

Entorno de Desarrollo del Control de Sistemas de Manufactura*

José L. Villarroel, José A. Bañares y Pedro R. Muro-Medrano

Dpto. de Ingeniería Eléctrica e Informática
 Centro Politécnico Superior, Universidad de Zaragoza
 María de Luna 3, E-50015 ZARAGOZA, SPAIN

Abstract: In this work, a software development environment for hierarchical control software design of flexible manufacturing systems is presented. The environment allows the software design for the coordination process in manufacturing as well as decision making, on the other hand, software prototypes can be tested by simulation. Special purpose graphic interfaces provide a user friendly interaction. A knowledge representation schema is used as a kernel in the environment. This schema integrates knowledge representation techniques based on frames, high level Petri nets and it follows the OOP paradigm.

Resumen: En este trabajo se presenta una aplicación informática para el diseño de software de control jerárquico de sistemas flexibles de fabricación. Dicha aplicación aporta componentes que permiten el diseño de software de coordinación y de toma de decisiones, así como la simulación de los diseños realizados. La creación de un modelo es facilitada por interfases gráficas con el usuario. Se propone un esquema de representación de sistemas de manufactura como núcleo de la aplicación que integra técnicas de representación estructurada del conocimiento basadas en frames, redes de Petri de alto nivel y diseño orientado a objeto como base para la metodología de modelado.

Palabras clave: Control de SFF, Redes de Petri, Diseño orientado a objeto.

*Este trabajo ha sido parcialmente financiado por los proyectos IT 10/91 del CONAI de la Diputación General de Aragón y ROB91-0949 de la CICYT del Estado Español.

1 Introducción

El diseño de las diferentes funciones involucradas en el control de un Sistema Flexible de Fabricación (SFF) ha sido abordado desde múltiples aproximaciones, entre las que cabe mencionar las basadas en técnicas de Inteligencia Artificial y las basadas en modelos formales como las redes de Petri.

En el campo de la Inteligencia Artificial se han desarrollado diversas aproximaciones para el modelado de sistemas de fabricación. Inicialmente las aproximaciones basadas en reglas fueron las más utilizadas, pero éstas están siendo desplazadas por aproximaciones basadas en el modelo [FOX 83], [SMIT 86], donde se emplean técnicas de representación estructurada del conocimiento.

Por otra parte, el modelado y análisis de sistemas de fabricación han constituido una aplicación de especial interés en el campo de las redes de Petri [ALJA 88], [SILV 89]. El objetivo es la construcción de modelos de sistemas de fabricación que verifiquen ciertas propiedades de tipo cualitativo, como ausencia de bloqueos, limitación de capacidades, exclusión mutua en el uso de máquinas, etc. Para tratar modelos complejos, como el de un SFF, se han empleado formalismos basados en redes de alto nivel, donde la composición modular es la metodología predominante [GENT 88], [MART 87]. Esta aproximación es la adoptada en este trabajo para el diseño del modelo de coordinación de un SFF.

En este trabajo se presenta una aplicación informática para el diseño del software de

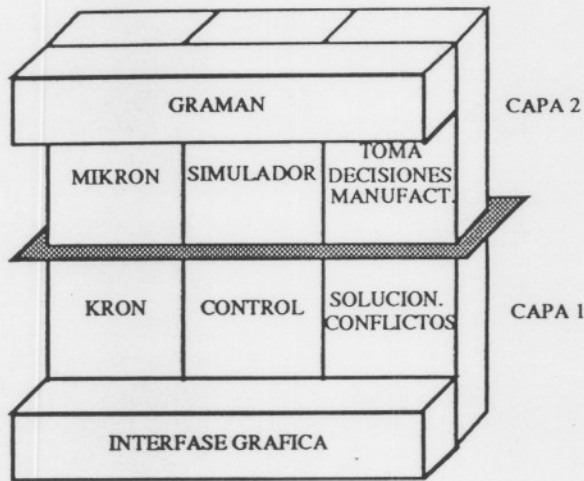


Figura 1: Arquitectura del entorno del sistema de desarrollo.

control jerárquico de SFF para la arquitectura jerárquica definida en [MART 89]. Dicho control tiene como objetivo el funcionamiento autónomo, sin intervención humana, de un SFF. Además de la utilización conjunta de técnicas de representación estructurada del conocimiento y de redes de Petri, se propone el diseño orientado a objeto como base de la metodología de modelado para el entorno de diseño que se presenta. Cada elemento de un sistema de fabricación es representado por un objeto cuya dinámica se describe mediante una red de alto nivel. De esta forma se combina la metodología y posibilidades de representación y manipulación del diseño orientado a objeto con el formalismo de las redes de alto nivel.

