

DISEÑO Y ANÁLISIS DE SISTEMAS PRODUCTIVOS UTILIZANDO LA OPTIMIZACIÓN MEDIANTE SIMULACIÓN BASADA EN INTERNET [♠]

PRODUCTION SYSTEM DESIGN AND ANALYSIS USING INTERNET- ENABLED SIMULATION BASED OPTIMIZATION

Matías Urenda Moris^{♠*}, Amos H.C. NG¹, Jacob Benedixen¹, Ainhoa Goienetxea Uriarte¹

RESUMEN

Los ingenieros de producción buscan continuamente configuraciones óptimas de sus sistemas productivos, con el fin de alcanzar el máximo rendimiento en los mismos. Este es un trabajo que consume mucho tiempo en la industria, y rara vez se le conceden los recursos necesarios para que las decisiones a tomar sean las acertadas. Para poder tomar estas decisiones de una manera más eficiente, es necesario disponer de herramientas nuevas e innovadoras. Este artículo presenta la segunda generación de FACTS Analyzer, una herramienta cuya función se basa en la integración de una variedad de metodologías y técnicas, incluyendo: el modelado rápido por medio de un nuevo concepto de agregación y funciones simplificadas, la generación de modelos de simulación de manera automática y optimización mediante simulación apoyada por una innovadora arquitectura basada en Internet. Dichas metodologías y técnicas permiten que la simulación sea mucho más fácil de utilizar, y aceleran el tiempo de diseño y desarrollo de modelos, la fase de experimentación y los procesos de optimización, con el fin de apoyar el proceso de toma de decisiones. FACTS Analyzer es el resultado de varios años de investigación y desarrollo, en estrecha colaboración con la industria automotriz en Suecia.

Palabras clave: Optimización mediante Simulación, optimización multiobjetivo, diseño de sistemas productivos.

ABSTRACT

Production engineers struggle continuously in seeking the optimal configuration in order to maximize the performance of their production systems. This is a time-consuming work in which they seldom get the necessary resources in order to make the right decisions. New and innovative tools are therefore needed to support high-quality decision making in a rapid manner. This paper introduces the second generation of FACTS Analyzer, a toolset developed based on the concept of integrating model abstraction, automatic model generation and simulation-based optimization under an innovative Internet-based platform. Specifically, it addresses a novel model aggregation and generation method, which when combined together with other system components, like optimization engines, can synthetically enable simulation to become

[♠]IX Congreso del Instituto Chileno de Investigación Operativa, 26-29 de octubre, Pucon. Chile

¹Virtual Systems Research Centre. University of Skövde. PO Box 408, 54128 Skövde, Sweden, amos.ng@his.se, jacob.benedixen@his.se, ainhoa.goienetxea@his.se

Autor para correspondencia: ^{*}matias.urenda.moris@his.se

Recibido: 22.09.2011 Aceptado: 03.05.2012

much easier to use and speed up the time-consuming model building, experimentation and optimization processes, in order to support optimal decision making. Such a toolset is the result of several years of research and development in close collaboration with the Swedish automotive industry.

Keywords: Simulation-based Optimization, Multi-objective optimization, Production Systems Design.

INTRODUCCIÓN

Los sistemas de producción industriales son a menudo sumamente complejos. La dimensión y complejidad de estos sistemas, junto al carácter dinámico y estocástico de los componentes individuales, así como sus dependencias e interacciones, hacen muy difícil obtener una configuración óptima del sistema en cuestión. Asimismo, la limitación del tiempo siempre está presente en el entorno productivo, por lo que los ingenieros de producción muchas veces basan sus decisiones sólo en la intuición o en sencillos análisis. A pesar de que la Simulación de Eventos Discretos (DES, en sus siglas en inglés) posee una técnica y metodología establecida, y cuyas ventajas en el diseño y análisis de sistemas complejos de producción están bien reconocidos, su uso como herramienta para la toma de decisiones no ha sido exitoso en la industria manufacturera, especialmente durante la primera fase conceptual de diseño de sistemas productivos (McNally & Heavey 2004). Algunas de las causas por las cuales la industria no hace uso de DES son: (1) el extenso tiempo que toma el desarrollo de modelos DES; (2) la falta de experiencia en simulación, y (3) que los modelos de DES, por naturaleza son evaluadores y no necesariamente proporcionan al analista una solución óptima. Además, el proceso de búsqueda de un diseño lo suficientemente bueno puede tomar mucho tiempo, y muchas veces puede ser imposible si el espacio de búsqueda es extenso y el tiempo disponible limitado. La optimización mediante simulación es una técnica relativamente nueva, que se aplica para buscar la configuración “óptima” a un sistema complejo, basándose en la evaluación de uno o varios parámetros del sistema. Dichos parámetros son generados a partir de la simulación mediante el uso de diversas metodologías de búsqueda. La optimización mediante simulación es una tecnología que ofrece un enorme potencial para resolver problemas de sistemas reales y se ha aplicado con éxito en diversos ámbitos (April *et al.*, 2004).

Sin embargo, prácticamente todos los paquetes de software comerciales de optimización mediante simulación que existen hasta el momento poseen aún varias limitaciones importantes: (1) trabajan en un modo determinista, sin tener en cuenta los resultados estocásticos de la DES; (2) no tratan explícitamente problemas multiobjetivo, y (3) similar a la mayoría de los paquetes de DES, las herramientas de optimización mediante simulación como otros software tradicionales, se deben instalar y ejecutar localmente en los equipos de los usuarios. En la década de los 90 se hicieron esfuerzos importantes centrados en la Simulación Basada en la Web (WBS, en sus siglas en inglés). Dichos esfuerzos estaban fuertemente incentivados por la visión de que Internet y las tecnologías Web podrían permitir el crecimiento exponencial de oportunidades comerciales y de investigación (Fu *et al.*, 2000) (Boesel *et al.*, 2001) (Miller *et al.* 2001). A pesar de este comienzo prometedor, Byrne *et al.*, (2010) afirman que la investigación en el área de la WBS se encuentra todavía en sus comienzos y que los esfuerzos prometidos no se han llevado a cabo hacia verdaderos consumidores/clientes, sino que más bien se han tratado de aplicaciones experimentales. Para que las herramientas de WBS realmente lleguen a tener un verdadero impacto en resolver problemas reales, se requiere un mayor trabajo de investigación para saber cómo han de diseñarse estas herramientas y apoyar no sólo la simulación sino también la optimización, en un entorno distribuido y multiusuario. El objetivo final de este trabajo es explorar las oportunidades que ofrece la tecnología de optimización

mediante simulación basada en la Web para la solución de problemas reales en la industria, a través del diseño e implementación de FACTS Analyzer.

