

308917

51
209



UNIVERSIDAD PANAMERICANA

**ESCUELA DE INGENIERIA
CON ESTUDIOS INCORPORADOS A LA
UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO**

**EXPERIMENTOS DE SIMULACION EN
LOS SISTEMAS PRODUCTIVOS
MODERNOS**

T E S I S

QUE PARA OBTENER EL TITULO DE

INGENIERO MECANICO ELECTRICISTA

AREA: INGENIERIA INDUSTRIAL

P R E S E N T A

FELIPE ANTONIO VILLEGAS MORAN

DIRECTOR DE TESIS: DR. STANISLAW RACZYNSKI

MEXICO, D. F.

1995



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

**Este trabajo está dedicado a mis Padres:
Como testimonio de mi cariño, y
agradecimiento por la educación que,
con tantos sacrificios, me han dado.**

Experimentos de Simulación en los Sistemas Productivos Modernos.

Índice General

| | |
|--|-----------|
| Introducción: La Simulación por computadora como herramienta del progreso | 1 |
| A. Definición de la Simulación | 2 |
| B. Las razones de la Simulación | 3 |
| | |
| Capítulo 1: La Simulación como Herramienta en la Industria | 5 |
| Introducción | 5 |
| 1.1 Principios básicos sobre la Simulación | 6 |
| 1.2 Metodología de la Simulación | 8 |
| Definición del problema. | |
| Construcción de un modelo de simulación. | |
| Especificación de variables y parámetros. | |
| Especificación de las reglas de decisión. | |
| Especificación de las distribuciones de probabilidad. | |
| Especificación del proceso de incremento de tiempo. | |
| Especificación de los valores de las variables y de los parámetros. | |
| Evaluación de Resultados. | |
| Verificación del modelo programado. | |
| Proponer un nuevo experimento. | |
| 1.3 Verificación, Validación y Credibilidad | 11 |
| Elementos de éxito. | |
| Iniciando el proceso. | |
| Obtención de Datos confiables. | |
| Las ideas clave. | |
| La verificación es trabajo del modelador. | |
| Validación, un esfuerzo de equipo. | |
| Limitaciones, peligros y últimos consejos. | |
| 1.4 Lenguajes de Simulación | 18 |
| Lenguajes para simulación continua. | |
| Lenguajes para simulación discreta y discreto-continua. | |
| Lenguajes Orientados a Objetos. | |
| Paquetes de simulación para sistemas de manufactura. | |
| 1.5 La Simulación y la Ingeniería Industrial | 22 |
| 1.6 Justificación Financiera de los Proyectos de Simulación | 22 |
| Descripción del Modelo. | |
| Retorno sobre la inversión de un proyecto de Simulación. | |
| Valor presente neto de un proyecto de Simulación. | |
| Tasa Interna de Retorno para los Proyectos de Simulación. | |
| Conclusiones | 25 |

Capítulo 2: Simulación de Células de Producción27

Introducción27

2.1 Tecnologías de Grupo, o Células de Producción27

Introducción.

Diseño de planta según la Tecnología de Grupos.

Células de Producción (GT) Conceptuales.

2.2 Caso: Células de Producción y Balanceo de Líneas29

Manufactura de Partes Automotrices.

Una Planta Fabricante de Cadenas.

Las Células de Producción en el Área de Ensamble.

Tópicos de la discusión.

Anexo 1.

Anexo 2.

Anexo 3: Propuesta para la distribución del área de ensamblado.

2.3 Solución del caso: Células de Producción y Balanceo de Líneas35

2.3.1 Respuesta a los Tópicos Presentados.

a. Otras áreas a examinar.

b. Cálculo de Número de Máquinas por Célula de Producción.

c. Preparándonos para la simulación.

Esquemas de la simulación.

Un ajuste "pequeño" en los Tiempos de Preparación.

Los últimos datos para la simulación.

2.3.2 Simulación en Xcell*.

2.3.3 Simulación en PASION.

2.4 Conclusiones52

2.4.1 Conclusiones sobre el caso presentado.

2.4.2 Conclusiones sobre la Simulación.

Capítulo 3: Balanceo de Líneas de Producción56

Introducción56

3.1 Teoría de balanceo de líneas de producción57

3.1.1 Introducción.

3.1.2 El diseño de lo sistemas de medición y control según TR.

3.1.3 Desbalanceo de Capacidades de Producción.

3.1.4 Cuellos de botella, y capacidad de los recursos restringidos.

3.1.5 Componentes del tiempo.

3.1.6 Localización del cuello de Botella.

3.1.7 Ahorrando tiempo en los cuellos de botella, Buffers y Enlaces entre las operaciones.

3.1.8 Importancia de la Calidad.

3.1.8 El tamaño de Lote.

3.2 Caso: Kyoto y Osaka, Ltd62

3.2.1 La compañía.

3.2.2 Operación de la producción.

3.2.3 Opciones.

3.2.4 Tópicos de discusión.

3.3 Planteamiento del modelo de simulación64

3.3.1 Respuesta a los Tópicos presentados.

| | |
|--|-----------|
| a. Análisis de la situación. | |
| b. Cálculos previos a la simulación. | |
| c. Formulación del modelo de la simulación. | |
| 3.3.2 Simulación en Xcell+. | |
| 3.3.3 Simulación en PASION. | |
| 3.4 Conclusiones | 75 |
| 3.4.1 Conclusiones sobre el caso presentado. | |
| 3.4.2 Conclusiones sobre la Simulación. | |

Capítulo 4: Líneas Automáticas de Producción78

| | |
|--|-----------|
| Introducción | 78 |
| 4.1 La manufactura Automática y los modelos de Manufactura Flexible | 78 |
| 4.1.1 Automatización. | |
| 4.1.2 Robots industriales. | |
| 4.1.3 Diseño asistido por computadora. | |
| 4.1.4 Sistemas de manufactura flexibles. | |
| 4.1.5 Manufactura integrada por computadora (CIM). | |
| 4.2 Caso: Área de ensamble Electrónico | 82 |
| 4.3 Planteamiento del modelo de Simulación | 83 |
| 4.3.1 Respuesta al Tópico presentado. | |
| a. Análisis de la situación. | |
| b. Cálculos previos a la Simulación. | |
| c. Formulación del modelo de la Simulación. | |
| 4.3.2 Simulación en Xcell+. | |
| 4.3.3 Simulación en PASION. | |
| 4.4 Conclusiones | 91 |
| 4.4.1 Conclusiones sobre el caso presentado. | |
| 4.4.2 Conclusiones sobre la Simulación. | |

Conclusiones de la Tesis94

| | |
|---|-----------|
| Generales | 94 |
| Los Sistemas Modernos de Manufactura | 94 |
| La simulación su realidad y expectativas | 95 |
| El trabajo presentado | 96 |

Bibliografía98

Anexo 1: PASION, un lenguaje de simulación100

| | |
|--|------------|
| Introducción | 100 |
| 1.1 ¿Porqué un nuevo Lenguaje? | 100 |
| 1.2 El lenguaje PASION | 101 |
| 1.3 Relación entre modelos de simulación y los programas de PASION | 102 |

| | |
|---|-----|
| 1.4 Funcionamiento de PASION | 102 |
| 1.5 PAT1: El traductor de PASION a PASCAL | 103 |
| 1.6 El ambiente PASION | 103 |
| 1.7 QMG: El Generador de los Modelos con Líneas de espera | 104 |
| 1.8 El Editor de programas QMG | 106 |
| 1.9 CMG: Generador de Modelos Continuos | 107 |

Anexo 2: Xcell+, Paquete de Simulación para Empresas de Manufactura

| | |
|---|-----|
| Introducción | 108 |
| 2.1 Orden de menú en Xcell+ | 109 |
| 2.2 La planta de producción en Xcell+ | 110 |
| 2.3 Los bloques básicos de diseño en Xcell+ | 111 |
| 2.4 Diseño de Costos de Producción en el Modelo de Simulación | 115 |
| 2.5 Los resultados de Xcell+ | 116 |

Anexo 3: Principios Fundamentales en la Dirección de Operaciones

| | |
|--|-----|
| 3.1 Flujo de Planta: Balanceo y cuellos de Botella | 117 |
| 3.2 Capacidad de almacenamiento en la Fábrica: Las reglas de utilización | 118 |
| 3.3 Algunos cálculos básicos anteriores a la simulación | 119 |
| 3.4 Conducta de un estudio de simulación | 120 |
| 3.5 Simulaciones veloces y lentas: hiper y turbo centros de trabajo | 121 |

Anexo 4: Análisis Crítico de la Teoría de Restricciones

| | |
|---|-----|
| Introducción | 123 |
| 4.1 Desarrollo de la discusión TOC vs JIT & MRP | 123 |

Introducción: La Simulación por computadora como herramienta del progreso.

La simulación por computadora se ha convertido en los últimos años, en una eficaz herramienta del progreso de las ciencias. La filosofía de solución de problemas que implica un análisis con base a la simulación de la realidad, ha mostrado ser una útil manera de encontrar las mejores opciones, a un problema, con la mayor rapidez y con un análisis inmediato de lo que puede suceder si el sistema es modificado de sus condiciones iniciales. El bajo costo, en comparación con las demás técnicas, implica el análisis de simulación por computadora, ha sido uno de los principales factores por los cuales el 80% de las empresas de manufactura en los Estados Unidos utilizan la simulación como método de análisis y toma de decisiones.

La tesis que aquí se presenta, como fruto de una primera investigación sobre el tema Simulación de Sistemas Modernos de Manufactura, tiene en su desarrollo la siguiente estructura: En el primer capítulo se explica lo que implica la herramienta de la simulación aplicada a un empresa de manufactura. Este primer capítulo no pretende explicar la complicada teoría de la simulación (ni siquiera la fácil), sólo es una introducción destinada a que el lector de este trabajo puede familiarizarse con algunos términos, si no lo está aún, y pueda adentrarse a los siguientes capítulos con algunas de las principales ideas sobre este tema, en claro. Los siguientes Capítulos contienen algunos casos concretos y prácticos sobre la aplicación de la simulación en los Sistemas de Manufactura, analizados con dos herramientas de simulación: XCELL+, y PASION. Se explica en cada capítulo el modelo propuesto para el caso concreto, se presentan las simulaciones en una de las dos herramientas o en las dos si este es el caso, y se termina con una pequeña comparación entre los resultados obtenidos con una y otra herramienta.

Si el lector gusta, y tiene interés, puede consultar algunos de los libros y artículos citados en la bibliografía de los capítulos, aunque muchos de ellos se consiguieron con verdadero esfuerzo de investigación y trabajo en bibliotecas (bajo la valiosísima colaboración del director de Tesis). Los libros y artículos de revistas, que tienen origen extranjero son difíciles de conseguir, se recomienda más bien acudir a las sociedades de Simulación por Computadora, para que facilite material actualizado sobre este tema.

A. Definición de la Simulación.

La simulación se define como una técnica numérica empleada para realizar experimentos con ciertos tipos de modelos matemáticos que describen el comportamiento de un sistema complejo, en una computadora digital y durante periodos prolongados. Esto significa que el punto inicial de cualquier experimento de simulación en computadoras es un modelo matemático del sistema que se va a simular; es decir, se supone que ya se ha formulado un modelo matemático que representa la realidad simulada, así como los parámetros que lo describen.

Analizando la definición propuesta se pueden encontrar las siguientes nociones claves:

a. Ya que la simulación es una técnica numérica, se usará sólo como "útil recurso" cuando no se disponga de técnicas analíticas apropiadas para encontrar la solución a un problema propuesto. Esto es, dada la complejidad que involucra el proceso de simulación por computadora es necesario que antes de adentrarse en tan apasionante empresa, el analista se asegure de que no existe otra forma más sencilla de llegar a la solución deseada, esto es para no caer en el tan conocido supuesto de "a quien le dan un martillo, a todo le ve cara de clavo"¹. El hecho de que sea una técnica numérica no implica que sólo se puede utilizar en aquellas ciencias que versan directamente sobre los números, pues es claro, que en una gran variedad de ciencias, incluso sociales, se pueden encontrar modelos matemáticos que describan algunos aspectos de la realidad, de hecho, las aplicaciones de la llamada "Inteligencia Artificial" pueden describir algunos comportamientos humanos, de toma de decisiones, con gran similitud².

b. Una simulación en computadora es un experimento. Con el gran recurso que significan las computadoras personales, la simulación se ha convertido en uno de los experimentos, con validez científica de más bajo costo. Ya no es necesario, para predecir el comportamiento de una sistema, lómese como ejemplo la suspensión de un automóvil, el construir modelos a escala, físicamente reales, lo que significa una gran reducción en el costo de la experimentación. Sin embargo, y puesto que la simulación es un experimento, se debe dar una especial atención, en el desarrollo de un proyecto de simulación a los problemas de *diseño experimental* y *análisis de resultados*³, un punto que se suele descuidar con demasiada frecuencia, y de tal gravedad, que puede invalidar todos los resultados de un estudio de simulación.

c. Aunque una computadora no es un instrumento indispensable para llevar a cabo un experimento de simulación con el modelo matemático de un sistema, me atrevería a decir que un estudio que, por su complejidad, no necesite de una computadora para realizar, con la eficiencia y rapidez necesarias, una simulación, es seguro que se ha utilizado la técnica equivocada para afrontar el problema, esto implica que la simulación no es el mejor método de solución, y por tanto se está perdiendo tiempo, dinero y esfuerzo.

d. Por medio de la computadora se pueden llevar a cabo experimentos con el modelo en un momento específico de tiempo, o bien, se pueden efectuar a lo largo de periodos prolongados. En el primer caso se dice que la simulación es estática o representativa, y en el segundo, que la simulación es dinámica o de series de tiempo. La simulación estática se logra reemplazando una corrida de simulación dada, es decir, cambiando una o más condiciones en las que se lleva a cabo dicha simulación. La dinámica es aquella en la que una simulación dada sólo se amplía en el tiempo, sin cambiar ninguna de las condiciones en las que se lleva a cabo dicha simulación.

e. Los modelos de simulación suelen ser de dos tipos, estocásticos, cuando el modelo matemático incluye al menos una variable aleatoria que se distribuye a lo largo del tiempo

¹ Frase hecha famosa por el Prof. Ing. Jesús Qui Nieto, catedrático de la Universidad Panamericana

² cfr. Mary Lou PADGETT. *Computational intelligence, neural networks and simulation applications*, Memorias del simposio

"Simulación por Computadora e Inteligencia Artificial", febrero de 1994, Universidad Panamericana México DF

³ cfr. MONTGOMERY. *Diseño y Análisis de Experimentos*. Grupo Editorial Iberoamericana. 1991 México

(también conocidos como Monte Carlo), o Determinístico, que se caracterizan por la ausencia del error aleatorio.

B. Las razones de la Simulación.

Para entender las razones que legitimen la simulación en computadora como medio de análisis para modelos de sistemas, es necesario entender la importancia de la experimentación para la solución de los problemas complicados y difíciles de predecir. La realidad es muy compleja, es necesario probar las teorías que se tienen antes de utilizarse, esto caracteriza a la mayoría de los prudentes en el mundo, sobre todo cuando se trata de vidas humanas, o muchos millones de pesos involucrados en una sola decisión.

La simulación por computadora no pretende desplazar a las personas que toman las decisiones en las empresas, o en la vida diaria, simplemente es una herramienta utilísima para la toma de decisiones. La simulación por computadora, aunque podría, no toma las decisiones por sí sola, si bien nos va sólo sugiere una posible solución. Esta solución tiene que ser cuidadosamente analizada, eliminando los posibles "ruidos" que pueden entrar en las variables del sistema. Se intenta que sea una solución que se "estabilice", y además, que sea óptima.

Tómese por ejemplo, el caso de un director de planta que utiliza la simulación para tomar la decisión de comprar una máquina cortadora u otra distinta. El director de producción debe contar con la simulación de la planta para poder decir, qué influencias tendrá la nueva máquina en el sistema productivo, qué sucede si es más rápida, hasta cuanto esto beneficia a la producción y en definitiva, cuánto vale la pena pagar por esa máquina para que sea rentable. No es suficiente que la máquina sea rápida, si la planta no es capaz de utilizar esa rapidez -para todo hay límites- y en definitiva, el cuello de botella puede cambiar de sitio. ¿Qué capacidad es la mejor para mi sistema productivo? esta es una de las preguntas que también se pueden solucionar mediante la simulación. Toda esta información desembocará en un análisis financiero y nos podrá sugerir posibles precios razonables para la máquina, esto es, la simulación de la planta no dice cuál comprar, simplemente da más herramientas para la toma de decisiones del director de la planta.

El método científico es rigurosamente utilizado por todo aquél que se llame simulador, por esto es tan aceptado por la comunidad científica. Los modelos de computadora pueden hacerse tan complejos y realistas como lo permitan las teorías, ya que son innecesarias las soluciones analíticas de ellos. Esto significa que el planteamiento matemático de los modelos a simular no necesitan de sus soluciones explícitas, ya que como se dijo antes, los modelos que son útiles en simulación, usualmente son tan complicados que no se pueden resolver. Por esto es que la simulación tiene una fuerte base en todo lo que son los *métodos numéricos* para la solución de ecuaciones diferenciales. Esta es la parte fundamental del porqué son necesarias las computadoras, sin su facilidad de programación, sería casi imposible encontrar todas las soluciones que implica una corrida de simulación basada en modelos matemáticos de la realidad.

La simulación permite el estudio y la experimentación con instalaciones internas complejas de un sistema dado, ya sea que se trate de una compañía, una industria, un sistema económico o cualquier subsistema de ésta. Mediante la simulación se pueden estudiar los efectos de ciertos cambios de información, organización y ambientales en el funcionamiento de un sistema, haciendo alteraciones en el modelo de dicho sistema y observando los efectos que los cambios tienen en su comportamiento.

La observación minuciosa del sistema simulado puede llevar a una mejor comprensión de dicho sistema y hacer sugerencias que lo perfeccione y que, de otra manera, no se podrían obtener. La simulación se puede emplear como dispositivo pedagógico para proporcionar a estudiantes y practicantes habilidades básicas en análisis teóricos, estadísticos y toma de decisiones.

La experiencia en el diseño de un modelo de simulación en computadora puede ser más valiosa que la simulación propiamente dicha. Los conocimientos obtenidos al diseñar un estudio de simulación sugieren a menudo cambios en el sistema simulado. Los efectos de estos cambios se pueden comprobar después mediante la simulación, antes de aplicarlos al sistema real. La simulación de sistemas complejos puede ayudar a incrementar los conocimientos sobre qué variables son más importantes que otras dentro del sistema así como su forma de interacción.

La simulación se puede utilizar para experimentar con situaciones nuevas sobre las que se tengan pocos datos o ninguno a fin de estar preparados para lo que pueda suceder. La simulación puede servir como una "prueba de prevención" para comprobar nuevas normas y reglas de decisión con el fin de manejar un sistema, antes de que se corra el riesgo al experimentar en el sistema real.

Para ciertos tipos de problemas estocásticos, la secuencia de los sucesos puede ser de vital importancia. Es posible que la información sobre los valores y los momentos esperados no sea suficiente para describir el proceso. En estos casos el método Monte Carlo puede constituir la única forma satisfactoria de proporcionar la información requerida. Es posible efectuar simulaciones tipo Monte Carlo para verificar las soluciones analíticas de algunos problemas.

La simulación permite estudiar sistemas dinámicos, ya sea en tiempo real, tiempo comprimido o extendido. Cuando se introducen elementos nuevos a un sistema, la simulación se puede emplear para anticipar estancamientos y otros problemas que puedan surgir en el comportamiento del sistema⁴.

⁴cf. NAYLOR, Thomas. Experimentos de Simulación en Computadoras con Modelos de Sistemas Económicos. Editorial Limusa, 1a. ed., 1977. México DF.

Capítulo 1. La simulación como Herramienta en la Industria.

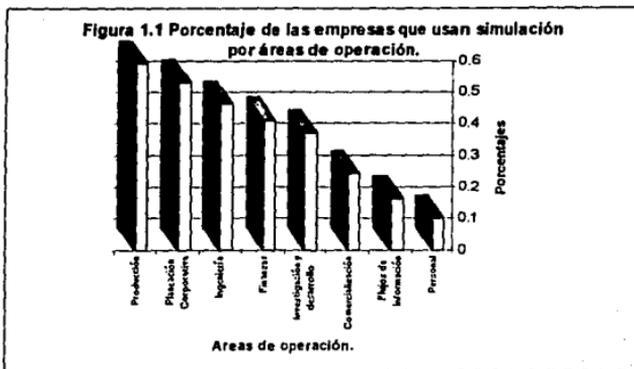
Introducción.

La simulación es una de las técnicas de análisis que en los últimos años han tenido el mayor crecimiento en su aplicación. Esto se debe al crecimiento que se ha experimentado a nivel mundial en lo referente al poderío de las computadoras. Esta corriente tecnológica en los últimos 10 años ha pasado de ser privilegio de unos cuantos super-expertos a realidad cotidiana de todos los estudiantes, su presencia no se limita sólo a las casas, sino también a los cuartos. Los nuevos procesadores que se ofrecen en el mercado llevan una marcada tendencia a la potencia cada vez mayor, y por lo mismo tienen una capacidad cada vez mas grande de ser aplicados en todo tipo de trabajos, desde los más convencionales, como puede ser un editor de texto, hasta los proyectos científicos mas complicados. Por decirlo en pocas palabras, las computadoras se han convertido en pieza fundamental de la vida cotidiana de los hombres del siglo 21, pero también en herramienta crucial en los continuos avances tecnológicos que se experimentan en todos los ámbitos de nuestra sociedad.

En lo que se refiera a la industria, la simulación se emplea fundamentalmente en la disciplina llamada Dirección de Operaciones, y se usa, principalmente, para determinar los esquemas productivos así como las necesidades de materiales; para analizar líneas de espera, niveles de inventarios y procedimientos de mantenimiento; para hacer la planeación de la capacidad, planeación de materiales y procesos; ... y una inmensa gama de otros factores productivos. Con cada vez más frecuencia la gente de producción prefiere "ver" el problema que resolver complicados sistemas matemáticos de los que sólo se pueden contemplar unas cuantas cifras, resultados de una obscurísima "caja negra". Una encuesta recogida por el Instituto de Ciencia en la Dirección¹ indica que el 89 por ciento de sus clientes usan la simulación, es claro que si esta encuesta se repitiera hoy en día no nos cabría la menor duda que la simulación es una herramienta tecnológica necesaria para la industria contemporánea.

| Áreas de Operación | Porcentaje |
|----------------------------|------------|
| Producción | 59% |
| Planeación Corporativa | 53% |
| Ingeniería | 46% |
| Finanzas | 41% |
| Investigación y desarrollo | 37% |
| Comercialización | 24% |
| Flujos de información | 16% |
| Personal | 10% |

¹David P. CRISTY and Hugh J. WATSON, "The Application of Simulation: A Survey of Industry Practice", Interfaces 13, no. 5 (October 1983).



Algunas empresas encuestadas respondieron que la mayoría de los modelos empleados para la simulación han sido desarrollados por ellas mismas.

Es por todo esto que para entender el contenido de los capítulos siguientes es necesario presentar una aproximación breve de lo que significa la poderosa herramienta de la simulación, sobre todo de su manera de funcionar así como de los pasos a seguir para realizar experimentos de simulación cuyos resultados tengan la suficiente validez como para ser creídos y aplicados.

1.1 Principios básicos sobre la simulación.

Bueno, sabemos que simular es importante en el nuevo contexto industrial, pero ¿que es simular?. Es claro que para seguir adelante tendremos que hacer algunas breves consideraciones sobre el significado del término empleado. Un diccionario dice:

Simular: La representación imitativa del funcionamiento de un proceso o sistema por el funcionamiento de otro (la simulación por computadora de un proceso industrial); el análisis de un problema con frecuencia no se realiza directamente, si no por el significado de un sistema que lo simula.²

Es claro que el significado primario de esta definición se comprende en el sentido de imitar la realidad¹. Cuando se estudia la realidad en la mayoría de las ocasiones resulta imposible, o muy costoso experimentar directamente sobre ella, es por esto que se desarrolla sistemas o aparatos que mediante su funcionamiento pueden representar o significar el comportamiento esperado de algunas variables del problema original. Estos aparatos se interpretan a la luz de materias como la Dinámica de Sistemas Físicos, y otras, que en sus correspondientes estructuras, establecen criterios de relación entre las variables. Claro que no siempre se simula mediante aparatos o maquetas, la modernidad que nos interesa en esta tesis comprende el estudio de todos aquellos problemas que se simulan a través del aparato conocido como Computadora. Claro, este aparato necesita ser programado, y estos programas deben contener funciones matemáticas que representen modelos reales del problema. Es ahora cuando llegamos al punto crucial del asunto, la simulación por computadora de sistemas productivos modernos tiene como su más profundo fundamento la definición de modelos matemáticos-probabilísticos con los cuales se puedan establecer relaciones objetivas entre las variables implicadas en un problema dado.

¹Webster's New Collegiate Dictionary, 9th ed.

²Si el lector quiere profundizar en las distintas definiciones de Simulación convalida que confrontara el artículo: Compilation of definitions of Simulation, de A. Alan B Pritsker, Simulation, Society for Computer Simulation, Agosto 1979.

Bueno, si no se tienen muchos elementos de programación, o incluso de modelación matemática, no hay de qué preocuparse. En la simulación se distinguen tres grandes técnicas:

1. Simulación por programación probabilísticas directa, también llamada Simulación Monte Carlo.
2. Simulación mediante interfaces visuales o lenguajes especiales de simulación.
3. Simulación mediante la programación de modelos matemáticos especiales.

El primer tipo de simulación comprende la etapa primaria de la ciencia, se trata de programar distintos comportamientos de funciones probabilísticas basados en variables aleatorias. Mediante una relación entre estas funciones y algunas variables se pueden representar eventos reales sujetos a probabilidad. Por ejemplo todos los juegos de "azar" se pueden simular de esta manera, así como problemas cuya raíz fundamental sea la probabilidad.

Claro, cuando los problemas se van haciendo complicados, es difícil desarrollar, para cada uno, un programa especial, de aquí nace la necesidad de hacer interfaces de programación, que mediante métodos sencillos, en su mayoría visuales, el usuario pueda concentrarse sobre todo en el problema simulado y no en la programación de éste. Estos son los sistemas más usados, tienen como base los del primer tipo, pero en la actualidad los analistas no profundizan en la programación ni en la modelación matemática.

El tercer tipo de técnica es usada principalmente cuando se trata de simular eventos tales como los Económicos, simulación de climas, de vientos, de sistemas ecológicos, de experimentos químicos, etc. Donde se pretende en primer lugar, mediante la simulación, más que reproducir la realidad, probar un determinado modelo matemático que establece ciertas relaciones de las variables que intervienen en un determinado fenómeno. Así es que en la actualidad, por el grado de complejidad que presenta, esta técnica de modelado matemático y su respectiva programación se realiza principalmente con fines pedagógicos y relacionales, y no con la intención de predecir lo que va a suceder. Para tener más claro el problema de la complejidad que presenta la realidad y su consiguiente resistencia a simularse basta tener presente el reciente auge de la teoría del Caos, y esfuerzos titánicos en lo referente a predicciones del clima.

Cuando un universitario se adentra en el estudio de la simulación es común que se encuentre con términos poco comunes. Por ejemplo, se denomina eventos estocásticos (o probabilísticos) a los que involucran en su desarrollo funciones de probabilidad; se les llama, por el contrario eventos Determinísticos a los que no incluyen experimentos con funciones de probabilidad. Hay otros términos comunes en el argot de la simulación, como eventos Discretos y Continuos. Los primeros hacen referencia a aquellos procesos o eventos que se desarrollan por "pasos grandes" en el tiempo, es decir, los cambios de estado del evento simulado acepta claras partes en el transcurso de su realidad; por el contrario los Continuos son los que no aceptan estos "pasos", o son tan "pequeños" que no vale la pena diferenciarlos. Para entender mejor estos términos convendría hacer referencia al Cálculo Diferencial e Integral.

Puesto que en este trabajo de Tesis el objetivo principal es mostrar las aplicaciones que tiene la Simulación en el Control de la Producción, es claro que nuestra técnica elegida será la segunda. Aunque no por esto se desechan las demás técnicas existentes, todas tienen sus aplicaciones en la producción, sólo difieren éstas en la complejidad de los problemas simulados, así como las finalidades que se persigan con un estudio de simulación determinado. En la tesis se utilizaron dos programas especiales para simular los sistemas de producción:

A. XCELL + , Factory Modeling System, release 4.0. que tiene como autores a Richard Conway, William L. Maxwell, John O. McClain, Steven L. Worona, de la Johnson Graduate School of Management & School of Operations Research and Industrial Engineering, de la Cornell University. Estos autores dicen:

Xcell + es un paquete de computadora que le ayuda a construir un "modelo lógico" de un proceso de manufactura. El modelo que usted puede construir tiene la capacidad de simular las operaciones de una fábrica, de manera que puede estimar la capacidad de producción, así como estudiar las alternativas para mejorar el funcionamiento.

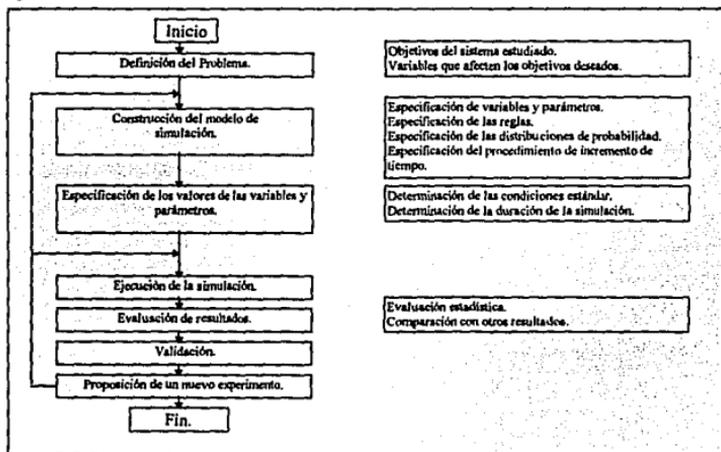
B. PASION, Pascal Simulation, que tiene como autor al Ph. Stanislaw Raczyński, profesor de la Universidad Panamericana y asesor de esta tesis. PASION es, en palabras del autor:

Es un lenguaje de simulación orientado a los procesos y eventos y diseñado para los usuarios de Pascal. El lenguaje tiene dos niveles de estructura (procesos/eventos) y permite usarse junto con todas las estructuras de Pascal. PASION también ofrece las ventajas de la programación orientada a objetos. Provee de elementos necesarios para analizar eventos aleatorios, colas, eventos cuasi-paralelos, discretos o continuos.⁴

1.2 Metodología de la Simulación.

En la siguiente figura⁵ se muestra un diagrama de flujo de las principales fases en el proceso de estudio de la simulación. En esta parte de la tesis se explicarán brevemente cada una de estas fases, sobre todo en lo que se refiere a la simulación de procesos de manufactura. Se sugiere poner especial atención en los factores claves que aparecen en el lado derecho de la figura.

Figura 1.2



Definición del problema.

Esencialmente consiste en especificar los objetivos del estudio, e identificar las principales variables controlables o incontrolables del sistema en cuestión. Estos objetivos podrían ser tales como: elevar la producción de una línea, optimizar los tiempos de mantenimiento, elevar el volumen de producción, disminuir los costos de producción y el número de personal, disminuir el volumen de material en proceso, aumentar los tiempos de producción de una máquina, disminuir los tiempos de reparación, evaluar una nueva inversión en maquinaria, evaluar un nuevo diseño de planta, evaluar una nueva política de producción, etc.

⁴PASION, Lenguaje Overview, Symposium on Computer Simulation, The Society for computer Simulation & Universidad Panamericana, México City, 1991.

⁵Production and Operation Management: A life Cycle Approach, Richard B. Chase y Nicholas J. Aquilano, 6a. de, De Irwin. 1992.

Construcción de un modelo de simulación.

La principal diferencia que existe entre la simulación y las técnicas de programación lineal o teoría de colas es el hecho de que cada modelo de simulación debe de estar hecho a la medida de un problema determinado. Por supuesto, la simulación asume las técnicas de programación lineal y la llamada Investigación de Operaciones, pero cuando estas técnicas se enfrentan a la realidad, la complejidad de las relaciones entre las variables aumentan de manera que no existen modelos matemáticos clásicos que solucionen todos los problemas existentes. Hay que destacar, además, que la simulación es eminentemente dinámica, esto es, los modelos cambian constantemente de un estado a otro en el transcurso del tiempo.

Especificación de variables y parámetros.

El primer paso que se debe dar en la construcción de un modelo de simulación es determinar cuáles propiedades del sistema real han de permanecer con valores constantes (parámetros), y cuáles deben de variar su valor durante la simulación (variables).

Especificación de las reglas de decisión.

Las reglas de decisión (o también llamadas reglas de operación) son condiciones fijas bajo las cuales el desarrollo del modelo de simulación es ejecutado. Es importante señalar que el establecimiento de estas normas o reglas que relacionan las variables y parámetros, constituyen directa o indirectamente el meollo de muchos estudios de simulación. En muchas simulaciones, las reglas de decisión son reglas de prioridad (por ejemplo, qué cliente debe ser servido primero, qué trabajo debe ser procesado primero), y en algunas situaciones puede ser muy complicado, teniendo en cuenta el gran número de variables en el sistema. Por ejemplo, las reglas de control de inventarios las órdenes de compra pueden ser declaradas en función del nivel de inventario actual, las órdenes previamente hechas pero no recibidas, el volumen de órdenes retrasadas y el valor del inventario de seguridad.

Especificación de las distribuciones de probabilidad.

Existen dos grandes categorías de distribuciones que pueden ser usadas en las simulación: las distribuciones de frecuencias empíricas y las matemáticas. Mientras más elementos matemáticos tenga la persona encargada de hacer el estudio de simulación estará en mejor posición de simular el evento con mayor exactitud. Cuando se hacen suposiciones de que un determinado experimento tiene una distribución de probabilidad "parecida" a la función gama, poisson, normal, se está realizando una importantísima suposición que puede afectar gravemente el resultado de la simulación. Pienso que este es el gran meollo de la simulación, en el que hay que tener mucho cuidado, experiencia y por supuesto, un basto conocimiento de matemáticas, estadística, etc., etc.

Especificación del proceso de incremento de tiempo.

En los modelos de simulación, el tiempo puede ser avanzado con uno de los siguientes métodos: (1) fijando los incrementos de forma constante o (2) asignar a los incrementos de tiempo un variable. Bajo cualquiera de estos métodos, el concepto de la simulación del transcurso de tiempo es importante. En el método en que los incrementos son fijos, los incrementos de reloj simulado son determinados (horas, minutos, segundos) y la simulación transcurre de un periodo de tiempo a otro. En cada uno de los momentos del reloj, el sistema debe analizarse para determinar si algún evento a sucedido, si no es así, el sistema da el siguiente paso temporal.

En el método en que los incrementos del tiempo son variables, el reloj de simulación avanza la cantidad necesaria para alcanzar el siguiente evento esperado. ¿Cuál de estos métodos es apropiado?, la experiencia determinará la decisión que el investigador tomó para una investigación concreta. En general cuando los eventos se verifican con gran frecuencia, y su variedad es grande conviene irse paso a paso en el transcurso del reloj de la simulación. Por el contrario el segundo método se recomienda cuando hay poca cantidad de eventos y éstos suceden de vez en cuando.

Especificación de los valores de las variables y de los parámetros.

a. Determinación de las condiciones de arranque.

Una variable, por definición, cambia su valor conforme la simulación se realiza, pero debe tener algún valor inicial. La determinación de estos valores es una decisión táctica en la simulación. Esto se debe a que el modelo se basa en una serie de valores iniciales de arranque, hasta que el modelo se alcanza un estado de comportamiento "normal" (sin embargo hay sistemas que nunca alcanza un estado de estabilidad, aunque este resultado puede inclinarnos a pensar que su "normalidad radica precisamente en su continua inestabilidad). Sin embargo hay valores de inicio mejores que otros; para entender este problema, los investigadores han formulado varias aproximaciones al "mejor" valor inicial y solucionar los problemas generados por el arranque: (1) descartar los valores generados antes de alcanzar la "normalidad" del sistema; (2) seleccionar los valores iniciales que reduzcan el tiempo de "normalización"; (3) seleccionar los valores que eliminen el arranque de la simulación. El uso de laguna de estas alternativas no quita la necesidad de que el analista tenga alguna idea del rango de valores que se espera obtener. Por otro lado, una de las características de la simulación es que el analista debe tener el suficiente criterio para evaluar la simulación así como sus resultados esperados. Esto nos lleva a un punto trascendental: el analista o investigador de la simulación, esto es, la persona que intente utilizar este poderoso instrumento, debe de tener amplios conocimientos de aquello que se simula (este es el principal atractivo y reto de los investigadores de la simulación, pues se exige de ellos un alto grado de interdisciplinaria científica).

b. Determinar la duración de la simulación.

El periodo de tiempo que dure la simulación depende de los objetivos del experimento. Lo más usual es que dure el tiempo suficiente para que el sistema alcance su estado de normalización (o condiciones de equilibrio). Otra posible opción que se tiene es ejecutar la simulación por un periodo de tiempo establecido, una semana, mes año, década, y observar si los valores de los estados finales son razonables, congruentes y parecidos. Una tercera opción es seleccionar un periodo de tiempo lo suficientemente largo para cumplir con los requisitos del estudio estadístico que se realiza en la validación, esta última se comentará más adelante.

Evaluación de Resultados.

a. Estudios estadísticos de los resultados.

Las conclusiones que se pueden formular de un análisis por simulación dependen fundamentalmente del grado en que el modelo de la simulación asemeja la realidad simulada, pero esto sólo se puede verificar por medio de una comparación estadística entre estos dos entes (la realidad y el modelo). De cualquier forma, muchos investigadores ven en la misma simulación una manera de comprobación del modelo, pues como se dijo antes, el analista debe tener conocimientos suficientes como para determinar si los resultados obtenidos son razonables. Para los más versados, de cada una de las ejecuciones del modelo simulado se pueden obtener suficientes datos como para realizar una comparación estadística formal, que nos lleve a un análisis formal del modelo⁵. Además, los métodos estadísticos pueden ser empleados para encontrar la mejor alternativa en un grupo de posibles soluciones propuestas, en estos casos, se puede incluso llegar a utilizar algunos métodos complicados de investigación de operaciones en el área estocástica.

⁵Algunos de los modelos estadísticos más usados en la evaluación de experimentos simulados son análisis de varianza, regresiones, y pruebas de t. Para más información de estas pruebas se sugiere al lector que se remita a la bibliografía específica de Probabilidad y Estadística, por supuesto, de grados superiores.

b. Comparación con otra información.

En la mayoría de las situaciones, el analista tiene información disponible con la cual puede comparar los resultados obtenidos de la simulación, por ejemplo, datos históricos del sistema real, y su propia experiencia. Probablemente esta información no es suficiente como para validar las conclusiones, de manera que, en última instancia, y después de toda la ciencia aplicada - ¡qué gran paradoja! - la única forma segura de comprobar que tan bien funcionan los resultados obtenidos es implementándolos en la realidad.

Verificación del modelo programado.

Dentro del contexto de la simulación, se entiende por validación a la comprobación del programa de computadora, para asegurarse que la simulación efectuada es correcta. Específicamente, es checar el grado en el que el programa de computadora se asemeja al modelo elaborado, y éste a su vez, comparado con el evento real simulado. Los errores pueden introducirse en el programa por equivocaciones en la codificación o equivocaciones en la lógica de programación. Las equivocaciones son relativamente sencillas de identificar y localizar, pues en la mayoría de los casos, sencillamente, el programa no es ejecutable. Sin embargo, cuando los errores que se cometen son de tipo lógico, el programa se ejecuta, pero los resultados de éste son erróneos.

Para resolver este problema, el de validación, el analista usualmente tiene tres alternativas: (1) Poseer una impresión de todos los cálculos y procedimientos hechos por el programa, paso a paso, y revisarlos a mano. (2) Simular las condiciones presentes y comparar los resultados con el sistema existente. (3) Tomar algún punto de la simulación y comparar este resultado resolviendo el problema matemático pertinente de la situación hasta ese punto dado. Es claro que las dos primeras opciones tienen inconvenientes, es deseable emplear la tercera opción, sobre todo porque se supone que el analista tiene un modelo matemático en mente, de manera que el problema concreto se supone sencillo de resolver sin ayuda de la simulación. Claro, es que la simulación no es una herramienta para analizar una situación concreta dentro de una amplia gama de posibilidades, sino que es la herramienta adecuada para analizar todas las posibles situaciones que se pueden presentar en la riqueza de la realidad.

Proponer un nuevo experimento.

Con base en los resultados obtenidos, un nuevo experimento de simulación puede ser considerado. Es claro que nos gustaría cambiar una serie de factores: parámetros, variables, reglas de decisión, condiciones de arranque, y tiempos de ejecución. Esta es la gran maravilla de la simulación, una vez que se tiene un modelo adecuado de la realidad, se puede "jugar" con él hasta encontrar las condiciones deseables del problema. Por ejemplo, en lo referente a los parámetros nos gustaría repetir las simulaciones con diferentes costos de los productos, y observar los resultados. Experimentar con diferentes reglas de decisión nos da la posibilidad de alcanzar mejores resultados y nuevas variantes de solución a un problema dado. Además, los resultados obtenidos de la última simulación pueden utilizarse como condiciones iniciales de la siguiente.

1.3 Verificación, Validación y Credibilidad².

Bueno, llegados a este punto, creo conveniente añadir al presente trabajo una pequeña discusión sobre los conceptos de verificación, validación, y credibilidad; porque una cosa es que el modelo parezca estar correcto, otra que en realidad sea correcto, y otra muy distinta es que el cliente crea que es correcto. El usuario final de un modelo tiene toda la razón de ser escéptico sobre la exactitud requerida, y de ser cuidadoso sobre las predicciones del modelo. Uno de los principales problemas del trabajo de simulación es la construcción de un modelo adecuado y convincente para los usuarios finales, de manera que sea una confiable fuente de información para la toma de decisión dentro de una empresa.

² Las ideas que se citan en este subcapítulo son un extracto de el artículo publicado Convincing Users of Model's validity is challenging aspect of modeler's job, escrito por John Carson, Iel, junio 1966

Para aclarar los conceptos que involucra esta discusión es necesario distinguir entre los términos *verificación, validación y credibilidad*:

Verificar es el proceso donde el modelador responde a la pregunta: *¿El modelo funciona como fue diseñado?, ¿el programa corresponde al diagrama del flujo de proceso?, La lógica de la codificación toman correctamente las suposiciones hechas?*

Validación es el proceso en el cual el modelador y el usuario se responden a las preguntas: *¿Qué tan exactamente el modelo representa la realidad?, Puede utilizarse el modelo como base para la toma de decisiones del sistema real?, Un modelo válido es aquel que tiene la suficiente exactitud que se necesita para llegar a nuestro objetivo planteado. Un modelo válido puede ser utilizado para sustituir la realidad en cuando se le exponen a cierto tipo de experimentaciones, que desde un principio, establecen el grado de exactitud del modelo propuesto, y son motivo de nuestro objetivo.*

Un modelo creíble es aquél que el cliente acepta como válido y es utilizado como herramienta en la toma de decisiones.

La validación y la verificación no deben de tomarse como un paso más que se debe realizar al final del proyecto. Son una parte integral del proceso, se utilizan desde el inicio del proyecto, continúan a través de la construcción del modelo y siguen aún en el período en que este modelo es utilizado. Tomar al cliente como miembro del equipo de diseño facilita la realización del proyecto, pues se le dará al modelo la exactitud necesaria y por tanto, será aceptado en la toma de decisiones. Es claro que la aceptación y utilización final del modelo por el cliente es la mejor prueba de credibilidad.

Elementos de éxito.

Los modelos, y proyectos, exitosos de simulación se caracterizan por la continua comunicación entre el analista y el cliente. Para la mejor aceptación del modelo, es mejor que no se formen en el cliente expectativas muy altas, pues el modelo no soluciona ningún problema, sólo proporciona una herramienta utilísima de análisis de los problemas. La simulación es cara y tardada, es por esto que el analista debe de decirle al cliente con claridad el valor real de estas dos variables, y además, durante el desarrollo del proyecto debe procurar mantener informado al cliente de los avances del proyecto, así se mantendrá su interés. Finalmente el analista debe dejar claro que él sólo es una parte de un equipo de trabajo, siendo la parte más importante el cliente, pues es el que tiene la necesidad y la mejor dimensión de lo que se quiere obtener del proyecto planteado. Esto último es fundamental para la credibilidad del modelo.

Una vez que la decisión de simular ha sido tomada, los directores de la empresa deberán de proporcionar toda la ayuda necesaria para el proyecto. Aunque en el proyecto, por un lado se encuentra el tiempo y el esfuerzo del analista (modelador), un modelo exitoso requiere de toda la asesoría operacional del funcionamiento del sistema modelado, esto es, en un proyecto de manufactura, es necesaria la colaboración de los ingenieros de planta y manufactura, supervisores, capataces y obreros, así como los ingenieros de diseño y de sistemas de la planta. Ellos son los que proporcionarán al analista de toda la información necesaria, y además son ellos los futuros usuarios del modelo de simulación construido, por esto, es de capital importancia que el modelo les proporcione toda la confianza para emitir un juicio de validación del modelo y pasen este juicio a los directivos superiores. Esta comunicación continua con el cliente juega un papel crucial en la formulación y credibilidad de un modelo de simulación.

Iniciando el proceso.

Las primeras dos preguntas que un analista necesita cuestionarse son:

- 1) *¿Cuál es el objetivo de este estudio?*
- 2) *¿Cómo es que el sistema (o diseño) real trabaja (o como se supone que debe de trabajar)?*

Junto con los directivos, el analista debe establecer objetivos reales para el proyecto de simulación. Usualmente, los directivos e ingenieros tienen muchas preguntas que les gustaría que fueran respondidas. Esto puede incluir tópicos como el de capacidad de la planta, cuellos de botella para un diseño dado, los efectos de no organizar los tiempos de mantenimiento, las ventajas y particularidades de una determinada política de producción, el funcionamiento de los sistemas de almacenaje, etc.

No todas estas cuestiones deben de ser atacadas por la simulación. La simulación es un buen instrumento para evaluar moderadamente un gran número de diseños de sistemas, pero el modelador y los usuarios finales deben de generar estas alternativas. A diferencia de los modelos de optimización matemáticos, la simulación no obtiene la mejor opción de un conjunto de opciones posibles.

Además, la simulación no es la mejor forma de atacar todos los problemas⁶. Para empezar, el analista debe estar seguro de que los objetivos son realistas y pueden alcanzarse en el tiempo y con los recursos disponibles. Este punto se lo deben de hacer ver a los directivos, para aclararles cuáles de sus preguntas pueden ser, o no ser, resueltas con un modelo de simulación.

El equipo de simulación debe de determinar los límites o fronteras del sistema simulado: Qué es lo que está dentro del sistema, y qué está fuera de él; cuáles son los factores externos que chocan con él. Además deben de decidir el nivel de detalle al que se llegará con el modelo. Esto depende en gran medida del propósito del modelo y de la información disponible del sistema. Hablando en grandes términos, el propósito debe ser clasificado en una de tres categorías, que requerirá un crecimiento en detalle de los modelos:

- 1) Determinar las bases de diseño de un nuevo sistema.
- 2) Evaluar los aspectos operativos del sistema existente.
- 3) Probar las estrategias de control.

Como es natural, el diseño de un nuevo sistema puede necesitar hasta del último detalle, con *peños y señales*, del funcionamiento del sistema. La estrategias de las pruebas de control requieren de una gran cantidad de detalles. La experiencia muestra que es muy raro encontrar a una persona que domine todos los detalles técnicos del sistema, en base a los cuales se pueda construir el modelo, por esto es necesario que el analista haga a veces una función detectiveza de recolección de información. En estos casos es una buena costumbre que el analista haga una lista de preguntas que sean contestadas por los empleados-clientes. Es curioso ver que con mucha frecuencia estas respuestas difieren unas de otras. La construcción de un buen modelo de simulación de un sistema de manufactura requiere de una buena combinación de la destreza y conocimientos del analista y el trabajador.

Obtención de Datos confiables.

Además de conocer con exactitud cómo es que el sistema funciona, el analista necesitará de buenos datos. Se entiende por datos los tiempos de proceso, los tiempos de transporte y espera, tiempos de falla, tiempos de mantenimiento y todos los demás valores de las variables y parámetros de importancia.

La buena calidad de los datos requeridos para un estudio de simulación con frecuencia no son conocidos rápidamente por el analista. Muchos datos son generados en empresas modernas de manufactura, sobre todo los que han implementado sistemas automáticos de producción, sin embargo estos datos con frecuencia están presentados en una forma poco precisa y legible, algunos son de variables sin importancia, y otros sólo presentan promedios tomados en largos periodos de tiempo, durante los cuales se han introducido serios cambios en la línea de producción y por tanto, en los procesos concretos de las máquinas en cuestión.

⁶En palabras del Profesor Jesús Qui Nieto, de la Escuela de Ingeniería de la Universidad Panamericana, "Al que le dan un martillo, a todo le ve cara de clavo".

Existen algunos métodos que nos pueden conducir a la obtención de los datos que necesitamos en un estudio de simulación:

- 1) Estudios de tiempos y movimientos.
- 2) Registros Históricos.
- 3) Demanda de vendedores.
- 4) La mejor suposición del cliente.
- 5) La mejor suposición del modelador.

Un estudio de tiempos de un sistema existente, si está diseñado apropiadamente, proporciona los datos más exactos posibles. Si el analista no conduce este estudio de tiempos personalmente, debe de aclarar el tipo y exactitud de los datos requeridos. Si el estudio se va a realizar de un nuevo proyecto, el estudio de tiempos se debe de realizar sobre un sistema similar que pueda ser evaluado, si es que existe por supuesto, elementos comunes entre estos dos sistemas.

Los Registros Históricos como los reportes de producción o de mantenimiento, deben de ser cuidadosamente revisados. Esto se debe a que los términos de mantenimiento engloban distinta casuística, como pueden ser fallas, reparaciones o simple mantenimiento, cuando una máquina importante falla, se puede sustituir o no, incluso desviar el flujo de producción, de manera que se suple su falla y se convierte en tiempo de mantenimiento, esto ocasiona que no exista un real desperdicio de tiempo. Cuando un estudio de tiempo no se puede realizar, las estimaciones provenientes de los ingenieros de la planta sobre y de los demás empleados pueden ser de mucha utilidad. Un persona de la planta puede estimar la frecuencia de las fallas, así como los límites superiores e inferiores del tiempo de reparación.

Raramente el analista posee la experiencias necesaria para hacer suposiciones adecuadas de los parámetros y variables de un sistema. De cualquier forma, el analista puede emplear su experiencia de fenómenos similares para formular funciones estadísticas aproximadas.

Las ideas clave.

Se ha mencionado la importancia de la comunicación, de mantener las suposiciones correctas y de los datos correctos. La conclusiones pueden ser formuladas de la siguiente manera:

- 1) Mantener los objetivos principales en mente.
- 2) Hacer una lista de los diseños alternativos del sistema que sean de interés.
- 3) Formular un modelo sencillo y funcionando tan pronto como sea posible para luego retocarlo. Este es un buen medio para mantener la atención del cliente e involucrarlo.
- 4) Construir un balance entre el nivel de detalle requerido para alcanzar los objetivos y el requerido para convencer al cliente de la validez del modelo.
- 5) Mantener la atención del cliente, involucrándolo con consultas y comunicación frecuente.