2 Presentación del entorno de desarrollo

El entorno de desarrollo, objeto de este trabajo, está estructurado en dos capas (la figura 1 muestra dicha estructura), cada una con su respectiva interfase gráfica. La primera capa es una herramienta de análisis y diseño de sistemas de eventos discretos de propósito general, que está compuesta a su vez por cuatro componentes:

KRON: Constituye el núcleo fundamental del entorno. KRON es un lenguaje orientado a objeto para la representación

del conocimiento, especialmente orientado al modelado y manipulación de sistemas de eventos discretos.

CONTROL: Este componente constituye el mecanismo de inferencia específico y aporta las primitivas necesarias para la ejecución de los modelos construidos con KRON.

SOLUCIONADOR DE CONFLICTOS: Los modelos construidos pueden ser no totalmente deterministas, es decir, necesitar de decisiones exteriores que determinen su evolución. Este componente permite el diseño de estrategias de decisión y asiste al CONTROL en la ejecución de modelos no deterministas.

INTERFASE GRÁFICA: Permite la visualización e introducción de forma gráfico-textual de los elementos que componen un modelo y de sus interrelaciones.

La segunda capa, construida utilizando los objetos y primitivas definidas en la capa inferior, constituye una especialización para el dominio de aplicación de los SFF. Dispone también de cuatro componentes:

MIKRON: Es un lenguaje de representación que especializa KRON para SFF. MIKRON aporta los objetos y primitivas específicas para la construcción del modelo de coordinación de los elementos de un SFF y sus relaciones, en términos propios del dominio de aplicación.

SIMULADOR: Es un simulador de eventos discretos específico para SFF. Utiliza las primitivas aportadas por CONTROL para hacer evolucionar un modelo de coordinación simulando su interrelación con la planta de producción.

TOMA DE DECISIONES DE MANUFACTURA: Este componente, construido sobre el SOLUCIONADOR DE CONFLICTOS de la primera capa, aporta una biblioteca de políticas de decisión basadas fundamentalmente en reglas de despacho y de interpretación de un plan de operaciones. Esto facilita el diseño de estrategias de decisión en términos propios de SFF.

GRAMAN (GRAphic MANufacturing): Posibilita la utilización de las primitivas de diseño definidas en MIKRON de forma gráfica y la

animación de las simulaciones de los modelos construidos.

3 Modelado de sistemas de eventos discretos

KRON (Knowledge Representation Oriented Net) [MURO 89b] [MURO 90] [VILL 90], es un lenguaje de representación del conocimiento de sistemas de eventos discretos concurrentes. KRON está basado en la integración de técnicas de representación basadas en frames, programación orientada a objeto y redes de Petri de alto nivel. La programación orientada a objeto proporciona la base de la metodología de modelado, los frames aportan el esquema genérico de representación del conocimiento y las Redes de Petri de Alto Nivel (RAN) [JENS 91] constituyen el formalismo adoptado para representar el comportamiento dinámico.

De acuerdo con la metodología de la programación orientada a objeto, en un modelo KRON, cada entidad está representada mediante un objeto que recoge toda la información concerniente a la entidad que modela y a sus relaciones con otros objetos, estructurada en atributos. Si la entidad posee características dinámicas, su comportamiento se representa mediante una RAN integrada con el resto de conocimiento de la entidad. Desde el punto de vista de la dinámica se pueden distinguir las siguientes clases de objetos: 1) *Objetos de Marcado*. Representan entidades sin comportamiento dinámico y tienen la misma semántica que las marcas de las RAN. 2) *Objetos de Estado*. Son los objetos que representan entidades dinámicas y contienen la representación de su estado (lugares en una RAN). 3) *Objetos de Acción*. Las acciones que modifican el estado de una entidad dinámica (transiciones en una RAN), son representadas en KRON mediante objetos de acción.