Este artículo presenta la segunda generación de un software basado en Web, DES y optimización mediante simulación, llamado FACTS Analyzer (Análisis de Fábrica en la fase Conceptual utilizando Simulación), que se ha desarrollado específicamente para apoyar el diseño, análisis y mejora de los sistemas de producción, dentro de un contexto multiobjetivo. Implementado como un sistema cliente-servidor a través de Internet, FACTS Analyzer (a partir de aquí FACTS) es un software de optimización mediante simulación paralela y distribuida, capaz de ejecutar múltiples experimentos de DES y procesos de optimización mediante simulación de forma simultánea. En el servidor de FACTS se hayan integrados varios algoritmos de optimización, metamodelos basados en redes neuronales artificiales (ANN, en sus siglas en inglés), sistemas estocásticos de simulación y un sistema de base de datos SQL, todo puesto a disposición de los usuarios a través de servicios web.

Además de exponer la arquitectura del sistema de FACTS en la sección metodología, este artículo presenta las capacidades únicas proporcionadas por FACTS, que van más allá de las que se encuentran en los simuladores convencionales de DES; por ejemplo, la generación automática de modelos complejos en base a parámetros de optimización (sección generación modelo..). Asimismo, utilizando como base un ejemplo sencillo, este artículo presenta cómo FACTS puede apoyar la generación de soluciones óptimas de Pareto (el mejor trade-off) para la toma de decisiones en la mejora de los sistemas de producción (sección optimización multiobjetivo)

FACTS ANALYZER: METODOLOGÍA Y MATERIALES

Desarrollo

La arquitectura de FACTS y el paradigma de modelación de la aplicación cliente, primeramente descritos en (Ng *et al.*, 2007), (Urenda *et al.*, 2008) y (Ng *et al.*, 2009), se han ampliado y mejorado en los últimos años. Se ha hecho un esfuerzo concienzudo para mantener la simplicidad del FACTS de la primera generación, pero al mismo tiempo se ha buscado aumentar la capacidad de modelación y las posibilidades de análisis de la herramienta. Un número importante de estudios efectuados en la industria automotriz ha servido de ayuda en la identificación de carencias en el conjunto de objetos de modelado de la herramienta, identificando también características y funcionalidades necesarias en las mismas (Benjamin, 2010; Zomaya, 2010). Además, durante estos años un grupo de la industria automotriz ha utilizado de manera continua la versión inicial de FACTS para el diseño y análisis de sistemas de producción. Sus opiniones y experiencias han sido de gran valor en el reciente desarrollo de FACTS. No sólo han sido de gran ayuda en la adaptación y el aumento del conjunto de objetos que se utilizan para modelar, sino que más importante aún ha sido su aporte en la adaptación de las funciones especializadas del software. Estas funciones se han generado del conocimiento adquirido de los problemas comunes que se encuentran en el día a día de este grupo de ingenieros. En Pehrsson *et al.*, (2011), Pehrsson y Ng (2011), así como Bernedixen y Ng (2011) se presentan típicas situaciones donde se buscan mejoras a través de la optimización mediante simulación. Actualmente están siendo implementadas en la herramienta funciones adaptadas para resolver estos problemas. Sin esta integración estrecha entre los investigadores y los usuarios industriales, esta obra no habría llegado al nivel de aceptación y utilidad que tiene hoy entre nuestros socios industriales. Como resultado de sus experiencias positivas, la Universidad de Skövde ha desarrollado, junto con la industria, un curso intensivo de 5 días en diseño de sistemas de producción. Este curso comenzó a impartirse en otoño del 2011, y para mediados del 2012 ya ha formado a más de 100 ingenieros, utilizando FACTS como principal herramienta para el diseño, análisis y

optimización de sistemas productivos. El próximo otoño 2012 se seguirá con la formación de más ingenieros de la industria.

Arquitectura del sistema

Los principales componentes y protocolos de comunicación de la arquitectura del sistema de FACTS, la cual puede soportar simultáneamente múltiples usuarios a través de Internet, se ilustra esquemáticamente en la figura 1. En el núcleo de esta arquitectura se encuentra el sistema de componentes del servidor, que consta de cuatro sub-sistemas distribuidos: (1) el servidor Web, (2) el servidor de optimización, (3) el servidor de base de datos, y (4) el clúster de simulación. Para un proceso de optimización mediante simulación ejecutado a través de FACTS, el motor de optimización (*OptEngine*) en el servidor de optimización es el componente más importante, ya que proporciona la funcionalidad básica para el procesamiento de los principales algoritmos y actúa como el centro de coordinación de funciones, incluyendo la interacción con el usuario, el envío de los datos de simulación y optimización para los grupos de simulación y el servidor de bases de datos, así como para el uso ocasional de metamodelos. En principio, los componentes del servidor pueden ser accedidos por cualquier aplicación cliente a través de los servicios Web alojados en el servidor Web. En la implementación actual, la aplicación cliente de FACTS es la aplicación con mayor consumo de funcionalidades del servidor de FACTS a través del envío de peticiones XML y especificaciones del modelo en forma de archivos XML, poniendo en marcha/controlando los procesos de optimización mediante simulación (a través de *OptManager*) y recuperando los datos de la optimización de la base de datos de optimización (*OptDB*).

