



CAPITULO 3: SIMULACIÓN

3.1. Introducción

El estudio que se va a realizar, se basa en la comprobación del funcionamiento de la línea de producción con distintos parámetros de control. Es evidente, la imposibilidad de la experimentación directa. Así como, la experimentación a tiempo real también es inviable, pues la comprobación de cada configuración de control supondría periodos muy largos de funcionamiento. Estas circunstancias se solventan con el uso de un modelo.

El interés del modelado en este proyecto está en la experimentación de situaciones concretas y esto se consigue haciendo uso de un modelo basado en simulación.

Por otra parte, debido al carácter estocástico de los entornos productivos, un estudio analítico quedaría limitado a casos extremadamente sencillos y poco realistas. Por ello la herramienta más adecuada para el estudio de estos sistemas es la simulación.

El uso de simulación permite experimentación controlada, compresión de tiempo (una simulación se realiza en mucho menos tiempo que el sistema real que modela), y análisis de sensibilidad. Otra gran ventaja es que su uso no afecta al sistema real, que puede seguir utilizándose (o no existir). Finalmente, la simulación es una herramienta efectiva de entrenamiento. Algunos problemas que existen en el uso de simulación son su tiempo de desarrollo, en que los resultados pueden tener divergencia con la realidad (precisan validación), y en que para reproducir el comportamiento del sistema simulado se precisa colección extensiva de datos.

La simulación de una línea de producción como la que se va a estudiar, se hace partiendo de la identificación del sistema como un sistema dinámico de eventos discretos. La simulación se realizará con el programa Optquest, que es un módulo de Arena v.12.

En el presente capítulo, se exponen los aspectos más importantes de la herramienta empleada para la realización de los experimentos del proyecto, la simulación.

En primer lugar se analizan los aspectos más importantes de la simulación. Particularizando en la simulación de eventos discretos mediante el uso del ordenador.

Seguidamente, se presenta la herramienta Arena, con la que se modela el sistema.

Posteriormente, se presenta la herramienta Optquest, con la que se han realizado las simulaciones para la obtención de los resultados de los experimentos.



3.2. Aspectos generales de la simulación

Se define simulación como la reproducción del comportamiento dinámico de un sistema real en base a un sistema con el fin de llegar a conclusiones aplicables al mundo real [41].

Actualmente se emplea la simulación como herramienta en el proceso de toma de decisiones. Como medio para comprender la realidad y asumir su complejidad, sin necesidad de interactuar con el sistema real. Con el desarrollo de la tecnología de los ordenadores, la simulación, se ha convertido en una herramienta ampliamente extendida y eficaz, para prever el comportamiento de sistemas reales, para determinadas condiciones de operación. Por tanto, aparece un nuevo concepto, la simulación digital.

La simulación digital es una técnica que permite imitar (o simular) en un ordenador el comportamiento de un sistema real o hipotético según ciertas condiciones particulares de operación [42].

Para realizar una simulación, es necesario desarrollar previamente el modelo conceptual que describe el comportamiento del sistema. Para continuar con el estudio, se hace necesario aclarar los conceptos de sistema y modelo.

Según el DRAE [43] un sistema es un conjunto de cosas que ordenadamente relacionadas entre sí contribuyen a un determinado objeto. Guash et al. [42], definen un sistema como una colección de objetos o entidades que interactúan entre sí para alcanzar un cierto objetivo.

El estado de un sistema se puede definir por un conjunto mínimo de variables necesarias para caracterizar o describir todos aquellos aspectos de interés del sistema en un cierto instante de tiempo. A estas variables se las denominan variables de estado [42].

Según Guash et al., los sistemas se pueden clasificar según su comportamiento a lo largo del tiempo en:

- **Sistemas Continuos:** Son aquellos en las que las variables de estado evolucionan de un modo continuo a lo largo del tiempo.
- **Sistemas Discretos:** Son aquellos en los que las propiedades de interés del sistema cambian en un cierto instante o secuencia de instantes, que normalmente obedecen a un patrón periódico.
- **Sistemas Orientados a Eventos Discretos:** Al igual que los sistemas discretos, se caracterizan en que las propiedades de interés del sistema cambian únicamente en una secuencia de instantes de tiempo, permaneciendo constantes el resto del tiempo. La secuencia de instantes en los cuales el estado del sistema puede presentar un cambio obedece a un patrón aleatorio.
- **Sistemas Combinados:** Son aquellos que combinan subsistemas continuos o discretos respectivamente.

No es posible estudiar los sistemas tal y como se presentan en la realidad, ya que el número de variables que les afectan es incontable. Siempre existe la posibilidad de que aparezcan factores imprevistos que pueden alterar su funcionamiento. Es por ello, por lo



que se hacen simplificaciones de la realidad, de modo que un conjunto de ecuaciones pueda representar el funcionamiento de un sistema real. Esto es lo que se denomina modelo de un sistema.

Como modelo del sistema se entiende a la descripción de las características internas y mecanismos de interés del sistema. Al proceso de abstracción para obtener esta descripción se conoce como modelado. Un modelo debe representar aquellas características del sistema que son de nuestro interés y ser una representación abstracta de la realidad lo suficientemente sencilla como para facilitar su mantenimiento, adaptación y reutilización [42].

Aunque existe una gran diversidad de metodologías para el desarrollo de modelos matemáticos de sistemas físicos, existen un conjunto de consideraciones que se deben tener en cuenta para garantizar una representación eficiente del sistema real [42]:

- Un modelo se desarrolla siempre a partir de una serie de aproximaciones e hipótesis y, consecuentemente, representa tan solo parcialmente la realidad.
- Un modelo se construye para una finalidad específica y debe ser formulado para que sea útil a dicho fin.
- Un modelo tiene que ser por necesidad un compromiso entre la simplicidad y la necesidad de recoger todos los aspectos esenciales del sistema en estudio. Así pues, un buen modelo debe representar adecuadamente aquellas características del sistema que son de nuestro interés y ser una representación abstracta de la realidad lo suficientemente sencilla como para facilitar su mantenimiento, adaptación y reutilización [42].

Atendiendo a las características que debe poseer un buen modelo, así como a los objetivos del estudio de simulación, los modelos de simulación suelen clasificarse de la siguiente manera [42]:

- Modelos Estáticos frente a Modelos Dinámicos.

Los Modelos Estáticos suelen utilizarse para representar el sistema en un cierto instante de tiempo y en su formulación no se considera el avance del tiempo. Este tipo de modelos es muy útil cuando el sistema se encuentra en equilibrio, no evoluciona respecto al tiempo. Si se cambia el punto de equilibrio alterando uno o más de los valores del sistema, el modelo permite deducir el resto de los valores, pero no muestra la manera en que cambiaron.

Los Modelos Dinámicos, en contraposición a los anteriores, permiten deducir como las variables de interés del sistema evolucionan respecto al tiempo.

- Modelos Deterministas frente a Modelos Estocásticos.

Un modelo se denomina Determinista si su nuevo estado puede ser completamente definido a partir del estado previo y de sus entradas. Es decir, ofrece un único conjunto de valores de salida para un conjunto de entradas conocidas.

Los modelos Estocásticos requieren de una o más variables aleatorias para formalizar las dinámicas de interés. En consecuencia, el modelo no genera un único conjunto de salida cuando es utilizado para realizar un experimento, sino que los resultados generados son utilizados para estimar el comportamiento real del sistema.



- Modelos Continuos frente a Modelos Discretos.

Los Modelos Continuos se caracterizan por representar la evolución de las variables de interés de forma continua.

Los Modelos Discretos se caracterizan por representar la evolución de las variables de interés de forma discreta.

Es posible describir un sistema continuo mediante un modelo discreto y viceversa. La decisión de utilizar un modelo continuo o un modelo discreto, no depende tanto de las características del sistema sino de los objetivos particulares de cada estudio.