La Verificación es trabajo del Modelador.

El analista necesita de habilidades computacionales para trasladar las experiencias recogidas sobre las operaciones del sistema y los datos recogidos en un modelo de computadora que funcione. La verificación se refiere al procesos de probar y checar el programa, de manera que éste represente las suposiciones y datos recogidos. Muchas de las técnicas de verificación de un modelo de simulación para computadora son idénticas para cualquier tipo de software utilizado, estas técnicas incluyen:

- 1) Uso de programación por estructuras, que incluyen modularidad y diseño "top-down" (algunos de los lenguajes existentes tienen limitaciones intrínsecas en este renglón).
- 2) Pruebas extensas del programa, bajo una gran variedad de parámetros de entrada (esto es, análisis de sensibilidad).
- 3) Recolección y análisis de una gran cantidad de estadísticos, examinando su congruencia y razonabilidad.
- 4) Uso de diagramas de flujo (graficación).

5) Uso de animación.

Los lenguajes como GPSS o SLAM que automáticamente coleccionan e imprimen un gran número de datos de salida. Lenguajes como SIMAN y SIMSCRIPT (que proporcionan con gran facilidad datos de salida) son una gran ayuda en lo que se refiere a la verificación. Un análisis profundo de los estadísticos de salida puede, con mucha frecuencia, detectar el lugar donde se encuentran los errores en el modelo.

La graficación consiste en detallar la salida que representa el progreso paso a paso del modelo de simulación a lo largo del tiempo. La graficación puede tener una importancia mayor cuando se busca detectar la causa de errores sutiles o de verificar que el modelo maneje las excepciones correctamente. El ingeniero de la planta con frecuencia estará interesado en saber lo que sucede con el sistema en las ocasiones excepcionales o extremas, que sólo suceden en ocasiones muy raras, este es un método apropiado para analizar estos momentos, así como de proponer las soluciones adecuadas. Los programas modernos de simulación tienen la facilidad de la graficación, lo que ayuda enormemente al analista, evidentemente los utilizados en esta tesis tienen esta opción, XCELL +, y PASON, pero además GPSS/H, SIMAN, SIMSCRIPT.

La animación gráfica de una simulación es uno de los nuevos recursos de los lenguajes de simulación, la animación proporciona un equivalente gráfico de la graficación. Con la animación el analista puede verificar visualmente que el modelo se comporte con corrección en las circunstancias normales o en las excepcionales. Sin embargo la animación necesita de más trabajo por parte del analista, de manera que es un gran recurso para aplicarse cuando el modelo matemático parece ser el adecuado y se llega el momento de hacer la presentación al cliente.

Validación, un esfuerzo de equipo.

Conceptualmente se habla de la verificación como un trabajo exclusivo del analista, y en cambio de validación sólo se puede aplicar cuando la verificación ha terminado. En la práctica los dos actos se realizan simultáneamente. Cuando el analista está satisfecho de la verificación del modelo, entonces se llama a los clientes para realizar junto con ellos las pruebas de validación. Cuando el analista confía en su modelo significa que ha terminado la verificación, pero cuando el cliente confía en el modelo y lo aplica se dice que ha terminado la validación, esto es, el modelo es válido para su aplicación. Para este proceso se recomiendan los siguientes consejos:

- 1) Desarrolle un modelo de fácil validación.
- 2) Valide los supuestos del modelo.
- 3) Valide los resultados que proporcione el modelo.

Esto es, que sea razonable superficialmente para el usuario final (no muestres los cables, sólo la pantalla). Usando las técnicas discutidas anteriormente se puede dar al modelo la necesaria apariencia de seguridad. Por medio de diagramas y animación se puede explicar con cierta claridad el funcionamiento lógico del modelo, pero no se trata de entrar en detalles especializados sobre la programación de cada uno de los detalles del modelo (se trata de que los usuarios pierdan la impresión de estar usando una caja negra). Para este propósito también es de gran utilidad la comparación del modelo con el sistema real, por supuesto usando los mismos datos actuales en las variables y parámetros, sin embargo esto no se puede hacer cuando se trata de modelos de sistemas nuevos, además, típicamente la simulación es utilizada cuando se trata de rediseñar un sistema ya existente.

La idea principal de la validación es mostrar que para los valores cercanos de entrada al modelo y de la realidad, los resultados de ambos también serán cercanos, esto muestra que el modelo es sano y es una representación "válida" de la realidad.

Limitaciones, Peligros y últimos consejos.

En el mejor de los casos, cada validación de datos de entrada-salida sólo pide proveer un indicador de que el modelo está funcionando adecuadamente, para un grupo particular de parámetros y por un periodo histórico determinado. Típicamente es decisión de los ingenieros experimentados el uso del modelo para experimentar sobre los valores de configuración del sistema, de manera que el buen juicio y la experiencia nunca deben de ser despreciados, ninguna técnica de validación es útil si sólo se utiliza esa, hay que "oír" el parecer de las demás técnicas sobre este particular.

Además el proceso de validación nunca se termina. Si el modelo es utilizado por un periodo largo de tiempo es necesario que los datos se revisen, actualicen y prueben. Es un grave peligro que después de la primera validación el usuario utilice con los ojos cerrados todos los resultados que el modelo proporcione sin importar estos procesos de actualización, ya que con el tiempo, proporcionará datos poco fiables.

Cuando un modelo es utilizado en repetidas ocasiones para el mismo o nuevos procesos, se debe recordar que el sistema siempre está cambiando, y que los datos no son iguales para circunstancias distintas. Una validación periódica hace posible que el modelo se mantenga actualizado a través del tiempo y que siempre sea confiable.

El director de un proyecto de simulación nunca debe de olvidar que el proyecto siempre requiere de un esfuerzo en equipo, los modeladores modelando, los programadores programando y los estadísticos haciendo los análisis necesarios. Los ingenieros de planta y demás trabajadores proporcionarán la información del funcionamiento del sistema real y un gran interés por resolver los problemas que tienen. Haciendo que estos dos grupos trabajen juntos y proporcionando los recursos necesarios se maximizan las posibilidades de obtener un modelo preciso y válido.

1.4 Lenguajes de Simulación.

Hasta ahora se han citado algunos lenguajes clásicos de simulación, pero ha llegado el momento de hacer un breve listado de algunas de las herramientas más populares que se pueden encontrar en el mercado. Este listado no es de ninguna manera exhaustivo, ni siquiera enunciativo.

Hay dos áreas principales donde se desarrollan los lenguajes de simulación: el área de los sistemas continuos con parámetros concentrados o distribuidos, y el área de simulación discreta (simulación de eventos o actividades, estos últimos más propios para la simulación de sistemas de manufactura). Recientemente se pone más atención a los lenguajes orientados a objetos, los cuales permiten crear programas más flexibles, donde los objetos continuos y discretos corren en la forma paralela o semiparalela en el mismo modelo⁹.

Junto a estas dos áreas de los lenguajes de simulación, y además de los Lenguajes Orientados a Objetos, se añade una pequeña lista de los paquetes específicos, que hay en el mercado, para la simulación de sistemas de manufactura. Los paquetes que se utilizaron en el trabajo de esta Tesis, PAsION y XCELL+, son expuestos con un poco de más detalles en los Anexos I y II, al final de este trabajo.

Lenguajes para simulación Continua.

SIMULA. Éste es un lenguaje de simulación de alto nivel. Aplicado a simulación, SIMULA permite implementar el algoritmo IP (Interacción de procesos) y PE (programación de eventos). La versión más conocida y completa, llamada SIMULA 67, fue desarrollada por Ole-Johan Dahl, Bjorn Myhrhang y Kristen Nygaard en el centro computacional Noruego en 1967. El lenguaje es una extensión del ALGOL 60. En los años 70 se usaba mucho en Europa y fue implementado en casi

⁹RACZYNSKI, Stanislaw. *Simulación por Computadora*. Megabyte, Grupo Noriega Editores. 1993. 1a ed. México.

todas las computadoras de esa época. En América SIMULA no ganó mucho terreno por dos razones: primero, parece que fue un lenguaje demasiado sofisticado para cumplir con la demanda de herramientas de simulación durante los últimos 20 años. Segundo, SIMULA es una extensión de ALGOL, lenguaje que nunca fue popular en los EE.UU. Cuando aparecieron las computadoras personales, ninguna tenía un compilador de ALGOL y tampoco de SIMULA. Últimamente aparecieron algunas implementaciones de SIMULA para la IBM PC. Si se está interesado en profundizar más sobre esta poderosa herramienta, el lector puede confrontar las notas*.

GPSS. GPSS es un paquete para simulación de uso general (General Purpose Simulation System), desarrollado por Geoffrey Gordon en 1961. Éste es un paquete y no un lenguaje como SIMULA, porque lo que ofrece es un número de procedimientos típicos para la simulación de eventos. La función del usuario es la de establecer la estructura del modelo en forma gráfica y escribir una secuencia de instrucciones que invocan a los procedimientos correspondientes según la gráfica.

DYNAMO. Es un lenguaje para simulación de los sistemas continuos descritos con ecuaciones ordinarias diferenciales o de diferencias. Aunque es un lenguaje antiguo, vale la pena mencionarlo porque está basado en el método de simulación de Forrester que contiene algunas ideas valiosas de modelación matemática. El método fue desarrollado y publicado en 1961 por J.W. Forrester en su libro "Industrial Dynamics". La primera versión fue orientada a la simulación de las organizaciones industriales. Después fue adoptado para modelos de desarrollo urbano y sistemas económicos. Para más información se puede confrontar la bibliografía de la nota a, en las páginas 135-137.

ACLS. "Advanced Continuous Simulation Language". Un lenguaje para simulación de los sistemas descritos con ecuaciones diferenciales. Sirve para sistemas pequeños y grandes. Alcanzable en muchas computadoras.

CSMP. Un programa para simular sistemas continuos. Permite pasar de un esquema analógico que se usa para simular sistemas dinámicos en computadoras analógicas, a un programa de simulación en una computadora digital. Se especifican los bloques típicos como integradores, sumadores, multiplicadores y funciones, y sus entradas y salidas, así como las condiciones iniciales. El resto lo hace CSMP, calculando y graficando la trayectoria del modelo. Corre en la VAX 11/780.

MATRIX/WS. Ésta es una familia de programas para simulación de los sistemas dinámicos continuos, desarrollado por *Integrated Systems Inc.*, EE.UU. El sistema tiene enfoque en esquemas de bloques y muestra el esquema del sistema que se simula en la pantalla. El usuario puede usar bloques típicos, lineales o no lineales, así como definir nuevos bloques a través de las funciones de transferencia. El sistema corre en los terminales DEC, Apollo y SUN.

ISIM. Un lenguaje interactivo para simulación continua. Alcanzable para la IBM PC.

CSSL IV. "Continuous System Simulation Language". Un lenguaje de simulación continua. Además de resolver las ecuaciones del modelo, ofrece procedimientos para el análisis lineal, la transformada rápida de Fourier, ecuaciones rígidas, el análisis del estado estacionario, diagramas de Bode y Nyquist, lugar geométrico de las raíces y graficación. Alcanzable en muchas computadoras.

SYS/M. Para la simulación continua de sistemas grandes. Además de evaluar las trayectorias del modelo ofrece mecanismos para el análisis de estados estacionarios, linealización de modelos y gráficas. Alcanzable para VAX y la IBM PC, de *E² Consulting, Poway*, EE.UU.

ATOMICC. No es exactamente un lenguaje de simulación sino un generador de programas, desarrollado por Y.F. Chang, Claremont McKenna College, EE.UU. ATOMIC genera programas en Fortran para resolver sistemas de ecuaciones diferenciales ordinarias. El usuario crea un archivo que describe las ecuaciones y las condiciones iniciales. El generador ofrece mecanismos especiales para sistemas rígidos y un número de procedimientos sofisticados para resolver las

ecuaciones, basándose en series de Taylor. Se pueden manejar sistemas de ecuaciones en el plano complejo. Alcanzable para la IBM PC y computadoras más grandes.

STELLA^b. Es un lenguaje para simulación continua desarrollado para la computadora Macintosh, por High Performance Systems, Inc., Lyme, NH, EE.UU. Los principios de este lenguaje son casi los mismos que los de DYNAMO. Al definir gráficamente la estructura del modelo y determinar las condiciones iniciales, se calcula automáticamente la trayectoria del modelo. El lenguaje ofrece buenas facilidades para graficación y animación de los resultados.

TUTSIM^c. Un lenguaje para simulación continua en computadoras pequeñas. Es útil para sistemas con elementos no lineales. El modelo se define en términos de diagramas de bloques o gráficas llamadas "Bondgraphs". Las gráficas se traducen fácilmente en el programa correspondiente en TUTSIM.

MICRO-CAP II. Un programa para simulación y diseño de los circuitos electrónicos. Permite simular, graficar y analizar circuitos con transistores, amplificadores operacionales y circuitos lógicos. Desarrollado por "Spectrum", Sunnyvale, California, EE.UU.

SSDC2^d. Un programa para simulación continua de los sistemas dinámicos continuos. Relacionado a FORTRAN, aplica el método de Runge-Kutta de cuarto orden. Usa esquemas de bloques para representar la estructura del sistema. Desarrollado en el Instituto Politécnico Nacional, México.

COL^e. "Control Oriented Language" desarrollado por J.P. Siebert y D.J. Winning. Sirve para simulación de sistemas de control automático que contienen microprocesadores. El programa de simulación corre en tiempo real y consiste en un número de tareas "task", basadas en CSSI.

TDP^f. Es un paquete para simulación continua de los sistemas distribuidos descritos con ecuaciones diferenciales parciales. La solución de las ecuaciones del modelo se efectúa con el programa TWODEPEP. TWODEPEP utiliza un archivo de "entrada" preparado por el usuario, que describe el modelo. Este archivo se traduce a FORTRAN y las ecuaciones se resuelven con el método de elementos finito. TDP ofrece la posibilidad de simular sistemas distribuidos de control con retroalimentación. Está implementado en la computadora VAX 11/780.

SIMMON. Un lenguaje para simulación de los sistemas no lineales descritos con ecuaciones diferenciales o ecuaciones en diferencias. Desarrollado por Lun Institute of Technology, Suecia.

Lenguajes para simulación discreta y discreto continua.

PASSIM^g. Un paquete para simulación de eventos y sistemas continuos. Ofrece un número de procedimientos escritos en Pascal, a través de los cuales se lleva a cabo la simulación. El programa se escribe en PASCAL y consiste en una secuencia de llamadas a los procedimientos.

WITNESS. Un sistema interactivo que facilita la simulación de sistemas de manufactura, ofrece modelos listos para simular algunos elementos típicos como cintas transportadoras, puestos de producción, etc., relacionado a FORTRAN.

SIMSCRIPT^h. Un sistema de simulación orientado a eventos, desarrollado por Kiviat, Villanueva y Markowitz. Las unidades que representan a las componentes del modelo tienen sus atributos y pueden pertenecer a conjuntos. Hay dos tipos de unidades: permanentes, que permanecen en el modelo durante la simulación como puestos de servicio, almacenes, etc., y temporales, que son creados y destruidos, como por ejemplo, los clientes que entran y salen del sistema. Fácil de aprender para los usuarios de FORTRAN.

SIMONⁱ. Un lenguaje para la simulación de eventos, similar a CSL. Basado en ALGOL 60.

SIMAN^j. Un lenguaje para simulación de eventos, filas, sistemas de manufactura, desarrollado por C.D. Pegden.

GASP IV^A. Un lenguaje de simulación basado en FORTRAN, orientado a eventos y modelos discreto-continuos. Más parece un paquete que lenguaje, debido a que la simulación se realiza a través de la llamada a más de 20 procedimientos que manejan los eventos. Las versiones nuevas son GASPPLI relacionado a PL/I, GASP/E la versión interactiva y GASP V que ofrece más facilidades para los procesos continuos.

EMSS¹. "Expert Manufacturing Simulation System" es un sistema experto para simulación de los sistemas de manufactura, desarrollado por D.R. Ford y B.J. Schroer. Contiene algunos elementos de AI (Inteligencia Artificial), para interpretar los comandos del usuario, formulados en el lenguaje natural (ingles). El sistema contiene los siguientes elementos: el interface con el lenguaje natural, el generador del código de simulación, el analizador de simulación, y el lenguaje de simulación SIMAN. Después de interpretar y aceptar los comandos de entrada, el generador produce el código correspondiente en SIMAN. Luego el analizador proceso el código, checando su eficiencia y hace cambios necesarios para mejorar el programa y agregar procedimientos que analicen los problemas de prueba ("what if"). Corre en VAX 785.

SIMPAC. Un lenguaje de simulación de líneas de espera y planeación, desarrollado por M.R. Lackner y J. Kagdis, para la máquina IBM 7090.

GSP. "General Simulation Program". Desarrollado por K.D. Toecher. Sirve para simular plantas de manufactura. Se usaba en las máquinas Ferranti Pegasus y Elliot 503, principalmente en el Reino Unido.

SYSMOD. Un lenguaje de simulación para modelos continuos y discretos. Emplea la sintaxis similar a la de Pascal y ADA. Ofrece un gran número de mecanismos para simulación, como procedimientos de integración, ordenación automática de relaciones matemáticas del modelo, solución de ecuaciones diferenciales y algebraicas, graficación. Desarrollado por Systems Designer plc., Reino Unido. Corre en las computadoras VAX e IBM.

AXE^m. Un lenguaje para la simulación de los sistemas digitales de "Hardware". La compilación de AXE se lleva a cabo a través del lenguaje C. El lenguaje sirve para simular sistemas digitales con elementos sencillos (NAND, NOR, flip-flops, etc.), así como para simular sistemas compuestos de elementos con muy alto grado de integración (VLSI). Desarrollado por David E. Van den Boul.

VERISM. Un paquete para simulación de los procesos de toma de decisiones desarrollado por la AccuQuest Corporation, Texas, EE.UU. El usuario describe su modelo de una manera similar a la que se usa en los modelos de tipo "hoja extendida". Para cada una de las variables del modelo se determina el rango correspondiente de los valores posibles basándose en la experiencia e intuición del usuario. El paquete evalúa las estadísticas sobre posibles eventos cuya ocurrencia depende de dichas variables. los resultados permiten minimizar riesgos que corresponden a las decisiones que toma el usuario.

MODSIM II. Un lenguaje de simulación orientado a objetos. Es similar a SIMULA pero relacionado con el lenguaje de programación C.

SLAM II^o. Un lenguaje de simulación orientado a objetos a procesos y eventos. Por el proceso se entienda una secuencia de eventos, actividades y decisiones. Los componentes u objetos (entités) atraviesan el proceso, como lo hacen transacciones de GPSS atravesando el esquema gráfico. SLAM II usa esquemas que tienen la forma de redes. Los nodos representan creación de los objetos, terminación de actividades, líneas de espera, toma de decisiones, etc. Las ramas representan actividades, por ejemplo el servicio dura cierto tiempo. Cambios continuos del estado de modelo pueden simularse usando las ecuaciones diferenciales o en diferencias. Esto permite simular sistemas continuos o discreto-continuos. Los eventos de SLAM describen los posibles cambios en el estado del modelo. Las reglas matemático lógicas de la ocurrencia de los eventos las programa el usuario en FORTRAN. SLAM proporciona un conjunto de procedimientos típicos que facilitan la simulación, incluso el "mecanismo reloj" para controlar la ejecución de los eventos. La primera versión de SLAM fue desarrollada en 1979. La nueva versión, llamada SLAM II 2.3, elaborada por Pritsker & Associates, EE.UU., corre en la IBM PC y otras computadoras.

Lenguajes Orientados a Objetos.

No todos los aspectos de los Lenguajes Orientados a Objetos (LOO) son de igual importancia para la simulación. La principal aportación de este tipo de lenguajes consiste en la idea de crear objetos independientes en la memoria y construir clases de objetos con propiedades nuevas usando el mecanismo de herencia. Los lenguajes LOO no son, en principio, lenguajes de simulación, sin embargo hay que reconocer que existe una gran similitud entre los conceptos de LOO y el modelado de sistemas en la simulación. Es claro que siempre es más fácil utilizar la facilidades de los programas de simulación que programarlo todo desde el principio en un lenguaje orientado a objetos, sin embargo las facilidades de estructuración que ofrecen los LOO son de suficiente importancia para tomarse en cuenta.

SMALLTALK[®]. Es una implementación típica y completa de LOO (Lenguaje Orientado a Objetos), con fuerte separación de objetos y protección de datos. Facilita la programación a través de modelo interactivo con "ventanas" y menús. Es relativamente lento y necesita casi 1 Mb de memoria para trabajar normalmente.

ACTOR. Un lenguaje interactivo para la IBM PC. Tiene todos los mecanismos de LOO y ofrece un software adicional para editar y verificar programas. Desarrollado por Whitewater Group Inc., Evanston IL, EE.UU.

OBJECT PASCAL. Es una extensión de PASCAL. La palabra *Object* permite declarar objetos como se declaran los records en Pascal. El objeto además de una estructura de datos, contiene métodos que se declaran como procedimientos dentro de la estructura del objeto.

NEON. Está relacionado a FORTH. Usa la sintaxis de FORTH, es relativamente rápido, aunque no es una implementación completa de LOO.

ExperCommon LISP. Relacionado con el lenguaje de Inteligencia Artificial LISP, es una completa implementación de LOO.

OBJECTIVE-C. Relacionado con el lenguaje C. La extensión consiste en un nuevo tipo de expresión que es el mensaje, y en las declaraciones de clases.

Object Assembler. Relacionado a Assembler, corre en el procesador Motorola 68000. Es rápido pero más difícil de aprender que otros lenguajes de tipo LOO.

OBJECT-ORIENTED FORTH. La versión de LOO del lenguaje Forth.

Paquetes de Simulación para Sistemas de Manufactura.

Actualmente, debido al gran auge que está teniendo la práctica de la ciencia de la simulación en las empresas modernas, es fácil nombrar más de 50 paquetes de simulación que se pueden encontrar con mucha facilidad en el mercado. A modo de ejemplo comentaremos sólo algunos de ellos.

Achilles, The Factory Model. Modelo y proporciona un análisis detallado de las operaciones de manufactura. Facilita la planeación dinámica a corto plazo y los experimentos "what if". Proveedor: In-Motion Technology, CA, EE.UU.

ALSS II. Es un simulador de líneas de ensamble. Tiene interfaces con el sistema de animación CINEMA y Proof Animation. Proveedor: Productive Systems, CA, EE.UU.

AutoMod. Simulación y animación tridimensional de los modelos de procesos complejos de manufactura. Proveedor: AutoSimulations Inc., UT, EE.UU.

Automation Master. Un software integrado para simulación, monitoreo e integración de plantas automatizadas y almacenes. proveedor: HEI Corporation IL, EE.UU.

Autoshed. Es una herramienta para *scheduling*, facilita el uso de recursos de la planta y el manejo de inventarios. Proveedor: AutoSimulations, Inc., UT, EE.UU.

CINEMA. Un sistema de simulación y animación basado en el lenguaje SIMAN. Sirve para mostrar el comportamiento dinámico de los sistemas de manufactura con detalles de realismo. Proveedor: System Modeling Corporation, PA, EE.UU.

Extended+Manufacturing. Simulación de los sistemas de manufactura, manejo de materiales, transporte, distribución etc. Proveedor: Imagine That, Inc., EE.UU.

FACTOR. Un sistema de software integrado para simulación, planeación de producción y manejo de las plantas productivas. Proveedor: Pritsker Corporation, IN, EE.UU.

Flow Simulation Software. Simulación de los sistemas de manufactura, cálculo de la capacidad de la planta, búsqueda de los cuellos de botella, manejo de inventarios, planeación. Proveedor: LOG'IN, Cesson-Sevigne, France.

GENETIK. Simulación de los eventos discretos y procesos continuos. Tiene un *scheduler* para planeación de producción, produce diagramas de Gantt. Proveedor: Insight International, Ltd., Ontario, Canadá.

MAST. Simulación de las líneas de espera, producción just-in-time, sistemas flexibles de manufactura con gráficas animadas. Proveedor: CMS Research, IN., WI, EE.UU.

PC Model. Un sistema de simulación de eventos discretos orientados a los sistemas de manufactura. Proveedor: SimSoft, Inc., OR, EE.UU.

Plan-IT. Sistema de simulación de manufactura. Analiza el uso de maquinaria y personal, hace un análisis "what if", funciona con Windows. Proveedor Ivy Systems, Inc., VA, EE.UU.

ProModel PC. Combina la flexibilidad de un sistema de simulación con un amigable simulador de los sistemas de manufactura. Proveedor: Promotel Corp., UT, EE.UU.

Simfactory con Simgraphics. Un sistema de simulación con una interface gráfica fácil de usar. Planeación de tipo just-in-time, diseño de layout de la planta, sistemas con líneas de espera, cuellos de botella y utilización de recursos. Proveedor: CACI Corp., CA EE.UU.

SIMPLE y SIMPLE++. Lenguaje de simulación para sistemas de manufactura. Simple ++ es un sistema más avanzado, orientado a objetos, relacionado a C++. proveedores: IPA FhG, AESOP GmbH, Stuttgart, Alemania.

TAYLOR II. Simulador de los sistemas de manufactura y de transporte. Flexible, amigable, fácil de usar, equipado con gráficas y animación. Proveedor: F & G Logistics and Automation, Tilburg, Holanda.

Whitnes Simulation Software. Modelación y Simulación gráfica de los sistemas de manufactura. Proveedor: AI & T ISTEEL, OH, EE.UU.

La comparación de los productos y una decisión razonable sobre la compra e implantación no es tarea fácil. Los precios fluctúan entre los 30 dólares (SIMPLE) y 18,000 dólares (TAYLOR II), y el rendimiento de los paquetes es rara vez proporcional al precio.

Sólo nos queda recordar, para cerrar la sección correspondiente de los paquetes y lenguajes de la simulación, que los paquetes utilizados para esta tesis, PASON y XCELL+, están expuestos brevemente en el Anexo Y, y el Anexo II de este trabajo.

1.5 La Simulación y la Ingeniería Industrial.

Una vez dejado atrás todo lo que se refiere a la metodología de la simulación, una amplia discusión sobre la validación y verificación de los modelos, y una breve lista de los principales lenguajes y paquetes que están disponibles en el mercado para aplicar en simulación, llega el momento de discutir el motivo por el que la simulación es un herramienta utilísima para los Ingenieros Industriales, y además en qué forma se relaciona con esta carrera en particular.

En la siguiente página se presenta la figura 1.3 que muestra las materias de la carrera de Ingeniería Industrial (según el plan de estudio de 1980) y se resaltan todas aquellas que tienen que ver en forma directa con la simulación. Con esta figura se trata de indicar que, para que un analista sea un gran simulador, necesita de un amplio conocimiento de estas ciencias indicadas en la figura, entre más amplio sea este conocimiento mejores modelos obtendrá.

Es además oportuno indicar que la simulación siempre presenta el reto de dominar aquello que es simulado. No se puede simular lo que no se conoce, por tanto, todo lo que se intenta simular exige necesariamente de un alto grado de comprensión del sistema real. El analista de la simulación se convierte de esta manera en un "todólogo interdisciplinario", lo que implica una gran ambición de ciencia, un constante reto intelectual y la continua emoción de conocer siempre más de la realidad que nos rodea. Por ejemplo si el analista pretende hacer un modelo de un fenómeno climatológico es necesario que conozca estos fenómenos con profundidad para poder elaborar un modelo que sea válido, de la misma manera lo que se refiere a experimentos científicos, y lo mismo para los industriales.

1.6 Justificación Financiera de los Proyectos de Simulación¹⁰.

Los programas de simulación son fundamentalmente usados como una herramienta analítica para estudiar nuevos sistemas y reducir los costos de experimentación. Esta herramienta puede ser usada para la verificación de apropiaciones de capital, distribución de mano de obra, añadir equipo automatizado, pruebas de esquemas y distribuciones de producción, pruebas de nuevos diseños productivos y muchas otras áreas de las operaciones de la empresa. El poder en los proyectos de simulación radica en la posibilidad de dar a los directores una herramienta que se pueda utilizar como prueba de los sistemas alternativos que se investigan de manera que se pueda optimizar el dinero antes de que se tomen las decisiones de inversión.

La compra de paquetes de programas de simulación como los listados anteriormente, puede ser financieramente justificado porque el impacto que puede llegar a tener un análisis de este tipo puede ser enorme en los proyectos de inversión. Como método de análisis de justificación económica de los proyectos de inversión usualmente se utilizan tres métodos: Período de Tiempo de Retorno de la Inversión, Valor presente neto (VPN), y tasa interna de retorno (TIR). A continuación se presenta un modelo que puede ser utilizado en el análisis de invertir en un proyecto de simulación, aunque existen muchos beneficios que se desprenden de la simulación que no son contabilizables numéricamente).

Descripción del Modelo.

En primer lugar, el ingeniero debe de estar seguro de los costos estimados de producción, de manera que el análisis de la inversión que se realizará sea lo más real posible. Existen muchos componentes que intervienen en la evaluación económica de un proyecto de simulación, algunos de ellos son:

¹⁰ MATWJEC T., KALASKY D R., Financial Justification for Simulation Software, SIMULATION 1992, pp 114.

Ingeniería Mecánico Electricista.

Area Industrial.

| | | | | | | | |
|-----------|--------------------------------|-----------------------------|---------------------------------|--------------------------|---|--------------------------------|---|
| 1 | | Dibujo | Física Experimental | Álgebra y Geometría | Cálculo Diferencial e | Computadoras y Programación | Introducción a la Ingeniería |
| 2 | | Problemas Internacionales | Mecánica I | Álgebra Lineal | Cálculo Vectorial | Computadoras y Programación II | Introducción a la Ingeniería III |
| 3 | | Sociología de México | Mecánica II | Termodinámica | Electricidad y Magnetismo | Ecuaciones Diferenciales y | Computadoras y Programación |
| 4 | Administración, Contabilidad Y | Introducción a la Economía | Fundamentos de Mecánica de | Elementos de Mecánica de | Dinámica de sistemas Físicos | Métodos Numéricos | Balace de Materia y |
| 5 | Administración y Costos II | Sistemas de Computo | Introducción a la Tecnología de | Análisis de Circuitos | Probabilidad y Estadística | Investigación de Operaciones I | Principios Morales de la |
| 6 | Técnicas de Evaluación | Diseño de elementos de | Electrónica Básica | Elementos de Control | Estadística Aplicada | Ingeniería Industrial | Optativa Cultural Sistesis de Teologia |
| 7 | Empresa y Humanismo | Evaluación de Proyectos | Procesos de Manufactura | Máquinas térmicas e | Medición e Instrumentación | Técnicas de Optimización | Investigación de Operaciones II |
| 8 | Estudio del Trabajo | Diseño de Sistemas | Componamiento Humano en las | Máquinas Eléctricas | Investigación de Operaciones III | Planeación y Control de la | Finanzas I |
| 9 | Gestión de Proyectos | Ingeniería Industrial | Técnicas de administración | Instalaciones Electro- | Seminario de Tesis I Sem. Problemas Int. | Planeación y Copntrol de la | Finanzas II |
| 10 | Recursos y Necesidades de | Sistemas de Comercializació | Planeación Estratégica | Seminario de Ingeniería | Introducción al Derecho | Administración de Personal | Téc.de Comunicación Análisis de Sistemas |

A. Costos Típicos de Simulación.

Costo del software.

Costo de capacitación.

Tiempo de recolección de datos y formulación del plan del proyecto (semanas hombre de Ingeniería).

Costos de computadoras (no siempre se gasta en este concepto).

Tiempo para construir un modelo completo (semanas hombre de Ingeniería).

Costo de contratar un equipo de especialistas en simulación.

B. Costos Típicos que se pueden ahorrar o disminuir por medio de un proyecto de simulación.

Reducción en los costos de mano de obra.

Mejoramiento de la capacidad de producción en los sistemas.

Mejorar la fluidez en líneas de producción.

Disminución del tiempo de paro de maquinaria.

Eliminación de la compra de maquinaria innecesaria.

Mejoramiento de los esquemas de operación.

Eliminar retrabajos.

Reducción de inventarios.

Retorno sobre la inversión de un proyecto de simulación.

El método de período de retorno sobre la inversión es un análisis aproximado sobre la rapidez con que un proyecto recobra su inversión inicial, en este caso inversión en software y analistas. Período de retorno se calcula con el número de años que le toma a los ingresos estimados del proyecto igualar a la inversión inicial del proyecto. En base a la experiencia de muchos de los analistas de la simulación, que han conducido proyectos exitosos, se puede decir que, en la mayoría de los casos, todos los rubros mencionados en el modelo anterior en la parte A se recuperan con el primer o segundo proyecto de simulación.

Típicamente los costos de simular (A) se encuentran entre las decenas de miles de dólares. Por otro lado, la cantidad de dólares que uno se puede ahorrar con los resultados exitosos de un proyecto de simulación se encuentran entre los cientos de miles de dólares. En la mayoría de los proyectos de simulación el período en el que se recupera la inversión es de 2 años o menos. Hay que notar que estos datos varían según el tamaño del modelo y la complejidad del sistema que se analice. Si el sistema es muy grande y complejo, los costos de simularlo serán mayores, pero se supone que los ahorros producto de un buen proyecto también crecerán. En la mayoría de los proyectos "normales", el tiempo de recuperación de la inversión es de menos de un año. Un factor importante en este impacto es la frecuencia con que la empresa utilice la herramienta de simulación durante el año, y lógicamente la habilidad con la que la utiliza.

Sólo para recordar el concepto de Retorno sobre la inversión mencionamos su fórmula clásica:

$$\text{Tiempo de retorno sobre la inversión} = \frac{\$ \text{ costo neto de la inversión}}{\$ / \text{año de ingresos (ahorros) producto del proyecto}}$$

Valor presente neto de un proyecto de simulación.

La principal diferencia entre esta herramienta de análisis de proyecto de inversión y la anterior es que el VPN (valor presente neto) tiene en cuenta el cambio del valor del dinero a través del tiempo. Este tipo de herramienta incluye un grado mayor en la complejidad de su aplicación, ya que el analista tendrá que estimar los valores de los posibles ahorros, año por año (y de igual manera con los gastos), antes de tener un modelo. La idea principal de estimar el VPN de un proyecto consiste en pasar cada uno de los flujos de caja esperados, productos del proyecto, a una fecha concreta en el tiempo. La conclusión es que, si el valor de todos estos flujos es mayor que cero, el proyecto debe aceptarse, pues lo que se obtiene y gasta del proyecto en cuestión, en

cada año de vida, afectado por la tasa de interés variable, es positivo, esto es, se obtiene más de lo que se gasta.

$$VPN = -C_0 + C_1/(1+r) + C_2/(1+r)^2 + C_3/(1+r)^3 + \dots + C_n/(1+r)^n$$

Donde C_0 es la inversión inicial del proyecto. C_i son los flujos de caja del año i . La r representa el interés o costo de capital del proyecto, este puede variar a lo largo del tiempo y se llamará entonces r_t . La n representa el número de periodos que el proyecto tiene de vida.

Tasa Interna de Retorno para los Proyectos de Simulación.

TIR es la tasa en la cual el VPN tiene valor de 0. Esto es, si en la ecuación anterior $VPN=0$, entonces r es TIR. La TIR significa la tasa de descuento en la cual ni se gana ni se pierde por la inversión. Sirve como herramienta para la toma de decisiones de los proyectos de inversión, ya que, si la tasa del mercado (costo del dinero) es mayor que la TIR del proyecto no conviene invertir, ya que existe una inversión en el mercado, con mayor certidumbre y que nos da más rendimiento. Si la TIR es mayor que r significará, por tanto, que el proyecto es una mejor inversión que los otros proyectos que ofrece el mercado.

Para encontrar el valor de la TIR se necesita encontrar las raíces de la siguiente ecuación:

$$0 = -C_0 + C_1/(1+TIR) + C_2/(1+TIR)^2 + C_3/(1+TIR)^3 + \dots + C_n/(1+TIR)^n$$

El gran inconveniente de esta herramienta para la evaluación de un proyecto de inversión es que un mismo proyecto puede tener varios valores de TIR. Esto es, la ecuación anterior es, en realidad, un polinomio de n 'simo grado, el cual puede tener n raíces. Esto se complica si además, por el teorema principal del álgebra, recordamos que un polinomio con estas características puede representar cualquier función en el espacio, dependiendo de los valores correctos de los coeficientes del polinomio. Todo esto no viene a decir que un mismo proyecto puede tener varias TIR, y la pregunta siguiente es ¿Cuál de ellas elegir?, esta pregunta y otras escapan de las intenciones de la presente Tesis.

Conclusiones.

Los flujos de caja estimados para evaluar estas herramientas de evaluación financiera son producto de la realidad, pero pueden variar dramáticamente. Es decir, nunca dejan de ser estimaciones. El ingeniero que conoce bien los costos de sus operaciones debe de ser el principal colaborador para la aplicación de estas herramientas, sobre todo aportando datos de ahorros en operaciones de producción que tendrá como resultado el proyecto de simulación. Sin embargo, en la mayoría de las ocasiones, estas herramientas son relativamente fáciles de evaluar, y el camino aceptado para presentar un nuevo proyecto de inversión, será de la naturaleza que sea.

Es importante insistir en que la clave que muestra la conveniencia de usar la simulación radica no tanto en los gastos que causa, sino en los ahorros que produce, esto es la parte B del modelo descrito anteriormente. Estos ahorros son los que en realidad van a justificar los gastos producidos por el proyecto de simulación. De esto se concluye que el analista debe buscar, para hacer el proyecto financieramente más atractivo, que los ahorros producidos como consecuencia de la simulación sean mayores, y no tanto que el proyecto de simulación sea en sí mismo barato. Esto implica que la ciencia nunca debe ser malbaratada¹.

¹BRIGHAM, E.F., *Financial Management Theory and Practice*, The Dryden Press, ed. 6th.

-
- ^aRACZYNSKI, S. Simulación por Computadora., Grupo Noriega Editores (Megabyte), de 1993, pp. 99-106.
- ^bROBINSON S.B., 1986, Stella. Byte 11 (13)
- ^cTHOMA J.U., Block Bond Graphs for Simulation of Nonlinear Systems: A technical note, Transactions of the Society for Computer Simulation 2(3), September 1985, pp 249-255.
- ^dARCE MEDIN E., SSDC2: un simulador de sistemas dinámicos, Revista de computación 010, 7(4), Septiembre 1984, Fundación Arturo Rosenblueth, México.
- ^eSIEBERT J.P., WINNING D.J., The Control Oriented Language, SIMULATION, 48(1), 1987.
- ^fAALTONEN M., TANTTU J.T., POHJOLAINEN S.A., Idp - A DISTRIBUTED PARAMETER SIMULATOR, Third IFAC/IFIP International Symposium on computer Aided Design in Control and Engineering Systems, The Technical University of Denmark, 1985, Lyngby, Copenhagen, Denmark 1985.
- ^gUYENO, D.H., VAESSEN W., Passim: a discrete-event simulation package for Pascal, SIMULATION 35(6), December 1980, pp. 183-190.
- ^hKIVIAT P.J., VILLANUEVA R., MARKOVITZ H., The Simgscript II Programming language, Prentice-Hall, 1969.
- ⁱHILLS P.R., Simon - a computer Simulation Language in Algol 60, Digital Simulation in Operational Research, de., English University Press, London, England, 1965.
- ^jLEDGEN C.D., Introduction to SIMAN, System Modeling Corporation, State College, Pennsylvania, 1985.
- ^kPRITSKER A.A.B., The GASP IV Simulation language, John Wiley & Sons, 1974.
- ^lFORD D.R., SCHROER B.J., An expert manufacturing simulation system, SIMULATION 48(5), pp 193-200.
- ^mVAN den BOUT, D.E. AXE: a language for the simulation of digital hardware, SIMULATION 35(6), December 1980, pp 183-190.
- ⁿPRITSKER A.A.B., Introduction to Simulation and SLAM II, John Wiley & Sons.
- ^oKOEHLER T. Patterson D., A Small Taste of Samitalk, BYTE 11(8), August 1986, pp. 145-159.

Capítulo 2: Simulación de Células de Producción.

Introducción.

En este capítulo se presenta el caso de una empresa manufacturera que tiene un diseño de planta tipo taller y se presenta el problema de convertir este diseño en otro orientado según la Tecnología de Grupos. Para esto se recurre al concepto de células de producción. Se forman tres y se propone un modelo de su funcionamiento. La simulación se aplica con éxito para evaluar el funcionamiento de estas células, de manera que se examine su viabilidad.

En la primera parte del capítulo se hace una introducción a los conceptos teóricos de Tecnologías de Grupo, Células de Producción. En la segunda parte se expone el caso a analizar. En la tercera parte del capítulo se presenta la solución del caso, el análisis del caso previo a la simulación, el modelo en Xcell+ y en PASION. Por último se presentan las conclusiones de la simulación en el caso concreto.

2.1 Tecnologías de Grupo y Células de Producción¹.

Introducción.

La Tecnología de grupos o células de producción es un sistema de diseño de planta que ubica máquinas distintas en células, para trabajar en productos que tienen formas y requerimientos de producción similares. La Tecnología de Grupos (GT) actualmente se usa mucho en la fabricación de metales, manufactura de componentes electrónicos, y trabajos de ensamble. El objetivo es añadir a los beneficios de la producción tipo taller, los beneficios de producir en líneas continuas. Estos beneficios incluyen:

1. Mejores relaciones humanas. Las células necesitan de unos cuantos obreros que forman un equipo de trabajo, el equipo procesa completamente un producto.

¹R. CHASE & N. AQUILANO, *Production and Operations Management: A Life Cycle Approach*, Irwin Co., Boston 1992.

2. Se mejora la experiencia del obrero. Los obreros sólo ven un número limitado de partes diferentes en un ciclo de producción finito, de manera que la repetición garantiza el aprendizaje rápido.
3. Menores inventarios de productos en proceso y traslado de material. Una célula enlaza algunas fases de producción, de manera que menos partes circulen por el piso.
4. Los tiempos de preparación de máquinas son más rápidos. Entre más especializada esté una célula en un producto determinado, los tiempos de preparación disminuirán, pues hay menos cambios de producto.

Diseño de planta según la Tecnología de Grupos.

Cambiar de un diseño de planta tipo proceso (taller) a uno tipo GT significa tres pasos:

1. Agrupar las partes en familias que tengan secuencias similares de procesos de producción. Estos pasos de procesos se sistematizan en códigos de clasificación que significan los pasos a los que pertenecen. Esto implica una continua actualización en archivos computarizados. Este paso es el más costoso en la mayoría de los sistemas, aunque muchas de las compañías tienen registros de secuencias de procedimientos similares, lo que facilita la identificación de familias.
2. Identificar los flujos dominantes de familias de partes para que sirva como base para relocalizar los procesos.
3. Agrupar físicamente las máquinas y los procesos en células. Algunas veces existen partes que no pueden ser asociadas con familias y las máquinas especializadas no se pueden localizar en ninguna célula porque son de uso común. Estas partes y máquinas especiales son colocadas en una célula especializada.

De acuerdo con Richard Schonberger² se puede realizar un diseño de planta tipo células de producción en una empresa que tenga las siguientes características:

- a. Que existan familias de partes.
- b. Que existan varias máquinas del mismo tipo, de tal manera que tomar una máquina de un centro no robe toda su capacidad al centro.
- c. Que sean fáciles de mover y separar los centros de trabajo.

El problema de la tecnología de grupo (GT) tiene en tres fases:

1. Mejorar la clasificación y codificación de esquemas de los productos (forma, tamaño, material, etc.)
2. De los requerimientos de los procesos y rutas, agrupar la partes en familias para formar células de producción.
3. Crear el diseño de planta físico, posicionando células relativas unas con otras.

La primera fase es más que un problema ingenieril de describir piezas y partes. La tercera fase es esencialmente solucionable usando evaluaciones de diseños de planta como CRAFT³ para minimizar el flujo de material de una célula a otra. La segunda fase es lo que generalmente se conoce como GT, es la base de cómo formar las células.

Para formar las células⁴, la primera cosa que hay que hacer es crear un flujo matricial mostrando cuáles componentes fluyen a cada máquina. Esta matriz se forma escribiendo en la primera columna los materiales y en la primera fila los centros de trabajo. Si se logra ordenar la matriz de tal manera que se forman grupos, el proceso termina. Se recomienda ampliamente revisar la bibliografía sobre el tema de manera que se pueda evaluar el método binario de reconocimiento de grupos, este método y los demás, por lo pronto trascienden los objetivos de la tesis.

²Richard J. SCHONBERG. *World Class Manufacturing. The Lessons of Simplicity Applied*. New York: Free Press, 1986, p. 112.

³R.L. FRANCIS & J.A. WHITE. *Facility Layout and Location: An Analytical Approach*. Englewood Cliffs, N.J.: Prentice Hall, 1974.

⁴J.L. KING. *Machine Component Group Formation for Group Technology*. Proceedings of the Fifth International Conference on Production Research, p. 193

Otra manera de crear células de trabajo, a parte del método binario es el propuesto por Askin & Subramanian⁵ usando los siguientes costos:

1. Costo de preparación.
2. Costos variables de producción.
3. Costos de inventarios del ciclo de producción.
4. Costos del inventario de productos en proceso.
5. Costos de acarrear material.
6. Costos fijos de máquinas.

Estos autores solamente asumen las siguientes condiciones:

- a. Cada parte tiene una secuencia predeterminada deseada con preparaciones y transformaciones asociadas.
- b. Planear sobre horizontes infinitos y demandas constantes.
- c. No se permite que haya carestía de materiales.
- d. Un lote entero de partes debe estar en una estación de trabajo, si está por comenzar su proceso, no como espera o cola.
- d. Cada parte corresponde a un código inicial de familia, que tienen similares procedimientos de preparación.

Los grupos son creados considerando combinaciones de partes y máquinas con costos resultantes. Este enfoque permite concentrarse en los costos, minimizar los costos mediante una correcta selección de familias y células de producción. El punto de arranque es arreglar los grupos en orden a la similitud de máquinas. Inicialmente cada parte corresponde a un grupo, y el costo total se calcula en base a la lista de costos de procedimientos. Entonces, la evaluación se hace probando las combinaciones posibles de los grupos adyacentes, de manera que se seleccione la secuencia más económica.

Células de Producción (GT) Conceptuales.

Cuando el equipo instalado no se puede mover con facilidad, muchas compañías dedican una máquina, o un grupo de máquinas, a un producto determinado. Es tanto como asignar máquinas concretas de cada taller, únicamente en producir una determinada pieza, de manera que, aunque las células no se formen físicamente, hay máquinas especializadas en un producto en cada taller.

2.2 Caso: Células de Producción y Balanceo de Líneas.

John Wells revisaba el piso de producción. Es un ingeniero con un MBA. Estaba trabajando para una empresa manufacturera de partes automotrices. Como jefe de Producción y administración de Materiales, él era responsable del trabajo que estaba observando.

Su objetivo era modificar el sistema de producción, para reducir inventarios, particularmente los inventarios de productos en procesos (WIP), y mejorar así la calidad del producto. Acorde con estas metas habría más capital libre que proveerá de beneficios a largo plazo para la compañía.

Manufactura de Partes Automotrices.

La compañía de John produce artefactos de medición de tiempo y cadenas de transmisión. Ellos sirven a dos mercados primarios: fabricantes de equipos originales ("original equipment manufacturers OEM's") como General Motors, Ford y Chrysler, y mercados de segundo nivel que proporcionan las piezas de refacciones. Cada cadena de transmisión tiene el largo, ancho y variaciones de diámetro, basadas en un específico año y tipo de carro para las que fueron diseñadas.

⁵R. ASKIN & S.P. SUBRAMANIAN, *A Cost-Based Heuristic for Group Technology Configuration*, International Journal of Production Research 25, no. 1 (1987), pp. 101-103.

Abastecer a las grandes empresas de automóviles es un negocio cambiante y vertiginoso. Las órdenes se reciben semanalmente desde varias plantas OEM por todo el país. Estas órdenes incluyen: estimaciones de tres meses de las cantidades y los tamaños necesitados (subdivididas en proyecciones mensuales y semanales), las órdenes liberadas para materiales brutos, las cantidades específicas de cadenas, y los datos requeridos de empaque.

Desafortunadamente, los estimados de las OEM son altamente inexactos. Con frecuencia las estimaciones de demanda varían más de 100%. Esto crea serios problemas de producción, porque órdenes inacabadas o atrasadas pueden llevar a problemas con las plantas OEM: penalizaciones de contratos, y pérdida potencial de los consumidores. Además, John con frecuencia tiene que pagar primas extras por transportar productos por aire, para cumplir los requerimientos de fechas de entrega. Definitivamente, él está interesado en que la compañía deje de tener problemas de producción por muchos años.

Las órdenes para los mercados secundarios o de refacciones son muy distintas. Las órdenes son recibidas con menos frecuencia y siempre proveen de cantidades específicas y fechas de entrega exactas. De cualquier forma, este mercado sólo representa el 25% del negocio de la empresa.

El incremento de la competencia foránea ha demandado de esfuerzos renovados por la firma para reducir costos, mejorar la calidad del producto, y mantener la competitividad en el mercado de partes automotrices. Una rígida medición en los costos está siendo implementada para rastrearlos a lo largo de la planta y sacar a la luz las áreas de problemas. La calidad ha llegado como un concepto que la compañía encarnó como el costosísimo proceso de inspeccionar cada una de las cadenas antes de empaquetarlas. A pesar de que el 100% de las cadenas se revisaba, algunas cadenas defectuosas llegan, de todas formas, a los consumidores. John ya ha examinado muchos planes para mejorar los sistemas de producción con la intención de mejorar la calidad y reducir los costos, pero todos coinciden en ser extremadamente caros para una justificación económica viable.

Una Planta Fabricante de Cadenas.

La capacidad instalada de la planta produce aproximadamente 27,000 cadenas de tiempo y alrededor de 100,000 cadenas de transmisión por día. La planta está dividida en dos áreas principales: la Fábrica de partes y el Ensamblado. La Fábrica de partes convierte la materia bruta en tornillos y enlaces, mientras que el área de Ensamble combina esas partes en cadenas. Cada una de las áreas está organizada con una distribución (diseño de planta) funcional, de manera que todas las máquinas con funciones similares están agrupadas juntas en un área referida a esa estación. Cada estación provee uno de los pasos en el proceso de manufactura de una cadena, y muchas de las máquinas en la estación son intercambiables. Todos los productos siguen los mismos pasos.

Las estaciones de fabricación de partes emplean un sistema de proceso por tandas (lotes) por el cual se transportan grandes cantidades del producto por los procesos de una estación a la vez. El tamaño típico de los lotes es el suficiente para satisfacer la demanda de esa parte por días. Las estaciones usualmente trabajan con muchas partes distintas simultáneamente. Las partes son almacenadas en cajas apiladas en el piso de las áreas de trabajo. Ahí hay, ordinariamente, algunas semanas de trabajo en inventarios WIP en cada estación de manufactura.

Los tornillos o pemos, las partes internas de las cadenas, se producen a partir de rollos de alambre de acero que es puesto en una máquina cortadora, del tipo Lewis, la cual desenrolla el alambre hasta la medida exacta y corta el pemo por medio de un proceso de guillotina. A continuación sigue un tratamiento térmico, calor, en grandes hornos para regresar sus propiedades térmicas al acero. Algunos de los pemos son entonces rebajados a tamaños específicos. Los inspectores examinan los pemos para confirmar que tengan la longitud y dureza necesaria, antes de colocarlos en el inventario de WIP.