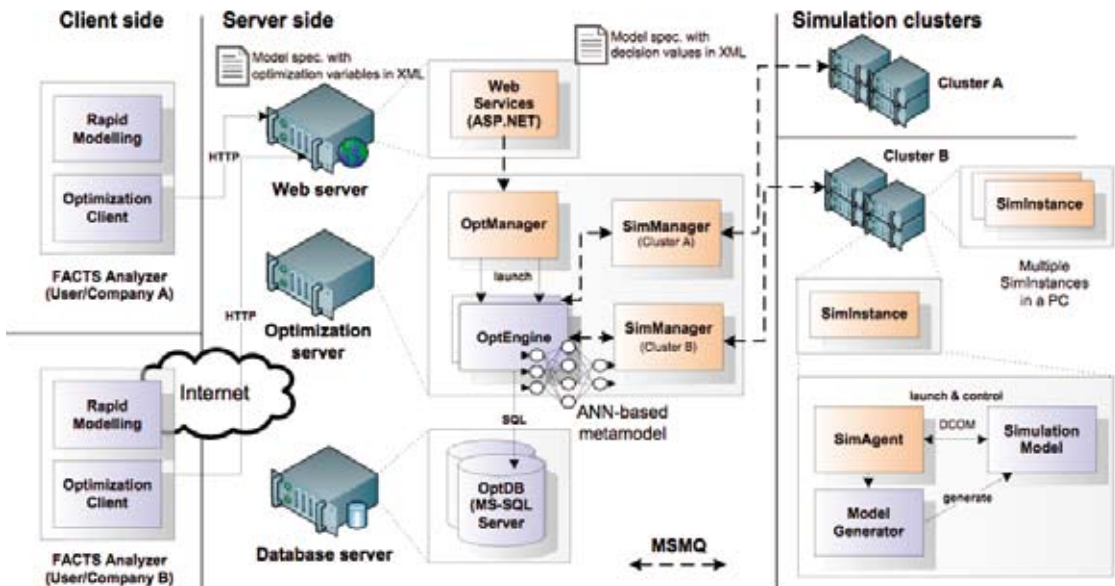


Figura 1. La arquitectura del sistema de FACTS Analyzer en un entorno multiusuario

El gestor de optimización (*OptManager*) es un servicio de Windows que escucha las peticiones desde el servidor Web para lanzar *OptEngines* diferentes de acuerdo a las preferencias y los parámetros especificados por el usuario a través de FACTS. Los datos requeridos para iniciar un procedimiento de optimización mediante simulación pueden incluir: (1) la configuración de los parámetros de simulación, tales como el tiempo de calentamiento (warm-up), el tiempo de simulación y el número de repeticiones, (2) las funciones multiobjetivo; (3) la lista de variables de entrada, (4) la lista de variables de salida, (5) las restricciones; (6), el algoritmo

de optimización a utilizar, y (7) los parámetros de optimización, como por ejemplo el tamaño de la población o la tasa de cruce (crossover rate) si se selecciona un algoritmo evolutivo (EA, en sus siglas en inglés). Las dos últimas opciones son especialmente interesantes para los investigadores que trabajan con optimización mediante simulación, ya que la herramienta les permite seleccionar y comparar el rendimiento de diferentes algoritmos o los diferentes parámetros de optimización. En la actualidad, el servidor FACTS puede trabajar con algoritmos evolutivos asistidos por metamodelos, pero los nuevos algoritmos se pueden añadir fácilmente compilando en la plataforma el núcleo del algoritmo modificado con librerías desarrolladas con orientación a objetos (OO).

FACTS permite a todos los *OptEngines* activos guardar en la base de datos central, *OptDB*, sus trayectorias de optimización y otros resultados de los experimentos. Esto le permite a FACTS disponer de las siguientes características avanzadas:

- La calidad y diversidad de las soluciones iniciales juegan un papel crucial en el rendimiento de un proceso de optimización, especialmente cuando se utiliza un método de búsqueda en una población determinada (population based search method). FACTS guarda todos los resultados experimentales en *OptDB* y, al iniciar un nuevo proceso de optimización, permite al usuario escoger el conjunto de soluciones iniciales de los registros obtenidos en experimentos previos que se hayan realizado. Esto se utiliza normalmente en conjunto con las funciones de Diseño de Experimentos (DoE, en sus siglas en inglés) del cliente de FACTS.
- Cambiar dinámicamente los algoritmos metaheurísticos en un proceso de optimización. Esto es especialmente útil cuando un método de búsqueda global como los algoritmos evolutivos, es usado inicialmente para exploración y seguidos de métodos de búsqueda locales, como puede ser el del algoritmo descenso por gradiente (Hill Climbing, en inglés) para poder conseguir un resultado de optimización aún mejor.
- La tolerancia a fallos. Mientras que los fallos en una simulación pueden ser fácilmente detectados y recuperados volviendo a iniciar la ejecución con un *SimAgent*, los fallos que pueden aparecer en el *OptEngine* pueden causar un punto único de fallo (single point of failure) y perder el tiempo empleado en las simulaciones previas, si los datos de optimización no están almacenados. La arquitectura de FACTS indirectamente facilita la recuperación del error, permitiendo al usuario comenzar un nuevo *OptEngine* y recuperando los registros de la optimización mediante simulación guardados en el *OptDB* como soluciones iniciales y conjunto de datos (training data set) para el metamodelo, para cuando el proceso de búsqueda se vuelva a iniciar.

Componentes de simulación

Los componentes de simulación están separados de los componentes del núcleo del servidor, porque pueden ser distribuidos en los clusters de ordenadores para soportar ejecuciones paralelas de simulaciones que consumen muchos recursos en el procesador y para poder así acelerar los procesos de optimización. El diseño de la plataforma permite trabajar con varios tipos de simulaciones para conectar el *OptEngine* a través de la tecnología *SimAgent*, la cual facilita a los sistemas heterogéneos de simulación ser conectados a *SimManager* en un protocolo unificado utilizando Microsoft Message Queues (MSMQ), tal y como se puede observar en la Figura N°1. Dependiendo de los métodos de interfaz de la aplicación soportados por el sistema de simulación objetivo, *SimAgent* puede comenzar con el correspondiente objeto *BackEnd* para iniciar/lanzar, interactuar y controlar el software de simulación. En la implementación actual, FACTS admite dos tipos de generación de modelos: (1) modelos DES generados en software

comerciales y utilizando algunos generadores de modelos personalizados, o (2) modelos binarios con librerías dinámicas vinculadas. La primera opción permite generar un modelo DES basado en las especificaciones del modelo FACTS y posteriormente poder ser modificado para incluir algunas lógicas específicas o detalles. Mientras que, en la segunda opción, la principal preocupación del usuario es conseguir una velocidad óptima de ejecución. Para el primer caso es necesario un objeto *BackEnd* para comunicarse con el software DES; por ejemplo, a través de DCOM. Los datos de salida provenientes de las evaluaciones de la simulación son “ensamblados” y enviados de nuevo en un formato estándar mediante *SimManager*, en forma de MSMQ, y devueltos al correspondiente *OptEngine* para su posterior procesamiento y registro de datos.

Más de un *SimInstance*, que consiste en un *SimAgent* controlando una única ejecución de la simulación, puede comenzarse en un único nodo con procesadores multi-núcleo en el clúster de simulación. Cuando se inicia un *SimAgent*, se registrará en el *SimManager* para anunciar su existencia. Sabiendo el número de *SimAgents* disponibles, el *SimManager* puede enviar múltiples trabajos recibidos de los *OptEngines* a las múltiples simulaciones que se ejecutan en paralelo.