Los objetos están organizados en jerarquías de especialización, en las que cada objeto hereda todos los atributos de sus objetos padre, incluida la dinámica, es decir, la RAN subyacente. La metodología de modelado subyacente en KRON consiste básicamente en

los siguientes pasos:

1. Identificación de las entidades, dinámicas y no dinámicas, que componen el sistema a modelar y su agrupación en clases.
2. Diseño de un modelo o prototipo para cada clase a partir de la especialización de prototipos ya existentes. Si se trata de una entidad no dinámica el prototipo será un objeto de marcado y si se trata de una entidad dinámica el prototipo estará compuesto por un objeto de estado y un conjunto de objetos de acción.
3. Creación de una instancia para cada entidad identificada en el sistema a partir del prototipo correspondiente.
4. Representación de las interacciones entre las diferentes entidades del sistema mediante el establecimiento de relaciones de *sincronización* entre objetos de acción.
5. Establecimiento del estado del sistema mediante la asignación de objetos de marcado a los lugares de los objetos de estado correspondientes.

Para permitir la ejecución de los modelos construidos con KRON el componente CONTROL aporta un conjunto de primitivas de manipulación que implementan las reglas de evolución de una RAN (disparo de objetos de acción, actualización del marcado, etc.) que se encuentran desarrolladas en detalle en [VILL 90].

Un modelo KRON puede especificar indeterminismo en la dinámica. Así, pueden existir situaciones en las cuales sea necesario tomar una decisión que determine la evolución del modelo. Este tipo de situaciones están reflejadas en la RAN subyacente de un modelo mediante estructuras denominadas *conflictos*. Un conflicto es un objeto de acción o agrupación de ellos que constituye un punto potencial de decisión. Cada conflicto de un modelo KRON está representado por un objeto denominado *objeto conflicto*. Cuando en la evolución de un modelo surge un problema de decisión, éste es resuelto por el objeto conflicto que lo representa mediante la aplicación de una política de

control que retorna una solución. El módulo SOLUCIONADOR DE CONFLICTOS dispone de una librería de políticas de control de la que el usuario puede disponer para diseñar sus propias estrategias de decisión.

4 Modelado de sistemas de manufactura

Un Sistema Flexible de Fabricación es un sistema de eventos discretos y, por lo tanto, se puede abordar el diseño de su modelo de coordinación y toma de decisiones mediante la primera capa del entorno. Sin embargo, y dado la complejidad inherente de estos sistemas, se ha ampliado el entorno con una segunda capa especializada en SFF.

MIKRON (Manufacturing Intended KRON) [MURO 89a] [MURO 90] [VILL 90] es el resultado de la ampliación de KRON con un conjunto de objetos y primitivas específicas del dominio de los SFF. El modelo de coordinación de un SFF se construye en torno a varios tipos de primitivas de representación: recursos, materiales, productos, operaciones y planes de proceso. MIKRON aporta también una metodología específica de modelado. En ella se contempla los sistemas de fabricación desde dos puntos de vista diferentes: (1) *Funcional u orientado a las actividades*: se realiza la descripción de las actividades y operaciones y su encadenamiento para realizar la producción; y (2) *Estructural u orientada a los recursos*: se describen los modelos de funcionamiento de los recursos de producción y las interconexiones entre ellos.

Las primitivas de representación de MIKRON se implementan mediante objetos prototípicos predefinidos que se organizan en jerarquías de especialización, ampliables por el diseñador, y que parten de las jerarquías existentes en KRON.

Para la creación del modelo deseado de una planta de producción se sigue el siguiente proceso: 1) se instancian los prototipos de los recursos que componen el sistema a modelar y se rellenan los contenidos de los atributos

necesarios y 2) se relacionan las instancias creadas mediante la sincronización de objetos de acción que modelan la misma actividad.

La construcción de un modelo de plan de proceso se realiza de forma similar a como se construye el modelo de una planta: 1) se realizan instancias de los prototipos de operaciones que vayan a formar un plan y 2) se sincronizan estas operaciones, formando la secuencia de operaciones deseada.

El modelo de coordinación de un SFF debe reunir las restricciones de la perspectiva estructural, correspondiente al modelado de los recursos y su sincronización, y de la funcional, correspondiente a los planes de proceso, en un único modelo. Para ello se sincronizará el modelo del sistema de fabricación con el plan de proceso. De esta forma ambas perspectivas evolucionarán acompasadamente.

El objetivo del módulo SIMULADOR es facilitar el análisis del comportamiento de un SFF controlado por un modelo de control diseñado con el entorno. Mediante MIKRON es posible diseñar el modelo de coordinación de un SFF que, aumentado con una estrategia de toma de decisiones constituye el modelo de control del SFF. El simulador del entorno de diseño, realizado mediante técnicas de simulación de eventos discretos, tiene como objetivos específicos: 1) sustituir a la planta real de producción, simulando el intercambio de señales con el modelo de control, 2) recolectar datos de la simulación y generar los informes oportunos y 3) definir las condiciones de los experimentos estableciendo: el estado inicial, las condiciones de final de la simulación, los informes a generar, etc.