Los modelos de eventos discretos son modelos dinámicos, estocásticos y discretos en los que las variables de estado cambian de valor en instantes no periódicos de tiempo. Estos instantes de tiempo se corresponden con la ocurrencia de un evento. Un evento es una acción instantánea que puede cambiar el estado del modelo.

Para un modelo de simulación de eventos discretos, los elementos más significativos son [42]:

- Variables de estado: son el conjunto mínimo de variables que describen todos los aspectos de interés del sistema. Cada conjunto de valores que toman estas variables definen cada estado del sistema.
- Eventos: Son acciones instantáneas que pueden cambiar el estado del modelo.

No consumen tiempo y se pueden clasificar en: eventos condicionados y no condicionados.

Los eventos condicionados, son aquellos eventos que para que ocurran es necesario que se den ciertas condiciones.

Los eventos no condicionados, son aquellos que están planificados de antemano y no necesitan de condición alguna para que ocurra.

Los eventos, también se pueden clasificar como: eventos endógenos, aquellos causados por condiciones en el modelo y eventos exógenos o externos, eventos externos al modelo.

- Actividades: Son las tareas o acciones que tienen lugar entre dos eventos.

Generalmente, tienen duración temporal conocida, aunque esto no significa que sea constante. En procesos estocásticos su duración, y por lo tanto su instante de finalización se determina a partir de una distribución de probabilidad.

- Entidades: Son los objetos que constituyen o se mueven a lo largo del sistema.

Se pueden agrupar en dos grupos: entidades permanentes y entidades temporales.

Los recursos o entidades permanentes, son los medios gracias a los cuales se pueden ejecutar las actividades. Definen quién o qué ejecuta la actividad, su número permanece constante a lo largo de la simulación y suelen parametrizarse por características tales como capacidad, velocidad o tiempo de ciclo.

Las entidades temporales, son los objetos que llegan, se procesan y salen del sistema. Se crean y se destruyen a lo largo de la simulación.



- Atributos: Son las características que caracterizan a las entidades según propiedades de estas como puede ser el tamaño, precio, prioridad, etc. Los atributos son imprescindibles para controlar el flujo de entidades en el sistema.
- Colas: Son estructuras que quedan determinadas a partir de una colección de entidades temporales, ordenadas de una forma lógica. Las entidades que están en una cola sufren retardos de una duración indeterminada.

En el campo de la producción, la simulación se utiliza para determinar los niveles de inventario, los procedimientos de mantenimiento, los programas de producción, la planificación de los procesos... [44].

El modelado del sistema, debe representar las dinámicas de interés del sistema bajo las condiciones de un estudio particular. La simulación consiste en realizar experimentos sobre el modelo del sistema. De esta forma, los resultados generados por el simulador, permiten prever con cierta exactitud los resultados que se obtendrían si se realizara el experimento sobre el sistema real.

Para experimentar con el modelado de simulación de un sistema orientado a eventos discretos, existen una serie de alternativas [42]:

- Simulación del modelo estático.
- Simulación a mano.
- Simulación digital mediante un lenguaje de propósito general.
- Experimentación mediante un entorno de simulación.

La técnica de la simulación ofrece como principales ventajas [44]:

- Desarrollar el modelo de un sistema conduce con frecuencia a una mejor comprensión del sistema real.
- El espacio temporal se puede comprimir, reduciendo años de experimentación en la realidad a minutos de computación.
- La simulación no interrumpe las actividades en curso del sistema real.
- La simulación es mucho más general que los modelos matemáticos y puede utilizarse cuando las condiciones no son apropiadas para un análisis matemático típico.
- La simulación puede utilizarse como un juego para la experiencia de capacitación.
- La simulación provee una duplicación más realista de un sistema que el análisis matemático.
- La simulación puede utilizarse para el estudio de situaciones transitorias mientras que las técnicas matemáticas no pueden hacerlo.



- Comercialmente están distribuidos muchos modelos de paquetes estándar, que cubren una amplia gama de temas.
- La simulación permite responder a las suposiciones de lo que puede llegar a ocurrir en determinadas circunstancias.

Y como principales inconvenientes:

- El tiempo y el esfuerzo requeridos para el desarrollo de un modelo de simulación, no supone una garantía de que del modelo, se obtengan buenos resultados.
- No existe modo alguno de probar que el modelo de simulación es totalmente confiable. Un sistema aparentemente estable, con una determinada combinación de eventos, puede volverse inestable. La simulación implica numerosas repeticiones de secuencias que se basan en ocurrencias generadas de manera aleatoria, que en algún momento generen una situación en la que el sistema se vuelva inestable.
- La construcción del modelo de simulación, supone un elevado número de horas de trabajo. Con lo que puede resultar muy costoso.
- La simulación puede ser menos exacta que el análisis matemático, pues se basa en ocurrencias aleatorias.
- El tiempo de computación de modelos complejos, puede llegar a ser muy elevado.
- La técnica de simulación no tiene un enfoque estandarizado. El modelo de simulación diseñado por un analista, puede diferir considerablemente, respecto al de otro.

3.3. Simulación de eventos discretos.

Para la simulación de la línea objeto del presente proyecto se parte de la identificación de ésta como un sistema dinámico de eventos discretos.

Un paradigma de simulación para sistemas dinámicos de eventos discretos asume que el sistema simulado sólo cambia de estado en puntos discretos del periodo simulado. Es decir, que el modelo cambia de estado ante la ocurrencia de un evento.

Por tanto, el modelo de simulación posee un estado S definido en algún instante en el tiempo. Un estado del sistema, es un conjunto de datos tomados de las variables salientes del sistema y que permite describir la evolución del sistema en un instante de tiempo. En un programa de simulación por ordenador, el estado se almacena en una o más variables que representan varias estructuras de datos. Así el estado, dependiendo de las necesidades particulares del modelo y el nivel de detalle incorporado en el modelo, se puede definir de varias formas. Por ejemplo, si se considera una máquina, alimentada por un almacén de materia prima de trabajos. Un estado inicial del sistema es el número de trabajos en el almacén, pero esta manera de definir el estado, no permite calcular los tiempos de espera, porque no se mantiene la identidad de cada trabajo individual. Para



poder calcular los tiempos de entrega, se podrían considerar los distintos clientes en una cola con datos asociados a cada uno de ellos, como los tiempos de llegada de los clientes.

La función que define un estado durante un tiempo $S(t)$, se define como una función discontinua, cuyos saltos se producen por la ocurrencia de eventos discretos, que inducen a estados de transición o cambio del estado del sistema.

La implementación de los eventos es similar en todos los simuladores. Un evento es una estructura de datos que siempre contiene un campo que hace referencia al tiempo de ocurrencia y una serie de campos más. Además, la ocurrencia de un evento en una simulación de este tipo, se implementa como la ejecución de un procedimiento (código de computación) en el tiempo de ocurrencia del evento programado. Cuando el procedimiento ha pasado, se dice que ya ha sido procesado o ejecutado.

La evolución de estos modelos se rige por un reloj y una lista de eventos ordenados cronológicamente. Los eventos están relacionados en la lista de eventos, de acuerdo a su orden de ocurrencia. El evento que se encuentra al principio de la lista se denomina evento inminente. La programación de un evento supone que el evento está vinculado a la lista de eventos. Si un evento ocurre significa que el evento se desliga de la lista y se ejecuta. La ejecución de un evento puede cambiar el estado de las variables y posiblemente programar otros eventos en la lista.

Una característica fundamental del paradigma de los eventos discretos, es que nada cambia, a menos que ocurra un evento, en cuyo instante ocurre la transición a otro estado. Es decir, ante la ejecución de un evento es posible que cambie el estado, pero todo cambio de estado se produce por un evento. Entre los eventos el estado del sistema se considera constante.

