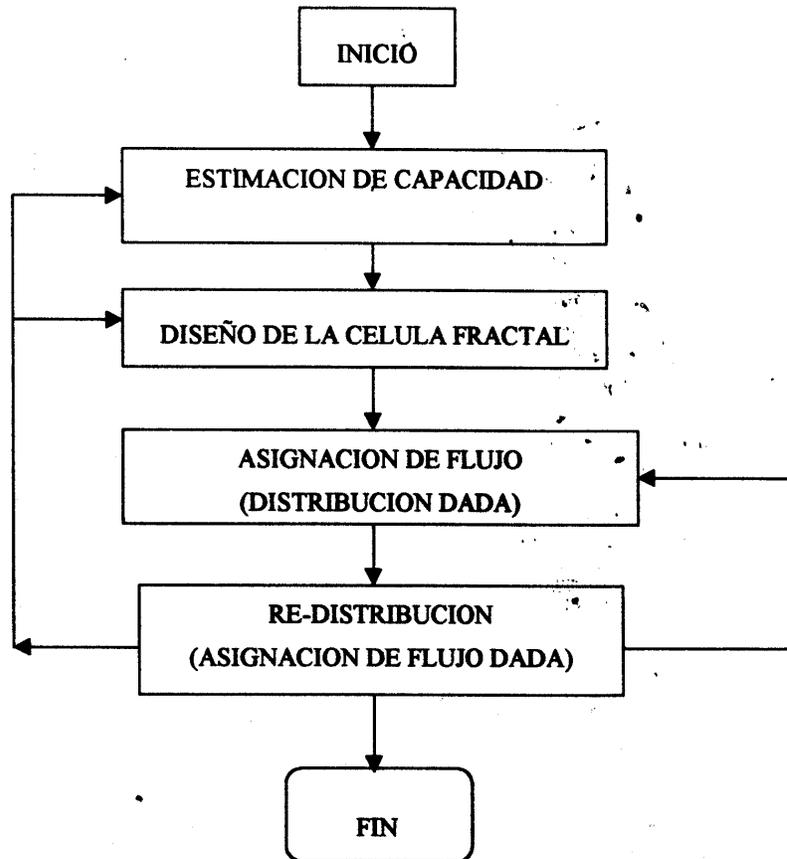


Fig. 6 Diagrama de flujo de la metodología de diseño

El último block en el esquema es el módulo de distribución basado en la formulación QAP. Una vez que los flujos entre las máquinas son conocidos al nivel de replicas, las distribuciones pueden generarse en un intento de minimizar las distancias de flujo conjunto, el

---

costo de transportación con base en la distribución actualizada, el proceso entero puede ser reiniciado, dando lugar a una nueva asignación de productos hacia las estaciones de trabajo. Existe una retroalimentación en todos los niveles de la metodología integrada. El diseñador puede observar la distribución actual generada cuando lo desee, revisar las capacidades de las células y la composición de las máquinas y volver a empezar.

No puede garantizarse que este proceso converja aunque así se desee. En computación, se encuentran situaciones raras donde una solución de una iteración previa era mejor que la actual. Esto puede ser debido a la naturaleza combinatoria bastante difícil de problema. La posición que proponen los autores es pragmática; dejar que esta metodología siga su curso mientras que se almacena la mejor solución encontrada hasta la fecha, si la solución final no es mejor la solución anterior que tiene un valor menor en la función objetivo puede recuperarse siempre.

#### **4.2.- RESULTADOS COMPUTACIONALES**

Se aplicó la metodología de diseño para obtener una distribución fractal en el problema ejemplo, y todos los otros casos en el cuadro 1. El primer ejemplo se discute en detalle y se incluye un resumen posterior de los resultados.

Después de que los pasos del análisis de capacidad y la creación de células en el ejemplo de Co y Araar (1988) cuyos resultados ya han sido vistos, la distribución inicial se proporcionó en la fig. 7 como el inicio del proceso interactivo, se obtuvo la distribución fractal mostrada en la fig. 8. La solución de la fig. 8 utiliza 33 réplicas de máquinas y tiene un puntaje de flujo de 2876 unidades de distancia de flujo rectilíneo. Recuerde que la distribución de grupo modifica de Co y

Araar (1988) utilizó 64 réplicas y tuvo un puntaje de flujo de 3996.

En el código de computadora se permitió la posibilidad ya sea de usar una distribución inicial aleatoria, o una distribución inicial desarrollada por cualquier usuario, reconociendo sus efectos en la solución final obtenida. La primera investigación de distribución fractal se enfocó no tanto a la calidad de los resultados obtenidos, sino más bien a si la metodología podría proporcionar resultados razonables y si la distribución fractal sería promisoría.