Los enlaces, es decir, las placas externas que fijan a los pemos, siguen un proceso de producción similar. Los rollos de tiras de acero que varían en anchos y tipos, son montados en largas máquinas que troquean los enlaces. Este proceso es altamente ineficiente, alrededor de un 50 % de material se desperdicia. Los enlaces son entonces tratados térmicamente en los hornos y luego tendidos cuidadosamente. Algunos de los enlaces son corridos a través de una pluma disparadora que golpea,

con bolas metálicas, los enlaces para proporcionarles rigidez, mientras que otras son recubiertas para darles mayor suavidad. Los enlaces son entonces puestos en inventarios de WIP.

Las áreas de ensamble también procesan por medio de lotes y un diseño de planta funcional. De la misma manera que en la fabricación, muchos días de trabajo de un producto determinado son almacenados entre las estaciones. Hay 4 estaciones, cada una dedicada a un paso de ensamble. El primer paso es ensamblar, donde la máquina conecta los enlaces con los pernos, haciendo grandes secciones de cadena. El siguiente es una máquina que ribetea los extremos de los pernos para asegurar los enlaces. La tercera estación es una máquina que divide las cadenas en las longitudes individuales requeridas, conecta los dos extremos de la cadena con un perno, ribetea este perno de conexión, y preestira la cadena. La preestirada tensa la cadena ligeramente. El paso final es la inspección, donde una máquina extremadamente exacta mide el largo de la cadena y realiza un prueba de sonido en los pernos. Si la cadena pasa la inspección, es transportada al área de empaque, donde se embarca y se manda al consumidor o cliente.

Las Células de Producción en el Área de Ensamble.

Hace algunos meses, John distribuyó un memo a sus supervisores pidiéndoles sugerencias sobre cómo disminuir el costo de los inventarios y mejorar la calidad. Una de las respuestas llegó de Mark Merritt, el jefe encargado del control de la producción, quien recomendó que algunas de las máquinas de ensamble fueran realineadas en "Células de Producción". El siguiente es un resumen de la sugerencia de Mark:

Las células de producción son configuraciones continuas de máquinas, las cuales desempeñan actividades según las políticas de JIT. Por ejemplo, la estación de ribeteado podría tener salida directamente de la estación de ensamble, y ésta directamente a la siguiente estación. Este sistema reduce significativamente los inventarios WIP entre las estaciones, porque se deja muy poco espacio entre las máquinas.

Una célula de producción se especializa en un pequeño número de productos; es, en esencia, una mini-fábrica. Las células de producción pueden también mejorar los resultados de la fábrica en la detección de problemas de calidad, ya que la inspección ocurre en un periodo de tiempo corto, desde que se inicia el ensamble.

Después de estudiar esta sugerencia, John llamó a Mark a su oficina. *"Mark, parece que hay una gran idea aquí. ¿Por qué no la realizas y trabajas en un plan para implementar esta política en otros productos?"*

Mark mandó un memo al primer jefe supervisor, Jim Hand, pidiéndole que recabara la información de costos de los cuatro tipos de cadenas. (Ver anexo 1). Después de examinar la información que Jim recolectó (anexo 2), Mark comenzó a diseñar una célula de producción de prueba, en el área de ensamble. Cuatro productos fueron seleccionados para la prueba.

Para formar las células de producción para los cuatro tipos de cadenas sin romper la producción de los otros productos, Mark sintió que podría dedicar nueve máquinas de ensamble, tres ribeteadoras, cinco máquinas preestiradoras y dos máquinas de inspección. La 3/8 x 7/8 x 64 pulgadas de la serie 2300 de cadenas de transmisión y la 3/8 x 7/8 x 62 pulgadas de la serie 2300 fueron los artículos que Mark eligió para ser producidos en células de trabajo separadas. Los otros dos productos, los 3/8 x 1/4 x 58 de la serie 3800 y 3/8 x 1/4 x 54 pulgadas de la serie 3800 serían ensamblados en la tercera célula. Después de hacer algunos cálculos sencillos, Mark decidió que cada uno de las células de trabajo deberían tener 7 máquinas: tres de ensamblar, una de ribetear, dos de estiramiento y una de inspección. Desafortunadamente, esto lo dejaba sin la suficiente capacidad de máquinas para dos células de trabajo: Tres ensambladoras, una ribeteadora y ninguna inspeccionadora (ver anexo 3). Era posible conseguir una máquina de inspección de una planta hermana. Para producir el máximo cambio de requerimientos para estas dos cadenas probablemente se necesitaría más que sólo un centro de inspección. El estaba inseguro sobre el lugar donde encontraría las otras máquinas o sobre cómo lo pagaría. A pesar de esto, sintió que rentando o comprando las máquinas adicionales era mejor que cambiar la máquina de inspección de la célula uno a la dos.

El control de la producción iba haciéndose más difícil que lo que él se había imaginado. Porque un pequeño porcentaje de productos fallaban la inspección y eran desechados, el beneficio de las células de ensamble era incierto. Por tanto, para complementar una producción JIT, algún método era necesario para determinar cuándo cambiar de un producto a otro. Los lotes fijos de tamaño no lo harán. La flexibilidad es necesaria para satisfacer este tipo de demanda.

Mark finalmente decidió que el control de la producción de las células, debería de ser acompañado de la preparación de un determinado nivel de productos buenos en inventario, de cada producto, y continuar la producción sin interrupción (excepto por cambios de turnos) hasta que estos dos niveles fueran alcanzados. Una vez que los requerimientos de una célula de producción fuera suficiente por el día, el mantenimiento preventivo podría comenzar. Como punto de partida, Mark decidió que cada producto debería ser preparado en la máquina, una vez por día, de manera que los lotes de producción deberían ser iguales a la demanda de un día.

El también reconoció dos problemas potenciales más en la construcción de estas tres células de producción: 1) Algunas de las estaciones tienen sólo una máquina; por tanto, las descomposturas de estas máquinas pueden anular todo el trabajo de la célula; y 2) la unión en células de producción puede afectar el actual sistema de pago a los trabajadores, en el que se les paga por cada pieza bien terminada, que funcionó muy bien con el sistema de lotes. Pero este sistema no es aplicable según el nuevo sistema, donde el trabajo de las estaciones precedentes afecta directamente en nivel de producción de las siguientes, pues están conectadas unas con otras. Por más que las estaciones de adelante se esfuerzen no podrán producir más que lo que las máquinas y operadores precedentes hacen.

Mark decidió hacer un alto en el día y analizar el problema por la mañana, cuando John le sugirió que utilizara la simulación para resolver el problema.

Tópicos de la discusión.

1. Además del problema de balancear las células de producción, ¿qué otras áreas mayores de los procesos de manufactura de la compañía deberían de ser analizadas?

2. ¿Cuántas máquinas de cada tipo se necesitan para el tercer centro de trabajo?. Explique el método para calcular esto, teniendo en cuenta el mantenimiento, los cambios de personal, y la naturaleza de JIT de las células de producción.

3. Diseñe un modelo usando la simulación para probar sus recomendaciones.

4. ¿Qué otros problemas puedes prever en tu sugerencia?.

Anexo 1.

"...Durante las siguientes dos semanas estaremos analizando nuestro sistema de producción de cadenas sincronizadoras y de transmisión. Para completar esta revisión, necesito que me digas nuestros costos de ensamblar, estos deben incluir:

- promedio de la tasa de producción de cada máquina por cambio de parte.
- el tiempo de preparación, por máquina para cada tipo de cadena.
- el número de empleados por máquina.
- el promedio de los costos de producción por hora.
- capacidad de las charolas de transporte (i.e. el número de cadenas por charola que pueden ser transportadas entre las estaciones de ensamble).
- número de trabajadoras en el equipo de mantenimiento.
- los requerimientos de servicio de las máquinas y tasas o porcentajes de falla.
- tasas o porcentajes de rechazo de producto.
- porcentaje de cambio de producción requeridos por producto.

Anexo 2.**A. INFORMACIÓN DE PRODUCCIÓN POR TIPO DE CADENA DE TRANSMISIÓN.**

1) 3/8 POR 7/8 POR 64 PULGADAS SERIE 2300, PRODUCCIÓN MÁXIMA REQUERIDA POR CAMBIO ES 2800 UNIDADES.

| ESTACIÓN | No. EMPLEADOS. | PROMEDIO DE PROD. | TPO. DE PREPARACIÓN |
|------------|----------------|-------------------|---------------------|
| | | PRO MAQ. POR MIN. | POR MAQ.(min) |
| ensamble | 1 | 2.708 | 5 |
| ribeteador | 2 | 10.833 | 10 |
| estirador | 2 | 3.75 | 20 |
| inspección | 2 | 12.5 | 25 |

2) 3/8 POR 7/8 POR 62 PULGADAS SERIE 2300, PRODUCCIÓN MÁXIMA REQUERIDA POR CAMBIO ES 2700 UNIDADES.

| ESTACIÓN | No. EMPLEADOS. | PROMEDIO DE PROD. | TPO. DE PREPARACIÓN |
|------------|----------------|-------------------|---------------------|
| | | PRO MAQ. POR MIN. | POR MAQ.(min) |
| ensamble | 1 | 2.5 | 5 |
| ribeteador | 2 | 10.833 | 10 |
| estirador | 2 | 3.75 | 20 |
| inspección | 2 | 12.5 | 25 |

3) 3/8 POR 5/8 POR 58 PULGADAS SERIE 3800, PRODUCCIÓN MÁXIMA REQUERIDA POR CAMBIO ES 1200 UNIDADES.

| ESTACIÓN | No. EMPLEADOS. | PROMEDIO DE PROD. | TPO. DE PREPARACIÓN |
|------------|----------------|-------------------|---------------------|
| | | PRO MAQ. POR MIN. | POR MAQ.(min) |
| ensamble | 1 | 2.083 | 5 |
| ribeteador | 1 | 10.833 | 10 |
| estirador | 2 | 2.708 | 20 |
| inspección | 1 | 12.5 | 25 |

3) 3/8 POR 3/4 POR 54 PULGADAS SERIE 3800, PRODUCCIÓN MÁXIMA REQUERIDA POR CAMBIO ES 1400 UNIDADES.

| ESTACIÓN | No. EMPLEADOS. | PROMEDIO DE PROD. | TPO. DE PREPARACIÓN |
|------------|----------------|-------------------|---------------------|
| | | PRO MAQ. POR MIN. | POR MAQ.(min) |
| ensamble | 1 | 2.5 | 5 |
| ribeteador | 1 | 10.833 | 10 |
| estirador | 2 | 2.708 | 20 |
| inspección | 1 | 12.5 | 25 |

* El tiempo de preparación es el tiempo requerido para cambiar de un producto a otro. No se requiere tiempo de preparación para continuar produciendo un mismo producto.

B. ESQUEMA DE MANTENIMIENTO.

El esquema de Mantenimiento ocurre una vez por cambio por cada máquina. El tiempo requerido para cada máquina es:

| | |
|--------------------------|--------|
| Maquinas Ensambladoras | 45 min |
| Máquinas ribeteadoras | 15 min |
| Máquinas de estiramiento | 15 min |
| Máquina probadoras | 30 min |

C. EL EQUIPO DE MANTENIMIENTO.

Usualmente el equipo de mantenimiento está formado por una cuadrilla de mecánicos. De cualquier forma para las células de producción, los operadores de máquinas son entrenados para dar diariamente mantenimiento al final de cada cambio.

D. TIEMPO QUE TARDA LA MÁQUINA CUANDO FALLA.

Desafortunadamente no hay registros de el tiempo que tarda una máquina en componerse. Las máquinas por lo regular siempre están funcionando. Parece ser que con sólo el mantenimiento preventivo se eliminan las fallas, de cualquier forma se hacen, en ocasiones, paros por reparaciones.

E. COSTOS DE PRODUCCIÓN.

Los pagos varían grandemente dependiendo de los años de experiencia y cambios diferenciales. Sin embargo, el promedio de paga está entre los \$11.00 y \$13.00 por hora.

F. TASAS DE RECHAZO DE PRODUCTOS.

De la experiencia pasada, estimamos que aproximadamente el 95% de la producción pasa la inspección, el resto se desperdicia.

G. CHAROLAS DE TRANSPORTE.

Las charolas que actualmente usamos guardan 50 cadenas cada una. De cualquier forma, en una células de producción, se sugeriría que sólo la primer máquina trabaje con charolas. Entre las máquinas, el producto se moverá uno a la vez. Desafortunadamente, los sistemas de acarreo de material automáticos pueden ser muy caros. La alternativa es continuar utilizando las charolas.

H. ESQUEMAS DE EMBALAJE.

Inicialmente el producto necesita ser empacado al final de cada cambio. El monto necesario está definido en el Inciso A. Para los nuevos tiempos, nos toparemos con más lotes por empacar al día.

Anexo 3: Propuesta para la distribución del área de ensamblado.

CÉLULA 1: 3/8 POR 7/8 POR 64 SERIE 2300.

| |
|----------|
| ENSAMBLE |
| 3 |
| MÁQUINAS |

| |
|-----------|
| RIBETEADO |
| 1 |
| MÁQUINAS |

| |
|--------------------|
| ESTIRADO DE PRUEBA |
| 2 |
| MÁQUINAS |

| |
|------------|
| INSPECCIÓN |
| 1 |
| MÁQUINAS |

CÉLULA 2: 3/8 POR 7/8 POR 62 SERIE 2300.

| |
|----------|
| ENSAMBLE |
| 3 |
| MÁQUINAS |

| |
|-----------|
| RIBETEADO |
| 1 |
| MÁQUINAS |

| |
|--------------------|
| ESTIRADO DE PRUEBA |
| 2 |
| MÁQUINAS |

| |
|------------|
| INSPECCIÓN |
| 1 |
| MÁQUINAS |

CÉLULA 3: 3/8 POR 5/8 POR 58 Y 3/8 POR 3/4 POR 54 SERIES 3800.

| |
|----------|
| ENSAMBLE |
| ? |
| MÁQUINAS |

| |
|-----------|
| RIBETEADO |
| ? |
| MÁQUINAS |

| |
|--------------------|
| ESTIRADO DE PRUEBA |
| ? |
| MÁQUINAS |

| |
|------------|
| INSPECCIÓN |
| ? |
| MÁQUINAS |

2.3 Solución del caso: Células de Producción y Balanceo de Líneas.

2.3.1 Respuesta a los Tópicos Presentados.

a. Otras áreas a examinar.

Mientras se estudió la simulación correspondiente a este caso, me pude dar cuenta de factores que se podrían investigar un poco más, como por ejemplo, las órdenes que liberan nuestros clientes, las empresas de manufactura final, el alto porcentaje que resulta desperdiciado por los procesos (5%), los grandes tiempos de preparación.

Para resolver los problemas derivados de las órdenes de producción de nuestros clientes, que como se mencionó, a veces varían de la realidad hasta un 100%, es necesario formular un mejor método de pronóstico. En el caso no se menciona ninguna razón de peso por la cual sea comprensible tan amplias fluctuaciones de los pronósticos, pero es de esperarse que dados los nuevos sistemas de administración de la producción se pueda conseguir una mejor predicción de la producción. Cuando la predicción presenta un rango de fluctuación tan grande, deja de ser predicción, y por tanto, deja de ser útil.

La gran cantidad de material que se desperdicia es un problema importante. Hoy en día no se puede tener el lujo de máquinas ineficientes. Es muy alto el porcentaje de desperdicio, se podría pensar en sistemas de reproceso o máquinas que sean mejores, con una tasa menor de desperdicio.

Cuando se estudió el tema referente a los tiempos de preparación, llegué a la conclusión de que es demasiado tiempo invertido. Es posible que las máquinas que se están utilizando en la planta sean demasiado obsoletas. Aún sincronizando los tiempos de preparación casi se pierde media hora en ajustar. Esto hace importante que sólo se ajusten las máquinas el número de veces necesario durante el día.

b. Cálculo de Número de Máquinas por Célula de Producción.

Los cálculos que a continuación se muestran tienen como finalidad mostrar la manera en que se determinó el número de máquinas indispensable para cada célula de producción. Se parte de que tenemos 480 hrs. por día, de manera que esas 480 hrs. menos los tiempos de producción, menos los tiempos de mantenimiento, nos da por resultado el número de horas que una máquina puede estar funcionando. En base a la demanda diaria y al tiempo por pieza, determinamos el número de horas que necesitamos para satisfacer la demanda; claro que hay que tomar en cuenta el porcentaje de horas que se desperdicia por procesar piezas que se rechazan. Estas horas necesarias de trabajo, entre el total de horas disponible al día nos da el número de máquinas que necesitamos de cada tipo, en la célula de producción en cuestión. Con estas máquinas se espera satisfacer la demanda diaria de una pieza determinada.

En los siguientes cuadros se muestra con cierto detalle el cálculo de las máquinas por centro de trabajo propuesto, para los cuatro productos seleccionados. Nótese que sólo hay 3 centros de trabajo, pues, dos productos se producirán en una sólo célula de producción.

Tabla 2.1 Primera (1) y segunda (2) célula de producción.

| | Tipo | Cantidad | Prim de Prd/min | Tmp Rq/mq | Mnrimto | Prep | Tpo turno | Tpo Neto | No. de Maqs | Total/cro |
|------------|--------|----------|-----------------|-----------|---------|------|-----------|----------|-------------|-----------|
| Ensamble | 2300-1 | 2800 | 2.708 | 1088.4 | 45 | 5 | 480.0 | 425 | 2.58 | 3.0 |
| | 2300-2 | 2700 | 2.5 | 1136.0 | 45 | 5 | 480.0 | 425 | 2.61 | 3.0 |
| Ribeteador | 2300-1 | 2800 | 10.833 | 272.1 | 15 | 10 | 480.0 | 445 | 0.61 | 1.0 |
| | 2300-2 | 2700 | 10.833 | 262.4 | 15 | 10 | 480.0 | 445 | 0.59 | 1.0 |
| Estirador | 2300-1 | 2800 | 3.75 | 786.0 | 15 | 20 | 480.0 | 425 | 1.85 | 2.0 |
| | 2300-2 | 2700 | 3.75 | 767.9 | 15 | 20 | 480.0 | 425 | 1.78 | 2.0 |
| Inspección | 2300-1 | 2800 | 12.5 | 235.8 | 30 | 25 | 480.0 | 400 | 0.59 | 1.0 |
| | 2300-2 | 2700 | 12.5 | 227.4 | 30 | 25 | 480.0 | 400 | 0.57 | 1.0 |

Tabla 2.2 Tercer célula de producción.

| | Tipo | Cantidad | Prim de Prd/min | Tmp Rq/mq | Mnrimto | Prep | Tpo turno | Tpo Neto | No. de Maqs | Total/cro |
|------------|--------|----------|-----------------|-----------|---------|------|-----------|----------|-------------|-----------|
| Ensamble | 3800-1 | 1200 | 2.083 | 606.4 | 45 | 5 | 480.0 | 425 | 1.43 | 2.81 |
| | 3800-2 | 1400 | 2.5 | 589.5 | 45 | 5 | 480.0 | 425 | 1.39 | 2.81 |
| Ribeteador | 3800-1 | 1200 | 10.833 | 116.6 | 15 | 10 | 480.0 | 445 | 0.26 | 0.57 |
| | 3800-2 | 1400 | 10.833 | 136.0 | 15 | 10 | 480.0 | 445 | 0.31 | 0.57 |
| Estirador | 3800-1 | 1200 | 2.708 | 466.5 | 15 | 20 | 480.0 | 425 | 1.10 | 2.38 |
| | 3800-2 | 1400 | 2.708 | 544.2 | 15 | 20 | 480.0 | 425 | 1.28 | 2.38 |
| Inspección | 3800-1 | 1200 | 12.5 | 101.1 | 30 | 25 | 480.0 | 400 | 0.25 | 0.55 |
| | 3800-2 | 1400 | 12.5 | 117.9 | 30 | 25 | 480.0 | 400 | 0.29 | 0.55 |

c. Preparándonos para la simulación.

Esquemas de la simulación.

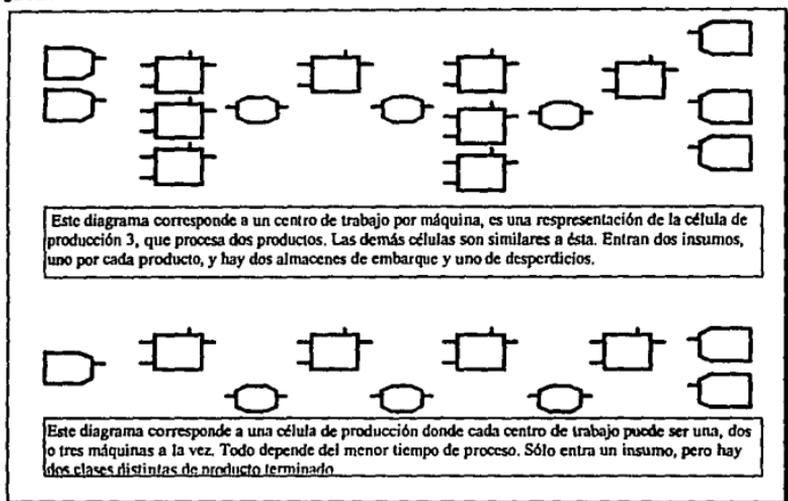
La simulación se puede esquematizar de dos maneras diferentes, una donde se maneja un centro de trabajo por cada máquina, y otra donde cada centro de trabajo representa una o varias máquinas (esto se hace posible mediante la idea de dividir los tiempos de procesos y de preparación, entre el número de máquinas que representa el centro de trabajo en cuestión). La ventaja de la segunda representación es, sobre todo, la rapidez con que corre el modelo. En cambio, la primera representación corre más lento, pero se puede obtener el seguimiento de materiales con más precisión, así como la reacción de cada una de las máquinas durante el día.

En base a los cálculos del número de máquinas necesarias para cada célula de producción, se concluye que la tercer célula, esto es, la que procesa dos productos, necesita una formación 3-1-3-1. Para hacer

un mejor seguimiento de material propongo para el diagrama de la operación, el que se muestra en la primer figura. Es importante hacer notar que los procesos están conectados directamente, entre sí, la salida de uno con la entrada de otro, y no debería de haber inventarios intermedios, sin embargo se pueden añadir para efectos de la logística de operación. Se podría considerar el experimento con inventarios intermedios con distintas capacidades, desde 0 hasta lotes de 50, y observar los cambios que se obtienen en los resultados de producción.

Para el experimento propongo que se simule en Xcell+ un modelo de las células de producción sencillas, de tipo 1 y 2, que sólo varían en la cantidad de productos a elaborar, y de la tercer célula, la que procesa dos productos. Esta última célula de producción se simula con sólo un centro de trabajo por tipo de máquina y con lotes de una sola pieza, de manera que la producción continúa paso por paso. Los inventarios propuestos no tienen capacidad de almacenamiento, por lo que pueden desaparecer del esquema final.

figura 2.1



Un ajuste "pequeño" en los Tiempos de Preparación.

Cuando se inicia el modelado de la tercera célula de producción, hay un detalle que puede pasar de ser percibido, hasta que no se obtienen los primeros resultados: es el de los tiempos de preparación. Si cuando se realiza el cambio de producción, del producto P1 a P2 (en la tercer célula de producción) se planea que los tiempos de preparación se efectúen hasta que llegue la nueva pieza a la nueva máquina, resulta ser que las primeras máquinas esperan una eternidad en volver a funcionar con eficiencia. El resultado es que la primer máquina espera más de 60 minutos antes de que vuelva a fluir por ella los productos con regularidad y sin bloqueos. Alrededor del minuto 217 se terminan de producir las 1200 piezas de P1, pero no es hasta el minuto 278 cuando se producen con fluidez las piezas P2.

Existen tiempos muertos en las máquinas posteriores a las primeras, pues cuando llegan las nuevas piezas se tardan en preparar el tiempo suficiente como para que el resto del sistema se vacíe de las últimas piezas P1 que quedaban en el sistema, este tiempo muerto no se aprovecha, y no es hasta que llega el primer producto P2 a cada máquina cuando se comienza la preparación de las máquinas. Así se

entiende porqué se pierde tanto tiempo. La idea para resolver este problema es aprovechar los tiempos muertos de las máquinas para iniciar la preparación y así ahorrar tiempo. Esta idea no es fácil de programar en Xcell+, mucho más directa de implementar en PASION. Para el modelo de Xcell+, se reprogramaron los centros de trabajo con nuevos tiempos de preparación que descontaban la utilización de los tiempos muertos de cada máquina.

El algoritmo matemático para este descuento es sumamente interesante, me costó algún tiempo en dar con él, y otro tanto en tenerlo claro como para poderlo explicar. Fundamentalmente consiste en el seguimiento de la última pieza P1 que pasa y la primer P2 que llega a una máquina, supongamos 3. Si le llamamos T3 al tiempo le toma a la última pieza P1 salir de la máquina 1 y pasar por la máquina 2, pasar por la máquina 3 y el tiempo de preparación de 3 para recibir un P2, su ecuación podría ser:

$$T3 = tA2 + tA3 + S3$$

Donde tAi representa el tiempo de proceso de la pieza P1 en la máquina "i", y S3 el tiempo real de preparación de la máquina 3 para recibir P2.

Pensemos ahora en el tiempo que le toma a la primer P2 ser procesada en 3. P2 es procesada en la máquina 1, previa preparación de 1, luego procesada en 2 (con su preparación), luego se prepara 3. La ecuación podría escribirse como:

$$T3 = S'1 + tB1 + S'2 + tB2 + S'3$$

Donde tBi es el tiempo de proceso de una pieza P2 en la máquina "i", y "S' i" es el tiempo de preparación que le falta a la máquina "i" para procesar un P2. Si se igualan las dos ecuaciones podemos encontrar los valores de las S's que significan los nuevos tiempos de preparación "descontados", que funcionarán en nuestro modelo Xcell+, para asumir que los tiempos muertos de las máquinas se utilizan para prepararlas. El cálculo implica el despeje (para este caso de S'3), aunque con esta idea se puede encontrar S'2 y S'4, de hecho no se puede encontrar la siguiente variable hasta que no se haya encontrado la anterior. El cálculo se muestra en el siguiente cuadro:

Tabla 2.3 Ajuste de los tiempos de preparación.

| | | Prep Real | Tpo min/prod. | Total | Acum | Preparación Descontada |
|------------|----|-----------|---------------|-------|-------|------------------------|
| Ensamble | p1 | 5 | 0.48 | 5.48 | 5.48 | 5.48 |
| | p2 | 5 | 0.40 | 5.40 | 5.40 | 5.40 |
| Ribeador | p1 | 10 | 0.09 | 10.09 | 15.57 | 4.6125 |
| | p2 | 10 | 0.09 | 10.09 | 15.49 | 4.6924 |
| Estirador | p1 | 20 | 0.37 | 20.37 | 35.94 | 10.2774 |
| | p2 | 20 | 0.37 | 20.37 | 35.86 | 10.2774 |
| Inspección | p1 | 25 | 0.08 | 25.08 | 61.02 | 4.7114 |
| | p2 | 25 | 0.08 | 25.08 | 60.94 | 4.7114 |

Los últimos datos para la simulación.

Para el cálculo de los tiempos de procesos que cada máquina tarda en un producto determinado se elaboró la tabla 2.4 de la página siguiente:

Tabla 2.4 Tiempos de proceso

| | Tp - Producción | Tp - montaje |
|------------|-----------------|--------------|
| Ensamble | 2.083 | 0.48 |
| | 2.5 | 0.40 |
| Ribeteador | 10.833 | 0.09 |
| | 10.833 | 0.09 |
| Estrador | 2.708 | 0.37 |
| | 2.708 | 0.37 |
| Inspección | 12.5 | 0.08 |
| | 12.5 | 0.08 |

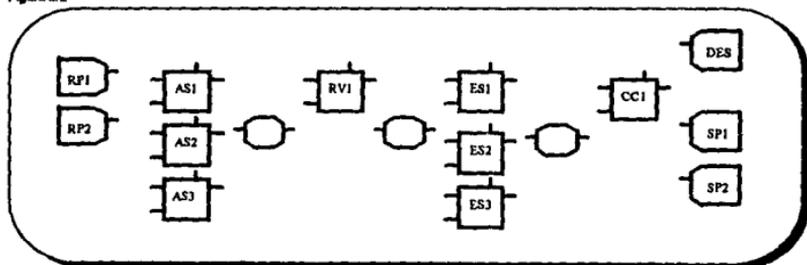
2.3.2 Simulación en Xcell+.

La simulación de las primeras dos células de producción, esto es, las que se refieren a la producción en exclusiva de los productos de la serie 2300, no se incluyen en el siguiente reporte. Realmente estas células de producción están sobradas de manera que su simulación carece de importancia. Cuando se habla de lo que pasará con la tercer célula de producción, entonces el problema toma interés. La tercera célula de producción trata de los productos de la serie 3800. Esta serie se producirá completamente en una misma célula de producción, esto implica, tiempos de preparación, cambio automático de producción, inventarios, lotes, y todos los elementos que se utilizarían en la simulación de las dos primeras células. Este es el modelo difícil del caso, el de mayor interés. Para esta célula de producción se proponen dos modelos, el sencillo y el complejo. El sencillo no considera la planeación del centro de mantenimiento y todos los inventarios que contiene tienen capacidad de 0. El complejo añade la consideración de los mantenimientos programados y además considera que los inventarios tienen capacidad de 1 unidad.

Los resultados que arrojan estas dos simulaciones no son muy distintos, y aunque la conclusión es que la tercera célula no podrá cumplir los requerimientos de producción que se necesitan, la discusión sobre este resultado se realiza en la parte conclusiva de este capítulo.

A continuación se muestra el modelo gráfico del modelo, en simbología de Xcell+, y luego se muestran los resultados de las simulaciones para ambos casos, durante un período de 4800 hrs., esto es, 10 días.

Figura 2.2



Los dos modelos analizados sólo difieren entre sí respecto las declaraciones de las características de los inventarios y la declaración del funcionamiento de centro de mantenimiento.

Primera Célula de Producción.

Resultados de la simulaciones en Xcellt

a) Simulación sencilla.

Factory C_PROD:

WorkCenter AS1; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 8523 8523
% of time (proc'g,setup,min): 78.03 1.98 0.00
costs (capital,unit operating): 500.00 0.12

Process P2 at WorkCenter AS1
inputs: X-input from ReceivingArea RP1; no Y-input
normal-output to: stock P2 at B1
reject-output: % reject: 0.00
group: 2 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.40
trigger: <HIGH>
invoke when qty at SPI increases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.18
units processed (period,cum): 4315 4315

Process P1 at WorkCenter AS1
inputs: X-input from ReceivingArea RP1; no Y-input
normal-output to: stock P1 at B1
reject-output: % reject: 0.00
group: 2800 major-setup: 3.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.48
trigger: <LOW>
invoke when qty at SPI decreases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4208 4208

WorkCenter RV1; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 25554 25554
% of time (proc'g,setup,min): 49.14 1.84 0.00
costs (capital,unit operating): 350.00 0.12

Process P2 at WorkCenter RV1
inputs: X-input from stock P2 at B1; no Y-input
normal-output to: stock P2 at B2
reject-output: % reject: 0.00
group: 2 major-setup: 4.69 minor-setup: 4.69
processing-time distn: <CONSTANT>
every proc'g time is: 0.09
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: IDLE, no material
units processed (period,cum): 12913 12913

Process P1 at WorkCenter RV1
inputs: X-input from stock P1 at B1; no Y-input
normal-output to: stock P1 at B2
reject-output: % reject: 0.00
group: 2800 major-setup: 4.61 minor-setup: 4.61
processing-time distn: <CONSTANT>
every proc'g time is: 0.09

trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 12641 12641

WorkCenter BS1; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 8518 8518
% of time (proc'g,setup,min): 65.53 4.07 0.00
costs (capital,unit operating): 250.00 0.12

Process P2 at WorkCenter BS1
inputs: X-input from stock P2 at B2; no Y-input
normal-output to: stock P2 at B3
reject-output: % reject: 0.00
group: 2 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.30
units processed (period,cum): 4298 4298

Process P1 at WorkCenter BS1
inputs: X-input from stock P1 at B2; no Y-input
normal-output to: stock P1 at B3
reject-output: % reject: 0.00
group: 2800 major-setup: 10.28 minor-setup:
10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4220 4220

WorkCenter AS2; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 8509 8509
% of time (proc'g,setup,min): 77.93 1.98 0.00
costs (capital,unit operating): 500.00 0.12

Process P2 at WorkCenter AS2
inputs: X-input from ReceivingArea RP2; no Y-input
normal-output to: stock P2 at B1
reject-output: % reject: 0.00
group: 2 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.40
trigger: <HIGH>
invoke when qty at SPI increases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.01
units processed (period,cum): 4298 4298

Process P1 at WorkCenter AS2
inputs: X-input from ReceivingArea RP1; no Y-input
normal-output to: stock P1 at B1
reject-output: % reject: 0.00
group: 2800 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.48
trigger: <LOW>
invoke when qty at SPI decreases to 1200

priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4211 4211

WorkCenter AS3; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 8525 8525
% of time (proc'g,setup,rtn): 78.07 1.98 0.00
costs (capital,unit operating): 500.00 0.12

Process P2 at WorkCenter AS3
inputs: X-input from ReceivingArea RP2; no Y-input
normal-output to: stock P2 at B1
reject-output: % reject: 0.00
group: 2 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.40
trigger: <HIGH>
invokes when qty at SP1 increases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.35
units processed (period,cum): 4303 4303

Process P1 at WorkCenter AS3
inputs: X-input from ReceivingArea RP1; no Y-input
normal-output to: stock P1 at B1
reject-output: % reject: 0.00
group: 2800 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.48
trigger: <LOW>
invokes when qty at SP1 decreases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4222 4222

WorkCenter ES2; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 8505 8505
% of time (proc'g,setup,rtn): 63.43 4.07 0.00
costs (capital,unit operating): 250.00 0.12

Process P2 at WorkCenter ES2
inputs: X-input from stock P2 at B2; no Y-input
normal-output to: stock P2 at B3
reject-output: % reject: 0.00
group: 2 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.13
units processed (period,cum): 4301 4301

Process P1 at WorkCenter ES2
inputs: X-input from stock P1 at B2; no Y-input
normal-output to: stock P1 at B3
reject-output: % reject: 0.00
group: 2800 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive

units processed (period,cum): 4204 4204

WorkCenter CC1; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 25552 25552
% of time (proc'g,setup,rtn): 42.59 1.86 0.00
costs (capital,unit operating): 380.00 0.12

Process P2 at WorkCenter CC1
inputs: X-input from stock P2 at B3; no Y-input
normal-output to: ShippingArea SP2
reject-output: % reject: 5.00; <SCRAPPED>
gone to: ShippingArea DESP
drift: initial: 5.00; % change: 0.00
group: 2 major-setup: 4.71 minor-setup: 4.71
processing-time distn: <CONSTANT>
every proc'g time is: 0.08
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.04
units processed (period,cum): 12911 12911

Process P1 at WorkCenter CC1
inputs: X-input from stock P1 at B3; no Y-input
normal-output to: ShippingArea SP1
reject-output: % reject: 5.00; <SCRAPPED>
gone to: ShippingArea DESP
drift: initial: 5.00; % change: 0.00
group: 2800 major-setup: 4.71 minor-setup: 4.71
processing-time distn: <CONSTANT>
every proc'g time is: 0.08
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 12641 12641

WorkCenter ES3; has Processes named:

P2 (current Process)
P1
no breakdowns or scheduled maintenance
units processed (period,cum): 8331 8331
% of time (proc'g,setup,rtn): 63.64 4.07 0.00
costs (capital,unit operating): 250.00 0.12

Process P2 at WorkCenter ES3
inputs: X-input from stock P2 at B2; no Y-input
normal-output to: stock P2 at B3
reject-output: % reject: 0.00
group: 2 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: IDE, no material
units processed (period,cum): 4314 4314

Process P1 at WorkCenter ES3
inputs: X-input from stock P1 at B2; no Y-input
normal-output to: stock P1 at B3
reject-output: % reject: 0.00
group: 2800 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4217 4217

Buffer B1; capacity: 0
withdrawal order: UNORDERED
minimum-holding-time: 0.00

| contents: | | min | max | average | | throughput | | | |
|-----------|------|-----|--------|---------|--------|------------|--------|-------|-------|
| Part | unit | now | period | cum | period | cum | period | cum | |
| P2 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 12913 | 12913 |
| P1 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 12641 | 12641 |
| totals: | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 25554 | 25554 | |

no request triggers specified
no delivery triggers specified
costs (capital, unit operating): 10.00 1.00

Buffer B2; capacity: 0
withdrawal order: UNORDERED
minimum-holding-time: 0.00

| contents: | | min | max | average | | throughput | | | |
|-----------|------|-----|--------|---------|--------|------------|--------|-------|-------|
| Part | unit | now | period | cum | period | cum | period | cum | |
| P2 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 12913 | 12913 |
| P1 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 12641 | 12641 |
| totals: | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 25554 | 25554 | |

no request triggers specified
no delivery triggers specified
costs (capital, unit operating): 10.00 1.00

Buffer B3; capacity: 0
withdrawal order: UNORDERED
minimum-holding-time: 0.00

| contents: | | min | max | average | | throughput | | | |
|-----------|------|-----|--------|---------|--------|------------|--------|-------|-------|
| Part | unit | now | period | cum | period | cum | period | cum | |
| P2 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 12911 | 12911 |
| P1 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 12641 | 12641 |
| totals: | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 25552 | 25552 | |

no request triggers specified
no delivery triggers specified
costs (capital, unit operating): 10.00 1.00

ReceivingArea RP1 supplies units to:

X input to P1 at AS1
 X input to P1 at AS2
 X input to P1 at AS3
 Part supplied is P1
 tag 1 unit in 100
 arrivals: <REGULAR-BATCH>
 range of batch-sizes: 1 1
 range of time between batches: 0.17 0.17
 storage-capacity: 2
 quantity-on-hand: 2
 next batch to arrive at: 4800.12
 units rejected (period,cum): 15593 15593
 units started (period,cum): 12641 12641

| WIP: | min | max | average |
|------|------|--------|---------|
| | now | period | cum |
| | 2 | 0 | 2 |
| | 0 | 0 | 2 |
| | 2 | 2 | 1.11 |
| | 1.11 | 1.11 | |

costs (capital, unit operating): 25.00 2.00

ReceivingArea RP2 supplies units to:

X input to P2 at AS1
 X input to P2 at AS2
 X input to P2 at AS3
 Part supplied is P2
 tag 1 unit in 100
 arrivals: <REGULAR-BATCH>
 range of batch-sizes: 1 1
 range of time between batches: 0.17 0.17
 storage-capacity: 2
 quantity-on-hand: 0
 next batch to arrive at: 4800.12
 units rejected (period,cum): 15320 15320
 units started (period,cum): 12916 12916

| WIP: | min | max | average |
|------|------|--------|---------|
| | now | period | cum |
| | 0 | 0 | 2 |
| | 0 | 0 | 2 |
| | 2 | 2 | 1.09 |
| | 1.09 | 1.09 | |

costs (capital, unit operating): 25.00 2.00

ShippingArea SF2 receives units of:

P2; normal-output from Proc P2 at WkCtr CC1
 all types of Parts accepted

b) Simulación compleja.**Factory C_PROD:****WorkCenter AS1; has Processes named:**

P2 (current Process)
 P1
 scheduled maintenance provided by M1
 CONSTANT interval between maintenance: 480.00
 CONSTANT service time: 45.00
 units processed (period,cum): 7072 7072
 % of time (proc%setup,inst): 65.99 1.98 8.44
 costs (capital,unit operating): 500.00 0.12

Process P1 at WorkCenter AS1
 inputs: X-input from ReceivingArea RP2; no Y-input
 normal-output to: stock P2 at H1
 reject-output: % reject: 0.00
 group: 2 major-setup: 5.00 minor-setup: 5.00
 processing-time distn: <CONSTANT>
 every proc% time is: 0.40

shipments: <REGULAR-BATCH>

range of batch-sizes: 1400 1400
 range of time between batches: 480.00 480.00
 storage-capacity: 1450
 quantity-on-hand: 0
 next shipment scheduled for: 5280.00
 shipment shortage (period,cum): 1740 1740
 units accepted (period,cum): 12260 12260
 tagged-unit avg time (total,in prc,wait): 1.14 0.94 0.20

| WIP: | min | max | average |
|------|--------|--------|---------|
| | now | period | cum |
| | 0 | 0 | 1324 |
| | 0 | 0 | 1324 |
| | 283.01 | 283.01 | |
| | 283.01 | 283.01 | |

costs (capital, unit operating): 25.00 2.00

ShippingArea DESP receives units of:

P1; scrap from Proc P1 at WkCtr CC1
 P2; scrap from Proc P2 at WkCtr CC1
 all types of Parts accepted
 shipments: <CONTINUOUS>
 storage-capacity: <UNLIMITED>
 units accepted (period,cum): 1285 1285
 tagged-unit avg time (total,in prc,wait): 0.97 0.97 0.00
 costs (capital, unit operating): 25.00 2.00

ShippingArea SP1 receives units of:

P1; normal-output from Proc P1 at WkCtr CC1
 Part accepted is P1
 shipments: <REGULAR-BATCH>
 range of batch-sizes: 1200 1200
 range of time between batches: 480.00 480.00
 storage-capacity: 1250
 quantity-on-hand: 6
 next shipment scheduled for: 5280.00
 shipment shortage (period,cum): 0 0
 units accepted (period,cum): 12006 12006
 tagged-unit avg time (total,in prc,wait): 1.02 1.02 0.00

| WIP: | min | max | average |
|------|--------|--------|---------|
| | now | period | cum |
| | 6 | 0 | 1206 |
| | 0 | 0 | 1206 |
| | 880.29 | 880.29 | |
| | 880.29 | 880.29 | |

costs (capital, unit operating): 25.00 2.00

trigger: <HIGH>

invokes when qty at SP1 increases to 1200
 priority: 100 batch-size: <NOT SPECIFIED>
 current state: BUSY processing until 4800.01
 units processed (period,cum): 2839 2839

Process P1 at WorkCenter AS1

inputs: X-input from ReceivingArea RP1; no Y-input
 normal-output to: stock P1 at B1
 reject-output: % reject: 0.00
 group: 2800 major-setup: 5.00 minor-setup: 5.00
 processing-time distn: <CONSTANT>
 every proc% time is: 0.48
 trigger: <LOW>
 invokes when qty at SP1 decreases to 1200
 priority: 100 batch-size: <NOT SPECIFIED>
 current state: inactive
 units processed (period,cum): 4233 4233

WorkCenter RV1; has Processes named:

P2 (current Process)

P1
scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 15.00
units processed (period,cum): 21119 21119
% of time (proc,gsetup,rtn): 40.61 1.84 2.81
costs (capital,unit operating): 350.00 0.12

Process P2 at WorkCenter RV1
inputs: X-input from stock P2 at B1; no Y-input
normal-output to: stock P2 at B2
reject-output: % reject: 0.00
group: 2 major-setup: 4.69 minor-setup: 4.69
processing-time distn: <CONSTANT>
every proc'g time is: 0.09
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: IDLE, no material
units processed (period,cum): 8478 8478

Process P1 at WorkCenter RV1
inputs: X-input from stock P1 at B1; no Y-input
normal-output to: stock P1 at B2
reject-output: % reject: 0.00
group: 2800 major-setup: 4.61 minor-setup: 4.61
processing-time distn: <CONSTANT>
every proc'g time is: 0.09
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 12641 12641

WorkCenter ES1; has Processes named:

P2 (current Process)

P1
scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 15.00
units processed (period,cum): 7005 7005
% of time (proc,gsetup,rtn): 53.89 4.07 2.81
costs (capital,unit operating): 250.00 0.12

Process P2 at WorkCenter ES1
inputs: X-input from stock P2 at B2; no Y-input
normal-output to: stock P2 at D3
reject-output: % reject: 0.00
group: 2 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.30
units processed (period,cum): 2811 2811

Process P1 at WorkCenter ES1
inputs: X-input from stock P1 at B2; no Y-input
normal-output to: stock P1 at B3
reject-output: % reject: 0.00
group: 2800 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4194 4194

WorkCenter AS2; has Processes named:

P2 (current Process)

P1

scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 45.00
units processed (period,cum): 7007 7007
% of time (proc,gsetup,rtn): 63.37 1.98 8.44
costs (capital,unit operating): 500.00 0.12

Process P2 at WorkCenter AS2
inputs: X-input from ReceivingArea RP2; no Y-input
normal-output to: stock P2 at B1
reject-output: % reject: 0.00
group: 2 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.40
trigger: <HIGH>
invoke when qty at SP1 increases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.35
units processed (period,cum): 2814 2814

Process P1 at WorkCenter AS2
inputs: X-input from ReceivingArea RP1; no Y-input
normal-output to: stock P1 at B1
reject-output: % reject: 0.00
group: 2800 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.48
trigger: <LOW>
invoke when qty at SP1 decreases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4193 4193

WorkCenter AS3; has Processes named:

P2 (current Process)

P1
scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 45.00
units processed (period,cum): 7043 7043
% of time (proc,gsetup,rtn): 65.71 1.98 8.44
costs (capital,unit operating): 500.00 0.12

Process P2 at WorkCenter AS3
inputs: X-input from ReceivingArea RP2; no Y-input
normal-output to: stock P2 at D1
reject-output: % reject: 0.00
group: 2 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.40
trigger: <HIGH>
invoke when qty at SP1 increases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.18
units processed (period,cum): 2828 2828

Process P1 at WorkCenter AS3
inputs: X-input from ReceivingArea RP1; no Y-input
normal-output to: stock P1 at B1
reject-output: % reject: 0.00
group: 2800 major-setup: 5.00 minor-setup: 5.00
processing-time distn: <CONSTANT>
every proc'g time is: 0.48
trigger: <LOW>
invoke when qty at SP1 decreases to 1200
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4215 4215

WorkCenter ES2; has Processes named:

P2 (current Process)
P1
scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 15.00
units processed (period,cum): 7045 7045
% of time (proc%setup,rtn): 54.20 4.07 2.81
costs (capital,unit operating): 250.00 0.12

Process P2 at WorkCenter ES2
inputs: X-input from stock P2 at B2; no Y-input
normal-output to: stock P2 at B3
reject-output: % reject: 0.00
group: 2 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.13
units processed (period,cum): 2830 2830

Process P1 at WorkCenter ES2
inputs: X-input from stock P1 at B2; no Y-input
normal-output to: stock P1 at B3
reject-output: % reject: 0.00
group: 2800 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4215 4215

WorkCenter CCI: has Processes named:

P2 (current Process)
P1
scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 30.00
units processed (period,cum): 21117 21117
% of time (proc%setup,rtn): 35.19 1.86 5.63
costs (capital,unit operating): 380.00 0.12

Process P2 at WorkCenter CCI
inputs: X-input from stock P2 at B3; no Y-input
normal-output to: ShippingArea SP2
reject-output: % reject: 5.00; <SCRAPPED>
gone to: ShippingArea DESP
drift: initial: 5.00; % change: 0.00
group: 2 major-setup: 4.71 minor-setup: 4.71
processing-time distn: <CONSTANT>
every proc'g time is: 0.08

trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: BUSY processing until 4800.04
units processed (period,cum): 8476 8476

Process P1 at WorkCenter CCI
inputs: X-input from stock P1 at B3; no Y-input
normal-output to: ShippingArea SP1
reject-output: % reject: 5.00; <SCRAPPED>
gone to: ShippingArea DESP
drift: initial: 5.00; % change: 0.00
group: 2800 major-setup: 4.71 minor-setup: 4.71
processing-time distn: <CONSTANT>
every proc'g time is: 0.08
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 12641 12641

WorkCenter ES3: has Processes named:

P2 (current Process)
P1
scheduled maintenance provided by M1
CONSTANT interval between maintenance: 480.00
CONSTANT service time: 15.00
units processed (period,cum): 7069 7069
% of time (proc%setup,rtn): 54.39 4.07 2.81
costs (capital,unit operating): 250.00 0.12

Process P2 at WorkCenter ES3
inputs: X-input from stock P2 at B2; no Y-input
normal-output to: stock P2 at B3
reject-output: % reject: 0.00
group: 2 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: IDLE, no material
units processed (period,cum): 2837 2837

Process P1 at WorkCenter ES3
inputs: X-input from stock P1 at B2; no Y-input
normal-output to: stock P1 at B3
reject-output: % reject: 0.00
group: 2800 major-setup: 10.28 minor-setup: 10.28
processing-time distn: <CONSTANT>
every proc'g time is: 0.37
trigger: <NONE>
priority: 100 batch-size: <NOT SPECIFIED>
current state: inactive
units processed (period,cum): 4232 4232

Buffer B1; capacity: 1
 withdrawal order: UNORDERED
 minimum-holding-time: 0.00

| contents: | | min | | max | | average | | throughput | | |
|----------------|----------|----------|----------|----------|----------|----------|-------------|-------------|--------------|--------------|
| Part | init | now | period | cum | period | cum | period | cum | period | cum |
| P2 | 0 | 0 | 0 | 0 | 1 | 1 | 0.05 | 0.05 | 8478 | 8478 |
| P1 | 0 | 0 | 0 | 0 | 1 | 1 | 0.12 | 0.12 | 12641 | 12641 |
| totals: | 0 | 0 | 0 | 0 | 1 | 1 | 0.17 | 0.17 | 21119 | 21119 |

no request triggers specified
 no delivery triggers specified
 costs (capital, unit operating): 10.00 1.00

Buffer B2; capacity: 1
 withdrawal order: UNORDERED
 minimum-holding-time: 0.00

| contents: | | min | | max | | average | | throughput | | |
|----------------|----------|----------|----------|----------|----------|----------|-------------|-------------|--------------|--------------|
| Part | init | now | period | cum | period | cum | period | cum | period | cum |
| P2 | 0 | 0 | 0 | 0 | 1 | 1 | 0.04 | 0.04 | 8478 | 8478 |
| P1 | 0 | 0 | 0 | 0 | 1 | 1 | 0.09 | 0.09 | 12641 | 12641 |
| totals: | 0 | 0 | 0 | 0 | 1 | 1 | 0.13 | 0.13 | 21119 | 21119 |

no request triggers specified
 no delivery triggers specified
 costs (capital, unit operating): 10.00 1.00

Buffer B3; capacity: 1
 withdrawal order: UNORDERED
 minimum-holding-time: 0.00

| contents: | | min | | max | | average | | throughput | | |
|----------------|----------|----------|----------|----------|----------|----------|-------------|-------------|--------------|--------------|
| Part | init | now | period | cum | period | cum | period | cum | period | cum |
| P2 | 0 | 0 | 0 | 0 | 1 | 1 | 0.02 | 0.02 | 8476 | 8476 |
| P1 | 0 | 0 | 0 | 0 | 1 | 1 | 0.06 | 0.06 | 12641 | 12641 |
| totals: | 0 | 0 | 0 | 0 | 1 | 1 | 0.07 | 0.07 | 21117 | 21117 |

no request triggers specified
 no delivery triggers specified
 costs (capital, unit operating): 10.00 1.00

ReceivingArea RP1 supplies units to:

X input to P1 at AS1

X input to P1 at AS2

X input to P1 at AS3

Part supplied is P1

tag 1 unit in 100

arrivals: <REGULAR-BATCH>

range of batch-sizes: 1 1

range of time between batches: 0.17 0.17

storage-capacity: 2

quantity-on-hand: 2

next batch to arrive at: 4800.12

units rejected (period,cum): 15593 15593

units started (period,cum): 12641 12641

| WIP: | min | max | average |
|----------------|------------|------------|------------|
| now period cum | period cum | period cum | period cum |
| 2 | 0 | 0 | 2 |
| | | | 1.12 1.12 |

costs (capital, unit operating): 25.00 2.00

ReceivingArea RP2 supplies units to:

X input to P2 at AS1

X input to P2 at AS2

X input to P2 at AS3

Part supplied is P2

tag 1 unit in 100

arrivals: <REGULAR-BATCH>

range of batch-sizes: 1 1

range of time between batches: 0.17 0.17

storage-capacity: 2

quantity-on-hand: 0

next batch to arrive at: 4800.12

units rejected (period,cum): 19755 19755

units started (period,cum): 8481 8481

| WIP: | min | max | average |
|----------------|------------|------------|------------|
| now period cum | period cum | period cum | period cum |
| 0 | 0 | 0 | 2 |
| | | | 1.41 1.41 |

costs (capital, unit operating): 25.00 2.00

ShippingArea SP2 receives units of:

P2; normal-output from Proc P2 at WkCr CC1

all types of Parts accepted

shipments: <REGULAR-BATCH>

range of batch-sizes: 1400 1400

range of time between batches: 480.00 480.00

storage-capacity: 1450

quantity-on-hand: 0

next shipments scheduled for: 5280.00

shipment shortage (period,cum): 5962 5962

units accepted (period,cum): 8038 8038

| WIP: | min | max | average |
|----------------|------------|------------|--------------------|
| now period cum | period cum | period cum | period cum |
| 0 | 0 | 0 | 1327 |
| | | | 1327 134.15 134.15 |

costs (capital, unit operating): 25.00 2.00

ShippingArea DESP receives units of:

P1; scrap from Proc P1 at WkCr CC1

P2; scrap from Proc P2 at WkCr CC1

all types of Parts accepted

shipments: <CONTINUOUS>

storage-capacity: <UNLIMITED>

units accepted (period,cum): 1072 1072

costs (capital, unit operating): 25.00 2.00

ShippingArea SP1 receives units of:

P1; normal-output from Proc P1 at WkCr CC1

Part accepted is P1

shipments: <REGULAR-BATCH>

range of batch-sizes: 1200 1200

range of time between batches: 480.00 480.00

storage-capacity: 1230

quantity-on-hand: 6

next shipments scheduled for: 5280.00

shipment shortage (period,cum): 0 0

units accepted (period,cum): 12006 12006

| WIP: | min | max | average |
|----------------|------------|------------|--------------------|
| now period cum | period cum | period cum | period cum |
| 6 | 0 | 0 | 1206 |
| | | | 1206 774.34 774.34 |

costs (capital, unit operating): 25.00 2.00

MaintenanceCenter M1

nbr of service-teams (total, idle): 8 8

% time busy (period,cum): 16.62 16.62

% time queued (period,cum): 0.00 0.00

status of WorkCenters assigned to M1

AS1 in production

RV1 idle

ES1 in production

AS2 in production

AS3 in production

ES2 in production

CC1 in production

ES3 idle

cost (capital): 25.00 - 10.00 per team

2.3.3 Simulación en PASION.

A continuación se presenta el listado del programa correspondiente PRO1, en el que se realiza la Simulación del modelo anterior en PASION. Este listado está codificado en el lenguaje PASION, y necesita de un traductor (PAT4¹) para generar los correspondientes programas en Pascal, los que son directamente ejecutables. Cuando se realiza la traducción a estos programas se obtienen los siguientes archivos:

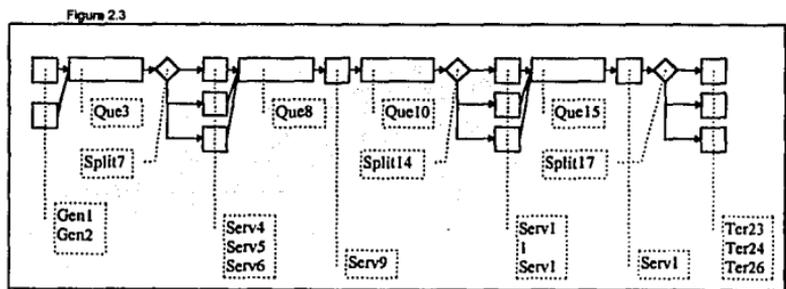
RPRO1.pas: Contiene algunos de los procedimientos principales del modelo (contiene el procedimiento SVOP²).