En general, un simulador de eventos discretos ejecuta el siguiente algoritmo [45]:

1. Fijar el reloj de simulación en un tiempo inicial, normalmente cero y generar uno o más eventos iniciales y programarlos.
2. Si la lista de eventos está vacía, termina la ejecución de la simulación. De lo contrario, encuentra el evento inminente y lo desliga de la lista de eventos.
3. Adelanta el reloj de simulación al tiempo del evento inminente y lo ejecuta. (El evento puede parar la simulación).
4. Vuelta al paso segundo.

Esta descripción es extremadamente simple. La complejidad se oculta en las rutinas que implementan la ejecución de eventos y en las estructuras de datos que utilizan.

Este tipo de simulación puede representar tanto sistemas grandes como complejos.

Para ampliar el conocimiento de estos sistemas, se pueden consultar varios autores, Fishman [46], Banks [47] y Law y Kelton [48].

En el presente proyecto, se han realizado la simulación a partir de un software de simulación que implementa el paradigma de simulación de eventos discretos.



Existen multitud de herramientas de simulación de este tipo, que ofrecen módulos para programar de forma rápida y fácil un modelo, reduciendo el tiempo de programación del mismo. Las características más destacadas de estas herramientas, que propician esta reducción son: la generación de números aleatorios que corresponden a ciertas funciones de distribución, la gestión automatizada del tiempo de simulación, la gestión automatizada de las rutinas de tratamiento de los eventos que determinan el comportamiento del sistema, los algoritmos de análisis de los resultados y la posibilidad de generación de informes. Además el mantenimiento del modelo de simulación se simplifica.

Existen numerosas herramientas de simulación comerciales orientadas a eventos discretos. Normalmente, los mecanismos internos son similares. Las principales diferencias entre ellas, residen en las funciones y metodologías de modelado y en las interfaces de usuario.

No existe un simulador que sea el mejor en el campo de simulación de modelos de eventos discretos. Si no que habrá que evaluarlos, dentro de los existentes, en función de una serie de criterios. Para seleccionar la herramienta se puede tomar como referencia la ISO/IEC 9126, que es un estándar internacional para la evaluación del software. Ésta norma considera aspectos como la funcionalidad, fiabilidad, usabilidad, eficiencia, portabilidad, mantenibilidad y portabilidad. Otros aspectos a considerar son la calidad de la empresa que desarrolla la herramienta o los costes [42].

3.4. Simulación en Arena v.12 CPR9

Arena es un potente software de modelado y simulación de diferentes áreas de negocio. Se ha diseñado para analizar el impacto de los cambios que suponen los complejos y significativos rediseños asociados a la cadena de suministros, procesos, logística, distribución y almacenaje y sistemas de servicio. Tiene gran flexibilidad y cubre gran cantidad de aplicaciones a modelar con cualquier nivel de detalle o complejidad.

Un escenario típico incluye [49]:

- Análisis detallado del tipo de sistema de manufactura, incluyendo el transporte manual de componentes.
- Análisis de servicio al cliente y sistemas de dirección orientados al cliente.
- Análisis de cadenas de suministro globales que incluyen almacenamiento, transporte y sistemas logísticos.
- Predicción del funcionamiento de sistemas en función de medidas clave como costes, tasa de salida de piezas, tiempos de ciclo y utilización.
- Identificación de los procesos cuello de botella como colas construidas con sobreutilización de recursos.
- Planificación del personal, equipos y requerimientos de material.

Arena se ha diseñado en base al lenguaje de programación SIMAN. SIMAN modela un sistema discreto utilizando la orientación al proceso; es decir, en un modelo de sistema particular, se estudian las entidades que se mueven a través del sistema. Una entidad para SIMAN es un cliente o un objeto que se mueve en la simulación y que posee características únicas conocidas como atributos. Los procesos



denotan la secuencia de operaciones o actividades a través de los que se mueven las entidades, siendo modeladas por el diagrama de bloques.

3.4.1. Guía resumen de Arena v.12

- Entorno

Arena es un entorno gráfico que asiste en la implementación de modelos en el paradigma orientado al proceso. Contiene un conjunto de bloques que representan gráficamente los diversos tipos de procesos agrupados en categorías en función de su complejidad. Los bloques se enlazan para representar el flujo de entidades entre ellos.

Asiste para representar gráficamente la animación del proceso, pudiendo fijar el aspecto gráfico de las entidades que fluyen. También permite fácilmente crear cuadros en los que se muestren los valores de variables del modelo según transcurre la simulación. Recopila datos de estadísticas de la simulación; los cuales se pueden visualizar mediante informes al final de la misma. Permite incluir módulos de usuario elaborados en otros lenguajes de programación, tales como FORTRAN, C o Visual Basic. Con este último tiene una fácil integración, ya que contiene una herramienta que permite editar, compilar y enlazar código realizado en este lenguaje.

Tiene incorporadas librerías Visual Basic que permite presentar o tomar datos de hojas de cálculo Microsoft Excel.

En la parte superior de la ventana principal de Arena, tiene una barra título con el nombre del modelo en la parte superior. Bajo la barra título, está la barra Menú, bajo la barra menú la barra Standard. Y bajo ésta todas las otras barras que el usuario ha seleccionado que desea visualizar.

La ventana del modelo, presenta básicamente tres zonas:

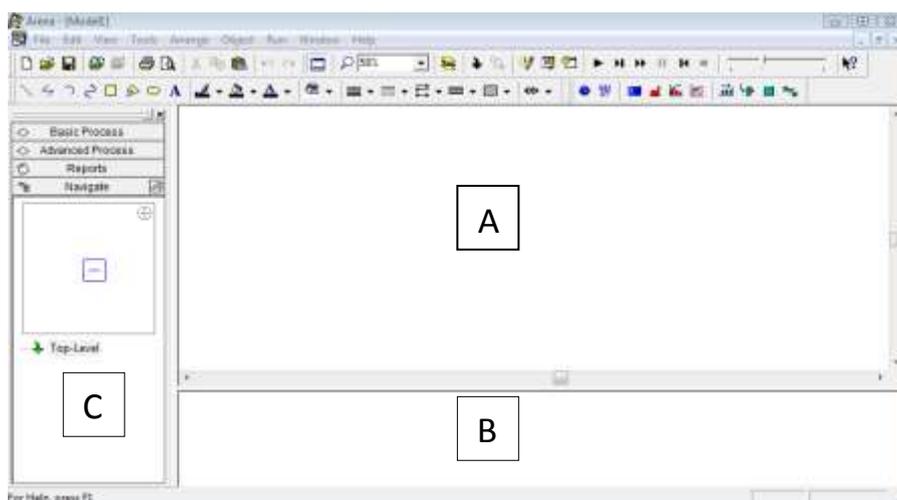


Figura 3. Ventana principal modelado Arena



A. Vista de diagrama de flujo. Contiene el diagrama, la animación y los elementos gráficos de la simulación.

B. Vista de hoja de cálculo. Contiene la información del modelo. A través de esta hoja, se puede acceder a los parámetros y elementos del modelo.

C. Barra de proyectos. Aquí se encuentran los paneles que sirven para diseñar los modelos. Los siguientes paneles, aparecen por defecto y contienen los elementos necesarios para configurar los modelos. El conjunto total de paneles de modelado para el diseño que Arena ofrece es variado, en la versión que se ha empleado en el presente proyecto, Arenav12.0 (CPR9), es de 13: “Basic Process”, “Advanced Process”, “Blocks”, “Advanced Transfer”, “Contact Data”, “CSUtil”, “Elements”, “Flow process”, “Flow Process Util”, “Packaging”, “Factory Blocks”, “AgentUtil”, “Script” y “UtlArena”.

A continuación se describen brevemente los paneles más usuales.

El panel Basic Process, contiene (módulos) elementos básicos de construcción de los modelos.

El panel Advanced Process provee de módulos básicos adicionales y otros avanzados, para la construcción del modelo.

El panel Blocks, contiene el grupo completo de bloques SIMAN.