La aplicación cliente de FACTS Analyzer

Para permitir el modelado rápido de los sistemas de producción, especialmente en la fase de diseño conceptual, FACTS proporciona un número limitado de objetos estándar de DES, combinado con una lista de objetos que tratan con los Mecanismos de Control de Producción (MCP), como se indica en la Tabla N°1. Los objetos MCP han sido desarrollados teniendo en cuenta la base de experiencias adquiridas en estudios de casos industriales. Sus funcionalidades se han diseñado de forma genérica y, al mismo tiempo, se ha buscado mantener su facilidad de uso.

Tabla 1: Objetos de modelado de FACTS (a la derecha objetos estándares y a la izquierda los de MCP).

Icono	Nombre	Descripción	Icono	Nombre	Descripción
	Fuente	Para controlar cómo el material entra en el modelo.		Horario	Para el modelado de jornadas de trabajo, es decir para controlar cuándo la producción en su totalidad o en partes de la línea de producción está permitida.
	Pozo	Para controlar cómo el material deja el modelo.		Ritmo	Sincroniza la producción de una línea en serie, puede usarse para secciones o para el flujo de toda la línea.
	Operación	Operación estándar de procesamiento del material.		Demanda	Para el modelado de demanda. La demanda puede ser satisfecha en uno o más lugares en un flujo de producción. Contiene estadísticas importante como retrasos pendientes, tardanza.
	Montaje	Operación en el cual dos o más partes son ensambladas, siguiendo la descripción del montaje.		Lote	Para agrupar variantes de productos en elegidas operaciones de la línea de producción; minimiza la cantidad de preparación de las celdas de trabajo. El tamaño y el orden de los lotes pueden basarse en niveles de seguridad de buffers y almacenes en el modelo.
	Desmontaje	Operación en el cual dos o más partes son separadas.		Kanban	Sistema de arrastre que autoriza la producción en una operación basada en la tarjeta Kanban recibida del subsiguiente buffer/operación.
	Buffer	Almacén que mantiene una o más piezas por un tiempo mínimo, la secuencia de las piezas se mantiene.		WIP Máximo	Mecanismo que limita la cantidad total de productos en proceso (WIP) en una parte o en toda la línea de producción.
	Almacén	Igual al buffer pero está permitido cambiar la secuencia de las piezas.		WIP Crítico	Mecanismo que limita la cantidad de WIP en una parte o en toda la línea de producción basándose en el nivel de un almacén/buffer.
	Componente	Objeto para modelar componentes personalizados, los cuales pueden repetirse en más lugares en la línea de producción.		Selección	Selección es una herramienta potente que permite, de una manera sencilla, la comparación y evaluación de diferentes configuraciones del sistema de producción.

Los objetos MCP permiten modelar los flujos complejos, que comúnmente se encuentran en las líneas de producción, sin la necesidad de personalizar la funcionalidad del modelo a través de la programación. A modo de ejemplo, en la figura 2 se muestra la utilización de los objetos estándar DES y los MCP para modelar una línea existente de mecanizado de motores.

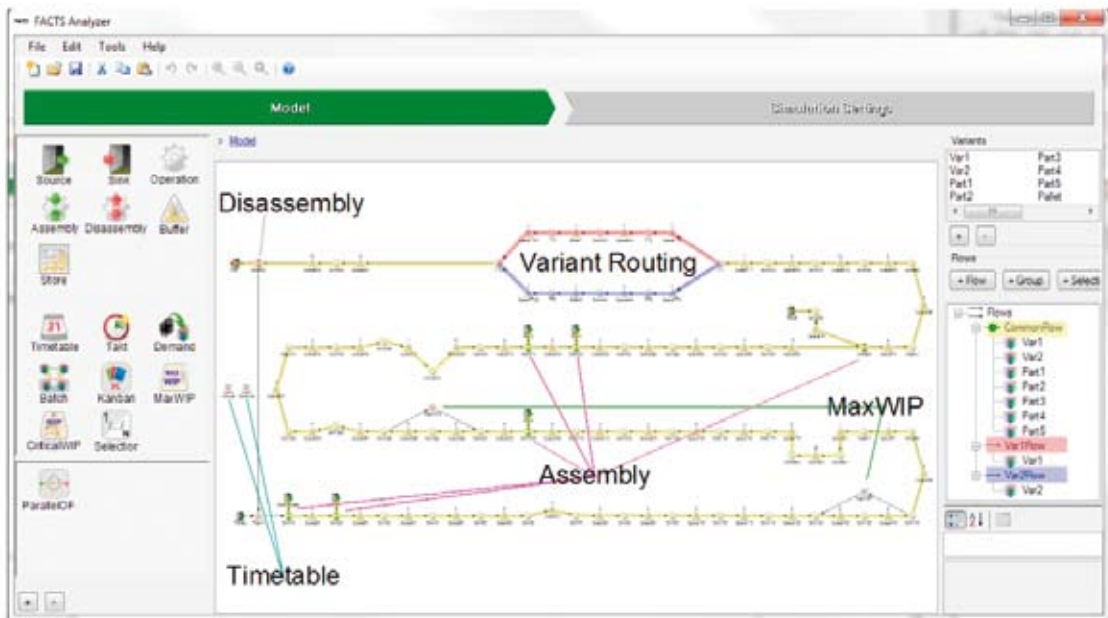


Figura 2. El uso de objetos de FACTS Analyzer para modelar una línea de mecanizado de automoción.

FACTS tiene una novedosa manera de modelar los flujos de producto dentro de la línea, tomando en consideración a diferentes variantes o grupos de variantes. Estos flujos de materiales se crean de forma similar a como los flujos de producción tradicionalmente se crean en los software de DES, es decir, mediante la conexión de los objetos en el modelo con conectores/flechas. Sin embargo, la principal diferencia es que FACTS permite la creación de múltiples flujos recogidos en una estructura de árbol (ver el lado derecho de la Figura N°2). Dentro de esta estructura de árbol también existe la posibilidad de crear la función de selección, similar al objeto MCP de selección, pero en este caso en vez de optimizar configuraciones de recursos, el proceso de optimización buscaría la combinación de flujo/producto más óptima de acuerdo a los objetivos. El concepto de selección en FACTS y su implementación permite que el usuario utilice la optimización mediante simulación para automáticamente evaluar diferentes alternativas de diseño; esto se ilustra más en detalle en el siguiente apartado:

Generación del modelo en base a parámetros de optimización: descripción de un ejemplo

En la fase de diseño conceptual de un sistema de producción es muy común que existan múltiples escenarios de cómo configurar este sistema de producción. Tener un modelo por escenario y, por lo tanto, un análisis por cada escenario, fácilmente limita la cantidad de escenarios que podrían ser evaluados debido al tiempo establecido que hay para la toma de decisiones en la industria. Lo ideal sería ser capaz de analizar todos los escenarios, representados en las diferentes alternativas de diseño, de una manera eficiente. En este tipo de escenarios es probable que gran parte del flujo de producción sea idéntico, con diferencias en lugares específicos. La función de selección (tanto la selección de objetos como la selección de flujo) permite que estas alternativas de diseño se construyan en forma de bloques o se definan en el árbol de selección de flujos, para que más tarde el proceso de evaluación cambie entre las diferentes combinaciones de forma automática. El análisis de este modelo y sus variantes se realiza a través del proceso de optimización mediante simulación para buscar la "mejor" combinación de alternativas de diseño, con sus valores óptimos. Pero también se buscan los

valores óptimos de otras variables pertenecientes a los objetivos de optimización. La solución “óptima” se puede, por lo tanto, alcanzar en una sola ejecución de optimización mediante simulación.

La figura 3 muestra cómo diseños alternativos fácilmente pueden ser modelados en FACTS gracias a la metodología de selección, ya sea a través de la selección de recursos o selección de flujos.

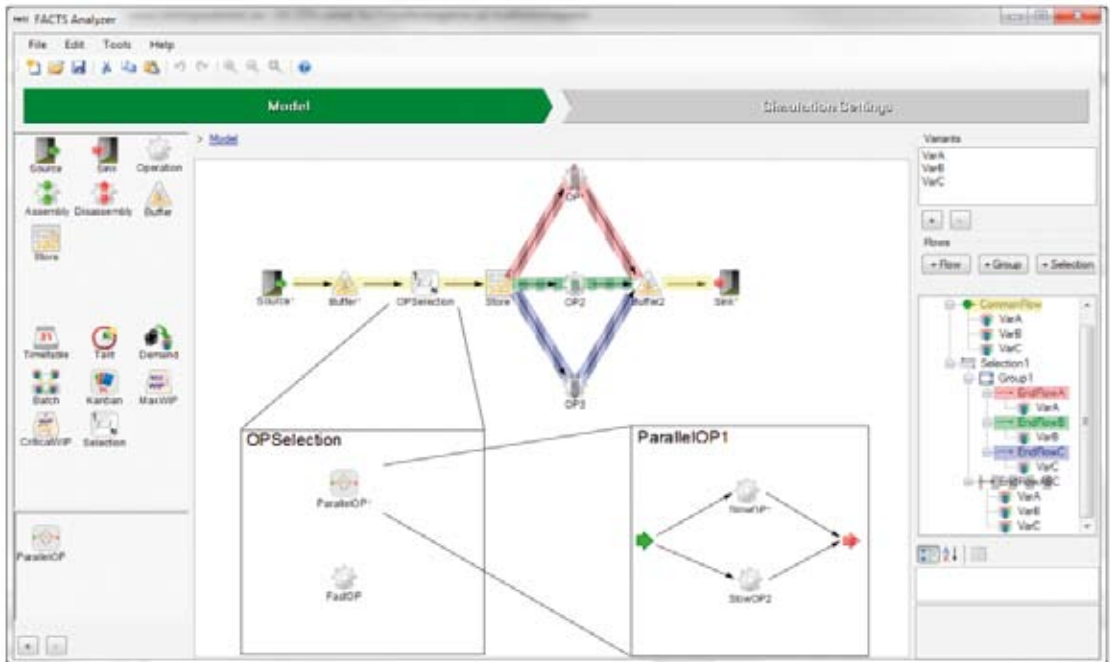


Figura 3. Ilustración de la función de selección: selección de objetos y selección de flujos en FACTS Analyzer

A continuación, para ilustrar su uso, utilizaremos un supuesto escenario considerando dos selecciones. En la primera se utiliza el *objeto de selección*, *OPSelection*. La selección constituye la opción de tener una máquina rápida en lugar de dos máquinas lentas en paralelo. Se desea considerar si la sustitución de las máquinas antiguas y lentas por una o más máquinas nuevas y rápidas conlleva la mejora esperada, este tipo de problema es común en la industria automotriz. La segunda selección es de flujo para variantes de productos. Para las tres operaciones, OP1, OP2 y OP3, que están conectadas a buffer2, hay una selección de flujo en el que dos alternativas están siendo evaluadas. La primera alternativa se basa en que las tres variantes de productos (VarA, VarB y VarC) tengan su propio flujo (marcados en color rojo, verde y azul). En la segunda opción se permite que las tres variantes sean procesadas en cualquiera de las tres máquinas (esto conlleva una inversión para obtener la flexibilidad necesaria), pero con la desventaja de que al cambiar variante en una operación se genera la necesidad de hacer una preparación (setup) de la operación, y esto conlleva un tiempo adicional. Por lo tanto, además de tomar la decisión de reemplazar o no las máquinas viejas por una nueva, se tiene que evaluar si es rentable invertir en máquinas flexibles para poder cambiar el flujo de productos a uno más flexible.

El modelo presentado en la figura 3 se utilizará para ilustrar la utilidad de la aplicación de optimización con el modelado de selección. La configuración original de este modelo se puede encontrar en la parte izquierda de la tabla 2. Las selecciones alternativas, junto con algunas

posibles mejoras adicionales, están todas modeladas como inversiones, con sus costos asociados. Estas se presentan en la parte derecha de la tabla 2.

Tabla 2. La configuración inicial (izquierda) y los costos de inversión (lado derecho) del modelo básico.