La toma de decisiones en manufactura surge como una especialización de la estrategia de resolución de conflictos comentada para la primera capa. Los conflictos, en el caso de un sistema de fabricación, están relacionados directamente con decisiones de planificación de operaciones ("scheduling"). El sistema de toma de decisiones dispone de una clasificación general de los conflictos de acuerdo con el problema de planificación que representan. Para cada conflicto, el

usuario tiene la posibilidad de elegir dentro de una biblioteca de políticas de control parametrizables, construidas en base a las reglas de despacho más comunes, cual es la política más adecuada para tratar de optimizar los objetivos de producción deseados.

Más adaptadas a la arquitectura jerárquica de control para la que se pretende diseñar software, son políticas de control que decidan tomando como referencia un plan de operaciones previamente construido. Actualmente esta posibilidad no está contemplada en el entorno, sin embargo, en un futuro próximo, se pretende incorporar a la toma de decisiones en manufactura un módulo de planificación de operaciones y una biblioteca de políticas de control que sean intérpretes de los planes generados. De esta manera, el software diseñado con el entorno, se adaptará perfectamente a la arquitectura de control.

GRAMAN, "GRAphic MANufacturing" es la interfase utilizada por el diseñador de modelos de manufactura. Está especializada en SFF y permite la manipulación gráfica de las primitivas que ofrecen los componentes de la segunda capa y, también, la realización de simulaciones animadas. El objetivo de su definición es el de facilitar la labor del diseñador y ocultar, al máximo posible, las características internas del modelo. En [VILL 88] se hace una definición preliminar de GRAMAN como una herramienta autónoma. En [VILL 90] se redefine GRAMAN como una interfase para MIKRON y en este trabajo se amplia su funcionalidad como interfase para la segunda capa del entorno.

En la figura 2 puede observarse un ejemplo de modelo de celda flexible realizado con GRAMAN. Cada recurso de un SFF se representa gráficamente mediante un bloque constructivo o icono. La colocación de un bloque constructivo en el plano de representación supone la creación de una instancia del prototipo deseado.

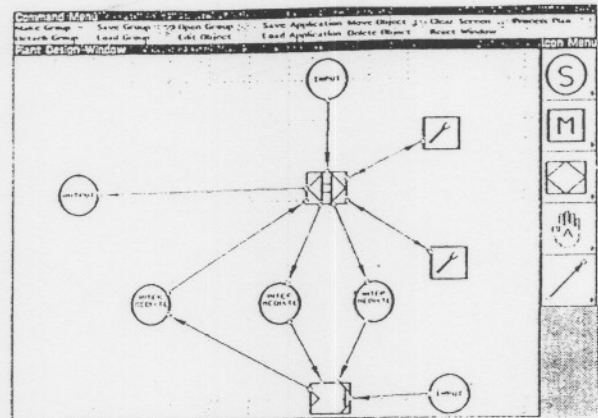


Figura 2: Ventana de GRAMAN que muestra un ejemplo de diseño de celda flexible.

5 Conclusiones

En este trabajo se ha presentado un entorno de diseño de software de control jerárquico de SFF, actualmente en desarrollo por el Grupo de Ingeniería de Sistemas e Informática del Centro Politécnico Superior de la Universidad de Zaragoza. El entorno está estructurado en dos capas, una genérica de diseño de sistemas de eventos discretos y otra especializada en SFF, que está construida sobre la anterior. Este tipo de estructura permite que la primera capa sea utilizada como una herramienta autónoma y como núcleo en entornos de diseño para otros dominios de aplicación. Además, ambas capas presentan interfases gráfico-textuales que facilitan el manejo de sus primitivas.

En el futuro se pretende ampliar la funcionalidad del entorno con: 1) la implementación de algoritmos de análisis formal de la red de Petri de alto nivel subyacente en los modelos que permitan detectar detectarmalos funcionamientos tales como bloqueos; 2) generadores de código tiempo real que permitan obtener programas de control a partir de los diseños realizados, en la línea de [VILL 90]; 3) un sistema de ayuda al diseño que guíe al diseñador y le informe del alcance de sus acciones (p.e. introducción de bloqueos); 4) un planificador de operaciones de manufactura.

Se ha desarrollado un prototipo del entorno presentado sobre LOOPS en una estación

de trabajo 1186 de Xerox. El entorno definitivo, con toda su funcionalidad, se está desarrollando sobre KEE en una SPARC-station de SUN.