El panel de Advanced Transfer consiste en una serie de módulos que soportan la transferencia de entidades en el modelo. Éstos pueden ser de transferencia ordinaria o empleando equipo de transporte manual de material.

El panel Elements contiene los elementos necesarios para declarar los recursos del modelo, colas, variables, atributos y recolección de estadísticas.

En algunas versiones, pueden aparecer otros paneles como son el Common, con módulos como Arrive, Server, Depart, Inspect, y módulos de elementos Stats, Variables, Expressions, and Simulate; y el Support que contiene un grupo de bloques SIMAN empleados con frecuencia.

Por otra parte, se encuentran el panel Navigate que se encuentra también en la barra Project, incluye las vistas del modelo. Pudiendo visualizar diferentes submodelos de forma jerárquica. Y el panel Reports, que es una lista de resultados de la simulación tras la ejecución, en relación a varios componentes del modelo.

Para incluir paneles seleccionar en el menú de Arena File>Template Panel>Attach. Entonces habrá que seleccionar el panel deseado en la capeta Template que se encuentra dentro de la carpeta de instalación del programa Rockwell Software>Arena. Estos archivos tienen extensión (.tpo).



Los más generales y que se pueden usar sin restricciones en la versión completa son los primeros. El panel que ofrece por defecto es “Basic Process”, que contiene bloques para un amplio rango de elementos frecuentes en modelado. “Blocks” son bloques de construcción más elementales que los del panel “Basic Process”. La funcionalidad que ofrecen los elementos del “Basic Process” requiere, si se quiere modelar con elementos de “Blocks”, del ensamblaje de varios elementos. Los elementos de “Blocks” son de utilidad cuando no hay elementos simples de “Advanced Process” que modelen el comportamiento deseado y sin embargo ensamblando elementos más simples de “Blocks”, se puede conseguir.

En la parte superior de la ventana del programa de la figura anterior se visualizan las barras de herramientas, las cuales se pueden seleccionar, en el menú superior View>Toolbars.

La duración de la simulación se puede fijar de forma temporal o bien indicando una condición. Se puede lanzar la simulación con animación o en modo Batch, en que la animación se suspende y la animación es mucho más rápida.

Además cuenta con una aplicación que puede realizar optimizaciones automáticas, Optquest, la cual se va a emplear en el presente proyecto y cuyo desarrollo se expondrá en el siguiente apartado.

Cuando se lanzan las simulaciones, Arena recopila de forma automática estadísticas sobre los distintos elementos del modelo, entre las que podemos destacar las de recursos y colas. De los recursos obtiene de forma automática, el porcentaje de utilización, el número total de veces que se han usado durante la simulación y el número de unidades puestas en juego. De las colas obtiene los tiempos mínimos, medios y máximos de espera; el número mínimo, máximo y promedio de entidades esperando. También admite que el usuario pueda definir estudios estadísticos.

- Nociones básicas

Los elementos básicos, con los que se construyen los modelos mediante el software Arena, son los módulos. El diagrama construido a partir de los módulos, ha de describir la dinámica de los procesos que tienen lugar en el sistema. Los módulos se conceptualizan como nodos de una red, a través de la cual fluyen entidades (clientes, productos...).

Las entidades son aquellas personas, objetos, piezas... reales o imaginarios, cuyos movimientos a través del sistema causan un cambio en su estado. Por tanto se consideran entidades, a aquellos documentos, clientes, piezas, que son producidos o atendidos o que de cualquier otra manera influyen en el proceso. Las entidades, se pueden entender, como objetos dinámicos que usualmente son creados, circulan por un tiempo a través del sistema y posteriormente se retiran del mismo o siempre se mantienen circulando en él. Las entidades serán creadas manualmente, por el modelador o automáticamente mediante el programa Arena.



Las entidades, contienen propiedades, llamadas Atributos, que permiten crear diferencias entre ellas. Ejemplos de estas diferencias son el tiempo de llegada, la prioridad, la fecha de vencimiento...

Los atributos que Arena asigna automáticamente a una entidad son: Entity Type (nombre del tipo de entidad), Entity Picture (dibujo que se mostrará durante la simulación) y Entity Create Time (tiempo en la simulación en que se crea la entidad).

Para modificar los atributos, es necesario acceder al panel de procesos básicos y hacer clic sobre el módulo de "Entity". Entonces, aparece en la vista de hoja de cálculo la información de los atributos.

Las entidades compiten por los servicios que les brindan los recursos. Los recursos pueden ser personal, equipo, espacio en un área de almacenamiento...

Una entidad puede ocupar un recurso (seize), en la medida que se encuentra disponible y lo libera (selease) cuando ya no lo necesita.

Siempre que una entidad ocupa un recurso, lo ha de liberar en algún momento. Mientras que éste no esté libre, cualquier otra entidad que necesite utilizarlo tendrá que esperar en una cola (queue).

Una fila es un área donde las entidades esperan mientras se libera el recurso que necesitan. Las filas tienen nombres y pueden tener capacidades.

Las variables son un conjunto de valores globales, que pueden alterarse en cualquier lugar dentro del modelo. Las variables puede definir las el usuario y pueden cambiarse durante el periodo de simulación. Ejemplo de estas variables pueden ser la tasa de llegadas, el inventario actual, el número de pacientes... Por otra parte, las variables pueden ser variables predefinidas por el comportamiento del modelo, que indican el estado de los componentes. Ejemplos: N° entidades esperando en una fila, valor actual de un contador...

Los módulos de datos, no tienen la representación en la vista de diagrama de flujo, aparecen en la vista de la hoja de cálculo. Sirven para definir las características de varios procesos y elementos del modelo [49].

- Menús

Cuando el software Arena arranca, están disponibles los menús File, View, Tools y Help. Una vez se abre un modelo, aparecen activos los menús Edit, Arrange, Object, Run y Window.

Algunas de las funciones que se encuentran en estos menús son similares a las estándar de Windows, por lo cual, solo se comentarán los que son específicos de Arena.

A partir del menú File, se crean nuevos ficheros de modelos Arena, se abren otros ya existentes, se guardan en disco o se crea un modelo de solo lectura. Además, se pueden importar dibujos CAD con formato DXF para emplearlos como fondo y, en algunos casos, como elementos activos. Como por ejemplo, caminos para vehículos dirigidos por cable.

Con el menú Edit, se pueden deshacer, Undo y rehacer, Redo, acciones previas. Existe la posibilidad de buscar los módulos y objetos de animación a partir de una cadena de texto, mediante la opción Find. Se pueden mostrar las propiedades de los objetos,



mediante Properties. Muestra los enlaces en el modelo a otros ficheros, como hojas de cálculo o ficheros de sonido y permite modificarlos.

La función Insert New Object, permite introducir objetos de otras aplicaciones, como gráficos u objetos multimedia y Object permite editar lo que se ha importado.

Desde el menú View, se controla cómo aparece el modelo en pantalla y las barras de herramientas que se muestran. Este menú permite hacer Zoom sobre el modelo. En Views se puede elegir entre varias vistas del modelo. Named Views permite definir, cambiar y utilizar vistas propias. Layers permite controlar el tipo de objetos que se mostrarán durante el modo de edición o de ejecución.

En el menú Tools, el Input Analyzer, encaja distribuciones de probabilidad sobre los datos observados para especificar las entradas al modelo. No Adds-In, lista los programas ejecutables que se han instalado en la carpeta Adds-in. Show Visual Basic Editor, abre una ventana en la que se puede escribir código de Visual Basic para completar el modelo. El elemento Options permite cambiar y personalizar muchas de las formas en que trabaja Arena y de su aspecto según las necesidades.

Por otra parte, las opciones del menú Arrange corresponden a las posiciones de los módulos del modelo y de los gráficos.

El menú Object, tiene la función Auto-Connect, que permite conectar automáticamente un nuevo objeto introducido con el que se encontrase seleccionado cuando se ha llevado a cabo la operación. Smart Connections permite que las conexiones nuevas que se realicen se dividan en tres segmentos en lugar de intentar llevarlas a cabo mediante una única conexión.