Objetos*	Configuración	Objetos	Inversión y costos
Modelo	El modelo se ejecuta con producción continua (sin horario de pausas) durante 31 días, de los cuales uno se utiliza como período de calentamiento e incluyendo 15 repeticiones para hacer frente a los elementos estocásticos en el modelo.	Buffer1, Store1 (almacén1)	Posibilidad de comprar desde 1 hasta 5 plazas adicionales cada una cuesta 10,000\$.
Source1 (fuente1)	Fuente infinita aleatoria de las tres variantes (VarA – 40 %, VarB – 30 %, VarC – 30 %).	OPSelection (selección de objetos)	La inversión en la nueva máquina (FastOP) coste 100,000\$.
Buffer1, Store1 (almacén1), Buffer2	Los almacenes tienen un tiempo de almacenaje mínimo (transporte) de 60 segundos y una capacidad para almacenar 5 unidades.	SlowOP1, SlowOP2, OP1, OP2, OP3	Mejora de la disponibilidad del 90% al 95% en pasos de 1%. Un coste de 10.000\$ por paso.
SlowOP1, SlowOP2	Tiempo de procesamiento constante de 7.000 segundos, con una disponibilidad del 90% y un tiempo medio de reparación (MTTR) de 5 minutos.	SlowOP1, SlowOP2, OP1, OP2, OP3	Mejora de MTTR de 5 minutos a 3 minutos en intervalos de 1 minuto. Un coste de 10.000\$ por minuto.
FastOP	Tiempo de procesamiento constante de 3.600 segundos, con una disponibilidad del 95% y un MTTR de 3 minutos.	Selection1 (selección de flujo)	Mejorar las máquinas a máquinas multiobjetivo/flexibles que puedan procesar todas las variantes. Coste 90.000\$.
OP1, OP2, OP3	Tiempo de procesamiento constante de 9.000 segundos para VarA y 1,2000 segundos para VarB y VarC y con una disponibilidad del 90% y un MTTR de 5 minutos. Tiempo de preparación (setup) de 10 minutos para pasar o dejar de producir VarC.	* Los nombres en inglés se han mantenido para poder relacionarlos con los términos de la Figura N°3.	
Sink1 (pozo1)	Demanda infinita.		

Como se ha mencionado anteriormente, los modelos creados mediante FACTS se almacenan y se envían al servidor utilizando XML. Estos modelos (archivos XML), pueden ser interpretados por un generador de modelo (Figura 1) en cada *SimInstance* del clúster de simulación distribuida. Sin embargo, antes de que el modelo se envíe a los generadores de modelo, la señal es procesada por el *OptEngine*, el cual establece los valores de las variables de decisión. A continuación, cada generador de modelos interpreta el archivo XML y genera el modelo de acuerdo a los valores que el algoritmo de optimización definió. Para los *objetos de selección* y las *selecciones de flujo*, un elemento de selección rodea todas las opciones disponibles (objetos/flujo), con el atributo *ActiveID*. Tal atributo define la/s selección/es que en tal instancia serán evaluadas. Esta manera de modelar y almacenar selecciones proporciona un modo sencillo de alternar entre las opciones disponibles y un aspecto clave en la incorporación de la función de selección en un proceso de optimización mediante simulación.

OPTIMIZACIÓN MULTIOBJETIVO

Se han ejecutado optimizaciones multiobjetivo (MOO, en sus siglas en inglés) para el caso hipotético descrito (en la sección generación de un modelo...) de este artículo. Para ayudar a tomar la mejor decisión, teniendo en cuenta las variables de inversión y rendimiento, se ha generado el diagrama de Pareto utilizando en este caso una variante del NSGA-II (Deb et al., 2002). FACTS Analyzer provee a los usuarios de las funcionalidades necesarias para dibujar la mejor, media o peor attainment surface (Knowles, 2005) de las soluciones óptimas de Pareto obtenidas de las varias ejecuciones de MOO. La figura 4 muestra cómo el usuario

puede agrupar los resultados de tres replicaciones de la optimización mediante simulación, para dibujar el mejor attainment surface para el ejemplo analizado (Figura 4 el camino de la izquierda). Para una ejecución de la optimización mediante simulación particular, el usuario puede navegar por los resultados de la optimización y seleccionar cualquier solución para analizar en detalle las medidas de rendimiento y sus estadísticas (Figura 4 el camino de la derecha). En este ejemplo, los óptimos de Pareto para una inversión de 90,000\$ han mostrado un salto del rendimiento (8.6% de mejora) y puede ser considerado como un punto de inflexión (knee point, en inglés) en la literatura de MOO. Este representa la mejor solución, ya que se puede obtener una mejora significativa del rendimiento con una inversión adicional de 10,000\$, la cual no se puede obtener con ninguna otra solución óptima de Pareto. La solución óptima de Pareto de inversión igual a 90,000\$ es únicamente atribuida a la inversión en máquinas flexibles, sin hacer ningún otro cambio en el modelo original.