PPRO1.pas: Es propiamente lo que se conoce como programa principal, es el directamente ejecutable.

DIS1.pas: Este programa no se genera automáticamente, sino que es un archivo que contiene las funciones de probabilidad que controlan los tiempos de los servidores, entradas y salidas del sistema, así como reglas especiales en la división de flujos de material).

Para este modelo se necesitó declarar un atributo nuevo como propiedad de las piezas, a este atributo se le dio el nombre de ID, y que significaba el número de pieza de su género que había entrado al sistema durante el día. Esto servía para identificar las primeras piezas de su género y efectuar los tiempos de SetUp adecuados, según el caso. Estos tiempos se tomaron según la versión corregida desarrollada en la parte teórica de la solución.

Todas las indicaciones que se encuentran entre llaves significan explicaciones del programa. Además el número de evento corresponde a el número de bloque según se realizó el modelo gráfico en QMG8. QMG8 es, como se explica en el anexo 1, el interfase gráfico para diseñar un modelo de simulación discreta. El diagrama de bloques que se genera es parecido al de Xcell+, a continuación se presenta el esquema en QMG:



PROGRAM PROD1;

(*SI DIS1.PAS*)

{PACION program generated by the QMG module}
{Use PAT4 to translate it to Pascal}
USES RPROD1;

PROCESS ENTITY,2000;
ATR TIMIN,TIMQ-REAL; SOURCE=INTEGER;

¹Aunque esta información se presenta en el anexo 1, correspondiente a PASION, es oportuno aclarar que el traductor directo de PASION a PASCAL es PAT4, aunque las operaciones se pueden resumir ejecutando SIMQ, que a su vez corre una serie de procedimientos que incluyen PAT4.

²Conviene revisar la información sobre PASION, en el anexo 1, donde se detalla información sobre el procedimiento SVOP.

ID-REAL;

```
EVENT EV3;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
WAIT(QUE3); TIMQ:=TIME;
PK:=FALSE;
IF SER4 THEN PK:=TRUE;
IF SER5 THEN PK:=TRUE;
IF SER6 THEN PK:=TRUE;
IF PK THEN BEGIN
OUT(QUE3,N,F);
IF N>0 THEN ENTITY[N] EV7:=TIME;
QN3:=QN3+1.0 END;
ENDEV;
```

```
EVENT EV7;
VAR BX:BOOLEAN;
BEGIN
BX:=FALSE;
IF SER4 THEN BX:=TRUE;
IF SER5 THEN BX:=TRUE;
IF SER6 THEN BX:=TRUE;
IF NOT BX THEN BEGIN KILL THIS; ABAND END;
IF BX THEN
IF SER4 THEN BEGIN EV4:=TIME; BX:=FALSE END;
IF BX THEN
IF SER5 THEN BEGIN EV5:=TIME; BX:=FALSE END;
IF BX THEN
IF SER6 THEN BEGIN EV6:=TIME; BX:=FALSE END;
```

ENDEV;

```
EVENT EV4;
BEGIN
IF NOT SER4 THEN BEGIN
LSER4:=LSER4+1; KILL THIS; ABAND END;
SER4:=FALSE; SV4:=TIME;
```

```
SVOP(4,SOURCE,TIMIN,TIMQ,JD);
ENDS4:=TIME-DIS(SOURCE,JD);ENDEV;
```

```
EVENT ENDS4;
VAR N:INTEGER; BEGIN
SER4:=TRUE;NSER4:=NSER4+1;COST:=COST+ 1.0000;
STM4:=STM4+TIME-SV4;
IF NR(QUE3)>0 THEN BEGIN
OUT(QUE3,N,F);
QN3:=QN3+1.0;
QU3:=QU3+TIME-ENTITY[N]TIMQ;
COST:=COST+(TIME-ENTITY[N]TIMQ)* 1.0000;
ENTITY[N] EV7:=TIME+0.0 END;
EV8:=TIME;ENDEV;
```

```
EVENT EV8;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
IF NR(QUE8)>=500 THEN
BEGIN LD8:=LD8+1;
KILL THIS; ABAND END;
WAIT(QUE8); TIMQ:=TIME;
IF SER9 THEN BEGIN OUT(QUE8,N,F); IF N > 0 THEN
BEGIN
QN8:=QN8+1.0;
```

```
ENTITY[N] EV9:=TIME END END;
ENDEV;
```

```
EVENT EV9;
BEGIN
IF NOT SER9 THEN BEGIN
LSER9:=LSER9+1; KILL THIS; ABAND END;
SER9:=FALSE; SV9:=TIME;
```

```
SVOP(9,SOURCE,TIMIN,TIMQ,JD);
ENDS9:=TIME-DIS(SOURCE,JD);ENDEV;
```

```
EVENT ENDS9;
VAR N:INTEGER; BEGIN
SER9:=TRUE;NSER9:=NSER9+1;COST:=COST+ 1.0000;
STM9:=STM9+TIME-SV9;
IF NR(QUE9)>0 THEN BEGIN
OUT(QUE9,N,F);
QN9:=QN9+1.0;
QU9:=QU9+TIME-ENTITY[N]TIMQ;
COST:=COST+(TIME-ENTITY[N]TIMQ)* 1.0000;
ENTITY[N] EV9:=TIME+0.0 END;
EV10:=TIME;ENDEV;
```

```
EVENT EV10;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
IF NR(QUE10)>=500 THEN
BEGIN LD10:=LD10+1;
KILL THIS; ABAND END;
WAIT(QUE10); TIMQ:=TIME;
PK:=FALSE;
IF SER11 THEN PK:=TRUE;
IF SER12 THEN PK:=TRUE;
IF SER13 THEN PK:=TRUE;
IF PK THEN BEGIN
OUT(QUE10,N,F);
IF N>0 THEN ENTITY[N] EV14:=TIME;
QN10:=QN10+1.0 END;
ENDEV;
```

```
EVENT EV14;
VAR BX:BOOLEAN;
BEGIN
BX:=FALSE;
IF SER11 THEN BX:=TRUE;
IF SER12 THEN BX:=TRUE;
IF SER13 THEN BX:=TRUE;
IF NOT BX THEN BEGIN KILL THIS; ABAND END;
IF BX THEN
IF SER11 THEN BEGIN EV11:=TIME; BX:=FALSE END;
IF BX THEN
IF SER12 THEN BEGIN EV12:=TIME; BX:=FALSE END;
IF BX THEN
IF SER13 THEN BEGIN EV13:=TIME; BX:=FALSE END;
```

ENDEV;

```
EVENT EV11;
BEGIN
IF NOT SER11 THEN BEGIN
LSER11:=LSER11+1; KILL THIS; ABAND END;
SER11:=FALSE; SV11:=TIME;
```

```
SVOP(11,SOURCE,TIMIN,TIMQ,JD);
ENDS11:=TIME-DIS(SOURCE,JD);ENDEV;
```

```
EVENT ENDS11;
VAR N:INTEGER; BEGIN
SER11:=TRUE;NSER11:=NSER11+1;COST:=COST+ 1.0000;
STM11:=STM11+TIME-SV11;
IF NR(QUE10)>0 THEN BEGIN
OUT(QUE10,N,F);
QN10:=QN10+1.0;
QU10:=QU10+TIME-ENTITY[N]TIMQ;
COST:=COST+(TIME-ENTITY[N]TIMQ)* 1.0000;
ENTITY[N] EV14:=TIME+0.0 END;
EV15:=TIME;ENDEV;
```

```
EVENT EV15;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
IF NR(QUE15)>=500 THEN
BEGIN LD15:=LD15+1;
KILL THIS; ABAND END;
WAIT(QUE15); TIMQ:=TIME;
```

```
IF SER16 THEN BEGIN OUT(QUE15,N,T); IF N > 0 THEN  
BEGIN  
QN15:=QN15+1.0;
```

```
ENTITY[N]EV16:=TIME END END;  
ENDEV;
```

```
EVENT EV16;
```

```
BEGIN
```

```
IF NOT SER16 THEN BEGIN  
LSER16:=LSER16+1; KILL THIS; ABAND END;  
SER16:=FALSE; SV16:=TIME;
```

```
SVOP(16,SOURCE,TIMIN,TIMQ,JD);  
ENDS16:=TIME+DIS4(SOURCE,JD);ENDEV;
```

```
EVENT ENDS16;
```

```
VAR N:INTEGER; BEGIN
```

```
SER16:=TRUE; NSER16:=NSER16+1; COST:=COST+ 1.0000;  
STM16:=STM16+TIME-SV16;
```

```
IF NR(QUE15)=0 THEN BEGIN
```

```
OUT(QUE15,N,T);
```

```
QN15:=QN15+1.0;
```

```
QU15:=QU15+TIME-ENTITY[N].TIMQ;  
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
```

```
ENTITY[N]EV16:=TIME+0.0 END;
```

```
EV25:=TIME;ENDEV;
```

```
EVENT EV25;
```

```
VAR KSUC:INTEGER;
```

```
BEGIN
```

```
KSUC:=DIVID(SOURCE);
```

```
IF KSUC=1 THEN EV23:=TIME;
```

```
IF KSUC=2 THEN EV24:=TIME;
```

```
IF KSUC=3 THEN EV26:=TIME;
```

```
ENDEV;
```

```
EVENT EV23;
```

```
BEGIN
```

```
TER23:=TER23+1;
```

```
SVOP(23,SOURCE,TIMIN,TIMQ,JD);
```

```
TSS(TIMIN); KILL THIS;
```

```
ENDEV;
```

```
EVENT EV24;
```

```
BEGIN
```

```
TER24:=TER24+1;
```

```
SVOP(24,SOURCE,TIMIN,TIMQ,JD);
```

```
TSS(TIMIN); KILL THIS;
```

```
ENDEV;
```

```
EVENT EV26;
```

```
BEGIN
```

```
TER26:=TER26+1;
```

```
SVOP(26,SOURCE,TIMIN,TIMQ,JD);
```

```
TSS(TIMIN); KILL THIS;
```

```
ENDEV;
```

```
EVENT EV12;
```

```
BEGIN
```

```
IF NOT SER12 THEN BEGIN
```

```
LSER12:=LSER12+1; KILL THIS; ABAND END;
```

```
SER12:=FALSE; SV12:=TIME;
```

```
SVOP(12,SOURCE,TIMIN,TIMQ,JD);
```

```
ENDS12:=TIME+DIS3(SOURCE,JD);ENDEV;
```

```
EVENT ENDS12;
```

```
VAR N:INTEGER; BEGIN
```

```
SER12:=TRUE; NSER12:=NSER12+1; COST:=COST+ 1.0000;
```

```
STM12:=STM12+TIME-SV12;
```

```
IF NR(QUE10)=0 THEN BEGIN
```

```
OUT(QUE10,N,T);
```

```
QN10:=QN10+1.0;
```

```
QU10:=QU10+TIME-ENTITY[N].TIMQ;
```

```
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
```

```
ENTITY[N]EV14:=TIME+0.0 END;
```

```
EV15:=TIME;ENDEV;
```

```
EVENT EV13;
```

```
BEGIN
```

```
IF NOT SER13 THEN BEGIN
```

```
LSER13:=LSER13+1; KILL THIS; ABAND END;
```

```
SER13:=FALSE; SV13:=TIME;
```

```
SVOP(13,SOURCE,TIMIN,TIMQ,JD);
```

```
ENDS13:=TIME+DIS3(SOURCE,JD);ENDEV;
```

```
EVENT ENDS13;
```

```
VAR N:INTEGER; BEGIN
```

```
SER13:=TRUE; NSER13:=NSER13+1; COST:=COST+ 1.0000;
```

```
STM13:=STM13+TIME-SV13;
```

```
IF NR(QUE10)=0 THEN BEGIN
```

```
OUT(QUE10,N,T);
```

```
QN10:=QN10+1.0;
```

```
QU10:=QU10+TIME-ENTITY[N].TIMQ;
```

```
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
```

```
ENTITY[N]EV14:=TIME+0.0 END;
```

```
EV15:=TIME;ENDEV;
```

```
EVENT EV5;
```

```
BEGIN
```

```
IF NOT SER5 THEN BEGIN
```

```
LSER5:=LSER5+1; KILL THIS; ABAND END;
```

```
SER5:=FALSE; SV5:=TIME;
```

```
SVOP(5,SOURCE,TIMIN,TIMQ,JD);
```

```
ENDS5:=TIME+DIS1(SOURCE,JD);ENDEV;
```

```
EVENT ENDS5;
```

```
VAR N:INTEGER; BEGIN
```

```
SER5:=TRUE; NSER5:=NSER5+1; COST:=COST+ 1.0000;
```

```
STM5:=STM5+TIME-SV5;
```

```
IF NR(QUE3)=0 THEN BEGIN
```

```
OUT(QUE3,N,T);
```

```
QN3:=QN3+1.0;
```

```
QU3:=QU3+TIME-ENTITY[N].TIMQ;
```

```
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
```

```
ENTITY[N]EV7:=TIME+0.0 END;
```

```
EV8:=TIME;ENDEV;
```

```
EVENT EV6;
```

```
BEGIN
```

```
IF NOT SER6 THEN BEGIN
```

```
LSER6:=LSER6+1; KILL THIS; ABAND END;
```

```
SER6:=FALSE; SV6:=TIME;
```

```
SVOP(6,SOURCE,TIMIN,TIMQ,JD);
```

```
ENDS6:=TIME+DIS1(SOURCE,JD);ENDEV;
```

```
EVENT ENDS6;
```

```
VAR N:INTEGER; BEGIN
```

```
SER6:=TRUE; NSER6:=NSER6+1; COST:=COST+ 1.0000;
```

```
STM6:=STM6+TIME-SV6;
```

```
IF NR(QUE3)=0 THEN BEGIN
```

```
OUT(QUE3,N,T);
```

```
QN3:=QN3+1.0;
```

```
QU3:=QU3+TIME-ENTITY[N].TIMQ;
```

```
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
```

```
ENTITY[N]EV7:=TIME+0.0 END;
```

```
EV8:=TIME;ENDEV;
```

```
PROCESS GENER.1;
```

```

EVENT GEN1;
VAR K:INTEGER; BEGIN
GEN1:=TIME+0.1;

FOR K:=1 TO 1 DO BEGIN
IF COUN1>1264 THEN ABAND;
NEWPR PENTITY;
COUN1:=COUN1+1;PENTITY.SOURCE:=1;

SVOR1,PENTITY.SOURCE,PENTITY.TIMIN,PENTITY.TIMO,
PENTITY.ID;
PENTITY.EV3:=TIME;
PENTITY.TIMIN:=TIME END; ENDEV;

EVENT FINAL;
VAR N:INTEGER; BEGIN
FORALL N ENTITY BEGIN KILL ENTITY[N] ENDALL;
FORALL N GENER BEGIN KILL GENER[N] ENDALL;
KILL PPRNT; FINST A,ENDEV;

EVENT FINN;
BEGIN
FINSTA ENDEV;

EVENT COSTFN;
VAR J:INTEGER; BEGIN
FORALL J IN QUE3 BEGIN
COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
FORALL J IN QUE8 BEGIN
COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
FORALL J IN QUE10 BEGIN
COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
FORALL J IN QUE15 BEGIN
COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
ENDEV;

EVENT GEN2;
VAR K:INTEGER; BEGIN
GEN2:=TIME+0.1;

FOR K:=1 TO 1 DO BEGIN
IF COUN2>1474 THEN ABAND;
NEWPR PENTITY;
COUN2:=COUN2+1;PENTITY.SOURCE:=2;

```

```

SVOR2,PENTITY.SOURCE,PENTITY.TIMIN,PENTITY.TIMO,
PENTITY.ID;
PENTITY.EV3:=TIME;
PENTITY.TIMIN:=TIME END; ENDEV;

```

```

PROCESS PRNT J;
ATR FIR TIM:BOOLEAN;

```

```

EVENT PR1;
VAR L,NQX:INTEGER; BEGIN
IF TIME+HPR<TFIN THEN PR1:=TIME+HPR;
REPQMO(FIR TIM);
VARN[1]:=-QUEUE3; VARN[2]:=-QUEUE8;
VARN[3]:=-QUEUE10;
VARN[4]:=-QUEUE15; VARN[5]:=-QUEUE;
VARN[6]:=-QUEUE;
VARN[7]:=-QUEUE; VARN[8]:=-QUEUE; VARN[9]:=-QUEUE;
VARN[10]:=-QUEUE; KX:=KX+1; IF KX<=100 THEN BEGIN
X1,KX):=-NR(QUE3); X2,KX):=-NR(QUE8);
X3,KX):=-NR(QUE10); X4,KX):=-NR(QUE15);
END;ENDEV;

```

```

START
INIT(TRUE);NQQUE3; NQUE QUE8; NQUE QUE10; NQUE
QUE15;
POW:=KPOW+1;
INQUE;
ININ;
REPSIM KPOW TFIN
REIN;
NEWPR PGENER;
PGENER.GEN1:=0*REPS;
PGENER.GEN2:=12*REPS;
PGENER.COSTFN:=TFIN-EPS;
PGENER.FINN:=TFIN+0.5*EPS;
NEWPR PPRNT; PPRNT.PR1:=2*EPS;
PPRNT.FIR TIM:=TRUE;
RESETS
$

```

A continuación se describe el archivo de procedimientos DIS1.pas, que comprende todas las funciones de distribución de los servidores del modelo, además de las reglas lógicas principales de distribución de flujo de material del modelo en su versión de PASION:

DIS1.PAS

```

funcion dis1(k:integer;id INTEGER) real;
Var x:real;
begin
if k=1 then x:=0.48;
if k=2 then x:=0.40;
if (id=1) or (id=2) or (id=3) then
BEGIN
(WRITELN (ID=,ID,CON=,K,);
x:=3*x
END;
dis1:=x
end;

funcion dis2(soucor:integer;id INTEGER) real;
Var x:real,y:integer;
begin
if soucor=1 then x:=0.09;
if soucor=2 then x:=0.09;
if id=1 then

```

```

BEGIN
(WRITELN (ID=,ID,CON=,SOURCE,DTRB=,2);
x:=4.612*x
END;
y:=soucor;
dis2:=x
end;

funcion dis3(k:integer;id INTEGER) real;
Var x:real;y:integer;
begin
if k=1 then x:=0.37;
if k=2 then x:=0.37;
if (id=1) or (id=2) or (id=3) then x:=10.277*x;
y:=k;
dis3:=x
end;

funcion dis4(k:integer;id INTEGER) real;
Var x:real;y:integer;
begin
if k=1 then x:=0.08;

```

```

if t=2 then x:=0.08;
if id=1 then x:=4.71+x;
y:=k;
dist:=x;
end;

function divid(source:integer):integer;
var xreal;Y:INTEGER;
begin
x:=random;
if (x<0.95) and (source=1) then Y:=1;
if (x<0.95) and (source=2) then Y:=2;
if x>=0.95 then Y:=3;
DIVID:=Y
end;

function arri(v:integer):real;

```

```

var xreal;gen1,gen2:integer;
begin
x:=0;

if (s=1) and (gen1<1264) then
begin
gen1:=gen1+1;
x:=0.03
end;

if (s=2) and (gen2<1474) then
begin
gen2:=gen2+1;
x:=0.03
end;
arriv:=x;
end;

```

2.4 Conclusiones.

2.4.1 Conclusiones sobre el caso presentado.

Una vez realizadas las simulaciones en Xcell+ y en PASION, es importante la comparación de los resultados arrojados en ambos programas. Para efectuar esta comparación con justicia, se tomó finalmente el modelo que mejor soluciona el caso presentado. Este modelo es una variante del sencillo, pero que considera que los inventarios entre los centros de trabajo tienen capacidad suficiente para amortiguar los efectos de los tiempos de preparación.

Cuando se considera esta opción, resulta que en realidad la tercera célula sí tiene la capacidad suficiente como para satisfacer la demanda que se necesita, incluso un poquito más. Es importante saber aprovechar el efecto de amortiguamiento entre los centros de trabajo colocando inventarios reales entre ellos. La decisión es: o cero inventario o llegar a la meta de producción. Cuando la meta de producción es importante se pueden soportar pequeños inventarios temporales en el proceso de producción. Estos inventarios desaparecerán en cada ciclo, de hecho su existencia es momentánea.

Este es pues el modelo que se compara, en igualdad de situaciones de Xcell+ y PASION. Hay que tomar en cuenta que cada programa arroja los resultados en distintos formatos. Los resultados resumidos en tablas de Xcell+ y PASION se muestran a continuación.

En primer lugar los de Xcell+:

Resultados en la fábrica C_PROD

Tabla 2.5 UTILIZACION de Centro de trabajo a tiempo 4800

| | %ocupación | | %En preparación | | %mantenimto | | %bloqueado | |
|-----|------------|-----------|-----------------|-----------|-------------|--------|------------|-----------|
| | periodo | acumulado | periodo | acumulado | periodo | acmld. | periodo | acumulado |
| ASI | 83.654 | 83.797 | 2.083 | 1.979 | 0 | 0 | 0.373 | 0.298 |
| RV1 | 52.933 | 53.046 | 1.938 | 1.842 | 0 | 0 | 2.001 | 1.601 |
| W3 | 70.239 | 70.452 | 4.282 | 4.068 | 0 | 0 | 3.657 | 2.926 |
| W10 | 83.658 | 83.797 | 2.083 | 1.979 | 0 | 0 | 0.36 | 0.288 |
| W11 | 83.666 | 83.8 | 2.083 | 1.979 | 0 | 0 | 0.36 | 0.288 |
| W12 | 70.263 | 70.475 | 4.282 | 4.068 | 0 | 0 | 3.636 | 2.909 |
| W8 | 45.617 | 45.72 | 1.963 | 1.865 | 0 | 0 | 5.415 | 4.423 |
| W13 | 70.33 | 70.536 | 4.282 | 4.068 | 0 | 0 | 3.615 | 2.892 |

Resultados en la fábrica C_PROD

Tabla 2.6 Flujo de material en el AREA de Embarque al tiempo: 4800

| | Salida | | almacenado | |
|-----------------|---------|-----------|------------|-----------|
| | periodo | acumulado | periodo | acumulado |
| SP2 | 11200 | 14051 | 0 | 0 |
| SS | 1093 | 1373 | 0 | 0 |
| SP1 | 9601 | 12006 | 0 | 0 |
| totales: | 21896 | 27432 | 0 | 0 |

Tabla 2.7

Productos en proceso al tiempo: 4800

| | actual | mínimo | | máximo | | promedio | |
|----------------|--------|---------|-----------|---------|---------|----------|-----------|
| | | periodo | acumulado | periodo | acmldo. | periodo | acumulado |
| B1 | 50 | 0 | 0 | 50 | 50 | 1.442 | 1.251 |
| B2 | 100 | 0 | 0 | 100 | 100 | 15.093 | 13.512 |
| B3 | 50 | 0 | 0 | 50 | 50 | 3.791 | 3.206 |
| RP1 | 2 | 0 | 0 | 2 | 2 | 1.391 | 1.444 |
| RP2 | 1 | 0 | 0 | 2 | 2 | 1.178 | 1.222 |
| SP2 | 51 | 0 | 0 | 1450 | 1450 | 636.11 | 625.863 |
| SP1 | 6 | 0 | 0 | 1206 | 1206 | 1146.3 | 1194.602 |
| totales | 260 | | | | | 1805.3 | 1841.099 |

Resultados en la fábrica C_PROD

Tabla 2.8

Tiempo: AREA de embarque al tiempo: 4800

de FLUJO:

(En base al número de unidades marcadas que llegan al área de embarque)

| | número | flujo promedio | procesamiento promedio | espera promedio |
|-----|--------|----------------|------------------------|-----------------|
| SP2 | 111 | 4.541 | 0.942 | 3.599 |
| SS | 9 | 1.796 | 0.959 | 0.837 |
| SP1 | 99 | 3.994 | 1.022 | 2.972 |

En segundo lugar los resultados de PASION. A continuación se muestra la tabla de resultados para las líneas de espera. La cola número 3 representa al almacén de materia prima, que en el modelo de XCELL+ no existe. Este almacén es una necesidad en PASION, de manera que se incluye para la debida comparación de los resultados entre ambos modelos.

Estadísticos finales a 20 ciclos.

Tabla 2.9

Tiempo final: 480 u.t.

| Colas | Tpo. Prom. de espera | L max | L min | Longitud Promedio | Pérdidas |
|--------|----------------------|-------|-------|-------------------|----------|
| Cola3 | 71.003 | 988 | 0 | 405.259 | 0 |
| Cola8 | 0.024 | 29 | 0 | 0.055 | 0 |
| Cola10 | 0.246 | 109 | 0 | 1.396 | 0 |
| Cola15 | 0.009 | 18 | 0 | 0.042 | 0 |

Ahora los resultados para los servidores del modelo, hay que recordar que los servidores 4,5,6 así como los 11,12,13 representan máquinas de la misma clase, por esto sus datos son muy parecidos. Se puede observar que por el servidor 9 y por el 16 pasan todas las piezas que se producen en la línea.

Tabla 2.10

| Servidores | Tpo. de Srv. | Tpo. de Espera | Espera % | Pérdidas | Piezas Servidas |
|------------|--------------|----------------|----------|----------|-----------------|
| SERV 4 | 399.612 | 80.388 | 16.75 | 0 | 913.4 |
| SERV 5 | 399.392 | 80.608 | 16.79 | 0 | 912.95 |
| SERV 6 | 399.336 | 80.664 | 16.81 | 0 | 912.9 |
| SERV 9 | 246.994 | 233.007 | 48.54 | 0 | 2740 |
| SERV 11 | 338.986 | 141.015 | 29.38 | 0 | 913.4 |
| SERV 12 | 338.838 | 141.163 | 29.41 | 0 | 913 |
| SERV 13 | 338.782 | 141.218 | 29.42 | 0 | 912.85 |
| SERV 16 | 219.603 | 260.397 | 54.25 | 0 | 2740 |

Finalmente los datos de estado final de los almacenes de producto terminado y las colas. Como da perfectamente bien el tiempo de terminar todo, debido a la capacidad en los inventarios para amortiguar las variaciones, los inventarios finalmente quedan vacíos.

Tabla 2.11

Tiempo promedio en el sistema: 74.24

| TER | Tipo | Cola | 0 | Cola | 0 |
|--------|------|-------|---|--------|---|
| TER 23 | 1210 | Cola3 | 0 | Cola10 | 0 |
| TER 24 | 1399 | Cola8 | 0 | Cola15 | 0 |
| TER 26 | 131 | | | | |

Se puede comprobar que se llega perfectamente a las metas de producción fijadas, en TER 23 se tienen las piezas de tipo 1 y en TER 24 las de tipo 2. TER 26 representa el 5% de piezas que se desecharon por errores en su producción.

2.4.2 Conclusiones sobre la Simulación.

Una vez que se ha conseguido con cierto éxito la simulación del problema modelado, es hora de reflexionar acerca de los resultados de la técnica en sí. Es decir, hay varias preguntas que me gustaría responder:

¿Fue la Simulación la mejor manera de resolver este problema, resolvió realmente el problema?

¿Son proporcionados los resultados obtenidos con el esfuerzo realizado?

¿Cuál de los dos paquetes aplicados fue más sencillo de utilizar, como fueron los resultados obtenidos?

Seguramente el lector añadiría a estas preguntas cuestiones adicionales, sin embargo, me parece que en estas tres breves preguntas se resume el centro de la cuestión.

¿Fue la Simulación la mejor manera de resolver este problema, resolvió realmente el problema?

Bien, respecto a la primera pregunta se puede decir que es realmente el meollo de la tesis. En el presente trabajo se intenta mostrar como es que la Simulación es la mejor manera de resolver los problemas complicados o complejos. Si bien no es siempre una forma sencilla, existe una gran cantidad de problemas cuya solución y análisis no son de ninguna manera sencillos. Para el problema en cuestión se demostró que el análisis técnico admitía la posibilidad de que la planta realmente llegara a la meta establecida, pero se vio que en la realidad, debido a las restricciones lógicas del problema, no iba a ser posible a menos que se corrigieran los tiempos de preparación y se aumentara el inventario entre las estaciones de trabajo. Claro que para llegar a esta conclusión se tuvieron que probar un gran número de hipótesis distintas, que si bien encerraban grandes innovaciones lógicas, los cambios en los modelos formulados tanto en XCELL+ como en PASION, no eran tan difíciles. Una vez que se tiene el modelo

base de Simulación Ingresado en algún paquete, es relativamente fácil jugar con variables y reglas para llegar a un modelo óptimo. La parte más difícil de la Simulación es la formulación del modelo, y la respectiva programación. La parte más sencilla es el análisis de opciones "what if?". Sin embargo es una enorme ventaja tener el lujo de analizar otras muchas opciones a un modelo dado.

Si la pregunta es si la Simulación sirvió en este caso en particular, la respuesta es sí. No sólo sí, sino que encontramos la solución óptima al problema presentado, una solución que no es visible desde un punto de vista teórico. Me parece que claramente, el esfuerzo valió la pena.

¿Son proporcionados los resultados obtenidos con el esfuerzo realizado?. Me parece que, honestamente, el esfuerzo realizado en este modelo no es muy grande. Esto es, el modelo no tiene una complejidad desmedida ni tampoco un tamaño exagerado. Esto se consigue formulando correctamente las cuestiones que nos interesa analizar. Para qué añadir complejidad y detalle al problema si esto no beneficia en nada al problema final, ni tampoco al análisis. Es importantísimo formular modelos que sean óptimos en dimensión y detalle. Esta es tarea de la gente especialista en Simulación y que tiene experiencia. Pienso, en resumen, que el trabajo fue relativamente poco comparado con la gran cantidad de información "segura" que se obtuvo como resultado.

¿Cuál de los dos paquetes aplicados fue más sencillo de utilizar, como fueron los resultados obtenidos?. Bueno, esta es una pregunta muy concreta en la que se pide un análisis comparativo. Yo diría en este problema XCELL+ ofrece la gran ventaja de que es más sencillo de utilizar con modelos que caben dentro de su estructura logística de posibilidades, sin embargo fuera de esta estructura, las cosas se vuelven imposibles. PACION tiene la característica de ser un programa que no está enfocado solamente a sistemas discretos de producción, sus esquemas son mucho más grandes y generales y se puede hacer prácticamente cualquier diseño en su estructura (claro, sabiendo PASCAL).

Basado en lo anterior, puedo concluir que, para el caso concreto, la Simulación, por su grado de complejidad, es más inmediata en XCELL+ que en PACION. Esto no significa que un paquete sea mejor que otro, ni mucho menos, simplemente es un ejemplo de aquel dicho mexicano: "A las moscas chicas se les mata con matamoscas chicos".

Capítulo 3: Balanceo de líneas de producción.

Introducción¹.

Desde 1970 hasta 1985 MRP (incluyendo tanto Planeación de Requerimientos de Material, como la Planeación de los Recursos de Manufactura) fue tomada como el camino que había que recorrer por parte de las empresas de manufactura. Si las compañías no recibirán los beneficios prometidos, concluimos que se debía a que tenían problemas con sus propias organizaciones. Esa era la pena de no estar preparados para implementar MRP, y no era problema de la técnica. Al paso de los años, en vez de alcanzar a los Japoneses en su tremendo camino de conquista de los mercados mundiales, las industrias americanas continuaron fracaso tras fracaso. Mayores esfuerzos se pusieron para hacer del MRP una técnica exitosa, pero eventualmente las compañías se dieron cuenta de que no era la solución de sus problemas, al menos de todos sus problemas, y que el sistema en sí mismo tenía limitaciones inherentes.

Todos los que en los Estados Unidos analizaron las razones posibles del éxito Japonés en la manufactura, citaron una variedad de causas: las diferencias culturales, círculos de calidad, participación del gobierno, una cultura de empleos para toda la vida. Todas estas explicaciones, que ahora conocemos, ignoraban una simple realidad: el sistema Japonés de manufactura JIT (Justo a Tiempo) y de control de sistemas estaban haciendo un trabajo excelente en las áreas de manufacturas repetitivas. Finalmente, a mediados de 1980, las empresas americanas comenzaron a copiar a los japoneses. Para completar la realización de su trabajo implementaron JIT en todas las plantas donde pudieron. Sin embargo, lo que ahora falló, fue el no entender que la implementación de la técnica de JIT es un proceso lento, que se realiza esencialmente mediante prueba y error. Esta no es una técnica adecuada si la meta es alcanzar y pasar a los Japoneses, y entonces recapturar los mercados perdidos.

Alrededor de 1980, el Dr. Eliyahu Goldratt, diseñó un programa que esquematizaba los trabajos de los procesos de manufactura, tomando en cuenta los que recursos son limitados: máquinas, personal, herramientas, materiales, y todas las demás variables que afectan la posibilidad de que la empresa funcione según un esquema rígido. El sistema tenía que ser confiable y correcto, y debería de correr en la computadora en una fracción del tiempo que le tomaba al sistema MRP. Esta es la razón de que el esquema lógico estuviera basado en la identificación del cuello de botella y los demás centros de trabajo. Después de que, aproximadamente, 100 compañías grandes habían instalado este programa, Goldratt y Robert Fox pasaron a promover la lógica de funcionamiento más que el programa en sí. Goldratt desde entonces formuló su "Teoría General de Restricciones", y, de hecho, ya no está muy asociado con la compañía de software.

¹R. CHASE & N. AQUILANO *Production and Operations Management: a Life Cycle Approach*, 2a edición, IRWIN DE 1992

En lo personal, creo en la lógica de Goldratt para entender la manufactura y enfocarla, esto es, acabar con las restricciones. Por esto me pareció conveniente incluir este capítulo en el trabajo presente, ya que la técnica de Goldratt debe ser una de las más usadas en el futuro de la planta productiva Mexicana.

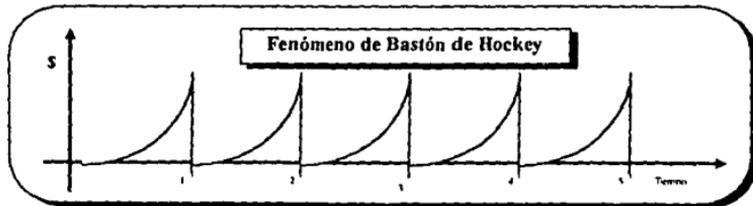
En este capítulo se discute básicamente lo que es la Manufactura Sincronizada. La **Manufactura Sincronizada** se refiere a conseguir que todos los procesos de producción trabajen armónicamente para conseguir las metas de la compañía. Cualquiera de los sistemas de producción consiste en una gran variedad de ciclos con distintos tiempos de duración. La lógica de sincronización de la producción intenta coordinar todos estos ciclos de manera que funcionen juntos y armónicamente, esto es, "sincronizados". En un estado de sincronía, se pone principalmente la atención en el diseño del sistema en su conjunto y no en partes particulares ni en su trabajo o utilización de la maquinaria.

3.1 Teoría Restricciones: Balanceo de Líneas de Producción².

3.1.1 Introducción.

Cada vez que una compañía enfrenta un problema, se parece al Fenómeno de Stick de Hockey³. Cuando se acerca el final del plazo para entregar un trabajo, un reporte, o dar cuentas de algo, todo mundo corre para alcanzar la meta propuesta. Si el período de tiempo es una semana parece el síndrome de fin de semana, si es de quince días aparece el síndrome de quincena, etc. Se le llama de bastón de Hockey pues la gráfica de los niveles de presión para los empleados tiene esa forma; baja al principio pero enorme al final. Una de las razones que se puede encontrar para explicar este fenómeno es la que nos ofrece la teoría del Caos: el caos sucede al final de la semana. Los procesos de entrega nunca se desarrollan de manera suave, todos trabajamos bajo presión mientras dure la recta final del proyecto de entrega. La causa del problema es básicamente humana: cuando se comienza la semana o el período de producción, uno se fija en hacer muy bien las cosas, en ahorrar (se mide y controla la calidad, variancias y costos locales de producción), pero cuando se acerca el final de período uno se preocupa principalmente por terminar el trabajo (se mide la cantidad total de producción y lo que significa en utilidades para la empresa). Es la diferencia entre la visión del contador, que cuida cada peso, y la del analista financiero de la empresa, que busca las utilidades. Se entiende que cuando tenga tiempo haga las cosas despacio, bien hechas y baratas, pero cuando se trata de terminar, de dar resultados, hay que hacerlo a toda costa. Una vez que pasa el cierre de ciclo, las cosas se tranquilizan, se vuelve a tener tiempo, y la presión se relaja. El ciclo vuelve a comenzar.

Figura 3.1



Para enfatizar el valor de la técnica de la Manufactura Sincronizada, Goldratt cita dos casos. El primero trata de una empresa que sufre un bajón de mercado. Los trabajadores son muy buenos y escasos, de manera que decide mantenerlos porque el director espera que en tres semanas el mercado se recupere. El director estima que mantener a los trabajadores tres semanas le cueste \$300,000. Pero los

²E. GOLDRATT *Computerized Shop Floor Scheduling*, International Journal of Production Research 26, no. 3 (1988), pp 443-55

³Nota: un stick de hockey es el bastón que se utiliza para jugar este deporte, ya sea en hielo o en pasto. Por cierto, es un deporte muy de moda en nuestros tiempos.

trabajadores al verse en riesgo de ser despedidos trabajan de una manera increíble, y tan sólo en inventarios acumulados le cuestan \$400,000 cada una de las dos primeras semanas. Hay demasiada presión y se sienten que tienen que estar trabajando, no conciben estar solamente sentados. Dos semanas más tardes la demanda aumenta pero no para los productos en los que se ha estado trabajando.

El segundo caso trata de una compañía que tiene problemas de flujo de efectivo, por lo que deciden despedir a algunos de los trabajadores. Como los trabajadores mejor pagados son los que se encargan de la preparación de las máquinas, deciden doblar el tamaño de los lotes de producción, de manera que se disminuyan las veces en que una máquina tendrá que ajustarse, y despedir a la mitad de los trabajadores que preparan las máquinas. El resultado es que la empresa quiebra. Doblando el tamaño de los lotes, el incremento de los inventarios de productos en proceso, WiP, consume todo el dinero de la compañía, de manera que no puede operar a largo plazo.

De esta forma, Goldratt plantea la principal idea que se tiene que mantener en un negocio, la meta del negocio: **"La meta de una compañía es hacer dinero"**. Una organización puede tener muchos propósitos: generar empleos, transformar materiales, incrementar ventas, hacer más grande el mercado, implementar tecnología, producir con una alta calidad, etc., pero todos estos propósitos no garantizan la supervivencia a largo plazo de la compañía. Si una empresa hace dinero, y sólo entonces, prosperará, o al menos, tendrá oportunidad de hacerlo. Una vez que una compañía hace dinero entonces puede hacer énfasis en otros objetivos.

3.1.2 El diseño de los Sistemas de Medición y Control según TR.

Para trabajar con un sistema de control y mediciones más eficaz dentro de un empresa, y que a la vez, se acople con la meta planteada para la compañía, hay que distinguir dos maneras de medir, cada una según un punto de vista: el financiero y el de operaciones.

El punto de vista financiero trata sobre ¿qué es lo que se va a hacer para ganar dinero?. En términos financieros se enfoca en:

1. Beneficio Neto, medida absoluta en dólares.
2. Retorno sobre la inversión, que es el punto de vista del inversionista.
3. Flujo de caja, que cuida la supervivencia de la empresa.

Las tres cosas son simultáneamente importantes. Se dice que una empresa gana mucho dinero, respecto al porcentaje que representa sobre la inversión realizada. Un buen negocio es aquel que gana mucho pero se invierte poco. El flujo de caja es necesario para saber si podemos o no afrontar los compromisos a corto plazo que tenemos con nuestros acreedores, sin ellos, no podremos hacer ningún negocio.

Mientras las medidas financieras funcionen bien a nivel directivo, no tendrán que aplicarse o influir en un nivel de operaciones, de manera extraordinaria. En el nivel de operaciones necesitamos otro tipo de instrumentos que nos den una pauta:

1. Producción, esto es, la tasa en la que el sistema genera dinero a través de ventas.
2. Inventarios: todo el dinero que el sistema tiene invertido en comprar cosas que intenta vender.
3. Costos de Operación: Todo el dinero que el sistema necesita para convertir el inventario en producto terminado.

La meta de los instrumentos de medición de las operaciones debe de ser para la compañía: **"Incrementar la producción mientras, simultáneamente, se reducen los inventarios y se reducen los costos de operación"**.

Típicamente la productividad se define como la salida de productos terminados por tiempo de trabajo empleado. Es claro que ésta no es la manera en que la compañía hará dinero. Para saber si nuestra lucha debe centrarse en ser más productivos, hay que responder a las siguientes preguntas: ¿Esta acción hará crecer mi producción?, ¿hará que disminuya mi inventario?, ¿hará que disminuyan mis

costos de operación?. Éstas respuestas nos dan una nueva definición de lo que significa para Goldratt la productividad: *"Productividad es toda aquella acción que coloque a la compañía más cerca de sus metas"*.

3.1.3 Balanceo de Capacidades de Producción.

Históricamente, todas las empresas de manufactura, cuando trabajan mediante líneas de producción, tratan de balancear las capacidades de las líneas según la demanda de productos en el mercado. Cuando esta idea se traduce, en el intento de que cada una de las máquinas que forma una determinada cadena, tengan la misma capacidad de producción, se comete un error grave. Aunque usted no lo crea, es mejor tener las líneas desbalanceadas, o más bien, balanceadamente desbalanceadas. Por ejemplo, cuando se forma una nueva cadena de producción, digamos de cuatro máquinas, los obreros instintivamente intentan balancear las capacidades de producción, ajustando las máquinas o el equipo usado, los métodos de carga, etc.

En la Manufactura Sincronizada se piensa que igualar las capacidades de producción de las máquinas de una cadena es una mala decisión. Un verdadero balanceo sólo podrá conseguirse, según este método, si las máquinas tienen un nivel de producción constante o poco variable. Esto sucede pocas veces en la realidad. Usualmente las máquinas tienen una distribución normal como nivel de producción, de manera que si las máquinas se igualan en capacidades, el efecto de variación de cada una de las máquinas se irá acumulando en la siguiente, de manera que al final se obtenga un verdadero caos. La única manera de amortiguar este efecto acumulativo de variaciones, es colocando inventarios entre las máquinas (esto parece contrario a nuestra idea de disminuir inventarios), o incrementar la capacidad de manera que se detenga la cadena el menor número de veces. La idea principal de esto es que las capacidades de las máquinas de una cadena de producción, no deben de ser balanceadas, lo que se debe de balancear es el flujo del producto dentro de la cadena. *"Cuando el flujo se balancea, las capacidades se desbalancean"*.

3.1.4 Cuellos de botella, y capacidad de los recursos restringidos.

Un cuello de botella se define como aquel recurso cuya capacidad es menor que la demanda que se le pide. Un cuello de botella, en otras palabras, es el proceso que limita la producción. Es el punto del proceso de manufactura donde el flujo adelgaza a causa de un conducto estrecho. Los cuellos de botella pueden ser máquinas, labores especializadas o herramientas especializadas. Las observaciones en la industria demuestran que la mayoría de las plantas tienen muy pocos cuellos de botella, siempre son sólo unos cuantos.

Capacidad se define como el tiempo disponible para producción. Este excluye mantenimientos y otros tiempos desperdiciados. Una operación que no es un cuello de botella, es un recurso cuya capacidad es mayor que la demanda que se le pide. Por tanto, una operación que no es cuello de botella, no tendrá que estar funcionando constantemente, pues puede producir más de lo que se necesita. Además tiene, por consecuencia, tiempo de ocio.

Un recurso con capacidad restringida, CCR¹, es aquel recurso que está trabajando cerca de su límite de capacidad y se puede convertir en un cuello de botella si no se planea su uso cuidadosamente. Es un cuello de botella en potencia activa.

3.1.5 Componentes del tiempo.

Los siguientes tipos de medición de tiempo se pueden encontrar en todos los ciclos de producción. Además, en la presente tesis, se ha hecho y hará uso de estos conceptos, por lo que es mejor definirlos de una buena vez:

¹N.T. del inglés "Capacity-Constrained Resource".

1. **Tiempo de Preparación:** Es el tiempo en que una parte espera en un centro de trabajo, mientras éste prepara el proceso requerido.
2. **Tiempo de proceso:** Es el tiempo que la parte es procesada en un centro de trabajo.
3. **Tiempo de cola:** Es el tiempo que una parte espera por un proceso mientras éste está ocupado trabajando en otra pieza.
4. **Tiempo de espera:** Es el tiempo que la parte espera no por un proceso ni recurso, sino por otra parte a la que necesita ser ensamblada.
5. **Tiempo de ocio:** Es el tiempo no usado por un centro de trabajo, esto es, el tiempo de ciclo menos la suma de el tiempo de preparación, el tiempo de proceso, el tiempo de cola, y el tiempo de espera.

En general, para una parte que pasará por un cuello de botella, el tiempo de cola es el mayor. Esto se debe principalmente a que es justamente frente al cuello de botella, donde se forman las colas más grandes de productos en proceso, WIP. Esta es una señal clara que se deriva de la intención de que el recurso más escaso siempre tenga trabajo. Para una parte que pasará por operaciones que no son cuellos de botella, el tiempo de espera puede ser el mayor.

3.1.6 Localización del cuello de Botella.

Existen dos formas de encontrar los cuellos de botella de una planta. Una es haciendo la simulación de la planta y analizar los resultados estadísticos. La otra es visitar la planta real, platicar con los operadores, y recorrer los pasillos. El perfil de los recursos de capacidad se obtiene poniendo atención en los lugares de carga de cada recurso por el que los productos van a pasar. Cuando se corre un perfil de capacidad, asumimos que los datos son razonablemente exactos, aunque no necesariamente perfectos. Es posible que un análisis de los perfiles de capacidad, sea demasiado complicado, y que sus resultados sean poco confiables. En estos casos es mejor usar la simple experiencia (también llamada *Knowledge*), y fijamos dónde es que los inventarios se amontonan, utilizando la técnica VAT (por la forma en que los procesos se entrelazan entre sí), y tomar decisiones correctivas.

3.1.7 Ahorrando tiempo en los cuellos de botella, Inventarios y Enlaces entre las operaciones⁵.

Cuando hemos localizado un cuello de botella, esto es, un proceso que tiene una capacidad menor que la demanda que se le pide, usualmente se restringe los materiales que se le mandan. Como la capacidad del cuello de botella es menor que la demanda del mercado, ya que si se pudiera se produciría más, hay que poner nuestra atención en cómo hacer que el tiempo en el cuello de botella sea óptimo. Hay varias formas de lograr esto: mejorando las máquinas, mayor calidad en las operaciones, lotes de producción grandes, reducir los tiempos de preparación, etc. Es importantísimo que obtengamos mas capacidad en nuestro principal recurso limitado, pues este es el que marca el paso de nuestra producción. Una hora ahorrada en el cuello de botella añade una hora extra a la capacidad productiva de todo nuestro sistema, en cambio, una hora ahorrada en una actividad que no es cuello de botella es sólo un espejismo, y sólo añade una hora más de ocio en esa actividad.