Por último el menú Run, contiene opciones para ejecutar la simulación, comprobarla, y pausarla o desplazarse paso a paso a través de ella para depurarla o para comprobar su correcto funcionamiento. Proporciona también varias formas alternativas de ver la ejecución, de ver sus resultados (o errores), y de especificar y controlar cómo evoluciona la ejecución y se muestra en la pantalla.

Por último, los menús Window y Help presentan las opciones típicas de este tipo de menús para cualquier aplicación “Windows”.

- Barras de herramientas

Arena posee varias barras de herramientas, cada una tiene varios grupos de botones, con menús desplegados para acceder rápidamente a actividades comunes. Las barras de herramientas disponibles en Arena se presentan a continuación [50].

Barra del menú Arena consiste en varios menús generales: File, Edit, View, Window y Help. Estos se hayan también en otros menús de Arena específicos.

El menú Tools provee el acceso a las herramientas de simulación y parámetros de Arena.

El menú Arrange contiene para realizar los gráficos de flujo y las operaciones de dibujo.



El menú Object contiene los módulos de conexiones y creación de submodelos.

El menú Run provee el control de la simulación. Su opción de Setup abre una forma que permite al usuario introducir la información, como los parámetros de proyecto (nombre, analista, fecha...), así como los parámetros de simulación (horizonte, periodo de calentamiento...). También tiene opciones que proveen el tipo VCR de funcionalidad para correr las réplicas de la simulación y las opciones de control para mostrar el movimiento de la entidad, asignaciones de variable y así sucesivamente.

La barra de herramientas Standar, que corresponde a la barra estándar de las aplicaciones de Windows. Contiene botones que soportan la construcción del modelo. Un botón importante en esta barra es el de Connect, que se utiliza para conectar los módulos de Arena con los bloques SIMAN y el resultado del diagrama describe el flujo del control lógico. El Time Patterns Editor, consiste en tres botones que permiten al que modela la programación de la disponibilidad de los recursos y su tasa de servicio. La barra Standard además provee de botones de estilo VCR, para poner el modelo Arena en modo pausa para señalar su evolución.



Figura 4. Barra de herramientas Estándar de Arena

La barra de proyectos de Arena permite acceder a los paneles donde se encuentran los módulos, bloques SIMAN y otros objetos. Los paneles se pueden seleccionar de la barra de Project haciendo clic en el botón Attach de la barra Standard. Cuando se hace clic en este botón, aparece una ventana de diálogo en la pantalla y muestra los archivos tipo correspondientes a cada panel. Eligiendo uno de los ficheros aparece el panel correspondiente en la barra Project. Los paneles que incluye fueron estudiados en el anteriormente. Posteriormente se describirán en profundidad las opciones de dos de los más utilizados.

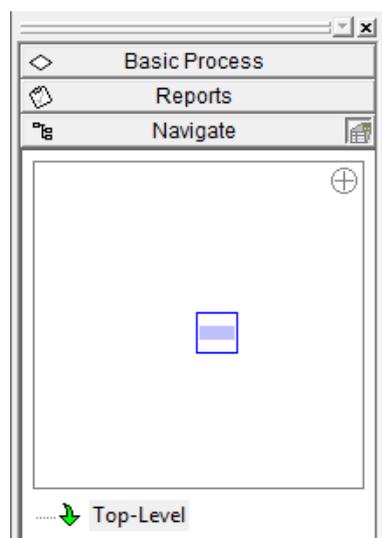


Figura 5. Barra de Proyectos de Arena



La barra View, corresponde a opciones del menú del mismo nombre. Asiste al usuario en la visualización del modelo. Sus botones incluyen zoom dentro, zoom fuera y vista previa. Estas funciones son útiles para ver grandes modelos en varios niveles de detalle.



Figura 6. Barra de herramientas View de Arena

La barra Arrange, tiene botones correspondientes al menú del mismo nombre.



Figura 7. Barra de herramientas Arrange de Arena

Los botones de la barra de Draw no tienen correspondencia con opciones de menú, por tanto, los dibujos únicamente se pueden realizar accediendo a la barra de herramientas. Con estas herramientas, se pueden dibujar líneas, poli líneas, arcos, etc., para vestir el modelo.

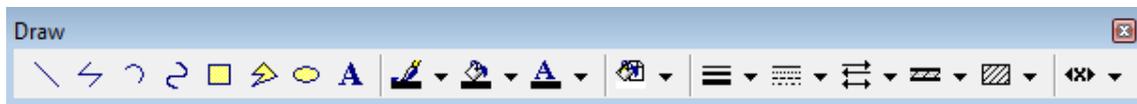


Figura 8. Barra de herramientas Draw de Arena

La barra Animate contiene elementos que permiten animar el modelo o mejorar la animación inherente a algunos módulos de Arena, mientras la simulación se está ejecutando. Los objetos de simulación incluye el reloj de simulación, colas, ventanas para monitorear las variables, gráficas dinámicas e histogramas.

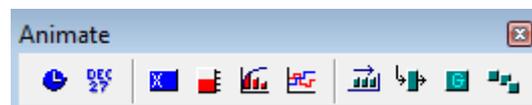


Figura 9. Barra de herramientas Animate de Arena

La barra Integration contiene botones relacionados con el asistente de transferencia de datos a módulos de Arena y VBA (el Editor de Visual Basic y el botón de Modo Diseño de VBA). Es decir, contiene las funciones de transferencia de datos a otras aplicaciones. Permite además la programación y diseño en Visual Basic.

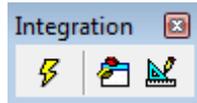


Figura 10. Barra de herramientas Integration de Arena

La barra Run Interaction corresponde a las opciones que ofrece el menú Run Interaction. Incluye las funciones de control de ejecución para seguir las simulaciones, como el acceso al código SIMAN y medios de depuración del modelo. Además incluye la visualización del modelo, así como el botón de Animate Connectors que activa u desactiva la animación del tráfico de entidades por las conexiones del módulo.



Figura 11. Barra de herramientas Run Interaction de Arena

La barra Animate Transfer sirve de interfaz con los objetos de animación de tipo transferencia, incluyendo el transporte manual, de que dispone Arena para las animaciones de objetos de este tipo.



Figura 12. Barra de herramientas Animate Transfer de Arena

En la parte izquierda de la aplicación, se muestran una serie de paneles que contienen los objetos de que dispone Arena para diseñar los diagramas de flujo y que a continuación se comentan con más detalle.

La barra Debug soporta la depuración de los modelos Arena mediante el control y monitoreo de la ejecución de la simulación. Consiste en varias subventanas. La subventana de la izquierda se utiliza en modo comando para seleccionar puntos de interrupción, asignar valores a la variable, observar las variables y trazar los flujos de entidad entre módulos. La ventana de la derecha, tiene pestañas para visualizar los eventos SIMAN en el modelo, mostrando los atributos de la entidad actual activa y pudiendo ver las expresiones definidas por el usuario en Arena.



Figura 13. Barra Debug de Arena



- Panel de procesos básico

En la siguiente figura, se observan los distintos módulos de flujos de datos y módulos de datos del panel de procesos básico.