Bibliografía

- [ALJA 88] R.Y. Al-Jaar and A.A. Desrochers: "A Survey of Petri Nets in Automated Manufacturing Systems". *Proc. of 12th World Congress on Scientific Computation, IMACS*, Vol. 3. Paris. 1988. pp.503-510.
- [FOX 83] Fox M.S.: "Constraint Directed Search: A Case Study of Job-Shop Scheduling." *PhD dissertation, Computer Science Dep., Carniage Mellon University. Pittsburgh., PA (USA).* 1983.
- [GENT 88] J.C. Gentina, J.P. Bourey and M. Kapusta: "Coloured Adaptive Structured Petri-Net: A Tool for the Automatic Syntesis of Hierarchical Control of Flexible Manufacturing Systems". *Computer-Integrated Manufacturing Systems*, Vol. 1, February 1988. pp. 39-47.
- [JENS 91] Jensen K. and Rozenberg G. (Eds.): "*High-level Petri Nets. Theory and Application*". Springer-Verlag, Berlin, 1991.
- [MART 87] Martínez J., Muro P.R. and Silva M.: "Modeling, Validation and Software Implementation of Production Systems using High Level Petri Nets". En *Proc. 1987 IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 999-1018 and 1160-1186. Raleigh, North Carolina, March-April 1987.
- [MART 89] J. Martínez, P.R. Muro, M. Silva, S.F. Smith, J.L. Villarroel: "Merging Artificial Intelligence Techniques and Petri Nets for Real Time Scheduling and Control of Production Systems". In *Artificial Intelligence in Scientific Computation: Towards second generation systems*. Scientific Publishing Co., IMACS, 1989, pp. 307-313.
- [MURO 89a] Muro P.R., Villarroel J.L., Martínez J. and Silva M.: "A Knowledge Representation Tool for Manufacturing Control Systems Design and Prototyping." *Proc. of IFAC/IFIC/IFORS/IMACS Symposium INCOM'89*. Madrid(Spain). September. 1989.
- [MURO 89b] P.R. Muro, J.L. Villarroel: "KRON: Redes Orientadas a la Representación del Conocimiento". *Actas de la III Reunión Técnica de Inteligencia Artificial, AEPIA '89*. Madrid. Noviembre 1989.
- [MURO 90] Muro P.R.: "*Aplicación de Técnicas de Inteligencia Artificial al Diseño de Sistemas Informáticos para el Control de Sistemas de Producción*". Tesis Doctoral. Depto. de Ingeniería eléctrica e Informática, Universidad de Zaragoza. Abril, 1990.
- [SILV 89] Silva M. and Valette R.: "Petri Nets and Flexible Manufacturing." In *Advances in Petri Nets 1989*, LNCS 424, Springer-Verlag. 1989. pp. 374-417.
- [SMIT 86] Smith S.F., Fox M.S. and Ow P.S.: "Constructing and Maintaining Detailed Production Plans: Investigations into the Development of Knowledge-Based Factory Scheduling Systems". *AI Magazine*, pp. 45-60, Fall. 1986.
- [VILL 88] J.L. Villarroel, J. Martínez, M. Silva: "GRAMAN: A Graphic System for Manufacturing System Design". *Proc. of the IMACS Symposium on System Modelling and Simulation*. Cetraro, Italy. September 1988. pp 311-316
- [VILL 90] Villarroel J.L.: "*Integración Informática del Control de Sistemas de Fabricación Flexible*". Tesis Doctoral. Departamento de Ingeniería eléctrica e Informática, Universidad de Zaragoza. Septiembre, 1990.



ROBÓTICA E AUTOMATIZAÇÃO

REVISTA DE INFORMAÇÃO TÉCNICA E CIENTÍFICA

PUBLICAÇÃO TRIMESTRAL
Nº 10 • Novembro 1992

ALEXANDRE TATO
Director

TENREIRO MACHADO
Director Científico

LUÍS SÁ
Director Adjunto

ANTÓNIO MALHEIRO
Editor

PAULO ANDREZ
Delegado em Lisboa

PAULO ALVES
Design

ALBANO PEREIRA
Marketing
Tel. (02) 510 05 18

SÉRGIO CASTRO
Assinaturas
Tel. (02) 57 77 28

CONSELHO EDITORIAL:
CAMARINHA MATOS
Robótica e CIM
EUGÉNIO OLIVEIRA
Inteligência Artificial
PINA CABRAL
Automação
PEREIRA DIAS
Electrónica Industrial
AMORIM MARQUES
Informática