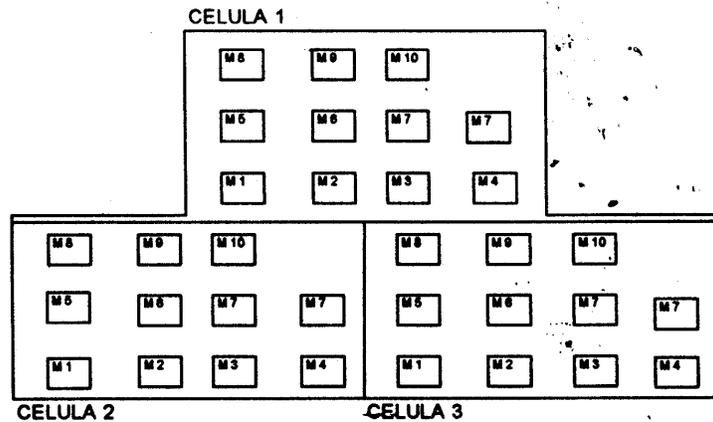


Fig. 7 Distribucion fractal inicial

La tabla 5 muestra como cuatro iteraciones fueron usadas para ir de la distribución inicial a la final. Al terminar la fase de asignación de flujo con la distribución inicial, el puntaje de flujo resultante fue de 3995 unidades. En cada iteración dejamos que el procedimiento de generación de columnas siguiera su curso y encontrar una solución óptima al problema de asignación de flujo. Un total de 252 columnas tuvieron que generarse en el curso de 4 iteraciones antes de

obtener la distribución final, la cual tuvo una distancia de flujo rectilínea de 2876 unidades. En la iteración final para la asignación de flujo ya no se encontraron intercambios que pudieran mejorar la solución.

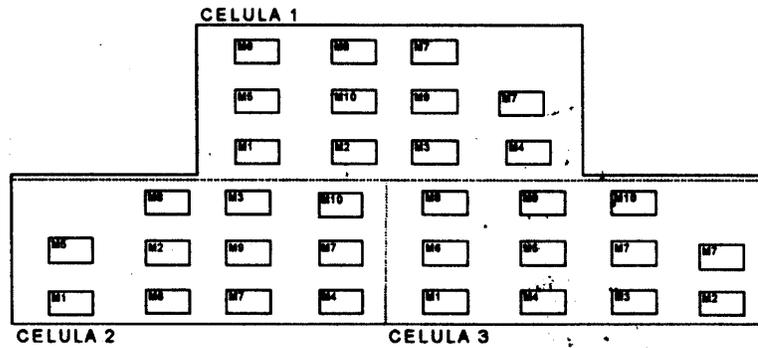


Fig. 8 Distribución final

NUMERO DE COLUMNAS GENERADAS	ITERACION	PUNTAJE DE FLUJO DESPUES DE LA ASIGNACION	PUNTAJE DE FLUJO DESPUES DE LA REDISTRIBUCION
62	1	3955	3404
63	2	3111	3107
64	3	3090	3032
65	4	876	2876

Tabla 5. Historia de la interacción

En la distribución final el producto 1 tiene asignadas la rutas que se muestra en la tabla 6. Los flujos implicados son usualmente muy complejos y se entrecruzan a través de la red. La red de flujo que resulta es compleja con productos asignados a muchas trayectorias, Esto refleja la

flexibilidad del sistema de producción. En la secuencia asignada para el producto 1, es interesante resaltar que las trayectorias fractales intracélulas fueron elegidas, antes de la trayectoria inter-fractales.

NUMERO DE RUTA	SECUENCIA DE LA REPLICA						FRACTAL UTILIZADO
	1-1	4-1	7-2	3-1	10-1	8-1	
1	1-1	4-1	7-2	3-1	10-1	8-1	1
2	1-2	4-2	7-4	3-2	10-2	8-2	2
3	1-3	4-3	7-6	3-3	10-3	8-3	3
4	1-3	4-2	7-4	3-2	10-2	8-3	2,3
5	1-3	4-2	7-3	3-2	10-2	8-3	2,3
6	1-3	4-2	7-3	3-2	10-2	8-2	2,3
7	1-3	4-2	7-4	3-2	10-2	8-2	2,3
8	1-3	4-2	7-4	3-3	10-3	8-3	2,3

Tabla 6 Asignación de flujo

La implementación fue escrita en lenguaje C y se utilizó el software versión 3.0 para resolver el programa lineal de flujo de la red de multi-productos. Para cada caso se grabó, el tiempo utilizado CPU en las dos fases principales de diseño, la asignación de flujo y distribución, estos aparecen en la tabla 7. El encabezado de columna que dice "Iteración hacia la convergencia" se refiere al número de iteraciones de asignación y distribución que se necesitaron para alcanzar la primera óptima local, como se puede observar la solución al problema de distribución de 44 réplicas, toma menos de 3 minutos de tiempo CPU, suficiente para despejar las dudas acerca de la factibilidad computacional.

				ITERACION
NUM. DE DUPLICADO	NUM. DE PRODUCTOS	ASIGNACION DE FLUJO (SEGUNDOS)	DISTRIBUCION (SEGUNDOS)	HACIA LA CONVERGENCIA
44	10	34.7	136.68	5
33	15	17.93	42.68	4
32	40	28.13	5.36	3
31	32	16.33	12.32	4
27	21	10.58	20.61	2
16	22	10.96	4.51	3
16	20	8.49	0.95	2

Tabla 7. Funcionamiento computacional