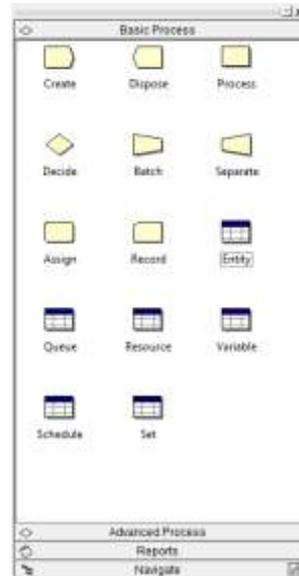


Figura 14. Panel de procesos básico de Arena

Los módulos de flujo de datos del panel Basic Process y sus características se recogen en la siguiente tabla [49]:

<p>Módulo Create</p>  <p>Create</p>	<p>Este módulo representa la llegada de entidades al modelo de simulación. Las entidades se crean usando una planificación o basándose en el tiempo entre llegadas. En este módulo se especifica también el tipo de entidad de que se trata. Una vez se incluye en el modelo a la derecha del símbolo aparece bajo una línea el número de entidades creadas.</p> <p><u>Posibles Usos.</u></p> <ul style="list-style-type: none"> • Punto de inicio de producción en una línea de fabricación. • Llegada de un documento (por ejemplo, una petición, una factura, una orden) en un proceso de negocio. • Llegada de un cliente a un proceso de servicio (por ejemplo, un restaurante, una oficina de información). <p><u>Parámetros</u></p> <ul style="list-style-type: none"> • Name: identificador único del módulo • Entity Type: nombre del tipo de entidad a ser generada. • Type: tipo de flujo de llegada a generar. Los tipos incluidos son:
--	---



	<p>Random (usa una distribución exponencial y hay que indicar la media), Schedule (usa una distribución exponencial pero la media se determina a partir del módulo Schedule especificado), Constant (se especifica un valor constante), o Expresión (se puede elegir entre distintas distribuciones).</p> <ul style="list-style-type: none"> • Value: determina la media de la distribución exponencial (si se usa Random) o el valor constante (si se usa Constant) para el tiempo entre llegadas. • Schedule Name: identifica el nombre de la planificación a usar. La planificación define el formato de llegada para las entidades que llegan al sistema. Sólo se aplica cuando se usa en Type, Schedule. • Expression: cualquier distribución o valor que especifique el tiempo entre llegadas. Se aplica sólo cuando en Type se usa Expression. • Units: unidades de tiempo que se usan para los tiempos entre llegadas y de la primera creación. • Entities per Arrival: número de entidades que se introducirán en el sistema en un momento dado con cada llegada. • Max Arrivals: número máximo de entidades que generará este módulo. • First Creation: momento de inicio en el que llega la primera entidad al sistema.
<p>Módulo Dispose  Dispose</p>	<p>Este módulo representa el punto final de entidades en un modelo de simulación. Las estadísticas de la entidad se registrarán antes de que la entidad se elimine del modelo.</p> <p><u>Posibles Usos</u></p> <ul style="list-style-type: none"> • Partes que abandonan un servicio • Finalización de un proceso de negocio • Clientes abandonando un comercio <p><u>Parámetros</u></p> <ul style="list-style-type: none"> • Name: identificador único del módulo. • Record EntityStatistics: determina si las estadísticas de las entidades entrantes se registrarán o no. Estas estadísticas incluyen value-added time, nonvalue-added time, wait time, transfer time, other time, total time, valueadded cost, non-value-added cost, wait cost, transfer cost, other cost, y total cost.
	<p>Este módulo corresponde a la principal forma de procesamiento en simulación. Se dispone de opciones para ocupar y liberar un recurso. Adicionalmente, existe la opción de especificar un “submodelo” y especificar jerárquicamente la lógica definida por el usuario. El tiempo de proceso se le añade a la entidad y se puede considerar como valor añadido, valor no-añadido, transferencia, espera u otros. Una vez se introduce en el modelo, aparece un número en la parte inferior del símbolo que indica el número de entidades que actualmente</p>



<p>Módulo Process</p>  <p>Process</p>	<p>están procesándose.</p> <p><u>Posibles Usos</u></p> <ul style="list-style-type: none">• Mecanizado de una parte.• Revisión de un documento para completarlo.• Rellenar órdenes.• Servir a un cliente. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name : identificador único del módulo• Type: método que especifica la lógica dentro del módulo. Un procesado Standard significa que toda la lógica se guardará dentro de un módulo Process y se definirá por una acción (Action) particular. Submodel indica que la lógica se definirá jerárquicamente en un “submodelo” que puede incluir un número indeterminado de módulos lógicos.• Action: tipo de proceso que tendrá lugar dentro del módulo. Existen cuatro tipos: Delay, Seize Delay y Seize Delay Release. Delay indica que solamente se llevará a cabo un proceso de retardo sin que existan restricciones de recursos. Seize Delay indica que un recurso será asignado en este módulo y que habrá un retardo y la liberación del recurso ocurrirá más tarde. Seize Delay Release indica que se asignará un recurso seguido por un retardo y luego, se liberará el recurso reservado. Delay Release indica que un recurso ha sido reservado previamente y que la entidad se retardará simplemente, y luego se liberará el recurso especificado.• Priority: valor de prioridad de la entidad que espera acceder en este módulo un determinado recurso si una o más entidades esperan el mismo recurso(s) en cualquier lugar en el modelo.• Resources: lista del recurso o conjunto de recursos utilizados para procesar la entidad. No se aplica cuando Action tiene el valor de Delay o cuando Type es submodel.• Delay Type: tipo de distribución o método de especificar los parámetros del retardo. Constant y Expression requieren valores simples, mientras que Normal, Uniform, y Triangular requieren varios parámetros.• Units: unidades de tiempo para los parámetros de retardo.• Allocation: determina cómo se asigna el tiempo de procesado y el coste del proceso a la entidad.• Minimum. valor mínimo en el caso de una distribución uniforme o triangular.• Value: valor medio para una distribución normal, el valor constante para un retardo de tiempo constante, o la moda para una distribución triangular.• Maximum: valor máximo para una distribución uniforme o triangular.• Std Dev: desviación estándar para una distribución normal.• Expression: expresión cuyo valor se evalúa y se usa para el procesado del retardo de tiempo.
--	--



<p>Módulo Decide</p>  <p>Decide</p>	<p>Este módulo permite a los procesos tomar decisiones en el sistema. Incluye la opción de tomar decisiones basándose en una o más condiciones (por ejemplo, si el tipo de la entidad es Gold Car) o basándose en una o más probabilidades (por ejemplo, 75% verdadero, 25% falso). Las condiciones se pueden basar en valores de atributos (por ejemplo, prioridad), valores de variables (por ejemplo, número de rechazados), el tipo de entidad o una expresión.</p> <p>Hay dos puntos de salida del módulo Decide cuando se especifica el tipo 2-way chance o 2-way condition. Hay un punto de salida para las entidades “verdaderas” y una para las entidades “falsas”. Cuando se especifica el tipo Nway chance o condition, aparecen múltiples puntos de salida para cada condición o probabilidad y una única salida “else”. Una vez incluido en el modelo, cerca de cada una de las ramas que salen del símbolo que representa el módulo, aparece un número. En la rama “True” corresponde al número de entidades que toman la rama de verdadero y el de la rama “False” el número de entidades que toman la rama de falso.</p> <p><u>Posibles Usos</u></p> <ul style="list-style-type: none">• Envío de partes defectuosas para que se vuelvan a hacer.• Ramas aceptadas frente a rechazadas.• Envío de clientes prioritarios a procesos dedicados <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: identificador único del módulo.• Type: indica si la decisión se basa en una condición o es aleatoria/porcentual. El tipo se puede especificar como 2-way o N-way. 2-way permite definir una condición o probabilidad (más la salida “false”). N-way permite definir cualquier número de condiciones o probabilidades, aparte de la salida “false”.• Conditions: define una o más condiciones que se usan para dirigir las entidades a los distintos módulos.• Percentages: define uno o más porcentajes usados para encaminar las entidades a los distintos módulos.• Percent True: valor que se comprobará para determinar el porcentaje de entidades que se han enviado a través de la salida True.• If: tipos de condiciones disponibles para ser evaluados.• Named: especifica el nombre de la variable, atributo, o tipo de entidad que se evaluarán cuando una entidad entre en el módulo.• Is: evaluador de la condición.• Value: expresión que se comparará con un atributo o variable o que se evaluará como una única expresión para determinar si es verdadero o falso.
--	--