ORGANIZAÇÃO E EDIÇÃO DESTE NÚMERO:
CAMARINHA MATOS

COLABORAÇÃO ESPECIAL:
STEIGER GARCÃO

PROPRIEDADE E ADMINISTRAÇÃO:
PUBLINDÚSTRIA
Produção de Comunicação, Lda
Rua Ferreira dos Santos, 124
4300 PORTO
Tel. (02) 56 54 82 Fax (02) 510 05 17

BRITISH REPRESENTATION:
ROBERT G. HORSFIELD
EDWARD J. KANIA
International Publishers Representatives
Daisy Bank - Chislebury
Via Stockport SK12 6DA - ENGLAND
Tel. (0663) 75 02 42 Fax (0663) 75 09 74

COMPOSIÇÃO, FOTOLITO E MONTAGEM:
PUBLINDÚSTRIA

IMPRESSÃO E ACABAMENTO:
PUBLICOR
Serzedo - V. N. GAIA
Tel. (02) 762 54 86

TIRAGEM:
3.000 EXEMPLARES

DEPÓSITO LEGAL:
113164



Os trabalhos assinados
são da exclusiva responsabilidade
dos seus autores

SUMÁRIO

3 EDITORIAL

5 APRESENTAÇÃO

6 CURRICULAVITAE

ANÁLISE DE SISTEMAS
DE MANUFATURA COM BASE
EM REDES DE PETRI

11 INTEGRACIÓN DE REDES DE PETRI
EN EL DISEÑO ORIENTADO
A OBJECTO DE SISTEMAS
DE FABRICACIÓN

19 ANÁLISE DE SISTEMAS INTEGRADOS
DE MANUFATURA BASEADA
NA METODOLOGIA MFG/PFS

25 ANÁLISE SEMI-AUTOMÁTICA
DE MARK FLOW GRAPHS

31 ESPECIFICAÇÃO E REALIZAÇÃO
DE CONTROLADORES UTILIZANDO
REDES DE PETRI COLORIDAS
E SINCRONIZADAS INTEGRANDO
LÓGICA IMPRECISA

ESCALONAMENTO E CONTROLE

39 UMA METODOLOGIA
PARA IMPLEMENTAÇÃO
DE SISTEMAS DE CONTROLE
E PROJEÇÃO TEMPORAL
DE ACTIVIDADES

45 UMA APROXIMAÇÃO
AO ESCALONAMENTO DINÂMICO
COM BASE EM NEGOCIAÇÃO

55 SCHEDULING EN CELDAS
AUTÓNOMAS DE ENSAMBLAJE
BASADO EN UN ALGORITMO
DE TIEMPO MÍNIMO EN R/dP

PLANEAMENTO E MONITORAÇÃO
EM ROBOTICA

61 PROGRAMACIÓN IMPLÍCITA
DE ROBOTS:
UNA APROXIMACIÓN FORMAL

69 PLANEAMENTO EFICIENTE
EM TAREFAS DE MONTAGEM

75 EXECUÇÃO, MONITORAÇÃO
SENSORIAL E REACÇÃO
A SITUAÇÕES DE EXCEÇÃO
EM ROBOTICA DE MONTAGEM

CONCEPÇÃO DE SISTEMAS
DE MANUFATURA

81 ENTORNO DE DESARROLLO
DEL CONTROL DE SISTEMAS
DE MANUFATURA

87 DISEÑO DE CELULAS FLEXIBLES
DE MANUFATURA

91 SADS INTELIGENTES
PARA PLANEJAMENTO
EM MANUFATURA:
UMA ABORDAGEM VIA MODELAGEM
VISUAL INTERATIVA

PROGNÓSTICO, VEÍCULOS
AUTÓNOMOS E SEQUENCIAMENTO
DE TRÁFEGO

95 CARACTERIZAÇÃO AUTOMÁTICA
DE PADRÕES EM SENSORES
DE MÁQUINAS CNC

107 SIMULAÇÃO DE VEÍCULOS
AUTÓNOMOS SUBAQUÁTICOS
CONTROLADOS
POR COMPORTAMENTOS

113 UMA APROXIMAÇÃO "MUNDOS
POSSÍVEIS" AOS PROBLEMAS
DE ESCALONAMENTO DE VOOS

Agradecemos à UNISYS a excelente colaboração
que prestou na informatização dos nossos serviços