Es importante notar que los cuellos de botella, y las operaciones que no lo son, pueden intercambiar papeles si los tamaños de los lotes de producción cambian. Cuando una actividad A tiene tiempos de preparación muy chicos y tiempos de procesos muy grandes, y otra operación B tiene tiempos de preparación muy grandes y de proceso muy chicos, entonces a la operación A le convendrá más hacer muchos lotes distintos y a B le perjudicará. Cuando los lotes sean pequeños B será cuello de botella por la gran inversión de los tiempos de preparación y cuando sean grandes A será el cuello de botella, pues será muy grande el tiempo de proceso empleado.

Cuando se tiene una línea de producción con cuellos de botella, es lógico que no entre en ésta más material que aquel que puede ser procesado por el cuello de botella. Si la línea se sobre carga, esto es, se llena de material en exceso, los inventarios y colas frente al cuello de botella crecerán, los WIP serán enormes. No se obtendrá ningún beneficio de producción y, al contrario, se perderá orden, espacio y dinero. Es claro que el ritmo al que deben de entrar los materiales en la línea de producción debe ser el

⁵ E. GOLDRATT & J. COX "La Meta: un proceso de mejora continua". Ediciones Castillo, primera edición 1984, primera edición mexicana 1991.

mismo al que puede trabajar el cuello de botella. Esto se puede decir también de la siguiente manera: "el cuello de botella lleva el ritmo de la producción, así como los tambores llevan el de la banda de música".

Cuando una línea de producción puede producir más de lo que la demanda del mercado necesita, se dice que no hay cuello de botella en el sistema. Se puede decir también que en realidad el cuello de botella se encuentra fuera del sistema de producción. En estos casos el ritmo puede fijarse con respecto a los recursos restringidos en capacidad, CCR, (aquellos que trabajan al borde de su capacidad y si no se les cuida y planifica bien pueden convertirse en cuellos de botella). Cuando tampoco existen éstos, el ritmo lo puede fijar cualquier elemento de la cadena productiva.

Cuando se localiza un cuello de botella, hay dos cosas que se deben hacer con él:

1. Colocar un pequeño inventario de material en proceso, WIP, frente al cuello de botella, de manera que estemos seguros de que siempre estará trabajando y nunca le faltará material que procesar. De esta manera sólo se detendrá cuando tenga que prepararse, o durante su mantenimiento.
2. Enlazar el cuello de botella con la primer actividad de la línea de producción de manera que sólo comience a procesar lo que el cuello de botella requiera. Esto es llevar el ritmo del cuello de botella. De esta manera se evita que entre al sistema más material que el que el cuello de botella puede procesar. Por ejemplo, cada vez que el cuello de botella toma un lote para procesar de su inventario, la primera operación de la línea acepta un lote más para procesar, siempre y cuando estos lotes sean equivalentes.

Una pregunta casi inmediata de lo anterior es: ¿qué tan grande debe de ser el inventario del cuello de botella?, tan grande como el cuello de botella lo necesite para mantenerse siempre ocupado. En general un buen número para empezar es $\frac{1}{4}$ de la carga actual del sistema.

Si el ritmo lo va a llevar un CCR, entonces serán necesarios dos inventarios, uno frente al CCR y otro al final de la línea con productos terminados. El inventario de productos terminados, al final de la línea, estará enlazado con el CCR, y el inventario del CCR estará enlazado con la primer actividad de la línea. De esta manera el recurso limitado en capacidad CCR, sólo trabajará cuando sea necesario, y la línea sólo trabajará cuando el CCR lo requiera. Cuando tenemos una línea de esta naturaleza se entiende que el mercado no puede tomar todo lo que la línea pueda producir, sólo hay que estar seguros de que cuando el mercado requiera producto, tendremos el suficiente a su disposición.

3.1.8 Importancia de la Calidad.

Un sistema MRP permite los rechazos de producción porque maneja lotes más grandes de los que en realidad necesitan. Un sistema JIT no puede tolerar ningún error en calidad, pues el éxito de JIT está basado en un sistema balanceado. Una parte defectuosa o deficiente puede ocasionar que todo el sistema JIT se resquebraje. La manufactura sincronizada, MS, en cambio, tiene exceso de capacidad en el sistema entero, excepto en el cuello de botella. Si una parte defectuosa se produce antes del cuello de botella sólo se pierde material. Porque el exceso de capacidad de las operaciones que anteceden el cuello de botella permite los errores. Sin embargo, para los cuellos de botella no existen los tiempos extra, de manera que hay que cuidar que el cuello de botella no procese algún material defectuoso pues sería perder miserablemente el tiempo más precioso de la línea. De manera que hay que poner un control de calidad justo antes del cuello de botella para garantizar que sólo trabajará en cosas útiles. De cualquier forma, hay que asegurar que después del cuello de botella no se arruine el material por algún otro factor.

3.1.8 El tamaño de Lote.

Cuando se trata de balancear líneas de producción cabe la discusión respecto a la producción continua: ¿el tamaño de lote debe ser uno o infinito?. Esta pregunta se me antoja por demás ociosa. Sin embargo me parece importante discutir si, cara al cuello de botella, es conveniente o inconveniente los lotes numerosos o reducidos.

Cuando hablamos de tiempo de preparación, debemos de entender, en este caso, tanto el tiempo de preparar una máquina para desempeñar una determinada función, como el tiempo que nos lleva transportar el material de una operación a otra. Los lotes de producción grandes requieren menos tiempos de preparación (preparación por pieza producida, pues preparan para todas a la vez) y por tanto pueden dedicar más de su tiempo a la producción. Para los recursos que son cuellos de botella los lotes de productos grandes son recomendables. Para los recursos que no son cuellos de botella, es decir, para los que de por sí tienen un exceso de capacidad, se les recomienda pequeños lotes de producción (para que se ocupen los tiempos de ocio en preparaciones), y así reducir el inventario en proceso de la línea.

Se puede distinguir en dos tipos de lotes dentro de una línea: los lotes de producción y los lotes de transportación. Por ejemplo, se pueden procesar lotes de 200 piezas, pero transportarlas cada 25 de una estación a otra, de manera que sólo se prepare a las máquinas una sola vez, pero no tengan que esperar a que se terminen las 200 en cada proceso. Un lote de transporte puede ser tan grande como uno de proceso, pero sería ilógico que fuera más grande.

Para finalizar, en el anexo 4 de la tesis se incluyen una discusión sobre el TOC frente a JIT y MRP, para todos los escépticos recomiendo vivamente su lectura. Por lo que a mí respecta paso a la materia de simulación presentando el siguiente caso.

3.2 Caso: Kyoto y Osaka, Ltd.

Cuando Akira Tsuruoka, un MBA recién graduado en EE.UU. regresó a Japón, se convirtió en un analista de operaciones en la empresa de electrónica en Kyoto que su padre y tío comenzaron hace 23 años. Su primer responsabilidad se centró en la administración de las líneas ensambladoras de teclados para computadoras.

3.2.1 La compañía.

La empresa manufacturera Kyoto y Osaka, Ltd., produce una gran cantidad de subcomponentes. Originalmente, la compañía produce partes para radios, toca cintas y televisores. A través de los últimos 10 años, la planta ha hecho énfasis en subcomponentes para microcomputadoras y accesorios relativos a ellas. Al final de último año, aproximadamente el 85% de su producción estaba relacionada con computadoras.

A pesar de que muchas de las plantas similares en EE.UU. y Japón están totalmente automatizadas, la empresa del padre de Akira es intensamente manual. Trescientos obreros de turno generan la producción de la planta. La mayoría de los empleados han estado con la compañía más de 10 años. Casi todos son mujeres.

La empresa opera un solo turno, seis horas al día, 250 días al año. Este esquema de trabajo fue instituido para que los trabajadores (particularmente las trabajadoras) pudieran regresar a sus casas antes que los demás miembros de su familia, particularmente los niños.

3.2.2 Operación de producción.

La planta de Kyoto produce 63 productos diferentes, que se realizan en 15 líneas de producción. Normalmente, cada uno de los productos es responsabilidad de un equipo de 4 trabajadores. Debido a las similitudes de las materias primas requeridas, la planta puede mantener un inventario bajo que se reparte a todos los equipos.

Además, cada producto difiere muy poco en su proceso respecto los demás, al igual que en sus tiempos de proceso y requerimientos de trabajo. Como cada uno de los equipos trabaja sólo en un producto, el tiempo de preparación para los cambios se elimina. Cada equipo se organiza como una línea de producción con cuatro estaciones. Existe alguna variabilidad en los tiempos de proceso de la estación, y este elemento aleatorio sigue una distribución uniforme.

Tanto el primer como el segundo trabajador de cada estación, desempeña una función que lleva 10 min. completarla. A la primera persona nunca le toma menos de 8 min. ni más de 12 min. La acción hecha por la segunda persona de la línea es más variable, requiere entre 6 y 14 min. completarla.

La tercera posición del equipo de ensamble es la menos difícil y usualmente es realizada por la persona con menos experiencia del equipo, toma entre 8 y 10 min.

La última persona de la línea tiene el más largo de los trabajos que completar. Este trabajo consiste en tres pasos distintos que deben de ser hechos antes de que las unidades estén listas para embarcarse. En total, los tres pasos requieren entre 9 y 15 min. Debido a la diferencia de los tiempos de proceso entre la tercera y cuarta estación, existe espacio suficiente entre ellas para almacenar una unidad de inventario en proceso (WIP). Este es el único lugar en la línea que puede almacenar. Las otras estaciones no tienen ningún tipo de almacén entre ellas, de manera que pasa de una a otra directamente.

3.2.3 Opciones.

Akira ha estado investigando algunas de las alternativas para hacer que la producción de la planta crezca. La primera posibilidad es colocar pequeños inventarios de WIP entre cada una de las trabajadoras de la línea. El objetivo es reducir la frecuencia con que un trabajador está desocupado, pero tiene que esperar a que el trabajador anterior termine.

Otra de las opciones es instituir una "línea dinámica de ensamble" entre los trabajadores 3 y 4. Esto significa aumentar la capacidad de almacenamiento entre ellos dos, de manera que el tercer trabajador pueda ayudar al cuarto. Esto es, el tercer trabajador debe conocer cómo se tiene que hacer uno de los tres pasos que normalmente realiza el cuarto. Esto recibe el nombre de "acción flotante" porque no siempre es realizada por el mismo trabajador. Bajo este plan, el tercer trabajador continuará haciendo su trabajo normal mientras que el inventario entre ellos permanezca vacío. Si el inventario no está vacío cuando el trabajador 3 comienza una nueva pieza, añade a su rutina la "acción flotante". El trabajador continúa haciendo su acción adicional hasta que no haya piezas de la operación 3 en el inventario. Tan pronto como no hay unidades en el inventario, deja de hacer la "acción flotante" y completa sólo su trabajo normal.

Este esquema requiere la misma cantidad de tiempo para cada unidad, pero altera el tiempo requerido por el tercer y cuarto operador. Cuando el tercer operador hace la acción flotante, requiere entre 8 y 14 min. para completar una unidad. Esto reduce el requerimiento de trabajo para el cuarto trabajador que necesita entre 7 y 11 min. para terminar esa unidad. Aunque el producto final es el mismo. Es casi como si los operadores tres y cuatro hicieran dos productos distintos.

El entrecruce de las operaciones de los trabajadores es muy común en las empresas japonesas, pero Kyoto y Osaka, Ltd., siempre ha usado una asignación de trabajo más tradicional. Akira está interesado en probar qué tanto la línea dinámica de balanceo es capaz de alcanzar una mayor cantidad de producción, antes de comenzar un programa de capacitación para implementarla. Él es consciente de la capacidad de decisión que este método requiere por parte del trabajador tres, y de las otras maravillosas y simples secuencias fijas que se pueden encontrar, de manera que se obtengan los mismos resultados.

Akira cree que el impacto de esta propuesta puede ser probado mediante la simulación. Decidió simular dos meses de producción usando la teoría de la línea dinámica de balanceo.

3.2.4 Tópicos de Discusión.

1. ¿Cuál es el nivel de producción presente en dos semanas para la línea, dada la estructura de funcionamiento actual?. Si es así, ¿qué capacidad debería de ser aumentada a la planta, y dónde debería de colocarse?.

2. ¿La capacidad de producción crecerá o decrecerá con la técnica de balanceo dinámico de la línea?. Si crece, ¿cuánto crecerá?. (Hay permanencia de trabajo directo en los trabajos 1,2,3).

Nota: El balanceo dinámico de línea requiere de dos procesos para los trabajadores 3 y 4. Desafortunadamente esto dificulta la simulación de el paso de material entre 2 y 3. La razón es la siguiente, en los paquetes de simulación, los procesos sólo pueden tener un sólo destino de salida para sus productos. De cualquier forma, los procesos del trabajador 3 utiliza el mismo material, esto es, la salida de 2. No existe, comúnmente, la manera de enlazar al trabajador 2 con los dos procesos del trabajador 3 (al menos directamente). La solución es un traslado de material indirecto, a través de un inventario, ya que la mayoría de los inventarios pueden surgir (tener salida a) una gran cantidad de operaciones y procesos. Colocando un inventario después del trabajador 2 con capacidad de almacenamiento de cero, parecerá que se conectarán directamente entre sí los trabajadores 2 y 3, pero no es así. Se debe de determinar cómo usar los gatillos del trabajador 2 para simular cero inventario en el inventario entre las estaciones 2 y 3. (este inventario es innecesario cuando se considera un inventario real en el tópico 3).

3. Con el balanceo dinámico de líneas para los trabajadores 3 y 4, investigue el efecto de usar inventarios entre los primeros 3 trabajadores. Determine el efecto que puede tener el aumentar la capacidad de almacenamiento entre los trabajadores 3 y 4.

4. Con el balanceo dinámico de líneas, investigue el efecto de especificar un tamaño de lote para cada proceso del trabajador 3. Considere sólo los tamaños de lotes de 1 y 2, pero los dos procesos necesitan no tener el mismo tamaño de lote. Encuentre la que crea la mejor combinación de tamaños de lotes y explique sus resultados. (para este problema, seleccione una combinación de capacidad de inventario para su investigación).

3.3 Planteamiento del modelo de simulación.

3.3.1 Respuesta a los Tópicos presentados.

a. Análisis de la situación.

1. En base a la simulación realizada de la situación actual de la planta, presentado en los resultados de la simulación, se concluye que el sistema tiene una capacidad de 1248 unidades, en 15,000 horas simuladas, una vez que el sistema se ha cargado "normalmente". Esto significa que hace una unidad cada 12.02 minutos, aproximadamente. Lo que está muy cerca de la máxima capacidad de cuello de botella, que sólo puede hacer una unidad cada 12 minutos. De hecho en los resultados de la simulación de KYOTO1, la planta que sólo simula el estado actual de distribución de trabajo, dice que la estación correspondiente al cuello de Botella, W1, trabaja al 100% de su capacidad.

Se puede concluir que si el cuello de botella, gracias al inventario, trabaja al 100%, la única manera de producir más, es modificando el estado actual de la línea, de manera que W1 tenga más capacidad. Esto se puede conseguir mediante la anexión de una nueva máquina de tipo W1 o mediante el balanceo dinámico de línea.

Es claro que no tiene ningún caso añadir más inventarios entre los centros de trabajo que no son cuellos de botella, pues de todas formas W1 no trabajará más, ya está al límite de su capacidad. De igual forma se concluye que no tiene caso añadir capacidad de almacenamiento al inventarios precedente al cuello de botella.

2. Cuando se realizó la simulación de la misma planta bajo la propuesta del balanceo dinámico de línea (que tiene el nombre de KYOTO2), se aumentó considerablemente la capacidad de la planta. La nueva capacidad es de 1351 unidades en el mismo período de 15000 horas. esto significa una mejora respecto al modelo anterior de 8.25%. La línea ahora producirá una unidad cada 11.103 minutos, lo que está muy cerca del tope óptimo de la línea. Este tope se obtiene bajo el supuesto de que tenemos una línea perfectamente bien balanceada en 10 un/min., lo que da un máximo alcanzable de 1500 unidades.

Es claro que con un solo inventario y la política de que el trabajador 3 le ayude al 4 en algunas ocasiones, se obtiene un muy considerable incremento en capacidad sin invertir un peso más. Digo "aparentemente" porque para hacer en la realidad una línea dinámica, es necesario tener operadores

muy bien capacitados, que puedan desempeñar tanto una función como la otra. Es claro que la mano de obra capacitada, cuesta a la empresa.

Es importante notar que cuando se abre la posibilidad de que T3 ayude a T4 nuestro nuevo objetivo será llevar a su tope a T3, ya que mientras no esté totalmente saturado significará que todavía puede ayudar a T4. En los resultados impresos se nota que T3 está trabajando a un 79.72%. Además está bloqueado, por no tener material con el cual trabajar, el 13.77% de su tiempo. Esto nos indica que si podemos añadir un inventario entre T2 y T3 con el objetivo de garantizar que no se bloquee T3. Actualmente hay un inventario "fantasma" entre T2 y T3, esto es, un inventario que no tiene capacidad de almacenamiento y sólo sirve para suministrar material a los dos procesos de T3 (ver la nota de la pregunta 2 del caso).

3. Cuando comenzamos a "jugar" con los inventarios, se plantea como una gran oportunidad la de saturar a T3 de trabajo, haciendo trabajar más a T1 y T2, por tanto, también se coloca un inventario entre T1 y T2, para "limar" las diferencias estadísticas de producción entre ellos. A este nuevo modelo se le bautizó con el nombre de KYOTO3, y los resultados de la simulación se muestran en las páginas siguientes.

En general podemos decir que con inventarios de capacidad 3,3,3 se obtienen los mejores resultados posibles. Esto significa en términos de la simulación de una producción de 1498 unidades, una unidad cada 10.01 minutos. Casi se llegó al máximo posible. Los trabajadores casi están al 100% de su capacidad.

En términos reales se mejoró con esta postura un 20% respecto a la situación original de la línea. Es impresionante saber que, basta con una buena sincronización entre los trabajadores, para conseguir capacidad casi gratis.

4. Es importante notar que, cuando se introduce el nuevo elemento de análisis, esto es, el tamaño de lote, ya no hay mucho que mejorar. Como se explicó en la teoría del capítulo respecto a TOC, cuando se "juega" con los tamaños de lotes se busca disminuir el inventario en proceso. A menor tamaño de lote menor inventario en proceso. Sin embargo actualmente tenemos un tamaño de lote mínimo de 1, el sueño de cualquier ingeniero. Además casi estamos en el límite de lo óptimo.

Sin embargo es importante jugar, a modo de "what if?" con esta nueva variable. Los resultados que se pueden obtener de esto podrían ser de alguna utilidad en términos ingenieriles y teóricos.

b. Cálculos previos a la simulación.

El promedio de tiempo de proceso es 10 para cada uno de los primeros 2 trabajadores, 8 para el tercero y 12 para el cuarto. Por lo tanto lo más rápido que esta línea puede operar es al ritmo del cuello de botella, 12 unidades por minuto. En las simulaciones se pide que se corran los modelos durante 2 semanas, esto es, 6 días a la semana y 8 horas diarias da un total de 15000 minutos, equivalente a 2 meses de trabajo con 8 horas diarias. Con este horizonte de horas por trabajar y tomando en cuenta el tope máximo impuesto por el cuello de botella se tiene una posibilidad de producir 1250 unidades en total. Si la producción es menor que esto, es posible que existan algunas interferencias entre los trabajadores, que se pueden corregir mediante el uso de inventarios.

De cualquier manera la línea puede ser balanceada, cada estación podría promediar 10 minutos por unidad y la salida máxima sería de 1500 unidades, esto significa un incremento de 20%. Para entender porqué es que mediante la actividad flotante de T3 se puede mejorar el sistema, hay que tomar en cuenta que T3 pasa de 8 a 11 minutos, pero T4 pasa de 12 a 9 minutos. Se equilibran entre sí, otra vez en 10 minutos.

Una interesante pregunta es: ¿Qué tan seguido T3 va a tener que hacer su operación especial?, es importante recordar que nuestro objetivo es que las actividades o centros de trabajo promedien 10 minutos, para conseguir así un balanceo perfecto. Si T3 trabaja siempre en la actividad especial, entonces tardará más de 10 minutos, cerca a 11 min., y por tanto no se conseguirá un balanceo de la línea. Para que T3 se balancee en 10 es necesario que pase de 8 a 10 en promedio, entre su actividad

normal y la extra. Por tanto T3 necesita sólo ocupar 2 de sus 3 minutos de diferencia. En otras palabras T3 sólo debe de estar 2/3 partes de su tiempo haciendo la operación extra, lo que también se puede expresar como hacer 2 unidades con la actividad especial por cada 1 que se hace normal.

En este sentido, en base a los resultados de KYOTO2 y KYOTO3, y las simulaciones realizadas, podemos concluir que el modelo muestra que, si el óptimo es 0.68 de piezas por el proceso dinámico, se alcanza 0.41 normales y 0.52 por el dinámico. Es curioso cómo en la simulación de KYOTO2, es decir, la que sólo implementa la capacidad de la actividad especial de T3 no se cumple con esta expectativa, sólo el 30% del tiempo lo dedica a estos productos especiales. Sin embargo cuando llegamos a la planta "óptima" es decir KYOTO3, sí se cumple con la expectativa pues se encarga de la actividad especial casi el 65.7% de su tiempo. Esto se puede explicar diciendo que debido a la condición de T3 para trabajar en la actividad especial, ésta no se activa pues no se cumple, es decir, como los problemas principales de T3 son de bloqueo (por eso sólo trabaja a un 79.8% de su tiempo, el 13.8% está bloqueado), cuando el trabajador 3 se desbloquea la condición de que en el inventarios entre T3 y T4 exista al menos 1 producto normal, no se suele cumplir en un 70% de las ocasiones, y tiene que hacer su trabajo normal.

c. Formulación del modelo de la simulación.

El modelo general para esta simulación cumple con las siguientes características:

T1: Es el trabajador 1.

PT = Uniforme (8,12)

T2: Es el trabajador 2.

PT = Uniforme (8,14)

Trigger low cuando P decrece a 0 en B2 en KYOTO2.

T3: Es el trabajador 3.

Tiene dos procesos, P (normal) y P2 (especial para KYOTO2 y KYOTO3)

Proceso P: PT = Uniforme (8,10), ST = 0, No trigger.

Proceso P2: PT = Uniforme (8,14), ST = 0, Trigger high cuando P crece a 1 en B1.

T4 : Es el trabajador 4.

Proceso P: PT = Uniforme (9,15), ST = 0, No trigger.

Proceso P2: PT = Uniforme (7,11), ST = 0, No trigger.

B1: Se localiza entre T3 y T4.

Tiene capacidad de 2, en KYOTO2 y KYOTO3

B2: Se localiza entre T2 y T3.

No tiene capacidad de almacenamiento, en KYOTO2.

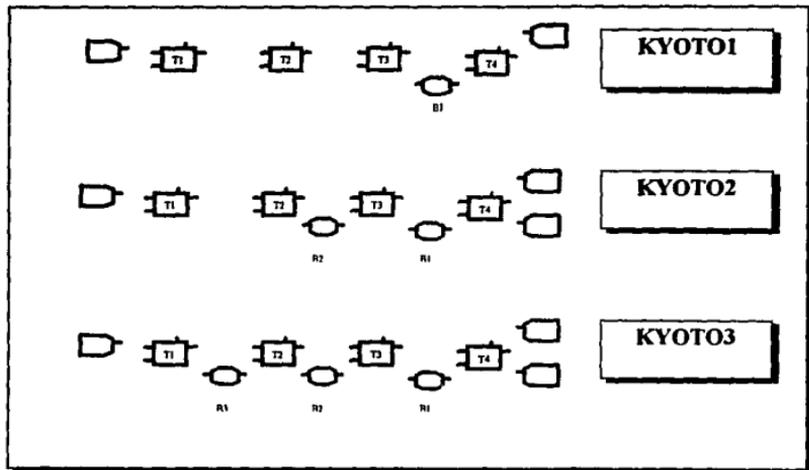
Tiene capacidad de 1 en KYOTO3.

B3: Se localiza entre T1 y T2.

Tiene capacidad de 1 en KYOTO3.

Además presento los siguientes diagramas de simulación que se emplearon en ambos programas:

Figura 3.2



3.3.2 Simulación en Excel+.

Factory KYOTO1:

WorkCenter T4, has Processes named:

P (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1248 1248
 % of time (procg,setup,mtn): 100.00 0.00 0.00

Process P at WorkCenter T4

inputs: X-input from stock P at B1; no Y-input
 normal-output to: ShippingArea EM
 processing-time distn: <FNORMAL>
 min,mean,max: 9.00 12.00 15.00
 trigger: <NONE>
 current state: BUSY processing until 15007.39
 units processed (period,cum): 1248 1248

WorkCenter T1, has Processes named:

P (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1249 1249
 % of time (procg,setup,mtn): 83.35 0.00 0.00

Process P at WorkCenter T1

inputs: X-input from ReceivingArea REC; no Y-input
 normal-output to: Process P at T2
 group: 1 major-setup: 10.00 minor-setup: 5.00
 processing-time distn: <FNORMAL>
 min,mean,max: 8.00 10.00 12.00
 trigger: <NONE>
 current state: BUSY processing until 15006.19

units processed (period,cum): 1249 1249

WorkCenter T2, has Processes named:

P (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1249 1249
 % of time (procg,setup,mtn): 83.47 0.00 0.00

Process P at WorkCenter T2

inputs: X-input from Process P at T1; no Y-input
 normal-output to: Process P at T3
 reject-output: % reject: 0.00
 group: 1 major-setup: 10.00 minor-setup: 5.00
 processing-time distn: <FNORMAL>
 min,mean,max: 6.00 10.00 14.00
 trigger: <NONE>
 current state: BUSY processing until 15007.44
 units processed (period,cum): 1249 1249

WorkCenter T3, has Processes named:

P (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1249 1249
 % of time (procg,setup,mtn): 66.71 0.00 0.00

Process P at WorkCenter T3

inputs: X-input from Process P at T2; no Y-input
 normal-output to: stock P at B1
 reject-output: % reject: 0.00
 group: 1 major-setup: 10.00 minor-setup: 5.00
 processing-time distn: <FNORMAL>
 min,mean,max: 6.00 8.00 10.00

trigger: <NONE>
current state: BUSY processing until 15004.02

units processed (period,cum): 1249 1249

Buffer B1; capacity: 1
withdrawal order: FIFO

| constants: | min | max | average | thruput | | | | |
|------------|------|-----|---------|---------|--------|------|--------|------|
| Part | init | new | period | cum | period | cum | period | cum |
| P | 0 | 0 | 0 | 0 | 1.00 | 1.00 | 1249 | 1249 |
| total: | | | | | 1.00 | 1.00 | 1249 | 1249 |

no request triggers specified
no delivery triggers specified
units in Buffer (Part names, oldest first):
(tagged units show RecArea of origin)
P.

ReceivingArea REC supplies units to:

X input to P at T1
Part supplied is P
tag 1 unit in 10
arrivals: <UNLIMITED>
storage-capacity: <UNLIMITED>
arrivals <LOGGED> in file
units started (period,cum): 1249 1249

ShippingArea EM receives units of:

P; normal-output from Proc P at WkCtr T4
Part accepted is P
shipments: <CONTINUOUS>
storage-capacity: <UNLIMITED>
shipments <LOGGED> in file
units accepted (period,cum): 1248 1248
tagged-unit avg time (total,in prev,wait): 60.27 40.18
20.09

Factory KYOTO2:

WorkCenter T4; has Processes named:

P2
P (current Process)
no breakdowns or scheduled maintenance
units processed (period,cum): 1352 1352
% of time (proc.g,setup,min): 99.81 0.00 0.00

Process P2 at WorkCenter T4

inputs: X-input from stock P2 at B1; no Y-input
normal-output to: ShippingArea EM2
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <FNORMAL>
min,mean,max: 7.00 9.00 11.00
trigger: <NONE>
current state: inactive
units processed (period,cum): 398 398

Process P at WorkCenter T4

inputs: X-input from stock P at B1; no Y-input
normal-output to: ShippingArea EM
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <FNORMAL>
min,mean,max: 9.00 12.00 15.00
trigger: <NONE>
current state: BUSY processing until 15010.14
units processed (period,cum): 954 954

WorkCenter T1; has Processes named:

P (current Process)
no breakdowns or scheduled maintenance
units processed (period,cum): 1357 1357
% of time (proc.g,setup,min): 90.22 0.00 0.00

Process P at WorkCenter T1

inputs: X-input from ReceivingArea REC; no Y-input
normal-output to: Process P at T2
reject-output: % reject: 0.00
group: 1 major-setup: 10.00 minor-setup: 5.00
processing-time distn: <FNORMAL>
min,mean,max: 8.00 10.00 12.00
trigger: <NONE>
current state: BUSY processing until 15007.60
units processed (period,cum): 1357 1357

WorkCenter T2; has Processes named:

P (current Process)
no breakdowns or scheduled maintenance
units processed (period,cum): 1356 1356
% of time (proc.g,setup,min): 89.97 0.00 0.00

Process P at WorkCenter T2

inputs: X-input from Process P at T1; no Y-input
normal-output to: stock P at B2
reject-output: % reject: 0.00
group: 1 major-setup: 10.00 minor-setup: 5.00
processing-time distn: <FNORMAL>
min,mean,max: 6.00 10.00 14.00
trigger: <NONE>
current state: BUSY processing until 15009.31
units processed (period,cum): 1356 1356

WorkCenter T3; has Processes named:

P2
P (current Process)
no breakdowns or scheduled maintenance
units processed (period,cum): 1355 1355
% of time (proc.g,setup,min): 79.88 0.00 0.00

Process P2 at WorkCenter T3

inputs: X-input from stock P at B2; no Y-input
normal-output to: stock P2 at B1
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00

processing-time distn: <FNORMAL>
 min,mean,max: 8.00 11.00 14.00
 trigger: <HGH>
 invcls when qty of P at B1 increases to 1
 current state: inactive
 units processed (period,cum): 400 400

Process P at WorkCenter T3
 inputs: X-input from stock P at B2; no Y-input

normal-output to: stock P at B1
 reject-output: % reject: 0.00
 group: 1 major-setup: 0.00 minor-setup: 0.00
 processing-time distn: <FNORMAL>
 min,mean,max: 6.00 8.00 10.00
 trigger: <NONE>
 current state: BUSY processing until 13006.64
 units processed (period,cum): 955 955

Buffer B1; capacity: 2

withdrawal order: FIFO
 minimum-holding-time: 0.00

| contents: | | min | max | average | | thruput | | | | |
|-----------|------|-----|--------|---------|--------|---------|--------|------|------|-----|
| Part | init | now | period | cum | period | cum | period | cum | | |
| P2 | 0 | 2 | 0 | 0 | 2 | 2 | 0.59 | 0.59 | 400 | 400 |
| P | 0 | 0 | 0 | 0 | 2 | 2 | 1.37 | 1.37 | 954 | 954 |
| totals: | | 2 | | 2 | 2 | 1.96 | 1.96 | 1354 | 1354 | |

no request triggers specified
 no delivery triggers specified
 units in Buffer (Part name, oldest first):
 P2, P2;

Buffer B2; capacity: 0

withdrawal order: UNORDERED
 minimum-holding-time: 0.00

| contents: | | min | max | average | | thruput | | | | |
|-----------|------|-----|--------|---------|--------|---------|--------|------|------|------|
| Part | init | now | period | cum | period | cum | period | cum | | |
| P | 0 | 0 | 0 | 0 | 0 | 0 | 0.00 | 0.00 | 1355 | 1355 |
| totals: | | 0 | | 0 | 0 | 0.00 | 0.00 | 1355 | 1355 | |

no request triggers specified
 no delivery triggers specified

ReceivingArea REC supplies units to:
 X input to P at T1
 Part supplied is P
 tag 1 unit in 10
 arrivals: <UNLIMITED>
 storage-capacity: <UNLIMITED>
 arrivals <LOGGED> in file
 units started (period,cum): 1357 1357

shipments: <CONTINUOUS>
 storage-capacity: <UNLIMITED>
 shipments <LOGGED> in file
 units accepted (period,cum): 953 953

ShippingArea EM receives units of:
 P, normal-output from Proc P at WkCtr T4
 Part accepted is P

ShippingArea EM2 receives units of:
 P2, normal-output from Proc P2 at WkCtr T4
 Part accepted is P2
 shipments: <CONTINUOUS>
 storage-capacity: <UNLIMITED>
 units accepted (period,cum): 398 398

Factory KYOTO3:

WorkCenter T4; has Processes named:
 P2
 P (current Process)

no breakdowns or scheduled maintenance
 units processed (period,cum): 1498 1498
 % of time (proc%setup,mntn): 100.00 0.00 0.00

Process P2 at WorkCenter T4

inputs: X-input from stock P2 at B1; no Y-input
 normal-output to: ShippingArea EM2
 reject-output: % reject: 0.00
 group: 1 major-setup: 0.00 minor-setup: 0.00
 processing-time dist: <FNORMAL>
 min,mean,max: 7.00 9.00 11.00
 trigger: <NONE>
 current state: inactive
 units processed (period,cum): 982 982

Process P at WorkCenter T4
 inputs: X-input from stock P at B1; no Y-input
 normal-output to: ShippingArea EM
 reject-output: % reject: 0.00
 group: 1 major-setup: 0.00 minor-setup: 0.00
 processing-time dist: <FNORMAL>
 min,mean,max: 9.00 12.00 15.00
 trigger: <NONE>
 current state: BUSY processing until 15006.93
 units processed (period,cum): 516 516

WorkCenter T1; has Processes named:

P (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1498 1498
 % of time (proc%_setup,stm): 99.96 0.00 0.00

Process P at WorkCenter T1
 inputs: X-input from ReceivingArea REC; no Y-input
 normal-output to: stock P at B3
 reject-output: % reject: 0.00
 group: 1 major-setup: 10.00 minor-setup: 5.00
 processing-time dist: <FNORMAL>
 min,mean,max: 8.00 10.00 12.00
 trigger: <NONE>
 current state: BUSY processing until 15004.85
 units processed (period,cum): 1498 1498

WorkCenter T2; has Processes named:
 P (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1496 1496
 % of time (proc%_setup,stm): 99.92 0.00 0.00

Process P at WorkCenter T2
 inputs: X-input from stock P at B3; no Y-input
 normal-output to: stock P at B2
 reject-output: % reject: 0.00
 group: 1 major-setup: 10.00 minor-setup: 5.00
 processing-time dist: <FNORMAL>
 min,mean,max: 6.00 10.00 14.00
 trigger: <NONE>
 current state: BUSY processing until 15003.39
 units processed (period,cum): 1496 1496

WorkCenter T3; has Processes named:

P2 (current Process)
 no breakdowns or scheduled maintenance
 units processed (period,cum): 1497 1497
 % of time (proc%_setup,stm): 99.26 0.00 0.00

Process P2 at WorkCenter T3
 inputs: X-input from stock P at B2; no Y-input
 normal-output to: stock P2 at B1
 reject-output: % reject: 0.00
 group: 1 major-setup: 0.00 minor-setup: 0.00
 processing-time dist: <FNORMAL>
 min,mean,max: 8.00 11.00 14.00
 trigger: <HIGH>
 invoke when qty of P at B1 increases to 1
 current state: inactive
 units processed (period,cum): 981 981

Process P at WorkCenter T3
 inputs: X-input from stock P at B2; no Y-input
 normal-output to: stock P at B1
 reject-output: % reject: 0.00
 group: 1 major-setup: 0.00 minor-setup: 0.00
 processing-time dist: <FNORMAL>
 min,mean,max: 6.00 8.00 10.00
 trigger: <NONE>
 current state: BUSY processing until 15007.70
 units processed (period,cum): 516 516

Buffer B1; capacity: 3

withdrawal order: FIFO
 minimum-holding-time: 0.00

| contents: | min | max | average | throughput | | | | | | |
|-----------|------|-----|---------|------------|--------|------|--------|------|--------|-----|
| Part | invl | now | period | sum | period | cum | period | cum | period | cum |
| P2 | 0 | 2 | 0 | 2 | 3 | 3 | 1.18 | 1.18 | 982 | 982 |
| P | 0 | 0 | 0 | 1 | 3 | 3 | 0.64 | 0.64 | 515 | 515 |
| totals: | 2 | | 3 | 3 | 1.82 | 1.82 | 1497 | 1497 | | |

no request triggers specified
 no delivery triggers specified
 units in Buffer (Part names, oldest first)
 (tagged units show RecArea of origin)
 P2; P2;

Buffer B2; capacity: 3

withdrawal order: FIFO
minimum-holding-time: 0.00

| contents: | | min | max | average | throughput | | | | | |
|-----------|------|-----|--------|---------|------------|-----|--------|------|------|------|
| Part | init | now | period | cum | period | cum | period | cum | | |
| P | 0 | 0 | 0 | 1 | 3 | 3 | 1.48 | 1.48 | 1496 | 1496 |
| totals: | | 0 | 0 | 3 | 3 | 3 | 1.48 | 1.48 | 1496 | 1496 |

no request triggers specified
no delivery triggers specified

Buffer B3; capacity: 3

withdrawal order: FIFO
minimum-holding-time: 0.00

| contents: | | min | max | average | throughput | | | | | |
|-----------|------|-----|--------|---------|------------|-----|--------|------|------|------|
| Part | init | now | period | cum | period | cum | period | cum | | |
| P | 0 | 3 | 0 | 1 | 3 | 3 | 1.20 | 1.20 | 1498 | 1498 |
| totals: | | 3 | 0 | 3 | 3 | 3 | 1.20 | 1.20 | 1498 | 1498 |

no request triggers specified
no delivery triggers specified
units in Buffer (Part names, oldest first):
(tagged units show RecArea of origin)
P, P, P

ReceivingArea REC supplies units to:

X input to P at T1
Part supplied is P
tag 1 unit in I0
arrivals: <UNLIMITED>
storage-capacity: <UNLIMITED>
arrivals <LOGGED> in file
units started (period,cum): 1498 1498

shipments <LOGGED> in file
units accepted (period,cum): 515 515
tagged-unit avg time (total,in prec,wait): 86.91 40.14
46.77

ShippingArea EM receives units of:

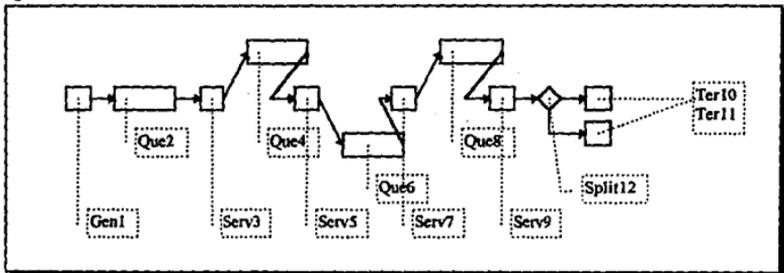
P, normal-output from Proc P at WkCr T4
Part accepted is P
shipments: <CONTINUOUS>
storage-capacity: <UNLIMITED>

ShippingArea EM2 receives units of:
P2, normal-output from Proc P2 at WkCr T4
Part accepted is P2
shipments: <CONTINUOUS>
storage-capacity: <UNLIMITED>
units accepted (period,cum): 983 983
tagged-unit avg time (total,in prec,wait): 83.83 40.04
43.79

3.3.3 Simulación en PASION.

A continuación se muestra el programa fuente en PASION (KYT6.pas), la 6 versión, que es la que corresponde al modelo comparado con el de XCELL+, este programa es traducido a PASCAL mediante PAT4. Recordamos que es gracias a QMG6 que no tenemos que programar directamente, que como se puede ver no parece muy amistoso, basta formar un diagrama de bloques y dar unas cuantas características. A continuación muestro el diagrama de bloques correspondiente para este caso en PASION QMG6:

Figura 3.3



PROGRAM KYT6;

```
[PACION program generated by the QMG module]
[Use PAT4 to translate it to Pascal]
USES RKYT6;
(*SI PROC2*)
```

```
PROCESS ENTITY,2000;
ATR TIMIN,TIMQ:REAL; SOURCE:INTEGER;
SO:INTEGER;
```

```
EVENT EV2;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
WAIT(QUE4,I); TIMQ:=TIME;
IF SER3 AND SEM3 THEN BEGIN OUT(QUE2,N,T);
IF N > 0 THEN BEGIN QN2:=QN2+1.0;
```

```
ENTITY[N].EV3:-TIME END END;
ENDEV;
```

```
EVENT EV3;
BEGIN
IF NOT (SER3 AND SEM3) THEN BEGIN
LSE3:=LSE3+1; KILL THIS; ABAND END;
SER3:=FALSE; SV3:-TIME;
SVOP3,SOURCE,TIMIN,TIMQ,SO);
ENDS3:=TIME+PRO1+RANDOM*EPS;ENDEV;
```

```
EVENT ENDS3;
VAR N:INTEGER; BEGIN
SER3:=TRUE;NSER3:=NSER3+1;COST:=COST+ 1.0000;
STM3:=STM3+TIME-SV3;
IF (NR(QUE2)=0) AND SEM3 THEN BEGIN
OUT(QUE2,N,T);
QN2:=QN2-1.0;
QU2:=QU2+TIME-ENTITY[N].TIMQ;
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
ENTITY[N].EV3:-TIME+0.0 END;
EV4:=TIME;ENDEV;
```

```
EVENT EV4;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
WAIT(QUE4,I); TIMQ:=TIME;
IF SER5 AND SEM5 THEN BEGIN OUT(QUE4,N,T);
IF N > 0 THEN BEGIN QN4:=QN4+1.0;
```

```
ENTITY[N].EV5:-TIME END END;
ENDEV;
```

```
EVENT EV5;
BEGIN
IF NOT (SER5 AND SEM5) THEN BEGIN
LSE5:=LSE5+1; KILL THIS; ABAND END;
SER5:=FALSE; SV5:-TIME;
```

```
SVOP5,SOURCE,TIMIN,TIMQ,SO);
ENDS5:=TIME+PRO2+RANDOM*EPS;ENDEV;
```

```
EVENT ENDS5;
VAR N:INTEGER; BEGIN
SER5:=TRUE;NSER5:=NSER5+1;COST:=COST+ 1.0000;
STM5:=STM5+TIME-SV5;
IF (NR(QUE4)=0) AND SEM5 THEN BEGIN
OUT(QUE4,N,T);
QN4:=QN4+1.0;
QU4:=QU4+TIME-ENTITY[N].TIMQ;
COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
ENTITY[N].EV5:-TIME+0.0 END;
EV6:=TIME;ENDEV;
```

```
EVENT EV6;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
WAIT(QUE6,I); TIMQ:=TIME;
IF SER7 AND SEM7 THEN BEGIN OUT(QUE6,N,T);
IF N > 0 THEN BEGIN QN6:=QN6+1.0;
```

```
ENTITY[N].EV7:-TIME END END;
ENDEV;
```

```
EVENT EV7;
```

```

BEGIN
IF NOT (SER7 AND SEM7) THEN BEGIN
  LSER7:=LSER7+1; KILL THIS; ABAND END;
SER7:=FALSE; SV7:=TIME;

```

```

SVOP(7,SOURCE,TIMIN,TIMQ,SO);
ENDS7:=TIME+PROX(SO)+RANDOM*EPS;ENDEV;

```

```

EVENT ENDS7;
VAR N:INTEGER; BEGIN
SER7:=TRUE;NSER7:=NSER7+1;COST:=COST+ 1.0000;
STM7:=STM7+TIME-SV7;
IF (NR(QUE6)>0) AND SEM7 THEN BEGIN
  OUT(QUE6,N,F);
  QN6:=QN6+1.0;
  QJ6:=QJ6+TIME-ENTITY[N].TIMQ;
  COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
  ENTITY[N].EV7:=TIME+0.0 END;
EV8:=TIME;ENDEV;

```

```

EVENT EV8;
VAR N:INTEGER; PK:BOOLEAN; BEGIN
WAIT(QUE,I), TIMQ:=TIME;
IF SER9 AND SEM9 THEN BEGIN OUT(QUE,N,F);
IF N >0 THEN BEGIN QN8:=QN8+1.0;

```

```

  ENTITY[N].EV9:=TIME END END;
ENDEV;

```

```

EVENT EV9;
BEGIN
IF NOT (SER9 AND SEM9) THEN BEGIN
  LSER9:=LSER9+1; KILL THIS; ABAND END;
SER9:=FALSE; SV9:=TIME;

```

```

SVOP(9,SOURCE,TIMIN,TIMQ,SO);
ENDS9:=TIME+PROX(SO)+RANDOM*EPS;ENDEV;

```

```

EVENT ENDS9;
VAR N:INTEGER; BEGIN
SER9:=TRUE;NSER9:=NSER9+1;COST:=COST+ 1.0000;
STM9:=STM9+TIME-SV9;
IF (NR(QUE8)>0) AND SEM9 THEN BEGIN
  OUT(QUE8,N,F);
  QN8:=QN8+1.0;
  QJ8:=QJ8+TIME-ENTITY[N].TIMQ;
  COST:=COST+(TIME-ENTITY[N].TIMQ)* 1.0000;
  ENTITY[N].EV9:=TIME+0.0 END;
EV12:=TIME;ENDEV;

```

```

EVENT EV12;
VAR KSUC:INTEGER;
BEGIN
KSUC:=DIVID(SO);
IF KSUC=1 THEN EV10:=TIME;
IF KSUC=2 THEN EV11:=TIME;
ENDEV;

```

```

EVENT EV10;
BEGIN
TER10:=TER10+1;

```

```

SVOP(10,SOURCE,TIMIN,TIMQ,SO);
TSS(TIMIN); KILL THIS;
ENDEV;

```

```

EVENT EV11;
BEGIN
TER11:=TER11+1;

```

```

SVOP(11,SOURCE,TIMIN,TIMQ,SO);
TSS(TIMIN); KILL THIS;
ENDEV;

```

```

PROCESS GENER.1;

```

```

EVENT EVX;
VAR N:INTEGER; BEGIN
EVX:=TIME+1000*EPS;

```

```

IF SER3 AND SEM3 AND NOT SEMP3
THEN BEGIN
  N:=GIVE(QUE3,1,F);
  IF N>0 THEN BEGIN
    OUT(QUE3,N,F); ENTITY[N].EV3:=TIME;
    SEMP3:=TRUE; ABAND END END;

```

```

IF SER3 AND SEM3 AND NOT SEMP3
THEN BEGIN
  N:=GIVE(QUE4,1,F);
  IF N>0 THEN BEGIN
    OUT(QUE4,N,F); ENTITY[N].EV3:=TIME;
    SEMP3:=TRUE; ABAND END END;

```

```

IF SER7 AND SEM7 AND NOT SEMP7
THEN BEGIN
  N:=GIVE(QUE6,1,F);
  IF N>0 THEN BEGIN
    OUT(QUE6,N,F); ENTITY[N].EV7:=TIME;
    SEMP7:=TRUE; ABAND END END;

```

```

IF SER9 AND SEM9 AND NOT SEMP9
THEN BEGIN
  N:=GIVE(QUE8,1,F);
  IF N>0 THEN BEGIN
    OUT(QUE8,N,F); ENTITY[N].EV9:=TIME;
    SEMP9:=TRUE; ABAND END END;
ENDEV;

```

```

EVENT GEN1;
VAR K:INTEGER; BEGIN
GEN1:=TIME+9.5;
IF SEM1 THEN BEGIN
  FOR K=1 TO 1 DO BEGIN
    IF COUNT>1500 THEN ABAND;
    NEWPR ENTITY;
    COUNT:=COUNT+1;PENTITY.SOURCE:=1;

```

```

SVOP(1,PENTITY.SOURCE,PENTITY.TIMIN,PENTITY.TIMQ,
PENTITY.SO);
PENTITY.EV2:=TIME+RANDOM*EPS;
PENTITY.TIMIN:=TIME END; END ENDEV;

```

```

EVENT FINAL;
VAR N:INTEGER; BEGIN
FORALL N ENTITY BEGIN KILL ENTITY[N] ENDALL;
FORALL N GENER BEGIN KILL GENER[N] ENDALL;
KILL PPRNT; FINSTA;ENDEV;

```

```

EVENT FINN;
BEGIN
FINSTA ENDEV;

```

```

EVENT COSTFN;
VAR J:INTEGER; BEGIN
FORALL J IN QUE2 BEGIN
  COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
FORALL J IN QUE4 BEGIN
  COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
FORALL J IN QUE6 BEGIN
  COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
FORALL J IN QUE8 BEGIN
  COST:=COST+(TIME-ENTITY[J].TIMQ)* 1.0000 ENDALL;
ENDEV;

```

```

PROCESS PRNT.1;
ATR FIRTIM:BOOLEAN.

```

```

EVENT PRI;
VAR L,NQX:INTEGER, BEGIN
IF TIME>HPR<TFIN THEN PRI:=TIME+HPR;
REQMAG(FIRTIM);
VARN[1]:='QUEUE1'; VARN[2]:='QUEUE4';
VARN[3]:='QUEUE5';
VARN[4]:='QUEUE7'; VARN[5]:='QUEUE'; VARN[6]:='QUEUE';
VARN[7]:='QUEUE'; VARN[8]:='QUEUE'; VARN[9]:='QUEUE';
VARN[10]:='QUEUE'; KX:=KX+1; IF KX<=100 THEN BEGIN
X[1,KX]:=NR(QUE1); X[2,KX]:=NR(QUE4);
X[3,KX]:=NR(QUE5);
X[4,KX]:=NR(QUE7);
END,ENDEV;
START

```

```

INIT(TRUE);NQJE QUE1; NQJE QUE4; NQJE QUE5; NQJE
QUE7;
POW:=-KPOW>;
BQJE;
DIN;
REPRM KPOW TFIN
REIN;
NEWPR POENER;
POENER.GEN1:=0+REPS;
POENER.EVX:=TIME+1000.0*EPS;
POENER.COSTFN:=TFIN.EPS;
POENER.FIN:=TFIN+0.5*EPS;
NEWPR PFRNT; PFRNT.PRI:=2*EPS;
PFRNT.FIRTIM:=TRUE;
RESETS
5

```

Ahora se presenta el programa correspondiente a las distribuciones de probabilidad de cada uno de los servidores, así como de las reglas de los divisores de flujo de materiales. Este programa recibió el nombre de PROC2.pas, pues representa la segunda versión:

```

FUNCTION DIVID(SO:INTEGER):INTEGER;
VAR X:INTEGER;
BEGIN
IF SO=1 THEN X:=1 ELSE X:=2;
DIVID:=X;
END;

FUNCTION PRO1:REAL;
BEGIN
PRO1:=RANDOM*(4)+8;
END;

FUNCTION PRO2:REAL;
BEGIN

```

```

PRO2:=RANDOM*(8)+6;
END;

FUNCTION PRO3(SO:INTEGER):REAL;
BEGIN
IF SO=2 THEN PRO3:=RANDOM*(6)+8
ELSE PRO3:=RANDOM*(4)-6;
END;

FUNCTION PRO4(SO:INTEGER):REAL;
BEGIN
IF SO=2 THEN PRO4:=RANDOM*(4)+7
ELSE PRO4:=RANDOM*(6)-9;
END;

```

Además se incluye un listado de las modificaciones que se hicieron en el procedimiento SVOP, que controlan los semáforos de los servidores, así como el proceso en turno para el balanceo dinámico de líneas. Este procedimiento está incluido en el programa RKYT6.pas, que tiene como contenido los procedimientos de control del programa principal. La primera línea del siguiente listado corresponde a la declaración de una variable global, que sirve para recordar cuál fue el último tipo de pieza hecho por el servidor 7 (trabajador 3).

```

var glob:integer;

PROCEDURE SVOP
(N,SOURCE:INTEGER; TIMN,TIMQ:REAL;
VAR SO:INTEGER);

BEGIN
IF (N=7) THEN BEGIN
IF (NR(QUE8)>=3) THEN BEGIN
glob:=2;
GOTOXY(1,20),WRITELN(' ');
GOTOXY(1,20),WRITELN('BIANDETA ON');
END;
IF (NR(QUE8)=0) THEN BEGIN
glob:=1;
GOTOXY(1,20),WRITELN(' ');
GOTOXY(1,20),WRITELN('BIANDETA OFF');
END;

```

```

so:=glob;
END;

sem3:=true; sem3:=true;
if (nr(que4)>=3) then sem3:=false;
if (nr(que6)>=3) then sem5:=false;

(* Insert here user-defined operations for all servers,
assembly operations and generators. N is the corresponding
block number. Use CASE or IF statements to identify blocks.)
If called from assembly op., this procedure is called for all
parts which enter it, and for the assembled new part. Remember
that each server generates and assembly op. can be open or closed
by SEMx boolean predefined variables (TRUE is ON, FALSE is OFF),
initially set to TRUE.
This procedure is called BEFORE the service time is calculated. *)

END;

```

Estas son las principales partes del programa de PASION para el modelo en cuestión, sólo faltan las correspondientes a la modificación de las pantallas de resultados, pero, como los resultados se presentan a continuación me parece que no hace falta abundar en detalles de programación.