<p>Módulo Assign</p>  <p>Assign</p>	<p>Este módulo se emplea para asignar valores nuevos a las variables, a los atributos de las entidades, tipos de entidades, figuras de las entidades, u otras variables del sistema. Se pueden hacer múltiples asignaciones con un único módulo Assign. Para añadir una nueva variable al modelo, simplemente se selecciona Add, Type: Entity, el nombre de la variable y el valor que se desea tome a partir de ese momento.</p> <p><u>Posibles Usos</u></p> <ul style="list-style-type: none">• Acumular el número de subensamblados añadidos a una parte.• Cambiar el tipo de entidad para representar una copia de un formulario multicopia.• Establecer una prioridad del cliente. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: identificador único del módulo.• Assignments: especifica la o las asignaciones que se llevarán a cabo cuando la entidad ejecute el módulo.• Type: tipo de asignación que se va a realizar. Other, puede incluir variables del sistema, tales como capacidad de los recursos o tiempo de finalización de la simulación.• Variable Name: nombre de la variable a la que se asignará un nuevo valor.• Attribute Name: nombre del atributo de la entidad al que se le asignará un nuevo valor.• Entity Type: nuevo tipo de entidad que se le asignará a la entidad cuando entre en el módulo.• Entity Picture: nueva imagen de la entidad que se le asignará.• Other: Identifica la variable del sistema especial a la que se le asignará un nuevo valor.• New value: Valor asignado al atributo, variable, u otras variables del sistema.
<p>Módulo Batch</p>  <p>Batch</p>	<p>Este módulo funciona como un mecanismo de agrupamiento dentro del modelo de simulación. Los lotes pueden estar agrupados permanente o temporalmente.</p> <p>Los lotes temporales deben ser divididos posteriormente usando el módulo Separate.</p> <p>Los lotes se pueden realizar con un número específico de entidades de entrada o se pueden unir a partir del valor de un determinado atributo. Las entidades que llegan a un módulo Batch se coloca en una cola hasta que se ha acumulado el número necesario de entidades. Una vez acumuladas, se crea una nueva entidad representativa. Cuando se incluye en el modelo, en la parte inferior del símbolo se representa el Número de entidades en espera de ser agrupadas.</p>



	<p><u>Posibles Usos</u></p> <ul style="list-style-type: none">• Recoger un cierto número de partes antes de empezar a procesar.• Reensamblar previamente copias separadas de un formulario.• Juntar un paciente con los resultados de sus pruebas antes de concederle una cita. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: identificador único del módulo.• Type: método de agrupamiento de las entidades.• Batch Size: número de entidades a ser agrupadas.• Save Criterion: método para asignar valores de atributos definidos por el usuario, representativos de la entidad.• Rule: determina cómo se agruparán las entidades que llegan. Any Entity tomará las primeras “Batch Size” entidades que lleguen y las unirá todas juntas. By Attribute significa que los valores de los atributos especificados deben coincidir para poder ser agrupados.
<p>Módulo Separate</p>  <p>Separate</p>	<p>Este módulo se puede usar para replicar la entidad entrante en múltiples entidades o para dividir una entidad previamente agrupada. Se especifican también las reglas de asignación de atributos para las entidades miembro.</p> <p>Cuando se segmentan lotes existentes, la entidad temporal que se formó se destruye y las entidades que originalmente formaron el grupo se recuperan. Las entidades saldrán del sistema secuencialmente en el mismo orden en que originalmente se agregaron al lote.</p> <p>Cuando se duplican entidades, se hacen el número de copias especificado.</p> <p>En el símbolo que representa este módulo, el original sale por la rama superior y el duplicado por la rama inferior.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Enviar entidades individuales que representan cajas eliminadas de un contenedor.• Enviar una orden tanto a realización y a facturación para un procesamiento paralelo.• Separar un conjunto de documentos previamente agrupados. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: identificador único del módulo.• Type: método de separación de las entidades entrantes.• Duplicate Original, toma la entidad original y realiza un cierto número de copias idénticas. Split Existing Batch, requiere que la entidad entrante sea una entidad agrupada temporalmente utilizando el módulo Batch. Las entidades originales se desagruparán.• Percent Cost to Duplicates: distribución de costes y tiempos de la entidad entrante en los duplicados salientes.• # of Duplicates: número de entidades salientes que dejarán el



	<p>módulo, además de la entidad entrante original.</p> <ul style="list-style-type: none"> • Member Attributes: método de determinar cómo asignar los valores de los atributos de la entidad representativa a las entidades originales. • Attribute Name: nombre del atributo(s) de la entidad representativa que se asignan a las entidades originales del grupo.
<p>Módulo Record</p>  <p>Record</p>	<p>Este módulo representa el final de entidades en un modelo de simulación. Las estadísticas de la entidad se registrarán antes de que la entidad se elimine del modelo.</p> <p><u>Posibles Usos</u></p> <ul style="list-style-type: none"> • Partes que abandonan el servicio modelado • Finalización de un proceso de negocio. • Clientes abandonando un comercio. <p><u>Parámetros</u></p> <ul style="list-style-type: none"> • Name Identificador único del módulo. • Record Entity • Statistics • Determina si las estadísticas de las entidades entrantes se registrarán o no. Estas estadísticas incluyen value-added time, nonvalue added time, wait time, transfer time, other time, total time, valueadded cost, non-value-added cost, wait cost, transfer cost, other cost, y total cost.

Tabla 1. Módulos de flujo de datos del panel Basic Process

Dentro de los módulos de datos, se encuentran [49]:

	<p>Este módulo de datos define los diversos tipos de entidades y su valor de imagen inicial en la simulación.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none"> • Elementos que se van a producir o ensamblar (piezas, pallets). • Documentos: formularios, e-mails, faxes, informes... • Gente que se mueve a través del proceso (clientes). <p><u>Parámetros</u></p> <ul style="list-style-type: none"> • Entity Type: nombre de la entidad que se va a definir. • Initial Picture: representación gráfica de la entidad al inicio de la simulación. • Holding Cost/Hour: coste por horas de procesamiento de la entidad a lo largo del sistema. Este coste se sufre siempre que la entidad se encuentre en cualquier sitio del sistema.
--	---



<p>Módulo Entity</p>  <p>Entity</p>	<p>Initial VA Cost: valor de coste inicial que se asignará al atributo value-added cost de la entidad. Este atributo acumula al coste sufrido cuando una entidad pasa tiempo en una actividad de espera; por ejemplo, esperando a ser metida en un lote o esperando un recurso(s) en un módulo Process.</p> <ul style="list-style-type: none">• Initial Transfer Cost: valor de coste inicial que se le asignará al atributo de coste de transferencia de la entidad. Este atributo acumula el coste sufrido cuando una entidad pasa tiempo en una actividad de transferencia.• Initial Other Cost: valor de coste inicial que se asignará al atributo other cost de la entidad. Este atributo acumula el coste sufrido cuando una entidad pasa tiempo en una actividad de transferencia.
<p>Módulo Queue</p>  <p>Queue</p>	<p>Este módulo de datos se puede usar para cambiar la regla para una determinada cola. La regla de la cola por defecto es First In, First Out salvo que se indique otra cosa en este módulo. Hay un campo adicional que permite definir la cola como compartida.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Cola de trabajos esperando un recurso en un módulo Process.• Área de almacenamiento de documentos que esperan ser cotejados en un módulo Batch. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: nombre de la cola cuyas características se van a definir.• Type: regla de encolado para la cola, la cual puede estar basada en un atributo. Los tipos incluyen First In, First Out; Last In, First Out; Lowest Attribute Value (primero); y Highest Attribute Value (primero). Un valor de atributo bajo puede ser 0 o 1, mientras que un valor alto puede ser 200 o 300.• Attribute name: atributo que se evaluará para los tipos Lowest Attribute Value o Highest Attribute Value. Las entidades con valores de atributos más bajos o más altos serán encoladas primero en la cola.• Shared: campo de selección que determina si una determinada cola se usa en múltiples sitios dentro del modelo de simulación. Sólo se puede usar en el caso de recursos de petición (es decir, con el módulo Seize del panel Advanced Process).
	<p>Este módulo de datos define los recursos en un sistema de simulación, incluyendo información de costes y disponibilidad del recurso. Los recursos pueden tener una capacidad fija que no varía durante la simulación o pueden operar basándose en una planificación. Los fallos y estados del recurso se pueden definir también en este módulo.</p>