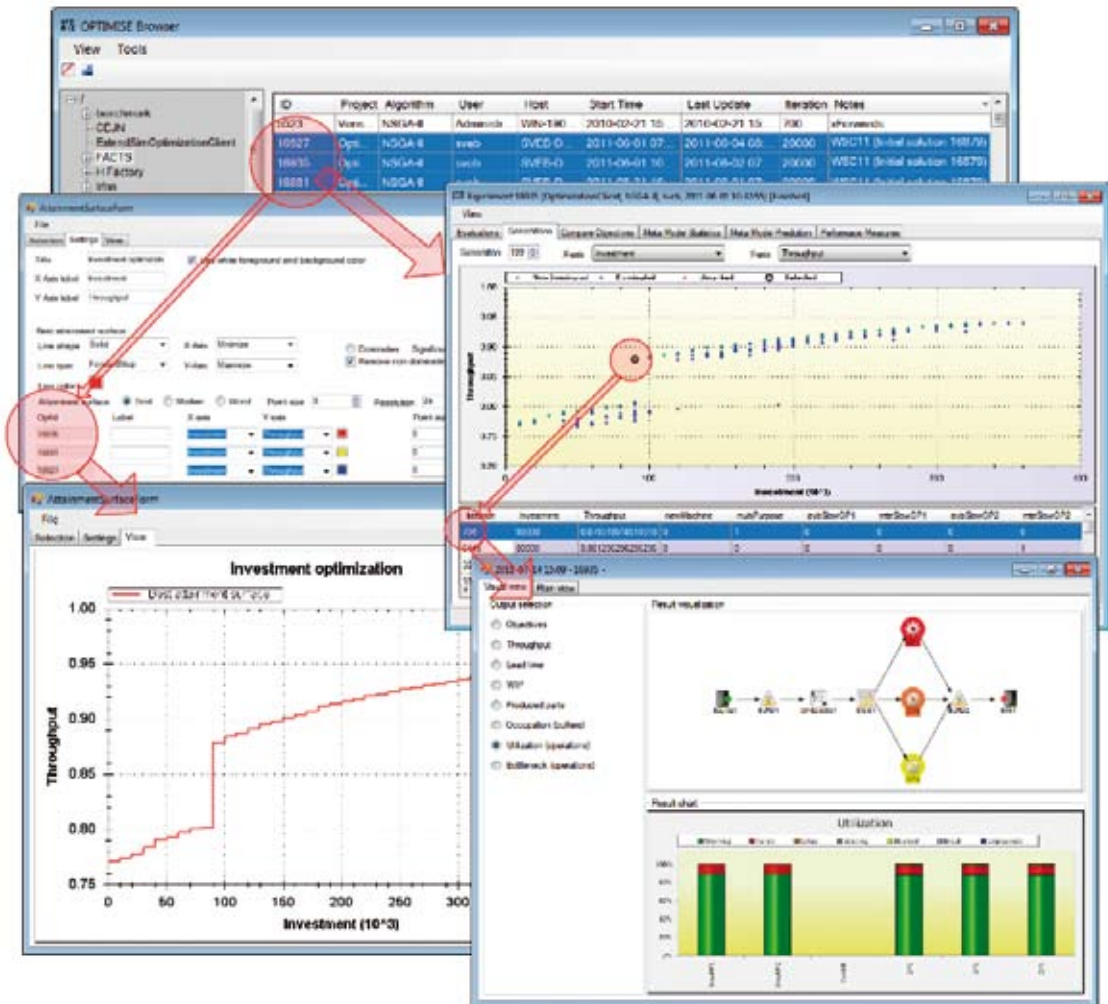


Figura 4. Imagen que muestra el uso de las herramientas de FACTS para navegar y analizar los resultados de los MOO.

El efecto de invertir en una máquina nueva o más rápida en la etapa 1 se puede comparar con el efecto de la actualización de las tres máquinas en la etapa 2 para ser flexible. Esto se puede realizar utilizando el gráfico en 4D (3D más color) y las coordenadas paralelas, de FACTS Analyzer (Figura 5). El gráfico de 4D en la figura N° 5 ilustra muy claramente que la no

linealidad en el eje de rendimiento se atribuye a la variable binaria que decide si las máquinas en la fase 2 son flexibles o no. El gráfico muestra que se puede hacer una mejora mayor del rendimiento haciendo cambios pequeños, que reemplazando la máquina vieja o lenta por una máquina nueva o más rápida.

Como se puede comprobar, la visualización proporciona información valiosa en la toma de decisiones para la mejora de la línea de producción.

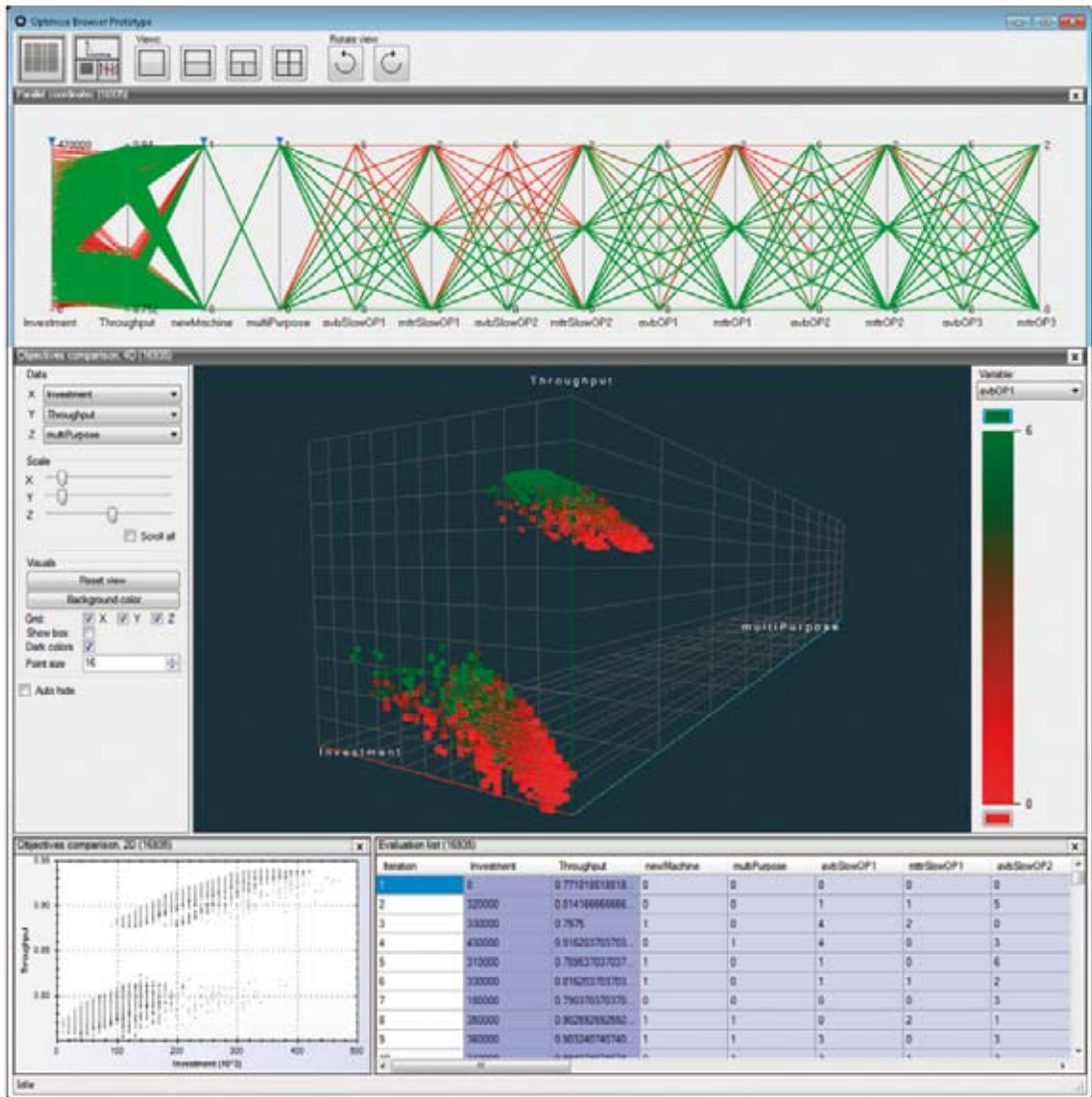


Figura 5. Visualización del efecto de las diferentes variables de decisión en las coordenadas paralelas y el gráfico 4D.