3.4 Conclusiones.

3.4.1 Conclusiones sobre el caso presentado.

Las conclusiones sobre el caso modelado, donde se evalúan las técnicas de JIT y balanceo dinámico de producción nos llevan a decir que, si bien la idea es bastante buena para los efectos que a producción corresponden, hay algunas variables cuya manipulación pueden hacer que se obtengan casi resultados óptimos. Esto quiere decir que, cuando se procura el balanceo dinámico de producción los centros de trabajo casi están equilibrados y trabajan al 80%, sin embargo, cuando se incrementan los inventarios entre ellos, se llega a periodos de ocupación de casi el 100%. Recordamos que en estos modelos que se comparan, hay igualdad de circunstancias, esto es, los tiempos de los servidores son los mismos, se tienen inventarios con capacidad de 3 cada uno y además las máquinas se bloquean cuando existe suficiente inventario como para hacerlo.

En primer lugar se muestran los resultados obtenidos en XCELL+ en el caso óptimo KYOTO3.

Resultados para la fábrica KYOTO3

Tabla 3.1

Utilización de Centros de Trabajo al tpo: 15000

| | %Ocupado | | %Preparación | | Mantenimiento | | %bloqueado | |
|----|----------|-----------|--------------|-----------|---------------|-----------|------------|-----------|
| | periodo | acumulado | periodo | acumulado | periodo | acumulado | periodo | acumulado |
| T4 | 99.777 | 99.777 | 0 | 0 | 0 | 0 | 0 | 0 |
| T1 | 99.063 | 99.063 | 0 | 0 | 0 | 0 | 0.937 | 0.937 |
| T2 | 98.237 | 98.237 | 0 | 0 | 0 | 0 | 0.612 | 0.612 |
| T3 | 96.558 | 96.558 | 0 | 0 | 0 | 0 | 0.836 | 0.836 |

Tiempo de flujo: Area de Embarque al tpo: 15000

(En base al arribo de unidades marcadas al Area de embarque)

Tabla 3.2

| | número | Flujo promedio | Promedio de procesamiento | Espera promedio |
|-----|--------|----------------|---------------------------|-----------------|
| EM | 55 | 84.685 | 40.213 | 44.472 |
| EM2 | 94 | 90.189 | 39.565 | 50.624 |

Tabla 3.3 Paso total por el Area de Embarque al tpo: 15000

| | salida | | almacén | |
|----------|---------|-----------|---------|-----------|
| | periodo | acumulado | periodo | acumulado |
| EM | 575 | 575 | 0 | 0 |
| EM2 | 911 | 911 | 0 | 0 |
| totales: | 1486 | 1486 | 0 | 0 |

Productos en Proceso al tpo: 15000

Tabla 3.4

| | actual | mínimo | | máximo | | Promedio | |
|---------|--------|---------|-----------|---------|-----------|----------|-----------|
| | | periodo | acumulado | periodo | acumulado | periodo | acumulado |
| B1 | 2 | 0 | 0 | 3 | 3 | 1.864 | 1.864 |
| B2 | 1 | 0 | 0 | 3 | 3 | 1.395 | 1.395 |
| B3 | 1 | 0 | 0 | 3 | 3 | 1.557 | 1.557 |
| totales | 4 | | | | | 4.815 | 4.815 |

A continuación se muestran los resultados en PASION:

Resultados para la simulación de KYOTO3:

Tiempo de la simulación: 15000 u.t.

Tabla 3.5 Líneas de espera:

| | Promedio de espera | Longitud máxima | Longitud mínima | Longitud Promedio | Pérdidas |
|------|--------------------|-----------------|-----------------|-------------------|----------|
| QUE2 | 484.139 | 107 | 0 | 51.297 | 0 |
| QUE4 | 12.544 | 4 | 0 | 1.323 | 0 |
| QUE6 | 8.069 | 3 | 0 | 0.759 | 0 |
| QUE8 | 15.687 | 4 | 0 | 1.505 | 0 |

Servidores:

Tabla 3.6

| | Tiempo de servicio | Tiempo perdido | %perdido | Pérdidas | Piezas trabajadas |
|--------|--------------------|----------------|----------|----------|-------------------|
| SERV 3 | 14387.122 | 612.885 | 4.09 | 0 | 1439.3 |
| SERV 5 | 14403.979 | 596.029 | 3.97 | 0 | 1438.4 |
| SERV 7 | 14142.289 | 857.718 | 5.72 | 0 | 1437.5 |
| SERV 9 | 14657.956 | 342.052 | 2.28 | 0 | 1436.5 |

Tabla 3.7 Time average in system: 568.91

| | PZs | QUE2 | PZs | QUE4 | PZs |
|---------|------|------|------|------|-----|
| TER10 | 590 | 25 | QUE4 | 2 | |
| TER11 | 878 | 1 | QUE8 | 1 | |
| Totales | 1468 | | | 29 | |

Cuando se comparan los resultados de las dos simulaciones se encuentra que tienen una muy pequeña diferencia entre sus resultados. Esta diferencia es explicable debido a que las distribuciones de los servidores son uniformes, y cuando se corren un número "pequeño" de veces no se comportan igual. Tal vez si este modelo lo ejecutamos, en ambos casos un gran número de veces, los resultados se parezcan más.

Sin embargo, las diferencias no son significativas. Los resultados presentados expresan lo mismo pero de distinta forma. Para encontrar la similitud hay que hacer algunos cálculos. Por ejemplo, PASION dice que el tiempo promedio total por pieza en el sistema es de 568.91 u.t., y XCELL+ dice que de las piezas muestreadas, el tiempo total promedio es de 86.20 u.t. (ver tabla de FLOW-TIME). Es sencillo darse cuenta de que el modelo de PASION contabiliza el almacén de entrada de materia prima, que no existe en el modelo de XCELL+, el tiempo que en la simulación de PASION las piezas están en este almacén es en promedio 484.139 u.t. (ver tabla de líneas de espera), si descontamos este tiempo al tiempo promedio en el sistema el resultado es 84.771, tiempo muy aproximado al presentado por XCELL+.

Tengo que reconocer que en la comparación de ambos lenguajes de simulación, al estudiar primero XCELL+, el comparado es PASION. Pero es importante hacer notar que PASION tiene una mayor flexibilidad en la presentación de los datos recogidos de la ejecución de la simulación, uno mismo puede formar las tablas de información que deseé, y además PASION cuenta con programas anexos como VARAN5, que gráfica el comportamiento estadístico de cada uno de los elementos simulados.

Para el caso es importante hacer notar que la idea de hacer Balanceo dinámico de líneas es realmente buena, y añadiéndole un poco de inventarios entre los trabajadores (cosa prohibida para JIT), se logra eliminar gran parte del

"nerviosismo" probabilístico, y se alcanza casi el óptimo teórico de producción (el 97%). Para finalizar, habría que analizar qué tan caro resulta aumentar este inventario de productos en proceso y tomar una decisión.

Me parece que el estudio del caso, y la correspondiente simulación han arrojado muchísima luz al análisis del problema, soluciones planteadas, análisis de técnicas de producción, y experiencia de la simulación.

3.4.2 Conclusiones sobre la técnica de la Simulación.

Cuando se trata sólo de comparar los modelos de simulación antes planteados, y sus correspondientes corridas, es importante señalar varias diferencias:

1. El tiempo de la formulación del modelo en XCELL+ fue breve respecto al invertido en PASION, sin embargo, en el tiempo de ejecución de cada uno de los modelos resultó ser más rápido PASION respecto a XCELL+. Esto nos puede indicar que los modelos "sencillos" son más fáciles de realizar en XCELL+, pero no los complicados.
2. XCELL+ muestra ciertas inconveniencias respecto al grado de complejidad lógica que permiten sus estructuras. PASION por el contrario presenta una gran flexibilidad, pues permite manipular directamente el programa (en PASCAL) del modelo. Se concluye que en PASION se puede desarrollar con gran facilidad modelos complicados.
3. PASION muestra gran flexibilidad en la presentación de los resultados, se pueden mostrar todas las variables involucradas en la ejecución del modelo, durante ella y a su fin. Esto es de gran utilidad, pues en la vida real, nos interesan muchas más variables que las restringidas por XCELL+. Claro que para esto es imprescindible saber programar.
4. La ventaja de trabajar con un programa de simulación diseñado en nuestra Universidad, es que se tiene al autor para consultar cualquier duda o problema¹. Este ejemplo ayudó a PASION a mejorar algunos detalles pequeños respecto a la logística de conteo en las colas y servidores, cuando se dan eventos simultáneos de arribo.

¹Para este capítulo en especial, se agradece las valiosísimas sugerencias del Dr. Raczyński que contribuyó enormemente a la formulación del modelo en PASION, y en que el caso resultara de un provecho enorme. Se invirtió mucho tiempo en el estudio.

Capítulo 4: Líneas Automáticas de Producción.

Introducción.

Justo después de la Revolución Industrial, las máquinas fueron sustituto del trabajo de los hombres, y la tecnología mecanizada reemplazó las labores manuales. Como el volumen de los productos estandarizados creció, fue más económico diseñar máquinas dedicadas a propósitos especiales de la producción de una sola parte o producto. Últimamente, en partes discretas de la manufactura, estas máquinas se enlazaron mediante artefactos automáticos de transporte. Ahora, gran parte del movimiento de material y producción directa se puede realizar automáticamente a través de la automatización.

Antes de la década de los 60's, la industria más avanzada en lo que se refiere a líneas de manufactura automáticas eran las de la industria automotriz. Ahora, aunque muchas más de las compañías automotrices siguen usando la manufactura automática, se ha extendido la automatización a otras ramas como la rama de los metales y la de componentes electrónicos. El objetivo principal en estas industrias es aproximarse a la rapidez y flexibilidad de estos procesos industriales como las empresas químicas y de comida. Los procesos industriales cada vez tienden más a adoptar este sistema de automatización pues requieren más bien de operaciones sin intervención de hombres, y les conviene mucho tener máquinas automáticas que pueden cambiar de proceso simplemente cargando un nuevo código de computadora.

4.1 La manufactura Automática y los modelos de Manufactura Flexible.

4.1.1 Automatización.

El término de automatización nos resulta muy familiar, pero cuando se trata de formar una definición nos resulta muy difícil. Algunos de las autoridades en producción ven detrás del término de automatización, un conjunto nuevo de conceptos que hacen referencia a las operaciones automáticas de los procesos productivos. Otros ven simplemente un desarrollo evolucionado en tecnología en el que las máquinas desarrollan todo o casi todo de las funciones de control de los procesos. Automatización es un conjunto de conceptos, pero es algo evolucionado en el sentido de que es un paso lógico y predecible en el desarrollo de los equipos de procesos.

Algunos de los mayores desarrollos en manufactura automática incluyen centros de maquinaria, máquinas controladas numéricamente, robots industriales, diseño y sistemas de manufactura asistidos por computadora

Los Centros de Maquinaria no sólo proveen de control automático de las máquinas, sino que también se refieren al traslado automático de los materiales. Por ejemplo, una máquina simple estará provista de un sistema disparador de dos mesas móviles que pueden moverse hacia dentro y hacia fuera de la

máquina. Cuando el trabajo ha sido hecho en una mesa, la siguiente parte se monta y prepara en la segunda mesa. Cuando el proceso de la primera mesa está completo, la mesa sale de la máquina, y la segunda pieza entra.

El Control Numérico de una Máquina (NC) trata del control digital de una máquina mediante un mecanismo de computadora. Los sensores de la computadora determinan la posición de la máquina durante el trabajo que desempeña, constantemente compara esta posición con la que originalmente se ha programado y la corrige si es necesario. Esto elimina el tiempo perdido durante las preparaciones y se puede aplicar a volúmenes muy altos de producción, estandarizando por tipos los productos y bajos niveles de producción ajustándolos a la medida.

4.1.2 Robots Industriales¹.

Los Robots Industriales son sustitutos de las manipulaciones humanas y otras operaciones altamente repetitivas. Un robot es una máquina programable con múltiples funciones que puede mover objetos en trayectorias muy especializadas y desarrollar cualquier tipo de tareas. Es esencialmente un brazo mecanizado que puede ser ajustado con una gran variedad de tenazas, brazaletes, o implementos sujetadores de objetos. Los robots pueden hacer cualquier tipo de procesos en una planta, desde procesos de maquinado hasta ensamblajes simples.

En la figura que a continuación se muestra se examina los movimientos humanos que un robot puede reproducir. Capacidades avanzadas han sido diseñadas en los robots para proporcionarles visión, tacto, y coordinación mano a mano. Adicionalmente, algunos modelos pueden seguir secuencias de movimientos en áreas tridimensionales. Un trabajador puede mover el extremo del brazo de un robot y éste registrará la secuencia de los movimientos y los grabará en su memoria de manera que los pueda repetir cuando se le dé un comando determinado.

Figura:

Los robots son caros, aunque con frecuencia pueden ser económicamente justificados ya que el tiempo de producción disminuye drásticamente y la exactitud y consistencia de producción se incrementan. Muchas de la compañías utilizan la siguiente expresión para evaluar los proyectos de inversión que involucran la compra de robots:

$$P = \frac{I}{L - E + q(L + Z)}$$

Donde:

P = Período de vida en años.

I = Capital total invertido en robots.

L = Costos anuales de trabajo reemplazados por el robot (Gastos y beneficios por tiempo del trabajador)

q = Factor de rapidez del robot, respecto el tiempo usual del trabajador.

Z = Depreciación anual.

En la planta de Forth Worth General Dynamics, el taladro computarizado Millicron T-3 perfora entre 24 y 30 partes por cargamento sin ningún defecto. Un operador humano puede producir sólo seis partes por carga, con 10% de rechazo. A pesar de que la compañía paga \$60,000 us por la compra de un robot, ahorra \$90,000 us el primer año.

Como la productividad aumenta mediante el uso de los robots los obreros pueden ser substituidos. Los robots desplazan entre 1.7 y 6 obreros; el potencial de aplicación es especialmente alto en las empresas metalúrgicas. Además, parece ser muy viable que en poco tiempo se tengan robots con sistemas de visión que se puedan utilizar en trabajos de ensamble delicados de otras industrias, como por ejemplo la electrónica. Algunos de los expertos estiman que en un futuro, los robots pueden desplazar 3.8 millones de trabajadores. Sin embargo hay muchísimas operaciones que en un futuro cercano, mediano y largo los robots no pueden hacer.

¹L.V. OTTINGER, "Robotics for the IE: Terminology, Types of Robots", Industrial Engineering, November 1981, p. 30.

Hay que tener en cuenta que los robots sólo pueden sustituir a los hombres en aquellas tareas altamente repetitivas y relativamente sencillas y poco racionales, operaciones donde el criterio no se utilice, operaciones que en el estudio del trabajo se pueden clasificar como poco humanas y siempre resultan ser agotadoras. Tenemos por ejemplo el obrero que pasa una vida apretando tuercas, o que pasa los días viendo botellas a contra luz para separar aquellas que tienen defectos. En el futuro se necesitarán obreros cada vez más capacitados, con la habilidad y conocimientos suficientes para controlar y programar a los robots, de manera que, a mi modo de ver las cosas, no se puede hablar en este tema como un franco desplazamiento de mano de obra, sino más bien, de una sustitución de aquellas actividades poco gratificantes para los obreros, y se exigirá de éstos, por otra parte, una mejor preparación y una participación racionalmente más activa en los procesos productivos.

4.1.3 Diseño Asistido por Computadora.

Uno de los avances contemporáneos más grandes en lo que se refiere al diseño de los productos es el diseño asistido por computadora (CAD). CAD puede ser definido como un traslado de todas las estructuras o mecanismos del los procesos de diseño de los productos o componentes a un equipo especial de computadora. Los ingenieros diseñan a través de los controles de la computadora y una pluma óptica que traza sobre la pantalla o plano electrónico. Las perspectivas de los diseños pueden ser visualizadas mediante la rotación automática del dibujo en la pantalla, y los pequeños componentes pueden ser analizados con detalle en sus características más ínfimas. Dependiendo de la sofisticación del software, las pruebas de pantalla pueden ahorrar gran parte de las fases de los prototipos y efectuar las modificaciones convenientes sin costos extras de materiales.

CAD ha sido usado para diseñar desde chips de computadoras hasta papitas fritas. Frito-Lay, por ejemplo, usa el CAD para diseñar sus Ruffles. El problema de diseñar papas es analizar si están cortadas correctamente, con el grosor indicado, pues en el momento de freírse pueden quedar quemadas por fuera y crudas por dentro, pueden estar muy quebradizas (y romperse cuando se embolsan) o tener algunas otras características que las hagan infuncionales para comerse, por ejemplo, con un dip de guacamole. De cualquier forma, mediante el uso de CAD, el ángulo adecuado y el número de ruffles pueden ser determinados matemáticamente, de manera que las papas pasen las pruebas adecuadas de textura, sabor y resistencia.

CAD actualmente se está usando para diseñar trajes de baño. Las tallas de la ropa son alimentadas en un programa de CAD, que contiene el diseño del vestido. Trabajando junto con el productor, el diseñador modifica el corte del vestido hasta que aparezca como si estuviera puesto en un cuerpo humano pero en la pantalla de la computadora. Una vez que se decide el diseño final la computadora imprime los moldes, y el vestido es cortado y cosido. Un buen ejemplo de el uso de CAD son los diseños de los circuitos electrónicos de Motorola, que trata de hacer el proceso de ensamble lo más sencillo posible.

Una limitación práctica de CAD es el gran tiempo de programación requerido para introducir todas las partes de un producto complicado en un programa de base de datos. En la actualidad gran partes de las empresas manufactureras invierten una gran cantidad de recursos en el uso de esta herramienta, como áreas prioritarias en el desarrollo de los productos.

El diseño y manufactura asistidos por computadora (CAD/CAM) usan a la computadora para hacer el diseño de las partes y elaborar instrucciones de procesos de fabricación. El diseño de algunos lotes de piezas sencillas por este sistema puede disminuir el costo de producción. Es importante hacer ver que la computadora no propone ideas, sólo aplica unas cuantas rutinas básicas de optimización, sin embargo el criterio y parte fundamental del uso de esta herramienta queda en manos del diseñador. Esta herramienta sólo es un asistente al diseños y no un sustituto del buen juicio y la experiencia.

A pesar de todo, el uso de CAD/CAM está limitado a las grandes compañías, pues sus costos de operación, capacitación e implementación son muy elevados. Por otro lado, esta herramienta eleva la productividad y calidad increíblemente, pues la filosofía de la simplificación en la mayoría de las ocasiones beneficia enormemente el proceso de manufactura. Los estudios de modificaciones así como los de costos de producción pueden ser también realizados con esta herramienta. Además, existe la llamada Planeación de procesos asistida por computadora CAPP, que en algunos casos puede facilitar los procesos de planeación convencionales.

4.1.4 Sistemas de Manufactura Flexibles².

Un sistema de Manufactura flexibles (FMS) se refiere a un número de sistema que difieren en su grado de mecanización, transferencia automática y control computarizado. Adrew Kusiak ha definido cinco sistemas de producción básicos, y cómo es que éstos se deben de aplicar según el número de partes distintas que se producen en una empresa. Estos cinco sistemas básicos son:

Módulos flexibles e manufactura.

Células de producción.

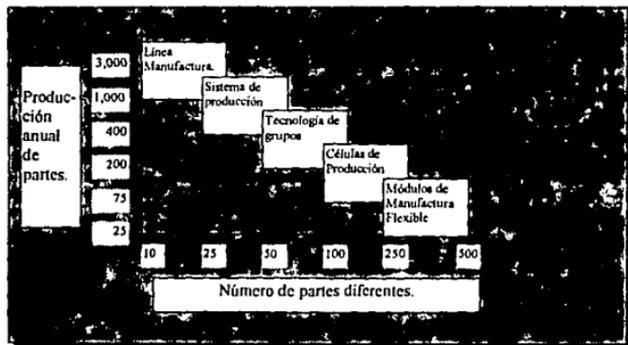
Tecnologías de grupos.

Sistemas de producción.

Producción en línea.

Un módulo flexible de manufactura es una máquina numéricamente controlada (NC) soportada con partes de inventario, herramientas de cambio, y charolas de cambio. Una célula flexible de manufactura consiste en conjuntos de módulos flexibles de manufactura organizados de acuerdo con los requerimientos particulares de un producto. Un grupo flexible de manufactura es una combinación de módulos flexibles y células localizadas en la misma área de manufactura y reunidas por un sistema de acarreo de material, como un riel automático.

Figura 4.1



Un sistema flexible de producción consiste en grupos flexibles de manufactura que conectan diferentes áreas de manufactura, las que fabrica, maquilan y ensamblan. Una línea flexible de manufactura es una serie de máquinas especializadas conectadas entre sí por rieles automáticos, robots, carros, o algún otro tipo de transporte automático.

² KUSIAK, Andrew, Flexible Manufacturing Systems: A Structural Approach, International Journal of Production Research 23, no. 6 (1985), pp. 1057-73.

Mientras lo convencional es creer que un sistema flexible de manufactura es lo mejor para bajos volúmenes y bajos números de partes, Cummins Engine Company³ sostiene que un alto grado de costos se pueden ahorrar cuando se aplica en altos niveles de producción.

4.1.5 Manufactura Integrada por Computadora (CIM).

La Manufactura Integrada por Computadora reúne todos los aspectos de la producción en un sólo sistema automático. El diseño, pruebas, fabricación, ensamble, inspección, y acarreo de material pueden tener funciones automáticas dentro de su misma área. De cualquier forma, en la mayoría de las compañías, la comunicación entre los departamentos se permanece mediante el flujo de papeles. En CIM, estas islas de automatización están integradas, por lo que se elimina el uso de papelería. Las computadoras enlazan todas los sectores, resultando una mayor eficiencia, menos papel de trabajo, y menos cantidad de personal.

El término islas de automatización se refieren a la transición de la manufactura convencional hasta la fábrica automatizada. Típicamente las islas incluyen herramientas numéricamente controladas, robots, almacenes automáticos, y centros de maquinarias.

Es ahora cuando el horizonte de este trabajo de tesis se vuelve a abrir en panoramas insospechados. Es claro para el lector que según la teoría podría seguir ahondando en la nueva tecnología y sistemas CIM. Pienso en este momento que es conveniente hacer un alto y presenta el caso que en este capítulo concreto nos ocupa. La simulación se analiza y aplica en una línea automática de producción.

4.2 Caso: Área de ensamble Electrónico.

En este ejemplo, las partes electrónicas son en primer lugar recogidas por la banda transportadora, entonces se ensamblan en 5 consecutivas operaciones (10, 20, 30, 40, 50), y entonces se descarga de la banda transportadora. Existen una estación para cargar la banda transportadora, para la operación 10, para la operación 20 y para descargar la banda al final del proceso. Existen tres estaciones para cada operación 30, 40 y 50, debido a que estas operaciones tienen un tiempo de proceso relativamente grande. Cada componente electrónico debe de pasar a través de las secuencias 30, 40, y 50.

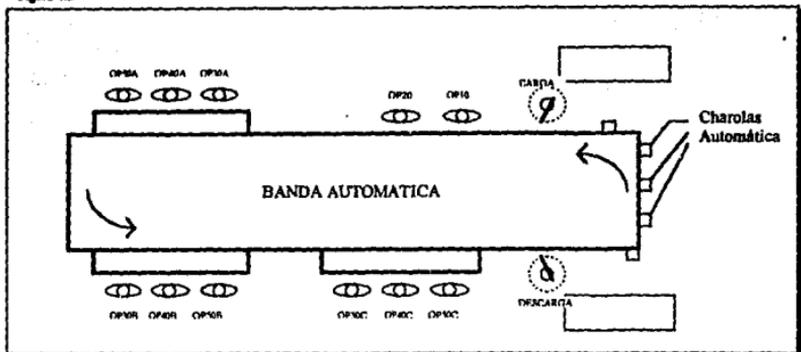
Adicionalmente, se requiere una charola automática de la banda para cada una de los elementos electrónicos de la producción.

| Operación | Tiempo de Proceso (Segundos por minuto) |
|-----------|---|
| Carga | 9 |
| 10 | 8.5 |
| 20 | 8.6 |
| 30 | 26.5 promedio (rango 25 a 28) |
| 40 | 26.5 promedio (rango 25 a 28) |
| 50 | 26.5 promedio (rango 25 a 28) |
| Descarga | 8.5 |

Tabla 4.1

³VENKATESAN, Ravi. Cummins Engine Fleets Its Factory. Harvard Business Review, Marzo/Abril de 1990, pp. 120-27.

Figura 4.2



Este caso es ilustrativo de las líneas de producción automáticas, y un indicio de lo que pueden ser los procesos de Manufactura Flexibles.

4.3 Planteamiento del modelo de simulación.

4.3.1 Respuesta al Tópico presentado.

a. Análisis de la situación.

El caso presentado tiene varios huecos de información. Por ejemplo, en lo que a las dimensiones de la línea de producción automática se refiere, no es lo mismo simular una de enormes proporciones de rieles, que una pequeña, ya que esto afecta en la rapidez y disposición de las charolas automáticas. En el caso concreto, la simulación se realizó de unas dimensiones fijadas libremente, y sobre esta base se hicieron los experimentos. El estudio consistió en encontrar el número óptimo de charolas, para que la producción alcanzara su máximo, dadas las condiciones actuales de producción.

No se tomaron en cuenta los tiempos de mantenimiento ni de preparación de las máquinas. Así como también se hizo a un lado todas las condiciones de costos de producción, acarreo, puntos de control, centros de trabajo, almacenes, etc. Tampoco se tomaron en cuenta los requerimientos de recargo de baterías de las charolas automáticas. Todo esto ayudó a simplificar el modelo y hacerlo del tamaño necesario para realizar el estudio central.

Se supuso que las máquinas de la serie 30, 40 y 50 tenían una distribución normal. Cuando el sistema es detenido, el único lugar donde se deben de colocar las charolas vacías es después de el robot de descarga y antes del de descarga, de otra manera, las condiciones lógicas de la líneas automática provocarán que el sistema se detenga.

b. Cálculos previos a la simulación.

Tomando en cuenta las dimensiones de la línea, velocidades de las charolas automáticas, las velocidades de los robots, y las condiciones de operación de las máquinas, se obtuvieron en un primer plano los siguientes resultados teóricos sobre la situación:

Centros de trabajo

Tabla 4.2

| | Tpo de Op | Tpo CyD | Total | Promedio | Distancia | D final | D total |
|-------------|-----------|---------|-------|----------|-----------|---------|---------|
| W10 | 8.5 | 2.00 | 10.50 | 10.50 | 3 | | |
| W20 | 8.6 | 2.00 | 10.60 | | 3 | | |
| W30 | 26.5 | 2.00 | 28.50 | | 8 | | |
| W40 | 26.5 | 2.00 | 28.50 | | 2 | | |
| W50 | 26.5 | 2.00 | 28.50 | 9.50 | 2 | 30 | 48 |
| W31 | 26.5 | 2.00 | 28.50 | | 20 | | |
| W41 | 26.5 | 2.00 | 28.50 | | 2 | | |
| W51 | 26.5 | 2.00 | 28.50 | 9.50 | 2 | 18 | 48 |
| W32 | 26.5 | 2.00 | 28.50 | | 30 | | |
| W42 | 26.5 | 2.00 | 28.50 | | 2 | | |
| W52 | 26.5 | 2.00 | 28.50 | 9.50 | 2 | 8 | 48 |
| Robot 1 Crg | 9 | 0.00 | 9.00 | 9.00 | 0 | | |
| Robot 2 Dsg | 8.5 | 0.00 | 8.50 | 8.50 | 0 | | |

Tabla 4.3

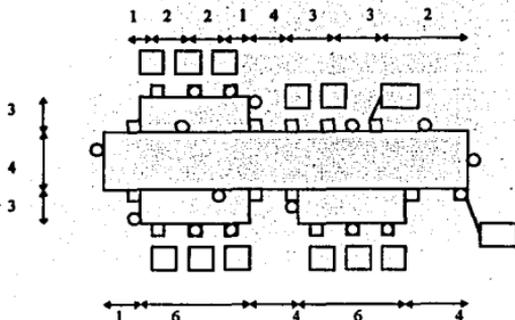
| | |
|----------------------------------|-------------|
| Tiempo que tarda una unidad | 12.72 un |
| Número máximo de unidades/1000ut | 94.34 un/ut |
| Cuello de Botella, W20, con | 10.60 un/ut |
| Velocidad de charolas | 1.00 d/ut |
| Distancia total que se recorre | 48.00 d |

En estos resultados se puede observar que el máximo valor de producción que se puede esperar en el intervalo de 1000 unidades de tiempo (ut), es de 94.34. Este valor se obtuvo dividiendo el tiempo total disponible entre el cuello de botella de la línea, que en este caso resultó ser el centro W20. Hay que aclarar que la tabla anterior llama tiempo de carga y descarga al tiempo que le toma a la charola ser cargada o descargada, en concreto 1 ut, este tiempo es independiente del tiempo que dura un proceso concreto, y es igual para todos los puntos de carga y descarga. Las demás conclusiones las dejo a la imaginación del amable lector.

c. Formulación del modelo de la simulación.

El modelo que se formuló para esta línea está resumido en el siguiente dibujo, que indica también la distancia de los rieles que unen a los centros de trabajo entre sí. Al momento de simular a los robots de carga y descarga, por simplicidad se eligió no representarlos con un centro de trabajo nuevo, pues al tener tiempo de proceso constante, y no tomar en cuenta los mantenimientos ni el tiempo de preparación, me pareció más conveniente aumentar simplemente el tiempo de carga y descarga para estos puntos de control, y desembarcarlos directamente en los centros de entrada y salida de material al sistema.

Figura 4.3



Modelo de la línea Automática de Producción.

Para el diagrama anterior, los cuadrillos son puntos de control, éstos están numerados del 1 al 19 comenzando por el punto de control de carga, y terminando con el punto de control de descarga del sistema, el 19. Cuadrados más grandes significan los centros de trabajo, según el diagrama presentado en el caso original. Los pequeños círculos representan las charolas de producto, algunas irán llenas y otras vacías. Es importante notar que las charolas vacías o están detenidas enfrente de un centro de trabajo esperando el producto procesado, o están entre el punto de descarga y el de carga. Si una charola vacía se mueve hasta llegar a un punto de control de carga para un proceso, al no tener nada que procesar, el sistema se bloquea. Los puntos de control que dirigen el tráfico de las charolas, son los siguientes: el PC 4, que envía alternativamente material al punto 5,10, 15. Esto asegura que alternativamente se enviará material a cada grupo de centros de trabajo, de manera que se equilibren. Los puntos de control que también controlan el tráfico son el 7,12,17, que envían los productos terminados directamente al PC 19, es decir, al de descarga del sistema. Si esto no fuera así, cada charola se detendría en todos los PC para ser procesada. Las charolas además están programadas para detenerse en los centros de trabajo por los que pasen si están llenas, y si están vacías esperar, pues si no, en cuanto la máquina empiece su trabajo se irían y no habría donde poner el producto una vez terminado el proceso. Además, los rieles por donde circulan las charolas están programados para sólo moverse en un sentido, de manera que ninguna charola puede jamás regresar. Esta logística es la parte fundamental del problema que se presenta, y cuesta tiempo determinar la manera más sencilla de funcionamiento. Es claro que entre menos reglas lógicas se tengan, con más facilidad se va a funcionar.

4.3.2 Simulación en Xcell+.

La siguiente es el resultado de la simulación del modelo en Xcell+, cuando se trató del modelo óptimo, encontrado con 16 charolas de transporte, en base a las dimensiones actuales de la línea. Estos resultados indican el estado final de la línea después de 1000 unidades de tiempo. Es importante notar que aunque el número de charolas es de 16, no se indica el estado actual mas que de algunas de ellas, sólo de las que se encuentran en algún punto de control. Después de estos resultados se muestran los experimentos que se realizaron en base al modelo, todos ellos con variante en el número de charolas, de manera que se obtiene el número óptimo de charolas a utilizar. Si la dimensión real de la línea se modifica, el sistema lo resentirá, actualmente se considera, por cada pie de riel, 1 unidad de tiempo, para efectos de la velocidad con que las charolas se transportan.

Tabla 4.4 Pruebas para distinto número de charolas.

| Número | %WC ocpdo | %charola Ocpdo | Producción |
|--------|-----------|----------------|------------|
| 5 | 25.0% | 36.0% | |
| 6 | 30.0% | 36.0% | |
| 8 | 40.0% | 35.0% | 46 |
| 10 | 50.0% | 35.0% | 57 |
| 12 | 60.0% | 36.0% | 71 |
| 14 | 69.0% | 36.0% | 82 |
| 16 | 80.0% | 36.0% | 92 |
| 18 | 82.0% | 41.0% | 95 |
| 20 | 82.5% | 47.0% | 94 |

A continuación se muestra una descripción del modelo en XCELL+, cada uno de los elementos contiene básicamente la información necesaria para leer la estructura simulada.

Factory FMS:

WorkCenter W10, has Processes named:
P (current Process)

ControlPoint 2 pickup all normal-output
ControlPoint 2 dropoff all X-input
units processed (period,cum): 92 92
% of time (procg,setup,min): 78.05 0.00 0.00

Process P at WorkCenter W10

inputs: X-input from ControlPoint 2; no Y-input
normal-output to: ControlPoint 2
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <CONSTANT>
every procg time is: 8.50
current state: BUSY processing until 1001.51
units processed (period,cum): 92 92

WorkCenter W20, has Processes named:
P (current Process)

ControlPoint 3 pickup all normal-output
ControlPoint 3 dropoff all X-input
units processed (period,cum): 92 92
% of time (procg,setup,min): 72.97 0.00 0.00

Process P at WorkCenter W20

inputs: X-input from ControlPoint 3; no Y-input
normal-output to: ControlPoint 3
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <CONSTANT>
every procg time is: 8.60
current state: BUSY processing until 1002.51
units processed (period,cum): 92 92

WorkCenter W30, has Processes named:
P (current Process)

ControlPoint 5 pickup all normal-output
ControlPoint 5 dropoff all X-input
no breakdowns or scheduled maintenance
units processed (period,cum): 30 30
% of time (procg,setup,min): 80.83 0.00 0.00

Process P at WorkCenter W30

inputs: X-input from ControlPoint 5; no Y-input
normal-output to: ControlPoint 5
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.30 28.00
current state: BUSY processing until 1005.93
units processed (period,cum): 30 30

WorkCenter W40, has Processes named:
P (current Process)

ControlPoint 6 pickup all normal-output
ControlPoint 6 dropoff all X-input
units processed (period,cum): 30 30
% of time (procg,setup,min): 81.23 0.00 0.00

Process P at WorkCenter W40

inputs: X-input from ControlPoint 6; no Y-input
normal-output to: ControlPoint 6
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.30 28.00
current state: BUSY processing until 1003.29
units processed (period,cum): 30 30

WorkCenter W50, has Processes named:
P (current Process)

ControlPoint 7 pickup all normal-output
ControlPoint 7 dropoff all X-input
units processed (period,cum): 30 30
% of time (procg,setup,min): 80.83 0.00 0.00

Process P at WorkCenter W50

inputs: X-input from ControlPoint 7; no Y-input
normal-output to: ControlPoint 7
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.30 28.00
current state: BUSY processing until 1002.91
units processed (period,cum): 30 30

WorkCenter W31, has Processes named:
P (current Process)

ControlPoint 10 pickup all normal-output
ControlPoint 10 dropoff all X-input
units processed (period,cum): 30 30
% of time (proc,g,setup,mtn): 80.31 0.00 0.00

Process P at WorkCenter W31
inputs: X-input from ControlPoint 10; no Y-input
normal-output to: ControlPoint 10
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.50 28.00
current state: IDLE, no material
units processed (period,cum): 30 30

WorkCenter W41; has Processes named:
P (current Process)
ControlPoint 11 pickup all normal-output
ControlPoint 11 dropoff all X-input
units processed (period,cum): 30 30
% of time (proc,g,setup,mtn): 80.94 0.00 0.00

Process P at WorkCenter W41
inputs: X-input from ControlPoint 11; no Y-input
normal-output to: ControlPoint 11
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.50 28.00
current state: IDLE, no material
units processed (period,cum): 30 30

WorkCenter W51; has Processes named:
P (current Process)
ControlPoint 12 pickup all normal-output
ControlPoint 12 dropoff all X-input
units processed (period,cum): 31 31
% of time (proc,g,setup,mtn): 80.67 0.00 0.00

Process P at WorkCenter W51
inputs: X-input from ControlPoint 12; no Y-input
normal-output to: ControlPoint 12
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.50 28.00
current state: BUSY processing until 1024.56
units processed (period,cum): 31 31

WorkCenter W32; has Processes named:
P (current Process)
ControlPoint 13 pickup all normal-output
ControlPoint 13 dropoff all X-input
units processed (period,cum): 31 31
% of time (proc,g,setup,mtn): 80.43 0.00 0.00

Process P at WorkCenter W32
inputs: X-input from ControlPoint 13; no Y-input
normal-output to: ControlPoint 13
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.50 28.00
current state: BUSY processing until 1017.13
units processed (period,cum): 31 31

WorkCenter W42; has Processes named:
P (current Process)
ControlPoint 16 pickup all normal-output

ControlPoint 16 dropoff all X-input
units processed (period,cum): 31 31
% of time (proc,g,setup,mtn): 80.30 0.00 0.00

Process P at WorkCenter W42
inputs: X-input from ControlPoint 16; no Y-input
normal-output to: ControlPoint 16
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.50 28.00
current state: BUSY processing until 1014.57
units processed (period,cum): 31 31

WorkCenter W52; has Processes named:
P (current Process)
ControlPoint 17 pickup all normal-output
ControlPoint 17 dropoff all X-input
units processed (period,cum): 31 31
% of time (proc,g,setup,mtn): 80.20 0.00 0.00

Process P at WorkCenter W52
inputs: X-input from ControlPoint 17; no Y-input
normal-output to: ControlPoint 17
reject-output: % reject: 0.00
group: 1 major-setup: 0.00 minor-setup: 0.00
processing-time distn: <UNIFORM>
min,mean,max: 25.00 26.50 28.00
current state: BUSY processing until 1014.63
units processed (period,cum): 31 31

ReceivingArea R1 supplies units to:
no cells supplied directly
supplies pickups for CtrPts: 1
Part supplied in P
tag 1 unit in 10
arrivals: <UNLIMITED>
storage-capacity: <UNLIMITED>
units started (period,cum): 91 91

ShippingArea S1 receives units of:
no cells supply unit directly
receives dropoffs from ControlPoints: 19
Part accepted in P
shipments: <CONTINUOUS>
storage-capacity: <UNLIMITED>
units accepted (period,cum): 92 92

ControlPoint 1 (row 9, column 25)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 9.00 1.00
pickup loads from ReceivingArea R1
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: incoming from CtrPt 19
Path at left: outgoing to CtrPt 2

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 91 91

Carrier 1 present:
loaded; contains P
shown on chart-line: 23

currently in pickup
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 33.26 33.26
units carried (period,cum) 34 34
cost (same for all Cars): 10.00

ControlPoint 2 (row 9, column 22)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W10
dropoff loads to WorkCenter W10
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: incoming from CtrlPt 1
Path at left: outgoing to CtrlPt 3

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 92 92

Carrier 2 present:

empty
shown on chart-line: 12
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 33.27 33.27
units carried (period,cum) 34 34
cost (same for all Cars): 10.00

ControlPoint 3 (row 9, column 19)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W20
dropoff loads to WorkCenter W20
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: incoming from CtrlPt 2
Path at left: outgoing to CtrlPt 4

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 92 92

Carrier 3 present:

empty
shown on chart-line: 16
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 36.12 36.12
units carried (period,cum) 34 34
cost (same for all Cars): 10.00

ControlPoint 4 (row 9, column 15)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <NOT WAIT>
automatic traffic-control rules:

empty Cars exit in clockwise rotation
loaded Cars routed to following CtrlPts, in rotation:
5,10,15

Path at top: outgoing to CtrlPt 5
Path at right: incoming from CtrlPt 3
Path at left: outgoing to CtrlPt 8

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 92 92

no Carrier present

ControlPoint 5 (row 6, column 14)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W30
dropoff loads to WorkCenter W30
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: incoming from CtrlPt 4
Path at left: outgoing to CtrlPt 6

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 30 30

costs (capital, unit operating): 20.00 1.00

Carrier 4 present:

empty
shown on chart-line: 13
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 37.19 37.19
units carried (period,cum) 34 34
cost (same for all Cars): 10.00

ControlPoint 6 (row 6, column 12)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W40
dropoff loads to WorkCenter W40
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: incoming from CtrlPt 5
Path at left: outgoing to CtrlPt 7

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 30 30

Carrier 5 present:

empty
shown on chart-line: 19
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 36.60 36.60
units carried (period,cum) 34 34
cost (same for all Cars): 10.00

ControlPoint 7 (row 6, column 10)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W50
dropoff loads to WorkCenter W30
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars routed to following CtrlPts, in rotation:
19

Path at right: incoming from CtrlPt 6
Path at left: outgoing to CtrlPt 8

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 30 30

Carrier 6 present:
empty

shown on chart-line: 20
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 34.72 34.72
units carried (period,cum) 34 34
cost (same for all Cars): 10.00

ControlPoint 8 (row 9, column 9)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <NOT WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at top: incoming from CtrlPt 7
Path at right: outgoing to CtrlPt 4
Path at left: outgoing to CtrlPt 9

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 91 91

no Carrier present

ControlPoint 9 (row 11, column 9)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <NOT WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrlPt 13
Path at bottom: outgoing to CtrlPt 10
Path at left: incoming from CtrlPt 8

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum) 91 91

no Carrier present

ControlPoint 10 (row 14, column 10)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W31
dropoff loads to WorkCenter W31
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrlPt 11
Path at left: incoming from CtrlPt 9

nbr of loaded Cars enroute here: 1
nbr Cars passed (period,cum): 30 30

no Carrier present

ControlPoint 11 (row 14, column 12)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W41
dropoff loads to WorkCenter W41
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrlPt 12
Path at left: incoming from CtrlPt 10

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 30 30

no Carrier present

ControlPoint 12 (row 14, column 14)
zone: 1 limit: 2000 Cars; contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W31
dropoff loads to WorkCenter W31
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars routed to following CtrlPts, in rotation:
19

Path at right: outgoing to CtrlPt 13
Path at left: incoming from CtrlPt 11

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum) 31 31

Carrier 7 present:
empty

currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum) 36.33 36.33
units carried (period,cum) 35 35
cost (same for all Cars): 10.00

ControlPoint 13 (row 11, column 15)
zone: 1 limit: 2000 Cars, contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <NOT WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrPt 14
Path at bottom: incoming from CtrPt 12
Path at left: incoming from CtrPt 9

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 93 93

no Carrier present

ControlPoint 14 (row 11, column 19)
zone: 1 limit: 2000 Cars, contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <NOT WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrPt 18
Path at bottom: outgoing to CtrPt 15
Path at left: incoming from CtrPt 13

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 92 92

no Carrier present

ControlPoint 15 (row 14, column 20)
zone: 1 limit: 2000 Cars, contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W32
dropoff loads to WorkCenter W32
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrPt 16
Path at left: incoming from CtrPt 14

nbr of loaded Cars enroute here: 1
nbr Cars passed (period,cum): 31 31

Carrier # present:
empty
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum): 37.67 37.67
units carried (period,cum): 34 34
cost (same for all Cars): 10.00

ControlPoint 16 (row 14, column 22)
zone: 1 limit: 2000 Cars, contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W42
dropoff loads to WorkCenter W42

Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrPt 17
Path at left: incoming from CtrPt 13

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 31 31

Carrier 9 present:
empty
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum): 37.40 37.40
units carried (period,cum): 34 34
cost (same for all Cars): 10.00

ControlPoint 17 (row 14, column 24)
zone: 1 limit: 2000 Cars, contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
pickup loads from WorkCenter W52
dropoff loads to WorkCenter W52
Cars <WILL NOT> engage WorkCenter
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <WAIT>; empty Cars <WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars routed to following CtrPts, in rotation:
19

Path at right: outgoing to CtrPt 18
Path at left: incoming from CtrPt 16

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 31 31

Carrier 10 present:
empty
currently waiting (for pickup)
run-time remaining: 200.00 percent battery charge: 100.00
percent time loaded (period,cum): 36.28 36.28
units carried (period,cum): 34 34
cost (same for all Cars): 10.00

ControlPoint 18 (row 11, column 25)
zone: 1 limit: 2000 Cars, contains: 16 Cars

transit,pickup,dropoff times: 1.00 1.00 1.00
Car traffic-control: <AUTOMATIC>
waiting: loaded Cars <NOT WAIT>; empty Cars <NOT WAIT>
automatic traffic-control rules:
empty Cars exit in clockwise rotation
loaded Cars exit in clockwise rotation

Path at right: outgoing to CtrPt 19
Path at bottom: incoming from CtrPt 17
Path at left: incoming from CtrPt 14

nbr of loaded Cars enroute here: 0
nbr Cars passed (period,cum): 92 92

no Carrier present

ControlPoint 19 (row 11, column 29)

zone: 1 limit: 2000 Cars, contains: 16 Cars
 transit_pickup/dropoff times: 1.00 1.00 2.50
 dropoff loads to ShippingArea S1
 Car traffic-control: <AUTOMATIC>
 waiting: loaded Cars <NOT WAIT>, empty Cars <NOT WAIT>
 automatic traffic-control rules:
 empty Cars exit in clockwise rotation
 loaded Cars exit in clockwise rotation
 Path at top: outgoing to CtrlPt 1
 Path at left: incoming from CtrlPt 18

nbr of loaded Cars enroute here: 1
 nbr Cars passed (period,cum): 92 92

Carrier 11 present:
 empty
 shown on chart-line: 14
 currently in dropoff
 run-time remaining: 200.00 percent battery charge: 100.00
 percent time loaded (period,cum) 36.41 36.41
 units carried (period,cum) 34 34
 cost (same for all Cars): 10.00

4.4 Conclusiones.

4.4.1 Conclusiones del Caso: Líneas Automáticas de Producción.

A continuación se muestran las tablas principales de los resultados obtenidos con el modelo en XCELL+, estas tablas resumen el comportamiento del sistema en 1000 u.f.. Es importante que se preste atención en la tabla que ilustra el estado de cada uno de los contenedores del sistema, así como su utilización.

La tabla que tenemos a continuación indica la utilización de cada uno de los centros de trabajo del modelo, es importante notar como es que nunca están bloqueados, ni tienen tiempos de preparación. Esto básicamente es porque lo que interesa en este modelo es la logística de funcionamiento de la banda transportadora y no el estudio de tiempos del sistema.

Tabla 4.3
 Resultados para FMS
 Utilización de Centros de Trabajo al tpo: 1000,0

| | %Ocupado | | %Preparación | | %Mantenimiento | | %Bloqueado | |
|-----|----------|-----------|--------------|-----------|----------------|-----------|------------|-----------|
| | Periodo | Acumulado | Periodo | Acumulado | Periodo | Acumulado | Periodo | Acumulado |
| W10 | 78.049 | 78.049 | 0 | 0 | 0 | 0 | 0 | 0 |
| W20 | 78.974 | 78.974 | 0 | 0 | 0 | 0 | 0 | 0 |
| W30 | 80.85 | 80.85 | 0 | 0 | 0 | 0 | 0 | 0 |
| W40 | 81.235 | 81.235 | 0 | 0 | 0 | 0 | 0 | 0 |
| W50 | 80.835 | 80.835 | 0 | 0 | 0 | 0 | 0 | 0 |
| W31 | 80.507 | 80.507 | 0 | 0 | 0 | 0 | 0 | 0 |
| W41 | 80.94 | 80.94 | 0 | 0 | 0 | 0 | 0 | 0 |
| W51 | 80.667 | 80.667 | 0 | 0 | 0 | 0 | 0 | 0 |
| W32 | 80.428 | 80.428 | 0 | 0 | 0 | 0 | 0 | 0 |
| W42 | 80.301 | 80.301 | 0 | 0 | 0 | 0 | 0 | 0 |
| W52 | 80.196 | 80.196 | 0 | 0 | 0 | 0 | 0 | 0 |

A continuación se muestra la tabla de utilización de los carritos o charolas de transporte de material dentro de la banda de trabajo. Cada uno de los datos muestran el estado final, a tiempo 1000 u.f., de los carritos. Es importante notar como es que los carritos están ocupados en promedio un 36% de su tiempo, este número es óptimo y máximo, es decir, no mejoraría de ninguna manera la eficiencia de la banda si aumentamos el número de carritos. Esta banda está funcionando al máximo de su capacidad, respecto a los datos de entrada de material al sistema.

Tabla 4.6 UTILIZACION de charolas

| | al tiempo : 1000 | | localización | | número de periodo | CARGAS acumuladas | % Tiempo de Carga | |
|----|------------------|----------|--------------|---------|----------------------|----------------------|-------------------|-----------|
| | Actual | Estado | Fila | columna | | | periodo | acumulado |
| 1 | vacío | espera | 9 | 22 | 34 | 34 | 35.27 | 35.27 |
| 2 | vacío | espera | 6 | 14 | 34 | 34 | 37.19 | 37.19 |
| 3 | vacío | descarga | 11 | 29 | 34 | 34 | 36.41 | 36.41 |
| 5 | vacío | espera | 9 | 19 | 34 | 34 | 36.12 | 36.12 |
| 8 | vacío | espera | 6 | 12 | 34 | 34 | 36.6 | 36.6 |
| 9 | vacío | espera | 6 | 10 | 34 | 34 | 34.72 | 34.72 |
| 11 | cargado | transito | 12 | 9 | 35 | 35 | 37.68 | 37.68 |
| 12 | cargado | carga | 9 | 25 | 34 | 34 | 35.26 | 35.26 |
| 13 | vacío | transito | 9 | 26 | 33 | 33 | 35.44 | 35.44 |
| 14 | cargado | transito | 9 | 12 | 36 | 36 | 35.72 | 35.72 |
| 15 | vacío | espera | 14 | 14 | 35 | 35 | 36.33 | 36.33 |
| 16 | vacío | espera | 14 | 22 | 34 | 34 | 37.4 | 37.4 |
| 17 | cargado | transito | 14 | 11 | 35 | 35 | 37.79 | 37.79 |
| 18 | vacío | espera | 14 | 20 | 34 | 34 | 37.67 | 37.67 |
| 19 | cargado | transito | 11 | 18 | 34 | 34 | 35.34 | 35.34 |
| 20 | vacío | espera | 14 | 24 | 34 | 34 | 36.28 | 36.28 |

Finalmente se muestran los tiempos de las piezas marcadas dentro del sistema, en promedio, según las piezas muestreadas, cuanto tiempo tarda una pieza desde que entra hasta que sale terminada. Nótese como dura 63 u.t. esperando ser recogida para llevarse al siguiente proceso. Este problema deriva de los encadenamientos de los centros de trabajo.

Tabla 4.7 Flujo de tpo, Area de Embarque: 1000
(En base al arribo de unidades marcadas al área de embarque)

| | Número | Flujo promedio | Promedio de proceso | Promedio de espera |
|----|--------|-------------------|------------------------|-----------------------|
| SI | 8 | 159.771 | 96.488 | 63.283 |

Para finalizar decimos que en un periodo de 1000 u.t. se procesaron un total de 91 piezas. Este número está a un 83 % del óptimo de la línea. Se sugiere corregir tiempos de recepción de material, así como mejorar la sincronización de los centros de trabajo. Añadir más carritos no va a solucionar el problema de producción de la línea automática.

4.4.2 Conclusiones de la técnica de simulación.

Este caso es sumamente ilustrativo en lo que se refiere a las líneas automáticas de producción. Cuando se trata de simular la logística concreta, al detalle, de qué pase con los recorridos de cada carrito dentro de una línea lógica-inteligente, de producción es importante hacerlo con la herramienta adecuada. En este caso PACION no pudo ofrecer mucho al respecto, pues PACION se utiliza principalmente en la simulación de modelos "estáticos" y no en la logística de comportamiento de cada uno de los carritos.

Este caso trata de comprobar cuál es el número óptimo de carritos y si las reglas lógicas de funcionamiento son las óptimas o no. Por esto el modelo está centrado en el comportamiento de los carritos y no en los centros de trabajo (por esta razón no se considera mantenimiento, tiempos de preparación, etc.). El modelo es del tamaño justo para investigar lo que nos interesa.

PACION podría ayudar a esta simulación con el análisis numérico de los carritos, con el subprograma ANIMATOR, pero escapa a las intenciones de esta tesis. Hay paquetes de simulación totalmente enfocados al estudio de sistemas automáticos de producción así como en la logística del control automático, se puede encontrar algunos ejemplos en los mencionados en el Capítulo 1 de el presente trabajo.

Conclusiones.

Generales.

Es tiempo de terminar con este trabajo. Cuando una persona se plantea investigar en torno a un tópico complicado, con frecuencia le queda el sabor de que nunca terminará, siempre existirá algo nuevo, un nuevo matiz, aplicación, etc., que se puede conocer, ideas nuevas que aplicar. Quedan muchas inquietudes en el ambiente, y sin embargo, alguna vez tiene que terminar.

Este trabajo representa la investigación de 1 año, con etapas muy intensas, muchas veces de reflexión de problemas. Ha logrado que, en momentos, los problemas logísticos angustien al investigador, otras distraerlo completamente de la realidad, quitarle sueño, etc. Pero pienso al final, que es un trabajo que ha valido la pena.

¿Porque elegí este tema?, bueno, porque la simulación representó durante la carrera una caja negra, una herramienta que sólo algunos pueden tocar, una ciencia que para utilizar con eficacia hay que ser genio. Por razones inversas a las que la luz atrae a la palomillas, me interesé por el tema. Algo de utilidad tenía que haber en el fondo de esta oscura herramienta. Lo encontré.

Este trabajo concluye en la realización de tres objetivos:

- a) Demostrar el funcionamiento real de sistemas modernos de control de producción. Mostrar que dependiendo del caso, una técnica es mejor que otra.
- b) Demostrar que la Simulación es una herramienta muy aplicable en los problemas de producción. Que no es una técnica abstracta, e inutilizable, que es utilísima y próxima a la realidad del ingeniero industrial.
- c) Demostrar que, dentro de los distintos programas de simulación, existen diferencias cualitativas que marcan profundas distinciones en el momento de elegir la mejor herramienta de simulación. Para este objetivo se utilizaron dos programas de simulación Xcell+, y PASION, en la presente tesis.

Los Sistemas modernos de manufactura.

El presente trabajo pretende ilustrar cómo es que los Sistemas Modernos de Manufactura funcionan en la realidad, en caso concretos y reales. Muchas de las técnicas actuales de control de la producción se anuncia como lo "Non, plus ultra", se atacan unas a otras, se pisan y viven tratando de mostrar cuál es mejor que las demás. Los entusiastas con frecuencia caen enredados bajo los encantos de alguna de ellas, y pierden la objetividad para poder voltear hacia las demás. Cada uno de los "inventores" de las

técnicas, da un toque de misticismo a su creación utilizando palabras nuevas que no significan más que cosas viejas. En los últimos años teorías como Calidad Total, Justo a Tiempo, Teoría de Restricciones, MRP I y II, Tecnología de Grupos, Reingeniería, Manufactura Exitosa, etc., no han hecho más que confundir a muchos directores de producción de nuestro país. Con mucha frecuencia las empresas americanas no tienen la suficiente paciencia para esperar los resultados de alguna de las técnicas de producción, y cambian rápidamente a otra nueva, con más brillo, esperando que ésta sí sea una receta mágica.

Sin embargo, ninguno de estos Sistemas Modernos de Manufactura es una receta mágica. Por más que algunos incluso las autodenominen "filosofías de vida", "filosofías organizacionales" o similares. Pienso que cada una de ellas tiene una importante aportación que dar, cada una tiene su mensaje. Sin embargo en los Sistemas de Producción, a diferencia de la Religión, no hay dogmas (la razón es sencilla, los sistemas de producción los ha elaborado el hombre a través del tiempo, en cambio la Religión la hizo Dios en un instante). Por tanto, siempre hay cosas nuevas que aprender, ideas nuevas que aplicar, enfoques nuevos que estudiar. En la Inmensidad de los problemas de producción, ensanchada constantemente por las innovaciones tecnológicas, cada problema tiene su traje a la medida. La habilidad del Director de Producción radica indiscutiblemente en la capacidad que tiene para aplicar la idea justa al problema exacto. Esto implica un conocimiento de los Sistemas Modernos de Manufactura, pero no significa que esté casado con ninguno de ellos. La mezcla de las aportaciones pueden hacer trajes a la medida estupendos, los esquemas generales nunca ajustarán a los problemas.

Como claras ilustraciones de lo que intento decir, están los casos presentados en esta Tesis. Especialmente el capítulo 3, que trata de JIT vs TOC. En estos capítulos se muestra cómo lo más importante no es el conocimiento de una técnica determinada de Producción. Más importante es el sentido común empleado cuando se implementa la técnica en cuestión. Por tanto, un director de producción contemporáneo debe trabajar en dos grandes oportunidades:

a) Conocer muy bien cuáles son las nuevas técnicas de producción, y evitar así una obsolescencia. Ser experto en lo nuevo, saber manejar herramientas modernas y poderosas de análisis. Tener los conocimientos académicos del mejor. **Ciencia.**

b) Tener el suficiente sentido común para saber aterrizar las teorías. Aportar, además, a las teorías su experiencia e inteligencia. No aplicar recetas, mejor aplicar la razón. No cerrarse a las nuevas ideas. **Visión.**

Claro que esto es difícil, y con frecuencia imposible que una sola persona domine todo esto. En casos como éste bastará la formación de un equipo de producción, capaz de cumplir estas características. Lo importante es cuidar de no tener miedo al progreso científico, y tener la suficiente visión para implementar racionalmente los nuevos modelos.

La Simulación su: realidad y expectativas.

En cuanto al segundo objetivo de la tesis diremos que: en este trabajo se pretendió ilustrar la aplicación de algunos Sistemas Modernos de Manufactura a casos concretos, se propició la comparación entre ellos y su mezcla. Además se intentó mostrar claramente el análisis de estos casos mediante la técnica de la simulación, que es una herramienta fundamental para las empresas de manufactura para alcanzar el "éxito" en la producción, llámense como le llamen.

Respecto a la técnica utilizada para el análisis de los casos, me pareció descubrir un mar sin orillas. Las técnicas derivadas de la adecuada explotación de la computadora son cada vez más sencillas de utilizar para los no expertos. Aunque siempre es necesario un grado adecuado de estudios. En los casos ilustrados es claro que los modelos eran relativamente sencillos, al menos pequeños. Claro que con cualquiera de los paquetes mencionados en la tesis se pueden simular plantas enteras de producción. Se pueden hacer modelos enormes y muy minuciosos, pero si no se tiene un motivo concreto de hacer esto,

no se sabrá que hacer con tanta información. En la gran mayoría de los casos conviene asilar el problema en un modelo sencillo, lo suficientemente detallista para que resuelva nuestro problema. Esto es lo que se intentó aplicar en los casos anteriores, pienso que se hizo con éxito.

La simulación, como se menciona en la introducción de la tesis, no es la piedra filosofal, no resuelve los problemas automáticamente, no hace propuestas innovadoras, no piensa. Sólo es una herramienta para probar, con ciertas restricciones, si la solución planteada funcionará o no. También sirve para entender mejor los problemas, y con frecuencia entender mejor sirve para aportar nuevas ideas. No se puede esperar que la simulación resuelva los problemas, el que los resuelve es el analista. Por esto es que la simulación exige de los analistas un profundo dominio tanto de simulación como de la disciplina que se pretende simular (son expertos en dos ciencias). Si el analista es una persona bien preparada, con visión o "talento", posiblemente encuentre en la simulación una técnica eficaz para corroborar sus teorías, y de mostrarlas visualmente a los demás. En conclusión, ser simulador exitoso implica saber de simulación y de producción, pero además, tener el suficiente "talento" para resolver creativamente los problemas.

La simulación no persigue la exacta imitación de la realidad. Esto es imposible. Sólo pretende contar con modelos suficientemente buenos como para utilizarlos en las pruebas. De la formación de los modelos de cada uno e los casos de la presente tesis, se puede basar el éxito de la técnica. Por esto es que dentro de la gran cantidad de modelos que se formularon para cada caso, sólo se mostraron los más relevantes. Es importante aclarar que quedaron muchas versiones en los archivos, que no se muestran en este trabajo, pero cada una de ellas aportó o significó un avance en una etapa determinada del proceso. Una etapa crucial en la elaboración de un modelo es la correcta elección de las variables importantes del fenómeno estudiado. Desde este punto de vista, los modelos de sistemas de producción son sencillos y bastante buenos. Pienso que sí, dentro de la simulación, hay algo que se puede aproximar a la realidad con mayor grado es la simulación de sistemas productivos, pues las variables importantes son fácilmente localizables, y son discretos. Sin embargo nunca servirán como bolas de cristal que nos digan el futuro, un modelo sólo dice lo que probablemente pase en el futuro, pero sólo eso, probablemente.

Siempre existe la posibilidad de fallar. La mayor aportación de la simulación de un sistema de producción es la mejor comprensión de un problema, de una técnica, de una circunstancia. Esta aportación no coloca en la posibilidad de hacer aportaciones que valen oro para una empresa. Una herramienta fundamental en cualquier grupo de consultores industriales tendrá que ser la simulación.

El trabajo presentado.

Respecto al tercer objetivo de esta tesis se ha hablado en las conclusiones de cada uno de los capítulos. Siempre, al finalizar el análisis se concluyen sobre el caso y sobre la técnica. Por esto, resumiré mis ideas sólo en los siguientes puntos:

a) Los programas, paquetes, y lenguajes de simulación difieren entre sí respecto a la capacidad y análisis que cada uno de ellos tiene. En ocasiones no se necesitan herramientas de simulación muy complicadas, basta con una buena modelación. Sin embargo existe una cantidad de problemas que necesitan de una complicada modelación y por tanto necesitan de mejores herramientas de trabajo, mas flexibles y completas. Esto es lo que hace diferentes a los programas, paquetes y lenguajes. Con un lenguaje orientado a objetos casi se puede programar cualquier cosa, aunque no siempre sea la manera más sencilla de hacerlo. Con un programa muy concreto se podrá abarcar sólo un área reducida de problemas, pero con mucha facilidad. Este es el caso de Xcell+, un programa concreto pero sencillo. PASION por su parte es relativamente mas complicado, pero más flexible y por tanto mejor para resolver problemas difíciles.

b) Por lo expuesta en el capítulo anterior, se concluye que en ciertos problemas de producción Xcell+ será mas sencillo y rápido de usar, pero necesita que nuestras ideas de solución no salgan de su esquema rígido. PASION permite implementar cualquier idea que se pueda programar, pero es más tardado. El tiempo en ocasiones es un factor decisivo para el analista. No podemos olvidar que, a fin de cuentas, la simulación es tan sólo una herramienta y no un fin en sí misma.

c) Como la simulación es una herramienta, hay que cuidar que los modelos sean de las dimensiones necesarias, no exorbitarse. Hay que cuidar también que el analista la vea como tal, herramienta, y no se eleve por el espacio sideral, realizando simulaciones que no tienen un objetivo concreto y claro. Se puede perder mucho el tiempo con esta herramienta, pues es bastante divertida.

d) Un punto importante en la comparación de herramientas de simulación es el despliegue de resultados. Ya se dijo, en el capítulo 1, que no basta con que el modelo sea confiable, es necesario que el que lo va a poner en práctica crea en el modelo. En ocasiones una buena presentación hace que los modelos luzcan verdaderos. Algunos paquetes son poco extraordinarios en lo que a presentaciones dinámicas se refiere, otros son muy restringidos, otros ofrecen mares sin orillas. Xcell+ es un paquete que tiene un esquema de presentación bonito pero fijo, es decir, presenta el movimiento de las piezas en pantalla a su paso por la planta, y las estadísticas finales, pero nada más. PASION además de esto, tiene un módulo de animación que permite dar mucha más vista al modelo a los resultados obtenidos, pero tiene la comodidad de un esquema rígido de presentación, que siempre se puede utilizar. Esto es muy práctico pues no interesa hacer un show con modelos que no estén terminados o aprobados.

e) Hay que cuidar también la diferencia de precios de la herramienta de simulación, hay algunos muy caros y poco útiles, pero eso sí, muy vistosos. Hoy en día se pueden obtener muy buenos paquetes pero no todos a un precio justo.

Bueno, es todo lo que tengo que decir respecto a este punto. Ojalá y tenga suerte aquel que tiene en sus manos la responsabilidad de elegir un paquete.

Bibliografía.

Textos y Libros citados:

BRIGHAM, E. Financical Management Theory and Practice. The Dryden Press, 6a edición.

CHASE, Richard and AQUILANO, Nicholas. Production and Operation Management: A Life Cycle Approach, 6a. edición Editorial Irwin, Boston 1992.

COSS BU, Raúl. Simulación, un enfoque práctico. Limusa Noriega Editores, México 1982.

CONWAY, MAXWELL, McClAIN WORONA. User's guide to XCELL+ Factory Modeling System, Release 4.0. The Scientific Press, Cornell University 1990.

FRANCIS and WHITE. Facility Layout and Location: An Analytical Approach, Prentice Hall, Englewood Cliffs, N.J. 1974.

FOGARTY, BLAKSTONE and HOFFMAN. Production and Inventory Management, 2a. edición, South Western 1991.

GOLDRATT and COX. La Meta: Un proceso de mejora continua, 1a. edición, ediciones Castillo, México 1991.

GORDON, Geoffrey. Simulación de Sistemas. Diana Editores. México 1980.

MONTGOMERY. Diseño y Análisis de Experimentos. Grupo Editorial Iberoamericana, México 1991.

NAYLOR, Thomas. Experimentos de Simulación en Computadoras con Modelos de Sistemas Económicos, 1a. Edición. Editorial Limusa. México DF 1977.

RAKZYNSKI, Stanislaw. Simulación por computadora. Colección Megabyte, Grupo Noriega Editores. México 1993.

SCHONENBERG, Richard. World Class Manufacturing: The Lessons of Simplicity Applied. Free Press. New York 1986. pp 112.

THE CHARLES STARK DRAPER LAB Inc. Flexibles Manufacturing Systems Handbook. Noyes Publication.

ZEIGLER B. Theory of Modeling and Simulation. Jhon Wiley and Sons.

Revistas y Ponencias Citadas:

ASKIN and SUBRAMANIAN. A Cost-Based Heuristic for Group Technology Configuration. International Journal of Production Research 25, no. 1 pp. 101. 1987.

CARSON, John. Convincing Users of Model's validity is challenging aspect of modeler's job. IIEI. Junio 1986.

CONWAY, MAXWELL, McCLAIN and THOMAS. The Role of Work In Process Inventory. Working paper 87-106. Johnson Graduate School of Management. Malott Hall, Cornell University, Ithaca, NY, 14853. Mayo 1987.

CRISTY, David and WATSON, Hugh. The Application of Simulation: A survey of Industry Practice. Interfaces 13, no. 5. October 1983.

GOLDRATT, E. Computerized Shop Floor Scheduling. International Journal of Production Research 26, no. 3, pp. 443. 1988.

KING J. Machines Component Group Formation for Technology. Proceedings of the Fifth International Conference on Production Research. pp. 193.

KUSIAK, Andrew. Flexible Manufacturing Systems: A Structural Approach. International Journal of Production Research 23, no. 6, pp. 1057. 1985.

MATWIJECT and KALASKY. Financial Justification for Simulation Software. Simulation, 1992 pp 114.

OTTINGER, L. Robotics for the IE: Terminology, types of Robots. Industrial Engineering, November 1981. pp. 30.

PADGETT, Mary Lou. Computational Intelligence, Neural Networks and Simulation applications. Ponencia del Simposium: "Simulación por Computadora e Inteligencia Artificial", Universidad Panamericana, México DF. Febrero 1994.

PRITSKER, Alan. Compilation of definitions of Simulation. Simulation, Society for Computer Simulation. Agosto 1979.

QUI NIETO, Jesús. Tesis: Fosisifa y Modelos Relativos a sistemas modernos de manufactura. Presentada en la División de Estudios de Posgrado de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México, como requisito para obtener el grado de Maestro en Ingeniería (Investigación de Operaciones). Ciudad Universitaria 1993.

RAKZYNSKI, Stanislaw. PASION. Lenguaje Overview. Ponencia del Simposium: "Symposium on Computer Simulation. The Society for Computer Simulation and Universidad Panamericana. México DF 1991.

RAKZYNSKI, Stanislaw. PASION-PASCAL related simulation language for small systems. Simulation 46(6). Junio pp 239. 1988.

VENIKATESAN, Ravi. Cummins Engine Flexes Its Factory. Harvard Business Review. Marzo-Abril pp. 120. 1990.

ANEXO 1: El Lenguaje PAsION¹.

Introducción.

PAsION es lenguaje de simulación orientado a procesos y eventos, diseñado para los usuarios de PASCAL. El lenguaje tiene dos niveles (procesos y eventos) en su estructura, y permite el uso de todas las herramientas y metodología de PASCAL. Además ofrece las ventajas de la programación orientada a objetos. PAsION proporciona las facilidades necesarias para el seguimiento secuencial de eventos aleatorios, colas y procesos cuasi-paralelos, pero discretos y continuos.

1.1 ¿Porqué un nuevo Lenguaje?²

Un vistazo a los catálogos de software para simulación puede resultar impresionante, uno se encuentra con una gran cantidad y variedad de lenguajes, y paquetes para la simulación. La idea de crear uno más resulta alocada. De cualquier forma, observando los software existentes y las recientes tendencias de programación, puede resultar que algunos de los programas existentes son obsoletos. Los nuevos programas de simulación (tanto lenguajes como paquetes) deben de ser orientados a objetos, no sólo porque es lo de moda si no porque esta técnica representa mayores ventajas que las técnicas tradicionales de simulación. Además, los nuevos software de simulación deben de estar orientados a objetos porque, sencillamente, el mundo real está orientado a objetos.

Segundo, no suele ser un buen camino para proporcionar herramientas de simulación, modificar lenguajes con más de 20 o 30 años de antigüedad, o crear extensiones de ellos. Un buen lenguaje de simulación, por su clara estructuración, favorable para los fines didácticos, es SIMULA, sin embargo este, no suele ser un lenguaje fácil ni rápido de enseñar, pues está íntimamente unido a ALGOL. Problemas similares resultan de MODULA-2 o MODSIM II (éste último programa es una nueva versión de SIMULA pero basada en C). Hablando claramente, un nuevo lenguaje de simulación debe de estar orientado a objetos y además tener las siguientes características:

- Procesos claros y eventos estructurados.
- Mecanismo de tiempo eficiente.
- Combinación de modelos continuos y discretos, hereditarios.
- Soporte gráfico.
- Simulación interactiva.
- Con módulos auxiliares (generadores de programas)

¹RACZYNSKI S. PAsION - Language overview. Symposium on Computer Simulation, The Society for Computer Simulation & Universidad Panamericana, Mexico Cd., Noviembre 1991.

²RACZYNSKI S. PAsION - PASCAL related simulation language for small systems. SIMULATION 46(6), Junio 1986, pp. 239-242.

1.2 El lenguaje PASION².

Recordemos algunos de los conceptos básicos de la simulación orientada a procesos. Para describir una secuencia de eventos debemos de especificar las operaciones de los eventos en el modelo temporal, y las interacciones entre los eventos. Un lenguaje orientado a procesos ofrece mucho más. A saber, define la estructura de un grupo de eventos mediante la introducción de diferentes procesos. Por "procesos" se entiende un segmento genérico de programas que declaran un tipo específico de objetos. Esta declaración describe las propiedades de los objetos que pueden ejecutar eventos dentro del marco de tiempo del modelo. De acuerdo con esta declaración, los objetos pueden ser creados en "tiempo corriente". Este funcionamiento nos aproxima a la programación orientada a objetos. Recordando que un lenguaje orientado a objetos debe de soportar el ocultamiento de información, abstracción de datos, esperas dinámicas y herencias, PASION no satisface completamente con todas estas características. Soporta la creación dinámica de objetos, ocultamiento de información y, en algunas modificaciones, la abstracción de datos mediante el mecanismo de procesos predefinidos y herenciales. El siguiente ejemplo muestra un programa en PASION, donde existen dos objetos que se activan:

Figura 1.

```
PROGRAM TRIGGER;
REF A,B:X; {A y B son nombres de objetos tipo X}

PROCESS X,2; {proceso tipo A}

    ATR N:STRING(6)
    EVENT ONE; {ONE es un evento}
    WRITELN(objeto activado:'.N');
    IF THIS=A THEN B.ONE:=TIME + 1.0
    ELSE A.ONE:=TIME + 1.0 ENDEC;

START {programa principal}
NEWPR A; A.N:='PEDRO';
NEWPR B; B.N:='JAIME';
A.ONE:=TIME;{el objeto A comienza inmediatamente. B espera}
$ {esto significa el final del programa}
```

Este es un programa completo en PASION. Las variables de referencia A y B son usadas para referir los dos objetos tipo X. La instrucción A.ONE:=TIME + 1.0 programa el evento ONE del objeto A para ser ejecutado en TIME + 1.0, donde TIME es el modelo de tiempo. THIS es una palabra reservada que se utiliza para referirse el objeto dentro de sus límites de alcance. Por consiguiente, THIS = A es verdadero cuando el objeto es llamado(conducido) por A y falso cuando el objeto es llamado (conducido) por B. Esto puede parecer que los dos objetos se activan uno al otro. La salida de este sencillo programa es la siguiente:

Figura 2.

```
Activa Object: PEDRO
Active Object: JAIME
Active Object: PEDRO
Active Object: JAIME
Active Object: PEDRO
... ETC.
```

Este es un ejemplo extremadamente sencillo. En la práctica, los programas de simulación pueden tener mas de 50 procesos declarados (tipos de objetos) y mas de 400 eventos en cada proceso. El número de objetos generados por un periodo de tiempo determinado depende de la complejidad de los objetos. Los

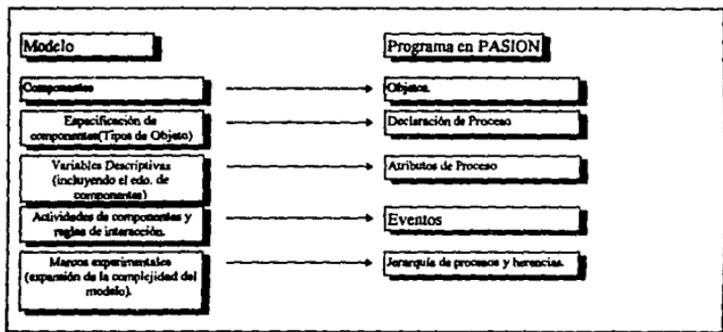
²RACZYNSKI S. Simulación por computador. Gpo. De Noriega (Megabyte), México 1983, pp. 106-126.

modelos con más de 4500 objetos han corrido con éxito en las computadoras IBM XT. La implementación de PASION para IBM RISC 6000 puede ejecutar modelos con más de 80,000 objetos, sin que sea notable una disminución del sistema.

1.3 Relación entre modelos de simulación y los programas de PASION⁴.

De acuerdo con la terminología común de la simulación, un modelo de simulación está integrado por componentes, como lo son los clientes en una tienda. El estado de cada uno de los componentes está descrito por el correspondiente conjunto de variables características y además por actividades que son definidas por las reglas de interacción entre los componentes del modelo. Los marcos experimentales definen el conjunto actual de las variables descriptivas y determinan la complejidad del modelo. El lenguaje PASION tiene todos estos elementos básicos para la simulación. Los componentes del modelo son objetos, la especificación de estos componentes está dada por la declaración del proceso, las variables descriptivas son atributos del proceso y las actividades de los componentes son eventos. Los marcos experimentales pueden ser ampliados mediante el uso de mecanismos de hereditarios.

Figura 3.



Las herencias permiten a los programadores crear clases, y por tanto, objetos que están especializados en otros objetos. Esto implica que el programador puede crear complejos modelos usando un código creado y probado con anterioridad. Las herencias en PASION pueden ser aplicadas mediante el uso la declaración de procesos prefijos. Digamos que PA es el nombre de un proceso existente. Suponga que deseamos crear uno nuevo, digamos PB, que tenga todas las propiedades del proceso PA (esto significa, todos los atributos y eventos). Esto puede realizarse usando el nombre PA/PB en vez de PB en la declaración del proceso. Mientras el proceso es declarado, el traductor busca el proceso PA (el proceso padre) e inserta todos los atributos y descripciones de eventos para PA en el nuevo proceso PB (proceso hijo o derivado). El proceso padre puede residir en archivo separado, o estar en el mismo archivo de uso. Así, el usuario puede preparar y guardar un conjunto de procesos padre, que sean los de más uso, y en base a ellos, cuando se necesite, crear nuevos procesos. Durante la creación de los nuevos procesos (derivados) algunos de los nombres usados en los procesos padre pueden cambiar. Esto incluye variables o tipos diferentes.

1.4 Funcionamiento de PASION.

No es sencillo exponer de una manera sucinta y sencilla la forma en que PASION funciona, pues es tanto una extensión de PASCAL, como un lenguaje de programación, como un paquete de simulación, como un interfaz de formulación de un modelo. Pienso que el siguiente diagrama, presentado en repetidas ocasiones por el autor de PASION, es el de más utilidad. Al siguiente diagrama le he añadido algunas notas que pueden servir de utilidad, y además las siguientes explicaciones.

⁴ZEIGLER B.P., *Theory of Modeling and Simulation*, John Wiley & Sons.

Figura 4.



Este diagrama significa que el lenguaje base es PASCAL, esto es, todas las simulaciones que se pueden hacer con PASION, cualquiera que sea la forma utilizada para definir el sistema simulado, termina en un programa ejecutable en PASCAL. Ahora bien, existen diversas formas de llegar a este programa ejecutable. El lenguaje PASION propiamente dicho es una manera inmediata superior de hacer, en este lenguaje se pueden crear procesos nuevos o utilizar antiguos modelos de procesos de la biblioteca MPPE, esto se realiza directamente con instrucciones de PASION, que posteriormente se tendrá que traducir mediante PAT4 a PASCAL. Un nivel más es utilizar los módulos de PASION para crear programas en PASION por medio de interfaces gráficas como lo son los módulos QMG y CMG. Estos módulos dan como resultado un programa en PASION, que puede o no hacer referencia a la biblioteca MPPE, y que luego, mediante el traductor PAT4 se traducirá en un programa compilable en PASCAL. El último proceso del ciclo de simulación consiste en ejecutar el compilador de PASCAL al programa traducido a PASCAL, para obtener así la configuración de un programa ejecutable desde PASCAL o desde el MSDOS.

El analista puede modificar o añadir la complejidad del modelo en cualquiera de estos niveles, esto es, modificando desde QMG o CMG, programando en PASION, o introduciéndose directamente al programa en PASCAL. Esta facilidad es una de las grandes bondades de PASION.

1.5 PAT4: El traductor de PASION a PASCAL.

PAT4 es un programa que traduce programas fuentes escritos en PASION al lenguaje PASCAL. Para correr el programa resultante en PASCAL hay que usar un compilador de PASCAL. En el proceso de traducción PAT4 usa la biblioteca de procesos predefinidos PASLIB.PAS. Algunos procedimientos adicionales, escritos en PASCAL se agregan al programa en la etapa de compilación del código correspondiente en PASCAL.

1.6 El ambiente PASION.

Existen algunos lenguajes de programación que pueden ser utilizados eficazmente con un ambiente apropiado. Pasion está equipado con el Ambiente Mínimo de Programación para Pasion (MPPE, Minimal Pasion Programming Environment), el cual consiste en una biblioteca de procesos predefinidos y otros módulos. Esta biblioteca aporta la simulación interactiva, gráficas, estadísticas de los resultados, procesos predefinidos para modelos dinámicos y de líneas de espera.

La parte medular de MPPE consiste en la biblioteca de Pasion, que contiene todos los procesos predefinidos. Estos son segmentos genéricos de programas que generan declaraciones de procesos (y

no de objetos). Los procesos predefinidos están escritos en PACION, que no es más que un metalenguaje, esto es una extensión de PASCAL que permite a los procesos tener parámetros formales. El usuario, de alguna forma (por cualquiera de las interfaces existentes ó mediante la programación en PACION) invoca un proceso predefinido de la biblioteca por su nombre y le especifica los parámetros actuales, que tienen relación con el proceso en cuestión, para que luego se realice la traducción a PASCAL. Estos parámetros pueden representar no sólo nombre de variables, sino además tipos, expresiones matemáticas completas, instrucciones, comentarios, etc. El usuario puede preparar su propios procesos predefinidos y añadirlos a la biblioteca.

La biblioteca standard de PACION LIB4.PAS contiene los siguientes procesos:

INTERP - Para una salida gráfica de resultados y una simulación interactiva.

INTERB - Para una salida gráfica de resultados (barras animadas), y una simulación interactiva.

INTERN - Para una salida numérica de resultados y una simulación interactiva.

SHOWP - salida gráfica de resultados a un instante dado.

STOR - (y el programa VARAN4) proporciona una salida de resultados una vez que la simulación ha terminado, promedio de trayectorias, análisis de varianza, intervalos de confianza, etc.

MONIT - despliega en pantalla los objetos existentes de modo gráfico (por bloques).

DYNAM - simula sistemas dinámicos continuos.

LSTAT - para análisis estadístico de líneas de espera.

BRDEA - proceso general de nacimiento y muerte con una distribución arbitraria del tiempo de vida.

ANIM1 - animación gráfica de líneas de espera.

WDOW - ventana de manipulación de procesos.

1.7 QMG: El Generador de los Modelos con Líneas de espera.

El lenguaje PACION tiene un ambiente bastante desarrollado que consiste en una biblioteca de procesos predefinidos para realizar la simulación interactiva, generar reportes, gráficas de resultados, estadísticas sobre resultados, etc. El ambiente también incluye algunos módulos auxiliares, como el Generador de los Modelos con Líneas de espera, descrito más adelante, que facilita considerablemente la simulación. Este programa puede ser usado aún sin ninguna programación en PACION o en PASCAL.

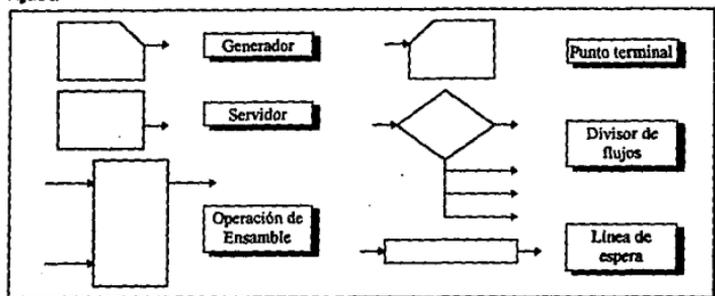
Los modelos con líneas de espera es una de las aplicaciones más populares de la simulación por computadora ya que es la idea más aplicada en lo que se refiere a las plantas de producción, tema de esta tesis. Muchos paquetes están relacionados a la representación gráfica del sistema que se modela. Así la gran parte del trabajo conceptual consiste en componer un diagrama de bloques o gráfica correspondiente del sistema modelado. Una vez concluido este trabajo, la programación del modelo resulta muy fácil y rápida. Esta misma estrategia fue la que se sirvió de modelo para crear el QMG (Queuing Model Generator).

Fuera del generador QMG, PACION tiene un tipo predefinido llamado Queue, el cual es una cola de tipo FIFO, LIFO o aleatoria. Este ofrece un número de procedimientos y funciones para formar líneas de producción con la técnica de líneas de espera. Así, el usuario puede declarar algunas colas y códigos de operaciones en el modelo fuera de QMG. De manera que si se prefiere no es necesario utilizar el módulo QMG.

QMG permite simular sin programar, usando la descripción gráfica de modelos en la pantalla del usuario. Cuando se aplica a los sistemas de espera, QMG ofrece nada nuevo que el GPSS y SLAM II. Note, sin embargo, que en algunos casos sí se tiene cierta ventaja. Por ejemplo en las simulación de Sistemas

Flexibles de Manufactura (SFM)⁵ que son controlados con algoritmos sumamente complejos y se desarrollan en ambientes computacionales sumamente complejos. Como consecuencia se tiene que cualquier paquete que se use para simular SFM debe de estar dentro de un ambiente apropiado. QMG satisface este requerimiento pues está relacionado con un lenguaje algorítmico y orientado a objetos. Los bloques que usa el QMG son:

Figura 5.



Como en esta tesis estamos precisamente interesados en este tipo de simulaciones, haré un alto para explicar el funcionamiento de cada uno de estos bloques.

Generador: es un bloque que crea los objetos o elementos del modelo. Estos pueden ser clientes en una tienda o en un banco, las piezas que entran a un puesto de trabajo o ensamble de una fábrica, o autos en un taller de servicio. Cada objeto puede tener un número de atributos que caracterizan sus parámetros y su estado actual. Los atributos (sus tipos y valores) están definidos por el usuario, que puede utilizar todos los tipos que permite PASCAL. los objetos pueden generarse con una cierta distribución de intervalos de tiempo entre llegadas y pueden aparecer uno por uno o en lotes. Todos los demás bloques describen todos los posibles eventos en la "vida" del objeto en el modelo. Así, los bloques del esquema gráfico corresponden a los eventos del modelo.

Sin un objeto entra en el bloque Terminal, entonces termina sus acciones y desaparece del modelo.

Servidor es un centro de trabajo o servicio. Este bloque puede ser disponible u ocupado en un momento determinado, y el objeto tarda un cierto tiempo para pasar por este bloque. El tiempo de servicio puede ser calculado por una expresión o puede ser aleatorio con una distribución predefinida o definida por el usuario. Servidor puede tener una o más entradas, donde el objeto a servir se puede escoger por prioridad, probabilidad o a través de cualquier otra regla definida por el usuario y descrita en PASCAL. Un retardo adicional puede agregarse al salir del Servidor.

En la **Operación de Ensamble (OE)** dos o más objetos se convierten en un nuevo objeto. Los objetos que entraron al bloque desaparecen y el nuevo objeto sigue por sus nuevos procesos (eventos o bloques). Este objeto puede recordar todos los atributos de los objetos que han sido usados para construirlo. El Servidor, la OE, el Generador, y el bloque Terminal pueden efectuar cualquier número de operaciones definidas por el usuario sobre los atributos de los objetos que los atraviesan. Estas operaciones llenan que ser programas en PASCAL.

Fila es una línea de espera donde los objetos pueden formarse antes de entrar en otros procesos (bloques). Filas puede ser de tipo FIFO (First in first out), LIFO (last in first out) o RANDOM (con salida aleatoria), y puede tener la longitud limitada o ilimitada. La Fila de tipo Random puede usarse como un lugar o sala de espera sin ningún orden de salida establecido.

⁵ THE CHARLES STARK DRAPER LAB Inc., 1984, Flexible Manufacturing Systems Handbook, Noyes Publication.

El flujo de objetos puede dividirse al entrar en un bloque de División. Este bloque tiene una entrada y algunas salidas que se determinan por probabilidad, por prioridad o a través de cualquier otra regla definida por el usuario y programada como una función de PASCAL.

La secuencia de eventos se define con flechas que conectan a los bloques. Antes de entrar al servicio o ensamble, el objeto puede esperar solamente en una de las Filas. Si un objeto no puede entrar en servicio y no puede formarse en una fila, entonces se pierde y desaparece del modelo. Para canalizar estos objetos a otros bloques se puede usar el bloque de División. En general, cada Fila debe corresponder a uno o más servidores.

1.8 El Editor de programas QMG.

Éste es un módulo de QMG dirigido por menús, que permite definir la estructura del modelo. El usuario puede mover el cursor en la pantalla usando las flechas u otras teclas típicas para editores como Wordstar, y poner o borrar los bloques. Los bloques pueden ubicarse en los nodos de una red con 80 nodos en horizontal y 30 en vertical. La pantalla es una ventana que muestra parte de esta área y se mueve automáticamente según los movimientos del cursor. La opción "Zoom", permite ver una mayor parte del modelo. El editor hace una verificación preliminar de la estructura y no permite crear modelos "obviamente" inconsistentes. Es posible, por ejemplo, ubicar una fila directamente después de un bloque terminal, o poner un generador de objetos de tal manera que nada pueda ser conectado a él. El diagrama queda definido mediante bloques y líneas (flechas) que los conectan. Si es difícil o imposible trazar una línea de conexión, se pueden usar etiquetas (pares de números enteros) para conectar cualquiera dos puntos del esquema. El editor es fácil de usar y se maneja a través de un sistema de menús. Con cierta experiencia se pueden manejar estructuras muy rápido, por ejemplo un modelo con 10 bloques se puede definir en menos de 40 segundos (según un experimento hecho por el creador de PASION). Después de generar la estructura, QMG pide al usuario que proporcione los datos necesarios (distribuciones de probabilidad y otros parámetros). Si el usuario tiene estos datos preparados, ingresarlos es sumamente sencillo. El modelo permite, como es lógico, almacenar en disco el modelo creado, leerlo y corregirlo.

Una vez definido y grabado el modelo, el resto del trabajo se hace automáticamente. Es importante notar que QMG es un generador de programas y no un programa de simulación, esto es, QMG sólo es un programa de interface. El precio que se paga para usar cualquier interface es el tiempo de reprocesamiento y compilación. La presente versión de QMG pertenece al ambiente del lenguaje PASION y produce un programa en PASION. Este programa se tiene que traducir a PASCAL y luego compilar este programa en PASCAL para poder ejecutarlo. Sólo el último módulo de QMG está relacionado a PASION, así que el generador podría ser adaptado a cualquier otro lenguaje de simulación. Esta relación entre PASION y PASCAL tiene fundamentalmente una razón didáctica. QMG puede ayudar a los alumnos a ver el programa (en PASION o PASCAL) de su modelo. Después de generar los códigos, el usuario puede trabajar con los programas, agregando, por ejemplo, más procesos al modelo, relacionándolo con bases de datos, poniendo características especiales, etc.

De esta manera se concluye que QMG puede ser usado por tres tipos de usuarios: los que no saben o no quieren programar; los que pueden programar en PASCAL y los que saben programar en PASION. Estos tres grupos pueden efectuar los siguientes trabajos:

1. Sin programar: Edición gráfica de modelos, salida estándar de QMG.
2. Programación en PASCAL: Selección de entradas y salidas para servidores y puntos de división y flujos a través de algoritmos definidos por el usuario, distribuciones no estándar de tiempos de llegadas y servicios, operaciones adicionales sobre los atributos de los objetos.
3. Programación en PASION: Se pueden agregar procesos adicionales de cualquier tipo (incluso procesos continuos) definidos por el usuario, se pueden usar otras facilidades de PASION, como la herencia de propiedades, procesos predefinidos de la biblioteca del traductor, etc.

Para finalizar con este módulo QMG, es necesario añadir que recientemente se ha presentado en la Universidad Panamericana una tesis que consiste en una nueva versión de QMG, que funciona en

WINDOWS, con todas las ventajas que esto lleva consigo. Este interfaz esta en su fase de prueba y se le ha bautizado con el nombre de PSI (ψ). Esta tesis ha utilizado este nuevo interfaz.

1.9 CMG: Generador de Modelos Continuos.

Este programa fue diseñado con el propósito de facilitar la simulación dinámica de sistemas continuos. El Generador de Modelos Continuos (CMG) es un programa generador cuyo producto es un programa en PAsION o en PAsCAL, de acuerdo a las especificaciones que el usuario determina en el módulo.

La salida de resultados de CMG es creada en forma de una declaración de procesos de PAsION que puede ser insertada (en forma continua, discreta o combinada) en otro modelo de PAsION. CMG también puede generar un programa completo en PAsCAL que puede correr mediante el uso de un compilador de PAsCAL. Los datos de entrada de CMG se formulan en términos de diagramas gráficos, los cuales describen la dinámica del modelo del sistema en cuestión. Por medio del diagrama gráfico podemos componer un red de nodos y sus correspondiente vínculos directos. Los nodos representan la señal y los vínculos (flechas) representan las funciones de transferencia.

CMG permite los siguientes tipos de vínculos: Lineares estáticos, no-lineares estáticos, lineares dinámicos (dad una función de transferencia), retraso de tiempo, muestreo y retención y supervínculos (un complejo sistema dinámico). El último tipo de vínculo (los supervínculos o superlinks) permiten incluir un modelo dinámico completo (especificado anteriormente y guardado en un archivo de disco) al modelo que está siendo creado en ese momento. Esta facilidad es usada mientras se desarrolla un modelo complejo, compuesto por submodelos creados y probados por separado.

Cuando se combinan simultáneamente (discreto/continuo) sistemas, es posible declarar "estados de eventos", esto es, eventos que ocurren cuando alguna variable continua de estado alcanza un nivel específico.

Anexo 2: Xcell+, Paquete de Simulación para Empresas de Manufactura.

Introducción.

Xcell+ es un paquete de computadora que le ayuda en la construcción de modelos lógicos de los procesos de manufactura. Un modelo construido con Xcell+ puede simular las operaciones de una fábrica, de manera que se puede estimar su capacidad de producción, y además es una forma alternativa para estudiar los nuevos diseños de procesos.

La simulación de los procesos de manufactura ha sido usada con un éxito considerable durante años. Pero, hasta ahora, el esfuerzo requerido para construir esos modelos era tan grande que la técnica no era muy usada como lo pudo haber sido. Mas aún, la simulación requería de expertos programadores en lenguajes de simulación, y la inversión de tiempo para alcanzar un nivel aceptable de competencia hacia que la simulación fuera practicada sólo por los especialistas, en vez de los ingenieros y administradores que eran los que enfrentaban los problemas.

Todo esto ha cambiado con rapidez. Xcell+ es uno del grupo de paquetes de simulación que son mucho más fáciles de aprender y usar que los lenguajes convencionales de simulación. Esta nueva generación permite a los ingenieros y administradores construir sus propios modelos y sólo llamar a los especialistas cuando se necesita un mayor grado de detalle o una mayor complejidad en el modelo.

Así como las modernas "Hojas de cálculo", los nuevos paquetes de simulación no hacen nada que hace 10 años no se podía hacer. De cualquier forma, mediante la reducción en el esfuerzo de programación de una gran cantidad de instrucciones, las hojas de cálculo han hecho posible - para los hombres corrientes - el uso de los métodos de proyección financiera, y esto ha ocasionado un gran crecimiento en el número de usuarios. Similamente, una reducción substancial en el esfuerzo requerido para la construcción de un modelo de simulación debe ocasionar un rápido crecimiento en el número de gente que use esta técnica.

Por ejemplo, en el pasado, a pesar de que muchas escuelas han tenido cursos especializados para la técnica de la simulación, era relativamente el uso de esta técnica en laboratorios de Ingeniería en manufactura y cursos de dirección de operaciones. Estos cursos no podían, en general, asumir que los estudiantes tenían un curso de prerrequisito en simulación, y no podían disponer de tiempo para enseñar un lenguaje de simulación. Ahora, de cualquier forma, con paquetes como Xcell+, es muy práctico hacer asignación de rutinas que requieren la construcción y operación de un modelo de simulación, sobre todo en un curso donde ni el instructor ni el estudiante tiene previo entrenamiento en la simulación.

La mayoría de estos paquetes de simulación son ejecutables en computadoras personales, cosa que ha contribuido en gran medida a su atractivo, pero no es lo fundamental de su contribución. La diferencia significativa en el uso de estos nuevos sistemas es el uso de interfaces de programación. Estos paquetes no son propiamente lenguajes de simulación, en el uso riguroso de este término. Estos nuevos paquetes tienen sistemas de base de menús, esto significa que en cada paso en su uso, el usuario tiene un menú de opciones - las acciones que pueden hacerse en el contexto determinado en que se encuentra el usuario. Hay muchos caminos distintos de implementar sistemas de menús - algunos muy efectivos y amistosos, y otros no tanto. Por supuesto, el usuario de la computadora tiene que probar ser increíblemente tolerante. Ellos aprenden a usarlo mediante el uso de aplicaciones notablemente tontas, pero en este caso, aún los peores menús del sistema de simulación permanecen significativamente fáciles de entender y usar, que los lenguajes de programación ordinarios.

Por otro lado, el precio de la simplicidad de estos paquetes de simulación es incuestionablemente bajo respecto al poder que tienen. Muchos de los nuevos paquetes, incluyendo a Xcell+, están enfocados a áreas concretas de aplicación, y modelar otros tipos de sistemas resulta pobre, si es que resulta. Este es el gran contraste respecto a los lenguajes de simulación que son, de hecho, lenguajes universales de programación y puede aplicarse a todo lo que puede modelarse. Esto es algo inevitable que los nuevos paquetes de simulación necesitan sacrificar. Estos son con frecuencia caracterizados como sistemas "rápidos y sucios", tratando de significar "rápidos y con un nivel limitado de detalle", pero esta no es necesariamente una característica mala. Los analistas de simulación, con frecuencia incluyen un mayor grado de detalle del que es realmente necesario, por el sólo hecho de que es posible, de manera que si los nuevos paquetes eliminan esta posibilidad, esta tendencia de complejidad se eliminará, y será simplificante para muchos problemas.

La analogía con las hojas de cálculo fácilmente exagerada. La modelación para simulación no es inherentemente un proceso trivial, y por tanto no es uno de esos que puede ser superficialmente utilizado como la proyección financiera. Lo más que uno puede esperar de paquetes como Xcell+ es que ellos pueden reducir los mecanismos de construcción de modelos de simulación hasta el punto de que la mayor atención sea puesta en los importantísimos aspectos conceptuales del modelado del sistema.

Una característica común de los nuevos paquetes de simulación es el uso de gráficos y animaciones. Muchos paquetes usan de estos gráficos no sólo en la construcción de los modelos, a modo de interfaz, sino como una manera sencilla de presentar los resultados de un proyecto de simulación. Xcell+ los usa durante la construcción del modelo, como durante la corrida del modelo, y en cierta forma para desplegar alguna información final del modelo. Xcell+ puede desplegar en pantalla tablas y listas de resultados como suplemento de los resultados gráficos.

Xcell+ usa gráficos simbólicos más que ilustraciones realísticas. Esto es, no hay necesidad de hacer que los elementos de Xcell+ aparezcan como en la realidad son. Algunos softwares usan gráficos tridimensionales más elaborados y producen presentaciones más elegantes. De cualquier forma, el precio de esta elegancia gráfica es exactamente proporcional a el equipo de cómputo que se necesita, de la ejecución sencilla del modelo, y el crecimiento en la dificultad de la construcción del modelo. Las gráficas de Xcell+ han sido diseñadas para un compromiso efectivo entre la presentación de las relaciones lógicas del modelo de una fábrica y los costos incurridos para dicha presentación.

2.1 Orden de menús en XCELL+.

Xcell+ presenta el menú de alternativas posibles en un renglón de ocho cajas a lo largo de la parte baja de la pantalla. Estas cajas indican la función que les es propia en ese momento a una tecla de función del teclado. Las cajas en la pantalla representan teclas de F1 a F8, de la izquierda a la derecha. Cuando una caja no tiene tecla asignada significa que en ese momento está desactivada.

Por supuesto que existen muchas más que 8 opciones que se pueden utilizar en Xcell+, pero sólo 8 pueden ser desplegadas en un solo momento. Para usar con eficacia Xcell+ se debe de entender muy bien cómo es que están administrados estos menús de opciones.

Las acciones y opciones en Xcell+ están agrupadas en paquetes llamados "menús". Cada menú puede tener hasta ocho opciones, mostradas en las ocho cajas del final de la pantalla (No todos los menús necesitan de las ocho cajas, de manera que en algunos menús existen cajas vacías). Por ejemplo cuando se acaba de entrar en Xcell+ se está automáticamente en el menú principal (Main Menu), y el renglón de control muestra las siguientes cajas:

Figura 1.

| | | | | | | | |
|------|-------------------|----------|----------------|-----------------|------|--|--|
| HELP | Change display | analysis | NEW FACTORY | FILE MANAGER | QUIT | | |
|------|-------------------|----------|----------------|-----------------|------|--|--|

Esto significa que sólo se puede escoger una de esas opciones, debido a lugar en el que se encuentra. Las opciones que están sombreadas sólo aparecerán cuando se tiene en memoria actual un modelo determinado, ya sea porque se acaba de crear o porque se ha leído de la unidad de disco.