Para este problema tan simple se puede apreciar que la visualización de datos mediante gráficos 4D y el computador/ordenador proporcionan un medio eficaz al usuario para comparar los atributos de las soluciones Pareto-óptimas. Pero, para problemas complejos que están compuestos por decenas o, incluso, cientos de variables de decisión con sus relaciones complejas, encontrar los atributos de los vectores de decisión que constituyen las soluciones

óptimas de Pareto, es un trabajo muy complejo. El implementar métodos eficientes para este tipo de análisis post-optimización, es una investigación muy importante e interesante que se está llevando a cabo en el desarrollo de la próxima generación de FACTS Analyzer.

RESULTADOS Y CONCLUSIONES

Hasta la fecha se puede afirmar que existen muy pocas aplicaciones y herramientas eficientes para ejecutar simulaciones y optimizaciones en Internet de sistemas reales. Este artículo ha presentado un sistema de DES y optimización mediante simulación basados en Internet, llamado FACTS Analyzer, que ha sido desarrollado específicamente para el diseño conceptual del flujo productivo, su análisis y optimización. Utilizando FACTS Analyzer, experimentos de simulaciones paralelas y distribuidas pueden ser ejecutadas y controladas remotamente por múltiples usuarios, en cualquier momento y desde cualquier lugar, mediante el uso de Internet. FACTS Analyzer es también compatible con MOO, por lo que los óptimos de Pareto pueden ser generados de forma eficiente para que los encargados de tomar decisiones puedan escoger aquella configuración que ofrezca el mejor trade-off entre los diferentes objetivos a la hora de diseñar y mejorar un sistema productivo. Actualmente, el propósito principal de la investigación es extender FACTS con técnicas de análisis post-optimización para descubrir la importancia de los atributos de los óptimos de Pareto para apoyar a la toma de decisiones en el diseño de sistemas productivos. Finalmente, cabe destacar que FACTS mejora el día a día de los ingenieros de producción en su trabajo de búsqueda de sistemas óptimos. Y aunque es muy pronto para decirlo, nuestros colaboradores de la industria están entusiasmados por el potencial que proporciona esta herramienta innovadora.

AGRADECIMIENTOS

Este trabajo se inició con el proyecto FACTS, el cual se llevó a cabo entre los años 2006-2008. Su continuación se basa en el proyecto FFI-HSO (2009-2012), ambos financiados por la agencia sueca VINNOVA. Los autores agradecen su apoyo con fondos de investigación durante todos estos años. Los autores agradecen también a los socios de la industria automotriz sueca, incluyendo Scania AB, Volvo Car Corporation y Volvo AB, por su apoyo en el desarrollo, prueba y evaluación de FACTS Analyzer.

REFERENCIAS

April, J; Better, M; Glover, F; Kelly, J. New advances for marrying simulation and optimization. En Ingalls, R. G., Rossetti, M. D., Smith, J. S.; Peters, B. A. (eds.) *Proceedings of the 2004 Winter Simulation Conference*, (Washington, D.C., Dec.5–8, 2004). P.80–86.

Benjamin, D. *Conceptual factory simulation analysis: prerequisites and specification*. MSc thesis. Skövde, Sweden: University of Skövde. 2010

Benedixen, J; Ng, A.H.C. Optimal Buffer Allocation for Semi-synchronized Automotive Assembly Lines using Simulation-based Multi-objective Optimization. En *Proceedings of the Industrial Simulation Conference*, (Venice, Italy, 6-8 June, 2011). P. 129-135.

Boesel, J., Jr; Bowden, R.O; Kelly, J.P; Westwig, F. Future of simulation optimization. En A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer (eds.). *Proceedings of the 2001 Winter Simulation Conference*, (2001).P.1466–1469.

- Byrne, J; Heavey, C.; Byrne. P.J. A review of Web-based simulation and supporting tools. *Simulation Modelling Practice and Theory*. 2010, vol 18 num 3.p. 253-276.
- Deb, K; Pratap, A; Agarwal, S; Meyarivan, T. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transaction on Evolutionary Computation*. 2002, vol 6, num 2, p. 181-197.
- Fu, M.C., Andradóttir, S., Carson, J. S., Glover, F., Harell, C. R., Ho, Y.-C., Kelly, J.P. & Robinson, S. M. Integrating optimization and simulation: research and practice. En J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick (eds.), *Proceedings of the 2000 Winter Simulation Conference*, (2000). P.610–616.
- Knowles, J. A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers. En *Proceedings of the 5th International Conference on Intelligent Systems Design and Applications (ISDA)*, (Wroclaw, Poland, 8-10 September 2005)
- Madan, M; Son, Y.J; Cho, H; Kulvatunyou, B. Determination of efficient simulation model fidelity for flexible manufacturing systems. *International Journal of Computer Integrated Manufacturing*. 2005, vol 18, num 2–3, p. 236–250.
- McNally, P; Heavey C. Developing simulation as a desktop resource. *International Journal of Computer Integrated Manufacturing*. 2004, 17, p. 435-450.
- Miller, J.A; Fishwick, P.A; Taylor, S.J.E. Benjamin, P; Szymanski, B. Research and commercial opportunities in Web-Based Simulation. *Simulation Practice and Theory*. 2001, vol 9, num 1-2, p. 55-72.
- Ng, A; Urenda, M; Svensson, J; Skoogh, A; Johansson, B. FACTS Analyser: An innovative tool for factory conceptual design using simulation, En *Proceedings of Swedish Production Symposium*, (Gothenburg, 28-30 August 2007).
- Ng, A; Urenda, M; Jägstam, M; Svensson, J. An internet enabled tool for multi-objective simulation optimization, En *Proceedings of the VIII Congreso Chileno de Investigacion Operativa, OPTIMA* (Chillan, Chile, 2009).
- Pehrsson, L; Ng, A.H.C; Bernedixen, J. Multi-objective production system optimisation including investment and running costs. En *Proceedings of the 4th Swedish Production Symposium*, (Lund 3-5 May 2011). P.55-62.
- Pehrsson, L; Ng, A.H.C. An applied framework for simulation-based multi-objective optimisation within production system development. En *Proceedings of the Industrial Simulation Conference*, (Venice, Italy. 8 June 2011). P.121-128.
- Urenda Moris, M; Ng, A; Svensson J. Simplification and aggregation strategies applied for factory analysis in conceptual phase using simulation. En: S. J. Mason, R. Hill, L. Moench, and O. Rose,(eds), *Proceedings of the 2008 Winter Simulation Conference*, (2008).P.1913-1921.
- Zomaya, R. Factory flow analysis: A case study for Volvo Cars Engines. BSc Thesis, Skövde, Sweden: University of Skövde. (2010)