Si se selecciona *File Manager* se presentarán las opciones de:

Figura 2.

| | | | | | | | |
|------|------------------|------------|------------|------------|---------------------|---------------|-----------------|
| HELP | Merge factory | Asign disk | Erase file | List Files | Retrieve factory | Store factory | Back to Main |
|------|------------------|------------|------------|------------|---------------------|---------------|-----------------|

La caja de *Store Factory* sólo aparecerá cuando se tenga en memoria actual un determinado modelo empresa, y no cuando todavía no se tiene ninguno. Este primer nivel de profundidad en lo que se refiere a sistemas de menús es fundamental para todas las opciones que se refieren a guardar y recuperar archivos de trabajo en los archivos de disco. A continuación entraremos a lo referente al diseño de un modelo, esto es, los menús que se derivan de <Design>:

Figura 3.

| | | | | | | | |
|-----------------|-------------------|------------------|-------------------|-------------------|-----------------|------------------|-----------------|
| HELP | Display detail | Change Symbol | Maintenance | Process design | | | |
| WkCtr design | Buffer design | Rec'gA design | Ship'gA design | MInCtr design | CtrlPt design | AuxRes design | Back to Main |
| HELP | Change display | analysis | NEW FACTORY | design | FILE MANAGER | QUIT | |

La caja correspondiente a F8 siempre está reservada para dejar el menú actual y regresar un nivel al menú anterior, también F1 siempre está reservada para la ayuda de programa. A esta estructura jerárquica, árbol se concluye que el menú base es el *Main Menu*. Por tanto se concluye que, por lo regular F8 sirve para bajar de nivel en el árbol. Sólo se puede subir y bajar en el árbol pero no cambiar, dentro del mismo nivel, de rama. Aunque no aparece en pantalla la tecla F10 está reservada para hacer saltos rápidos al menú de diseño de planta <design>, esto es, bajar con F8 tantas veces hasta <main> como sea necesario, y una vez ahí seleccionar <design>.

Esta es la manera de funcionar a grandes rasgos de Xcell+, lo demás lo dejamos en manos del lector interesado, ya que un manual completo sobre el funcionamiento de Xcell+ escapa de los fines e intenciones del presente trabajo.

2.2 La planta de producción en Xcell+.

Una característica distintiva de Xcell+ es el uso de una cuadrícula rectangular y uniforme como área de trabajo para definir, dentro de ella, el modelo de planta en cuestión. Esto es, la fábrica está representada en base a una cuadrícula de células, y cada elemento de la fábrica ocupa exactamente una célula. Esencialmente, la construcción de un modelo en Xcell+ es ir seleccionando los elementos y su posición dentro de esta cuadrícula (desafortunadamente para los autores de Xcell+ el término célula es aplicado

actualmente en las empresas de manufactura para designar un grupo integrado de máquinas y no un sitio concreto dentro de la planta de la fábrica).

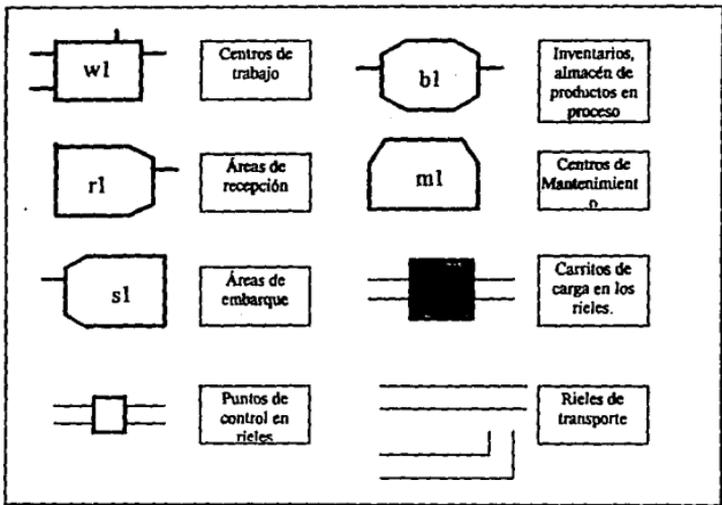
Nótese que desde que cada elemento del modelo ocupa una sólo célula, todos los elementos del modelo serán necesariamente mostrados del mismo tamaño, sin importar lo que suceda en la realidad. Xcell+ sólo intenta representar la relaciones lógicas entre los elementos, de manera que esta representación geométrica sólo es una herramienta. Xcell+ no intenta ser una herramienta de *layout* de la fábrica.

Sólo aparece el cursor en la pantalla de cuadrícula si el usuario se encuentra en un punto de los menús en que lo hace necesario, esto es, por ejemplo, cuando se está diseñando una planta, y no cuando se está en el menú principal. El cursor de la cuadrícula se mueve, como es lógico, con las flechas del teclado o con *mouse*.

2.3 Los bloques básicos de diseño en Xcell+.

Gran parte de la simplicidad de Xcell+ radica en el hecho de que sólo un número muy pequeño de tipos diferentes de elementos se utilizan en la construcción de modelos de fábricas. Los tipos de elementos que pueden ocupar una célula son:

Figura 4.



Los Centros de Trabajo (CT), son los lugares donde los procesos se realizan, un centro de trabajo puede desempeñar varios procesos, según el tipo de pieza que llegue. El CT detecta cuál es el proceso que debe aplicar a una pieza determinada según el nombre de la pieza. Los centros de trabajo tienen 2 entradas y 2 salidas, una de las cuales es de desperdicio o reproceso (según necesite el modelo). Cuando una pieza termina de ser procesada en un centro de trabajo puede tener un nombre distinto de que tenía cuando entró. Para cada CT se tienen que asignar los porcentajes de material reciclado (por rechazo a *scrap* o reproceso); el mecanismo de generación de los tiempos que tarda un proceso determinado del CT; los tiempos de preparación, esto es, cuando se cambia de un proceso a otro en el

mismo CT (esto se puede deber a que entra una nueva pieza, distinta a la anterior que exige un proceso distinto al anterior, o que una misma pieza sufre dos procesos en el mismo CT); además es necesario asignar las reglas específicas de como se cambia de un proceso a otro (qué procesos corresponden a cada pieza). Sólo un proceso a la vez puede ser activado en un mismo centro de trabajo. Puede ser que un mismo centro de trabajo reciba material de otros, más de dos lugares, en este caso a cada proceso sólo le pueden pertenecer 2 entradas a la vez. Los procesos pueden trabajar sobre tandas de material y no forzosamente de 1 en 1. Los centros de trabajo pueden tener condicionado su funcionamiento a lo que ocurra en alguna otra parte del sistema, por ejemplo en el nivel de un inventario determinado, cuando se cumple cierta condición puede comenzar a trabajar y terminar de trabajar. Esta característica de Xcell+ es fundamental en lo que se refiere al modelado de sistemas JIT.

Los Centros de Mantenimiento (CM) se encargan de detener periódicamente los procesos en CT para arreglar o ajustar los procesos. Los CT tienen especificado la frecuencia con que tendrán fallas de maquinaria, estas frecuencias se definen indicando la periodicidad de atención aleatoria de un CM en un CT determinado a esto se le llama *Random Breakdowns* de manera que un CT puede tener dos paros distintos, el de mantenimiento normal y el de falla y un sólo CM que se encargará, en ambos casos, de proporcionar el servicio. Nótese que el mantenimiento está asociado al CT y no a un proceso en particular, de hecho es cuando se declara un CT el momento para indicar la frecuencia del mantenimiento y el CM que se encargará de ese trabajo. Junto a la periodicidad de interrupción de los CT por los CM se debe también indicar el tiempo de servicio, esto es, el tiempo que dura cada interrupción. Nótese también que los paros de mantenimiento por falla se producen cuando el CT está operando, en cambio un paro por mantenimiento simple sólo se realiza cuando el CT ha terminado un proceso y va a comenzar otro. Cada CM tiene un determinado número de equipos de servicio, de manera que si los centros de trabajo asignados al CM exceden la capacidad de equipos de servicio, los turnos de servicio tendrán que hacer cola.

El diseño de Áreas de Recepción (AR) consiste fundamentalmente en declarar el tipo de material que entrará y la frecuencia con que se recibirá de fuera del modelo. Un AR no puede ser directamente enlazada a un inventario. Las llegadas de material pueden ser ilimitadas, regulares, probabilísticas, o manualmente establecidas. También hay que indicar las cantidades de piezas en que se recibe cada tanda de material. Incluso se puede asociar esta cantidad a funciones de probabilidad. Las AR tienen una determinada capacidad de almacenamiento, se puede asignar cualquier capacidad que se desee. Este nivel de almacén puede ser usado como detonador ("gatillo") que active un proceso determinado. Si llega más material del que se puede almacenar, el excedente se desecha. A diferencia de los inventarios, un área de recepción sólo puede recibir y almacenar un tipo de material determinado. Cuando el material entra en la planta se puede etiquetar periódicamente, con la finalidad de calcular cuánto tiempo tardará un determinado material en salir nuevamente de la planta. La AR debe de saber la frecuencia con la que se desea etiquetar el material. Es importante recordar que el seguimiento de un material marcado se interrumpe cuando entra en un inventario no ordenado (FIFO o LIFO), y no cuando cambia de nombre o sale de un proceso determinado.

Las Áreas de Embarque (AE) son los lugares donde el material producido por la fábrica es descargado al mundo exterior, esto es, sale del sistema. Las AE pueden estar conectadas directamente a procesos pero no a inventarios ya que, al igual que las AR se tiene cierta capacidad de almacenamiento de material. A diferencia de la AR, las AE pueden manejar cualquier tipo de partes, aunque si se desea sólo aceptará de un sólo tipo. En estos elementos es necesario indicar el tamaño y frecuencia de pedidos de salida, esto según distribuciones de probabilidad, valores predeterminados o constantes. Si la demanda de producción no tiene suficiente respaldo en el AE entonces el exceso de demanda se considera demanda perdida, que es irrecuperable aunque en momentos posteriores se tenga nuevamente material en el almacén. El nivel de este almacén también puede ser utilizado como detonador o activador (gatillo) de algún proceso o muchos procesos, esta es la herramienta necesaria para simular procesos "pull" en JIT. Los procesos que tienen una salida designada como scrap también sacan del sistema el desperdicio de material sin necesidad de un área de embarque, ya que este desperdicio no se considera como producto de la planta; de cualquier forma es útil conectar a AE los desperdicios pues es una manera efectiva de llevar la cuenta de estos.

Los Buffers o Almacenes (Bf) representan el inventario en proceso de la planta, esto es, todo aquel material que ya ha experimentado al menos una modificación por medio de algún proceso. Aunque

algunos "expertos" contemporáneos son partidarios de que estos inventarios en proceso deben desaparecer, la nueva tecnología de manufactura indica que los sistemas requieren desesperadamente de inventarios entre sus procesos, así por ejemplo, la Manufactura Flexible y los sistemas asíncronos de materiales. La principal disputa es la elección entre el costo de tener inventarios y el costo de detener los procesos por falta de ellos; la verdadera cuestión es ¿Cuánto debe ser el volumen del inventario en proceso (WIP)? Los BF son los elementos más sencillos de Xcell+ en términos de la cantidad de variantes que pueden tener, sólo se tienen que fijar dos factores: La capacidad y el orden en el que los materiales fluyen (FIFO, LIFO, RANDOM). Los BF pueden almacenar cualquier tipo de piezas, ya que distinguen dentro de sí la contabilización de cada una de ellas. La máxima capacidad de un BF es el límite total de todas las partes que puede almacenar, de manera que lo que importa para la cuenta de capacidad es la suma de piezas total. Suele ocurrir que una sola pieza es la que domina en el BF. Para efectos de detonadores, activadores o gatillos de procesos, se puede utilizar tanto el nivel de una pieza concreta como el nivel de BF en general. Dentro de los resultados de salida se puede obtener un diagrama del comportamiento de los BF en general o por pieza. En Xcell+ los BF se pueden utilizar como trampas para obtener múltiples salidas o entradas a centros de trabajo, esto se consigue mediante el uso de BF con cero capacidad.

Diseño de Ayudas Auxiliares (AA). Las AA son las ayudas extras para los CT, con ellas se puede simular típicamente los operadores requeridos para un proceso en particular. Son esencialmente los sitios donde uno o más recursos están detenidos mientras no están en uso. Se pueden especificar tres atributos de AA: la localización, el nombre, el número de recursos que contiene. La especificación más importante es, obviamente, el centro de trabajo que van a auxiliar, pero esto se especifica desde el diseño del proceso concreto. Se puede utilizar como una herramienta útil de administración de empleados, es un mecanismo interesante para restringir el número de centros de trabajo que puede funcionar simultáneamente. Los procesos no hacen cola, sólo cuando un recurso termina agota su trabajo en un CT revisa si no existe algún otro CT que necesite de su ayuda.

Los Puntos de Control tienen cuatro papeles distintos en el modelo:

- Son las juncturas de los segmentos de rieles, todos los rieles deben comenzar y terminar en un Punto de Control (PC).
- Son la conexión entre los elementos de recolección de material y los demás elementos, esto es, entre los PC, rieles, y carritos, y las AR, AE, BF, CT. Los PC son los únicos sitios donde los carritos pueden recoger material y dejar material.
- Son los que dirigen el tránsito que determina la ruta y acción que debe seguir un determinado carrito, según el material que carga.
- Pueden servir como puntos de recarga de baterías de los carritos.

El diseño de los PC pueden ser muy complicados, sobre todo cuando se trata de modelar un sistema moderno dirigido por computadora de acarreo de material. Aquí hay una gran idea sobre flexibilidad en la herramientas de modelado de Xcell+, pero es importante entender la variedad y funcionamiento de los mecanismos de control, así como su funcionamiento conjunto. Los PC tienen números como nombres, en vez de letras, esta numeración es asignada automáticamente según el orden en el que se crean los PC.

Los PC son los únicos lugares donde los carritos pueden recoger y dejar material. Si un carrito puede recoger material de un PC entonces este PC debe de estar a su vez conectado con algún elemento que lo provea de material (AR, CT, BF,). De la misma manera si está designado lugar de descarga de un carrito, el PC debe de estar a su vez, conectado con algún elemento capaz de recibir el producto (BF, CT, AE).

Un PC puede funcionar como lugares de recolección y desembarco de material, o como de almacén y recarga de los carritos, pero no tener estas dos funciones al mismo tiempo. Se asigna tiempo de recolección y descarga de material, que siempre será constante. Muchos PC se pueden conectar a los mismos BF o CT pero no al revés, esto es, a cada PC sólo se le puede conectar un CT o BF.

Los Rieles son las rutas por las cuales circularán los carritos de materiales de un PC a otro. Un PC puede tener hasta 4 rieles conectados al mismo tiempo. Los rieles tienen direcciones determinadas en su definición. Pueden ser de salida si el riel se origina en ese PC, de entrada si el riel termina en ese PC o

de reversa, si el riel cambia de salida a entrada, según los requerimientos del sistema. Los rieles se crean automáticamente cuando se define en el menú de PC el origen y fin de un segmento determinado, todas sus características se estipulan en la definición de los PC. Los rieles tienen dos características: la longitud (que está determinada por el número de bloques que abarca) y la velocidad de recorrido. La longitud define una cierta capacidad de almacenar carritos de material (uno por bloque). Un riel puede ser reversible o bi-direccional, esto es, si su sentido cambia según los requerimientos de la dinámicas del sistema, esta característica permite simular sistemas de monoriel, además permite construir sistemas extremadamente complicados, que por lo regular son difíciles de que funcionen razonablemente.

Los carritos de material son elementos con dos únicas características, son móviles y no tienen bloque o celdas propias en la planta. El número de carritos en el sistema depende, por supuesto, del número de veces que se pide uno más en el menú de diseño. Un carrito puede estar, o ser creado sobre un PC o un riel. Cada carrito se identifica con un número determinado, del que se detalla el comportamiento en los resultados. Todos los carritos son equivalentes, esto es, tienen la misma capacidad, corren a la misma velocidad y cuestan los mismo. Las imágenes de los carritos indican si está cargado, si tiene una dirección (indica el número del PC al que se dirige), el estado de su batería y su estado.

Reglas de Control de Tráfico.

Este es un tema que merece un alto. Es una de las principales opciones de trabajo de Xcell+ y muestra el grado de especialización en la simulación de sistemas de manufactura. Por esto, es interesante observar la logística de funcionamiento del paquete.

El aspecto más complejo de los PC es la especificación de las reglas de control de tráfico, esto es, que es lo que un carrito tiene que hacer cuando entra y sale de un PC. La primera cosa a entender es el control de rutas del carrito. La ruta de un carrito (vacío o lleno) es el destino, el número de PC al que se dirige mostrado en el gráfico del carrito. Un carrito (si está dirigido) sólo se detiene (a cargar o descargar) en el PC al cual se dirige, sin embargo, los PC en su camino pueden alterar su ruta a seguir (por ejemplo si hay tráfico), aunque siempre seguirá la ruta más corta. Si un carrito no está dirigido (puede estar vacío o lleno) toma la primera posibilidad de carga y descarga que encuentra, y por tanto todas las reglas de tráfico de los PC por los que pasa, mientras está sin dirección, le afectan. Estos dos tipos de carrito (dirigido y no dirigidos son compatibles en el mismo modelo). También existe la opción de asignar rutas manualmente, o por default, según el sentido del riel.

Las reglas de control automático para carritos cargados es el siguiente. Si un carrito que lleva la parte P entra al PC D se aplican las siguientes reglas:

1. Si existe D es un descargadero y si el elemento al que se descarga puede aceptar inmediatamente una pieza P se inicia el proceso de descarga.
2. Si el elemento al que se descarga no puede aceptar en ese momento una pieza P, pero la señal de espera está prendida en el PC D, el carrito espera hasta descargar.
3. Si no se descarga el carrito, permanece cargado con una parte P, y espera su tiempo de tránsito en D, se aplican las siguientes reglas:
 - a. Sigue al siguiente centro de trabajo que puede aceptar P, este orden puede aparecer en el PC D.
 - b. Si no hay lista de orden en D, puede consultar la secuencia de movimientos para los carritos llenos en D
 - c. Si no existe nada de esto, sale por el riel correspondiente según las manecillas del riel.

Las reglas de control automático de tráfico para carritos vacíos, cuando entran a un PC D son:

1. Si la opción de dejar pasar los carritos vacíos ("empty skip") está activada en D, y no han pasado un gran número de carritos vacíos por D desde la última vez que cargó (o descargó) uno, el carrito pasa de largo.
2. Si existe una unidad disponible para recoger (o descargar) en D inmediatamente la recoge (una vez transcurrido el tiempo especificado de carga o descarga).

3. Si no sucede ningún evento de carga, el carrito permanece en D mientras transcurre el tiempo de tránsito.

a. Sigue la salida indicada en la lista de D de PC a seguir, si no, sale por un riel según el orden de reloj.

Los puntos de Control de sólo demanda funcionan sólo si algún Bf del sistema, indica una demanda de material. Sólo funcionan para satisfacer la demanda de un Bf. Cuando hay muchas demandas, estas siguen un orden cronológico de satisfacción. Cada vez que un carrito vacío llegue a S (PC de demanda), se busca la lista de pedidos, y sigue la demanda más vieja que debe ser satisfecha (FIFO). Cada orden de entrega indica la información de destino. Se puede calcular un número aproximado de carritos que se necesitan para satisfacer la demanda adecuadamente. Los carritos pueden ser pre-posicionados en un PC determinado, cuando no tienen trabajo, a estos PC se les llama "Puntos de espera", o áreas de estacionamiento de los carritos. Cuando hay más de un "punto de espera" o para los carritos, Xcell+ automáticamente asigna ese punto como la mejor fuente de abastecimiento de carritos para cualquier otro PC. De manera que cada vez que S reciba una orden de demanda, un carrito irá a S desde el más cercano "punto de espera" de S, el sistema es casi automático.

Las Zonas de Control son conjuntos de PC que tienen en número determinado de carritos que pueden contener. Cuando se diseña una planta, se pueden añadir más carritos que los que una zona puede soportar, pero en el momento de la ejecución del modelo nunca habrá en la zona más de los permitidos. Una zona se define como un conjunto de PC y rieles que están conectados a esos PC. Esto significa que las puertas de entrada a las zonas son los PC y cuando una zona está saturada de carritos la puerta se cierra temporalmente para nuevos carritos.

Las baterías de los carritos se necesitan recargar, esta es una importante consideración en el modelado de plantas automáticas, ya que el tiempo en el cual un carrito está fuera de servicio puede ser significativo para el sistema. Más aún, la interacción de el fenómeno de recarga y otros fenómenos del sistema pueden producir disturbios considerables en el sistema. Para la recarga de las baterías de los carritos se tiene:

1. Uno o más PC pueden ser asignados como sitios de carga. Esto se define en el menú de opciones, estos PC asignados a cargar las baterías de los carritos sólo pueden servir para esta función.
2. El tiempo que tarda un carrito en descargarse puede ser especificado. Este tiempo es igual para todos los carritos. Cada uno de los carritos lleva su propia cuenta de descarga que depende sólo del número de celdas que ha atravesado, de manera que los carritos tienen distinto grado de descarga, según su uso.
3. El tiempo que tarda un carrito en solicitar su carga de baterías se especifica, fijando un nivel de batería en el cual el carrito, la próxima vez que pase por un PC se detendrá a cargar.
4. El tiempo que tarda un carrito en recargarse puede ser especificado. Este valor se aplica a todos los PC y carritos por igual. Sólo se puede recargar un carrito por cada PC de recarga.

El sistema chequea el nivel de baterías de los carritos cada vez que pasan por un PC, si lo necesita el carrito se enfila hacia el PC de recarga más cercano. Si no se definen ningún PC de recarga, el sistema asume que los carritos no se descargan, aunque todos los demás valores sí se definan. Como no se intenta que el sistema se bloquee cada vez que un carrito quede completamente descargado antes de alcanzar su PC de recarga (y se espera que se tenga cuidado en el diseño del modelo en este sentido), Xcell+ de hecho, no deja ningún carrito muerto en los rieles por descarga, la simulación continúa y sólo lleva la cuenta de el número de veces que un carrito se queda sin batería antes de llegar al PC de recarga.

2.4 Diseño de Costos de Producción en el Modelo de Simulación.

Cada uno de los elementos del modelo de fábrica tienen dos coeficientes de costo asociados con ellos:

- 1) El costo de Capital, que representa el costo fijo, o Inversión, es el costo de incluir algún elemento nuevo en la planta.
- 2) El costo de Operación, representa el costo variable, por unidad procesada, de trabajar una unidad en la planta.

Estos costos están generalizados en costos por unidad, que cada analista puede interpretar de la manera que más le convenga para resolver el problema que tenga entre manos. Xcell+ asigna valores fijos, aunque estos valores pueden ser modificados por el usuario. Este cambio se realizará desde cada uno de los menús de diseño de cada elemento de planta, desde una caja llamada <option>. Todos los segmentos de traslado y carritos de material tienen los mismos costos, sin importar la pieza concreta que trasladen. Los costos totales de capital son presentados en el menú de resultados, accesible desde <run>.

2.5 Los resultados de Xcell+.

Existe un menú separado de resultados, accesible cada vez que después de un corrida se hace una pausa, el usuario puede desplegar en pantalla o imprimir cualquiera de los siguientes tipos de resultados obtenidos de la última corrida del modelo:

1. **Resumen de costos:** Costo de capital y de operación de cada elemento del modelo.
2. **Entrada:** Para cada AR:
 - a. Unidades aceptadas.
 - b. Cargamentos de pedidos no satisfechos.
3. **Inventario de productos en proceso:** acumulado en los BF y en las AE y AR.
 - a. Inventario Total en el sistema.
 - b. Nivel máximo de inventario.
 - c. Nivel mínimo de inventario.
 - d. Nivel promedio de inventario.
4. **Utilización de los Centros de Trabajo:** para cada Centro de trabajo.
 - a. Porcentaje de tiempo de ocupación.
 - b. Porcentaje de tiempo de preparación.
 - c. Porcentaje de tiempo de mantenimiento (en servicio o en espera)
 - d. Porcentaje de tiempo bloqueado.
5. **Utilización de los Centros de Mantenimiento:**
 - a. Porcentaje del tiempo que por lo menos un equipo de servicio estaba funcionando.
 - b. Porcentaje de tiempo que por lo menos un CT estaba esperando mantenimiento.
6. **Utilización de Recursos Auxiliares:**
 - a. Porcentaje de tiempo en que por lo menos un recurso estaba en uso.
 - b. número promedio de recursos en uso.
7. **Utilización de carritos.**
 - a. Estado actual.
 - b. Localización actual.
 - c. Número de carritos cargados.
 - d. Porcentaje de tiempo en que estuvieron cargados.
8. **Tiempo de proceso, para cada AE.**
 - a. Número de unidades marcadas recibidas.
 - b. Promedio (de todas las unidades marcadas) de tiempo de proceso.
 - c. Promedio total de tiempo de proceso.
 - d. Promedio total de tiempo de espera.

Anexo 3: Principios Fundamentales en la Dirección de Operaciones.

1. Flujo de planta: Balanceo y cuellos de botella.

En las plantas de multi-operaciones, donde los materiales fluyen de un área de trabajo a otra antes de salir de la fábrica, la palabra Balanceo se refiere a la distribución equitativa de trabajo. Cuando el trabajo está mal distribuido, significa que una operación será el cuello de botella, porque es más lenta que el resto de las operaciones.

Considere simplemente el caso de dos trabajadores t1 y t2. ¿Cuál es la capacidad de producción si t1 requiere de 1 minuto y t2 de 2 min. por unidad fabricada?. El sistema no puede producir más de 30 unidades por hora pues el cuello de Botella es t2. Más aún, t1 estará la mitad del tiempo inutilizada mientras que t2 siempre estará saturada. Esta línea está Desbalanceada. Por otro lado, si la carga de trabajo se distribuye de tal manera que cada trabajador toma 1.5 min., la capacidad de producción crecerá a 40 unidades por hora (60/1.5) y ambos trabajadores estarán constantemente ocupados. Si es posible distribuir el trabajo de esta manera, la línea estará perfectamente balanceada.

el Balanceo es con frecuencia mucho más complicado que este sencillo ejemplo. Por ejemplo suponga que cada uno de los trabajadores anteriores son operadores de máquinas. Cada máquina necesita 2 min. de operación por unidad fabricada. La primera máquina es semiautomática; requiere un operador para sólo 1 min. por unidad, y la máquina se hace cargo del segundo minuto. La segunda máquina no es automática y requiere del operador los 2 min. ¿Está la fábrica Balanceada?. La capacidad de producción del sistema es de 30 unidades por hora, y ambas máquinas están siempre ocupadas, pero el trabajador 1 descansa la mitad del tiempo.

La lección en el segundo ejemplo es que hay muchos recursos en una fábrica, y nosotros tenemos que concentrarnos en aquél que limita la producción, si es que queremos incrementar la producción. Cualquier recurso limitado está en potencia de causar un Cuello de Botella.

Para resumir, el balanceo es más que un aspecto de equilibrio entre los trabajadores. Redistribuir la carga de trabajo puede incrementar la productividad de la fábrica sin la necesidad de recursos adicionales. Si se puede incrementar la capacidad de producción del Cuello de Botella, se tendrá mayor producción para toda la fábrica. Por ejemplo, si el Cuello de Botella es una máquina sujeta a descomposturas, un mantenimiento preventivo puede mejorar la producción y los beneficios.

Los Cuellos de Botella pueden ser identificados de muchas maneras. La característica más obvia es que no hay descanso para el cuello de botella. Siempre está completamente ocupado en producción, preparación o mantenimiento. Segundo, en la gran mayoría de las fábricas se encuentra mucho trabajo

pendiente para los cuellos de Botella. Esto con frecuencia acarrea que se forme una pila de inventario de trabajo en proceso (WIP) en la entrada del cuello de Botella. Por tanto, un alto grado de WIP y de utilización pueden ser usadas en la simulación de una fábrica para identificar los Cuellos de Botella. Este asunto es particularmente importante cuando se trata de análisis tipo "¿Qué pasa si?", para prever el efecto de un nuevo producto o nuevo equipo, por ejemplo.

2. Capacidad de almacenamiento en la Fábrica: Las reglas de utilización.

Los inventarios entre las operaciones son con frecuencia usados para suavizar el flujo de salida. La capacidad de almacenamiento es necesaria en caso de que el equipo falle, de manera que las demás operaciones puedan continuar mientras la falla de una máquina se repara. Si existe variación en el tiempo de proceso, el inventario entre las operaciones ayuda evitando los bloqueos y arranques de las operaciones no afectas. Pero ¿cuánto inventario es suficiente?

Las respuestas no son sencillas. Las siguientes reglas de utilización están basadas en un estudio¹ sobre las líneas de producción de un sólo artículo operadas según el modelo de empujar ("push mode", infinita disposición de recursos, y cada estación trabaja cuando el material está disponible). Estos principios pueden ser muy útiles, pero la experiencia nos indica que hay que ser muy cuidadosos cuando se extrapolan a situaciones nuevas.

- a. Los buffers son correctos sólo para nivelar pequeñas variaciones de flujo entre las áreas de una fábrica, y no para grandes desbalances entre los centros de trabajo.
- b. La capacidad de almacenamiento de un Buffer no necesita ser grande en la mayoría de los casos. El objetivo es prevenir el bloqueo y arranque de las operaciones del cuello de botella.
- c. En general, los buffers antes y después de una operación tienen valores iguales, pues el arranque y bloqueo de una máquina afecta en definitiva el nivel de producción de la máquina.
- e. Si la alimentación de material para una operación está sujeto a un esquema regular de interrupciones, el buffer de entrada de dicha operación necesita almacenar lo necesario para mantener el funcionamiento de la operación durante la interrupción de material. Por ejemplo, si el flujo se detiene por 10 min. cada hora, y el tiempo de producción es de 0.5 min., el buffer de entrada debe tener 20 piezas.

Por otra parte, si las interrupciones de material son aleatorias, pero permanecen en promedio 10 min. por hora, una capacidad de 20 piezas en el buffer de entrada no es suficiente porque algunas interrupciones podrían ocurrir tan juntas que el buffer no tendría la oportunidad de recuperarse. Quizá el peor de los casos ocurra cuando todas las estaciones de una línea balanceada estén sujetas a interrupciones aleatorias. En este caso, cerca de 10 veces la capacidad de buffer es requerida para obtener el 90% del crecimiento de la producción que puede ser alcanzada mediante los buffers.

f. Si el flujo de salida de una operación es interrumpido (por ejemplo, debido a la descompostura de la siguiente máquina), la capacidad del buffer es la misma que la de interrupciones de alimentación o entrada.

De cualquier forma, en este caso la capacidad del buffer es necesaria para tener espacio de almacenamiento para el material procesado que está saliendo mientras dura la falla de la siguiente máquina.

g. Si no existen interrupciones ni fallas, pero los tiempos de procesos son variables, la capacidad requerida del buffer es pequeña. En una línea balanceada con una variabilidad de 50% ± el promedio del tiempo de proceso, colocando buffers con capacidad de 6 entre cada operación se obtiene alrededor de un 90% de mejora (incremento de la tasa de producción) que puede ser alcanzada mediante el uso de los buffers.

¹CONWAY, MAXWELL, McCLAIN & THOMAS, "The Role of Work in Process Inventory", Working paper 87-06, Johnson Graduate School of Management, Malott Hall, Cornell University, Ithaca, NY, 14853 (May 1987).

h. Si existe una operación que sea Cuello de Botella, la necesidad de los buffers para la operación es menor. Los buffers deben de emplearse para mantener al Cuello de Botella en funcionamiento siempre. Sorpresivamente, a menor velocidad de producción del cuello de Botella, menos unidades se necesitan

3. Algunos cálculos básicos anteriores a la simulación.

¿Cuál es la capacidad de producción de la fábrica?. Esta pregunta es difícil de responder porque depende de muchísimos factores. De cualquier manera, en la simulación, muchos de estos factores están bajo nuestro control directo, de manera que es sencillo encontrar fórmulas y métodos simples para estimar la capacidad de producción. A continuación se describirán los cálculos para un centro de trabajo. Las fórmulas son típicamente utilizadas para cada centro de trabajo; con esto se puede determinar el Cuello de Botella de la operación y la capacidad del sistema en teoría. Estos cálculos son sólo una aproximación, y deben de verse como una guía por donde comenzar la simulación. Estos cálculos ignoran la variabilidad y limitaciones del sistema.

El objetivo es calcular producción máxima posible para un centro de trabajo que tiene múltiples procesos, con un tamaño específico de lotes de producción (productos que son procesados al mismo tiempo).

Usaremos los siguientes símbolos:

ST: Tiempo de preparación.

PT: Tiempo de proceso.

Q: Tamaño de lote.

R: El porcentaje de la salida que pasa por este centro de trabajo mediante por vía de reproceso. El rechazo puede ocurrir en este momento o en algún centro de trabajo posterior, pero R no incluye los productos que son desechados o retrabajados en otra línea.

S: Porcentaje de la producción de salida que es desechada.

U: Significa el tiempo que pasa entre una falla y otra.

D: Significa el tiempo que toma reparar una falla.

M: Intervalo de tiempo entre los mantenimientos planeados.

MT: Tiempo que tarda un mantenimiento planeado.

Primero, considere el efecto del retrabajo. Cuando un producto visita el mismo centro de trabajo más de una vez, el resultado es similar a tener un mayor tiempo de proceso y no retrabajo. Esta puede ser modelada dividiendo el tiempo de proceso entre (1-R). Podemos llamar a este resultado PTE (Tiempo de proceso efectivo).

$$PTE = Q \cdot PT / (1-R)$$

Ahora, considere un ciclo de producción para un centro de trabajo, consistente en mantenimiento y ejecución de un lote completo de un producto, incluyendo el retrabajo. El tiempo de ciclo debe incluir un momento para reparaciones, que ocurren a una tasa de D/U por cada unidad de tiempo del proceso, y un momento para el mantenimiento, que ocurre a una tasa de MT/M por cada unidad de tiempo que la facilidad está funcionando o no.

Para estimar el tiempo de reparación por ciclo, llegaremos a una proporción. Según la teoría de los promedios, D/U significa la tasa del tiempo de reparación respecto el tiempo de operación. De manera que para estimar el tiempo de reparación en un ciclo, multiplicaremos el tiempo de operación por ciclo (PTE) por D/U. De manera que, el ciclo incluya el tiempo de reparación más el tiempo de proceso, cuyo total es $PTE + (1-D/U) \cdot PTE$.

El mantenimiento por ciclo es estimado de la misma manera, excepto que el término $(1 + MT/M)$ multiplica al tiempo entero del ciclo, porque el mantenimiento está basado en el tiempo de reloj más que en el tiempo de operación. De otra forma, los días "durarían más".

La siguiente ecuación considera todos estas estimaciones en un sólo número:

$WC =$ Tiempo de ciclo de un centro de trabajo.

$$WC = (\text{Suma de los ST de cada producto} + (1+D/U)(\text{Suma de los PTE para cada producto})) \cdot (1+MT/MI)$$

Ahora que el tamaño del ciclo está establecido, consideremos la salida. Durante un ciclo, la salida de producción es un lote de cada producto, pero esta debe ser corregida por las unidades que se desperdician, entonces definiremos como el tamaño de lote efectivo, QE, a:

$QE =$ Promedio de salida de buenos productos (no desperdiciadas) por ciclo.

$$QE = Q \cdot (1-s)$$

La tasa promedio de salida de producción es la cantidad efectiva, QE, por ciclo dividida entre el tiempo de ciclo del centro de trabajo, esto es:

$ORI =$ Tasa de salida (promedio) del producto 'i' (unidades útiles).

$$ORI = \frac{QE \text{ (del producto 'i')}}{WC \text{ (del centro de trabajo)}}$$

Estas fórmulas muestran la manera de estimar el máximo volumen posible de producción de un centro de trabajo cuando los lotes de producción son específicos. El cuello de botella o proceso, se identificará como aquel centro de trabajo que tiene la más pequeña de las salidas de producción máximas.

Usar estas fórmulas es sencillo si el flujo de material es en serie. De cualquier manera, si existe un ensamble, y si algunas unidades se requieren en múltiples cantidades, estas fórmulas deben de ser usadas con sentido común. Por ejemplo, si dos tuercas son necesarias para hacer un artefacto, entonces el flujo no estará balanceado hasta que la tasa de salida sea el doble que el de los artefactos. Otro ejemplo sería, si existen muchas máquinas iguales que hacen la misma operación, la capacidad de producción de esa área dependerá de la asignación de trabajo; por ejemplo, cada línea hace sólo un producto, el tiempo de preparación será cero y la salida de producción de esta área será mayor.

Estas fórmulas no son sustituto de la simulación, pero pueden ahorrar tiempo. Por ejemplo, hay que usarlas para responder a preguntas como, por ejemplo, ¿cuántos trabajadores o máquinas son necesarias en un área determinada para aumentar o disminuir 1 unidad?. Con frecuencia se encuentran muchas situaciones donde las fórmulas anteriores o similares son utilizadas para reducir la duración de la simulación.

4. Conducta de un estudio de simulación.

En los proyectos de simulación existen maneras eficientes de conducir los estudios. Uno puede ahorrar tiempo y ser más eficaz si tiene presentes ciertos criterios que la experiencia ha dictado:

El primer paso es decidir qué es lo que uno espera aprender o enseñar (obtener), y a qué nivel de detalle lo necesita. Diseña el modelo más sencillo que le ayude a resolver su problema, con frecuencia, la tendencia nos lleva a hacer modelos complejos, más de lo que realmente necesitamos. En cierta forma la simulación es un trabajo que nunca se termina, siempre se puede mejorar, ir más a detalle. Por definición, ningún modelo puede estar 100% perfecto. Es importante tener presente que un modelo es una representación abstracta de la realidad y debe de ser tratada como eso. Por ejemplo, si usted necesita comprender la diferencia entre operar un banco con 25 máquinas rápidas y otro con 50 máquinas lentas, hay que preguntarse si la verdadera esencia del problema se alcanza comparando una máquina rápida con dos lentas, y si no es así, comparando 2 rápidas con 4 lentas, etc.

¿Cómo es que debe de planear el tamaño de diseño del sistema? Cuando uno se plantea el tamaño que desea simular, es necesario responder a la pregunta del grado de exactitud o detalle con que se dan los

resultados. Del diseño del modelo, que considera la exactitud y el objeto simulado, depende el tamaño del sistema.

Una vez que el modelo está diseñado, debe de ser muy cuidadoso checando cada elemento para asegurar que todos los datos son correctos. Esto es tedioso, pero no tanto como repetir toda la corrida de simulación después de descubrir el error.

Finalmente, cuando se trata de comparar resultados de las diferentes corridas, surge la pregunta sobre el tamaño de cada una de las corridas, qué tan típicos son los resultados obtenidos. Es claro que si esta pregunta exige una respuesta rigurosa es necesario acudir a fuertes herramientas estadísticas, como los análisis de experimentos. Sin embargo, cuando la ocasión no lo amerite, podrían ahorrarse unas cuantas horas de trabajo teniéndose en cuenta simplemente algunos principios básicos de obtención de resultados:

1. Los sistemas simulados, para poderse analizar, es necesario que tengan un periodo suficientemente largo de carga del sistema. Esto se debe a que básicamente, las empresas no funcionan igual el día de su inauguración a el día 1000 de funcionamiento. Esto se debe a que el sistema tiene que cargarse, esto es, se crean inventarios, se estabilizan comportamiento, etc., de una manera que no es fácil de predecir ni de imitar desde un inicio.
2. Una vez que el sistema ha alcanzado cierta estabilidad, entonces los datos recolectados por el programa necesitan reiniciarse. Nótese que sólo son los datos, y no el programa entero, pues esto volvería al sistema a su estado inicial. Cuando se tiene cargado el sistema es un buen inicio de la recolección de estadísticos.
3. Ahora, ¿qué tan grandes deben ser los periodos registrados?, los periodos deben de ser lo suficientemente grandes como para que engloben ciclos enteros de comportamiento del sistema. Por ejemplo, si un sistema en las primeras horas del día se comporta de una manera determinada, y al final del día de otra distinta, resulta evidente que el periodo debe de considerarse a lo largo de todo el día, para tener una clara muestra del comportamiento diario, y no sólo de unas cuantas horas. Estos tamaños de ciclos pueden identificarse contemplando distintos tamaños, y comparando los resultados obtenidos entre sí. Si los resultados varían considerablemente unos de otros, significa que hay necesidad de ampliar el tamaño de la corrida de la muestra, pues todavía no se llega a englobar un ciclo.

5. Simulaciones veloces y lentas: hiper y turbo centros de trabajo.

Un hiper-centro de trabajo es aquél que representa más que un elemento en la fábrica. Este particular es importante si se está utilizando un programa limitado, de tipo, versiones de estudiantes. Un banco con diez máquinas, por ejemplo, puede verse como un sólo centro de trabajo. Para esto, el tiempo de proceso del hiper-centro de trabajo debe de ser la décima parte del de una sola máquina, de manera que la capacidad de salida sea 10 veces mayor. Las cosas se complican si estas máquinas pueden especializarse en un sólo producto, de manera que se eliminen los tiempos de preparación. ¿Cómo es que se puede superar este problema?, es cosa que dejamos a la imaginación del amable lector.

Un turbo-centro de trabajo es aquél que está diseñado para correr rápidamente, de este modo recorta el tiempo en el que se espera llegar a un resultado final. Esto se logra redefiniendo la unidad del producto. Por ejemplo, sin un producto se produce en lotes de 20, lo más sencillo es especificar el tamaño de lotes en 20. Un turbo-centro de trabajo ignora las partes individuales y hace de un lote la unidad básica del

modelo. Para aplicar este principio es necesario multiplicar el tiempo del proceso por 20 y especificar el tamaño del lote en uno, en cada uno de los procesos por los que pase la pieza en cuestión. Hay que tener especial cuidado en la interpretación de los resultados, pues se refleja un comportamiento para lotes de 20 y no por unidad.

La razón de que un turbo-centro de trabajo es más rápido se debe a que la mayoría de los paquetes de simulación hacen el seguimiento de cada una de las piezas en individual, y esta información considerada simultáneamente crea colas en el orden de los monitoreos. Cuando se tienen menos piezas en consideración el monitoreo se agiliza. En nuestro ejemplo concreto alrededor de 20 veces.

Estos dos "trucos" utilizados para obtener mayor rapidez en el cálculo de la simulación son de muchísima utilidad. Hay que tener presente, cuando se realiza el modelado del sistema, qué es lo que realmente necesitamos saber del modelo, esto es, lo que realmente queremos obtener como respuesta de nuestra simulación. Esta información servirá para decidir el grado de detalle en que estamos interesados en obtener y simular, y repercutirá de una forma importante en el tiempo invertido de simulación y de la cantidad y calidad de los recursos utilizados.

Anexo 4: Análisis Crítico de la Teoría de Restricciones¹.

Introducción.

A finales de 1989 y principios de 1990, se iniciaron en México una serie de seminarios con la finalidad de promover la Teoría de Restricciones, además de que aparecieron también la traducción al español de dos libros, uno detrás de otro, que apoyaban esta filosofía cuyos títulos fueron "La Carrera" y "La Meta", a la fecha, por ejemplo, "La meta" llevaba publicados 800,000 ejemplares a varios idiomas siendo este dato por sí mismo un testimonio del éxito de esta filosofía de producción. Su iniciador es un físico que se llama Eliyahu Goldratt, autor además de los dos libros, y este físico se acredita la implementación exitosa de su filosofía en muchas empresas donde muchos de los casos se escriben en una revista especializada sobre el tema: "The Theory of Constraints Journal".

Es importante mencionar esta filosofía que tiene mucho que ver con los sistemas modernos de manufactura por muchos, muchos motivos: su aparente éxito por una parte y por otra la total o gran ignorancia que hay sobre esta teoría de restricciones; por ejemplo, se ha dicho que es mejor que el JIT, y no cabe duda que quien hace este tipo de afirmaciones (lo he comprobado personalmente por quien las hace), en realidad afirma dos cosas: que no sabe qué es JIT ni sabe lo que es TOC (teoría de Restricciones, por sus siglas en Inglés), es por esto que he decidido dar una breve explicación, estableciendo las similitudes y diferencias, así como las ventajas y desventajas con respecto al JIT.

4.1 Desarrollo de la discusión TOC vs JIT & MRP.

Tanto los que aplican JIT como MRP, se basan en la suposición de que la planta ideal es aquella que se encuentra balanceada, esto es, aquella en la que cada recurso tiene la misma capacidad de producción relativa a la necesidad de la planta. El enfoque de TOC es el de aceptar la existencia de una planta no balanceada, esto es, aquella en la que algún recurso tiene menos capacidad de producción que otros. El recurso más limitado se le llama restricción. TOC protege mediante inventario solamente la restricción. Las estaciones que no son restricciones tienen una protección "natural" por medio del exceso de capacidad. Añadir inventario a una estación que no es restricción provoca el incremento del tiempo de entrega (un costo) y que se incremente por lo tanto el inventario en proceso (un costo) sin proporcionar algún beneficio tangible. TOC coincide con JIT en que el inventario es un desperdicio, si es que el inventario está puesto en alguna estación que no es una restricción. Sin embargo, asigna un inventario a la restricción de manera que nunca se desperdicie su tiempo, esperando material que procesar.

¹Las ideas a las que se hacen referencia se obtuvieron de: J. QUI NIETO, Tesis: Filosofía y Modelos Relativos a sistemas modernos de manufactura. Presentada a la división de estudios de posgrado de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México, como requisito para obtener el grado de Maestro en Ingeniería (Investigación de Operaciones). Ciudad Universitaria 1993.

TOC no trata de eliminar todos los problemas, sólo aquellos que amenazan la restricción. Usando la misma terminología de JIT, un esfuerzo excesivo en la eliminación del problema es un desperdicio. Debe llegarse a un punto en el que es mucho más caro proporcionar un poco de inventario contra un problema en la restricción que eliminar el problema.

Contra estos argumentos, se hace la siguiente apología en favor de JIT:

1. En primer lugar, aunque idealmente el ingeniero de producción espera tener o encontrarse con una planta balanceada, ya se sabe que la realidad no va a ser así. Normalmente el ingeniero de producción tiene dos caminos, con dos impactos contablemente hablando: el primero es, efectivamente, protegerse con inventarios, el segundo sólo colocar inventarios frente a la restricción, TOC. El inventario por más pequeño que sea, resta liquidez a las empresas y además tiene un costo administrativo. Cuando se afirma que el inventario sólo se debe de colocar antes del cuello de botella, resulta ser fácil decirlo, pero no realizarlo. En general la gente de la planta no sabe dónde están las restricciones: SE HA DICHO QUE, EL PROBLEMA QUE LAS EMPRESAS TIENEN, ES EL NO SABER DONDE SE ENCUENTRAN SUS PROPIOS PROBLEMAS. El mejor mecanismo que tiene JIT para encontrar las restricciones es precisamente la disminución del inventario, porque en donde se reduzca el inventario que forma parte de una restricción, allí es donde va a aparecer la "gran roca", es en este punto donde roca y restricción se identifican.

2. Por otro lado, hay que considerar que la situación de la planta es totalmente dinámica, la demanda a final de cuentas es la que determina la existencia o la localización de las restricciones. Por ejemplo si se tiene una línea desbalanceada con tres centros de trabajo, con capacidades de 10 pz/hr, 15 pz/hr y 20 pz/hr, Cuando la demanda sea de 10 pz/hr no habrá restricciones, aunque se haya llegado al límite. Cuando sea de 15 habrá un cuello de botella, pero cuando llegue a ser de 30 habrá una línea completa de restricción, todo un centro de trabajo. ¿Qué pasa cuando se piensa que este aumento en la demanda será estable, o al menos por algún tiempo largo?, bueno, sería mejor invertir en maquinaria accesoría. Cuando este fenómeno en la demanda es transitoria se puede pronosticar, de manera que nos protejamos con inventarios de producto terminado. Sin embargo cuando la demanda normal es de 10 pz/hr y no hay forma de hacer inventarios de protección de producto terminado, ¿qué sucede?, este problema no es tan fácil resolver con inventarios de protección en las restricciones. La respuesta es: debemos de tener una planta algo excedida siempre. A final de cuentas nos enfrentamos más bien a un problema de planeación y pronóstico.

JIT reconoce que hay un momento en que es mucho menos caro proporcionar un poco de inventario contra un problema en la restricción, que eliminar el problema. ¿De qué forma lo hace?, con los Kanban. Mediante el uso de los kanban se hace un sistema de producción de "jalar", cada kanban representa un lote de producción que se pide una unidad a la anterior. Cuando esto sucede significa que el sistema tiene una variación no proyectada y por tanto se descubre un problema.

Se hizo una simulación² de los tres enfoques: MRP, JIT y TOC. El enfoque de MRP dio como resultado que no alcanzó la producción deseada y además tuvo un efecto lateral indeseable: si se evalúan las implicaciones de la variación en el inventario terminado para administrar los tiempos de entrega, se podrá entender porqué muchos administradores occidentales creen que el taller los está controlando a ellos en lugar de ser ellos los que controlan el taller. El enfoque JIT redujo por mucho la variabilidad de la producción de la línea. El tiempo de entrega fue muy bajo (un día menos). Desafortunadamente JIT alcanzó un promedio de producción más pequeño que el enfoque MRP, pero independientemente del costo alto de reducir la variabilidad, cuando se considera qué tan predecible puede ser el sistema, el enfoque JIT es claramente superior al enfoque anterior. Con respecto a TOC, mejoró los resultados obtenidos por JIT en aproximadamente 2% y requirió menos esfuerzo. Los resultados obtenidos por TOC se vieron en esta simulación superiores a los alcanzados por JIT, además, tiene menos inventario en proceso que JIT.

²FOGARTY, BLAKSTONE, HOFFMAN "Production and Inventory Management". 2a. edición, South Western, 1991, página 649.

Hasta este momento suena muy lógico que implementar un sistema basado en TOC sea lo mejor, pero *no cabe duda de que el mejor sinodal es siempre un sistema real y no una simulación* con un número mínimo de estaciones y con pocas complicaciones, así que el siguiente punto es considerar los aspectos de implementación de TOC:

1. El TOC no es para el iniciado. Las compañías necesitan conocer conceptos como la programación infinita, educar a la gente y apoyo de la alta dirección. Cuando una compañía está acostumbrada a actuar mal, es difícil reeducar. Sin embargo este problema también lo tienen las otras dos técnicas.
2. Otro aspecto que no se acepta es que los algoritmos que apoyan la filosofía y que se contienen dentro del programa computacional, OPT (Optimized Production Technology), que genera los planes de producción son un secreto. Una filosofía básica de un sistema moderno de manufactura debe sostener como principio la Transparencia de Sistemas, esto es, permitir la comprensión de los resultados por parte de los usuarios.
3. Se dan dificultades cuando la base de los planes de producción no están claras para la gente del piso encargada de su ejecución. Este problema se agrava cuando el desempeño de la gente del piso no está relacionada directamente con la ejecución del plan.
4. Otro problema son las suposiciones de certidumbre en el procesamiento de material. Si los datos fueran incorrectos, en cuanto a capacidad, requerimientos, el sistema producirá resultados erróneos. Esto es muy natural, sucede a todos los sistemas en general, no conozco ningún sistema de producción que cuando se alimenta con datos erróneos dé respuestas correctas.
5. También se descalifica la técnica TOC por su visión reduccionista, sólo concentra sus esfuerzos en los números y no en la mentalidad de las personas. *No basta producir más a costa del precio invisible que se ha de pagar, en este caso la dignidad de la persona (¿?)*.

Aún es muy pronto para saber el éxito de esta técnica, los resultados son relativamente pocos y por supuesto, no hay literatura existente sobre los fracasos, así que si hay alguien que no esté de acuerdo acerca de lo que dice Jesús Qui Nieto acerca de TOC (*respaldado la mayor parte la experiencia de otros practicantes y la mía propia*) dejemos que el tiempo sea el mejor juez.