<p>Módulo Resource</p>  <p>Resource</p>	<p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Equipamiento (maquinaria, caja registradora, línea de teléfono).• Gente (empleados, procesadores de órdenes, empleados de ventas, operadores). <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: nombre del recurso cuyas características se deben definir.• Type: método para determinar la capacidad de un recurso. Fixed Capacity no cambiarán durante la simulación. Based on Schedule significa que se usa el módulo Schedule para especificar la capacidad y duración del recurso.• Capacity: número de unidades de recurso de un determinado nombre que están disponibles en el sistema para el procesamiento.• Schedule: name Identifica el nombre de la planificación a usar por parte del recurso. El planificador define la capacidad del recurso para un periodo de tiempo determinado.• Schedule Rule: determina cuándo debe ocurrir el cambio de capacidad cuando se requiere una disminución de la capacidad para una unidad de recurso muy ocupada.• Busy/Hour: coste por hora de un recurso que está procesando una entidad.• Idle/Hour: coste por hora del recurso cuando está libre.• Per Use: coste de un recurso en base al uso, independientemente del tiempo durante el cual se esté usando.• StateSet Name: nombre de los estados que se le pueden asignar a un recurso durante la simulación.• Initial State: estado inicial del recurso.• Failures: lista todos los fallos asociados con el recurso.
<p>Módulo Schedule</p>  <p>Schedule</p>	<p>Este módulo de datos se puede usar en conjunción con el módulo Resource para definir una operación de planificación para un recurso o con el módulo Create para definir una planificación de llegada. Además, una planificación se puede usar y referir a factores de retardos de tiempo basados en el tiempo de simulación.</p> <p><u>Posibles Usos</u></p> <ul style="list-style-type: none">• Planificación del trabajo, incluyendo descansos, para la plantilla.• Esquemas de fallos del equipamiento.• Volumen de clientes que llegan a un comercio.• Factores de curva de aprendizaje de los nuevos trabajadores. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: nombre de la planificación que se va a definir.• Type: tipo de planificación que se va a definir. Puede ser relativa a Capacity (para planificaciones de recurso), relativa a Arrival (para el módulo Create), o Other.• Time Units: unidades de tiempo empleadas para informar de la



	<p>duración de tiempo.</p> <ul style="list-style-type: none">• Scale Factor: método de escalado de la planificación para incrementar o disminuir los valores de Arrival/Other. Los campos Value se multiplicarán por el factor de escala para determinar los nuevos valores.• Durations: lista los pares valor y duración para la planificación. Los datos de planificación se introducen gráficamente usando el editor de planificaciones gráfico.• Value: representa la capacidad del recurso, la frecuencia de llegadas, o algún otro valor.• Duration: duración de tiempo para la que un Value especificado será válido.
<p>Módulo Set</p> 	<p>Este módulo de datos define varios tipos de conjuntos, incluyendo recursos, contadores, cuentas, tipos de entidad, y figuras de entidad. Los conjuntos de recursos se pueden usar en los módulos Process (Seize, Release, Enter y Leave en el panel Advanced Transfer). Los conjuntos counter y tally se pueden usar en el módulo Record. Los conjuntos queue se pueden utilizar con Seize, Hold, Access, Request, Leave, y Allocate de los paneles Advanced Process y Advanced Transfer.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Máquinas que pueden realizar las mismas operaciones en un servicio de fabricación.• Supervisores, empleados de caja en un comercio.• Conjunto de figuras correspondientes a un conjunto de tipos de entidades. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: nombre del conjunto que se va a definir.• Type: tipo de conjunto que se va a definir.• Members: grupo repetido que especifica los miembros del recurso en el conjunto. El orden es importante si se emplean reglas de selección del tipo Preferred Order y Cyclical.• Resource Name: nombre del recurso a ser incluido en el conjunto de recursos.• Tally Name: nombre de la cuenta dentro del conjunto de cuentas.• Counter Name: nombre del contador dentro del conjunto de contadores.• Entity Type: nombre del tipo de entidad dentro del conjunto de tipos de entidad.• Picture Name: nombre de la imagen dentro del conjunto de imágenes.
	<p>Este módulo de datos se utiliza para definir una dimensión de la variable y su valor(es) inicial(es). Las variables se pueden referenciar en otros módulos, se les puede reasignar un valor nuevo y se pueden</p>



<p>Módulo Variable</p>  <p>Variable</p>	<p>emplear en cualquier expresión.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Número de documentos procesados por hora.• Número serie a asignar a partes para una identificación única.• Espacio disponible en un servidor. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: nombre de la variable que se va a definir.• Rows: número de filas en una variable con dimensión.• Columns: número de columnas en una variable con dimensión.• Statistics: caja de selección para determinar si se recogerán o no estadísticas• Clear Option: define el tiempo, en el caso de ser requerido, en que el valor(es) de la variable, se reinicia al valor(es) inicial(es) especificado. Si se escoge Statistics implica reiniciar esta variable a su valor inicial siempre que las estadísticas se borren. Si se escoge System se reinicia la variable a su valor inicial siempre que se reinicia el sistema. None indica que nunca se reinicia la variable a su valor inicial.• Initial Values: lista el valor(es) inicial de la variable.• Initial Value: valor variable al inicio de la simulación.
--	---

Tabla 2. Módulos de datos del panel Basic Process

- Panel de procesos avanzado

En este panel se encuentran funcionales adicionales para modelar el proceso. Ya que el objeto del presente proyecto no es la descripción detallada de la herramienta de simulación, si no su empleo para el desarrollo de modelos que simulen determinados sistemas, se van a explicar los módulos específicos de este panel que se han empleado en la construcción del mismo.



Figura 15. Panel de procesos avanzados de Arena



<p>Módulo Hold</p>  <p>Hold</p>	<p>Este módulo frenará una entidad en una cola para esperar a una señal, esperar una condición llegue a ser verdadera (scan) o sea detenida infinitamente, para que sea removida después con el módulo Remove.</p> <p>Si la entidad está detenida esperando una señal, el módulo Signal se utiliza en otro lugar en el modelo para permitir que la entidad pase al siguiente módulo. Si la entidad está esperando que una condición dada sea verdadera, la entidad permanecerá en el módulo, hasta que la condición/es llegue a ser verdadera.</p> <p>Cuando la entidad es un Hold infinito, el módulo Remove se utiliza en algún lugar del modelo para permitir que la entidad continúe procesándose.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Espera a que un semáforo se vuelva verde.• Deteniendo una pieza esperando una autorización.• Comprobando el estado de una máquina u operador para continuar un proceso. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: identificador único del módulo que se muestra en la forma del módulo.• Type: indica el razonamiento de espera de la entidad en una cola interna o especificada. Wait for Signal, esperará la entidad hasta que se reciba una señal del mismo valor. Scan for Condition esperará la entidad hasta que una condición específica sea verdadera. Con la opción Infinite Hold, la entidad esperará hasta que sea retirada de la cola por el módulo Remove.• Wait for Value: código de señal para la entidad en espera. Se aplica solo cuando Type es Wait for Signal.• Limit: número máximo de entidades en espera que serán liberadas una vez recibida una señal. Se aplica solo cuando Type es Wait for a Signal.• Condition: especifica la condición que será evaluada para que espere la entidad en el módulo. Si la condición evaluada es verdadera, la entidad deja el módulo inmediatamente. Si la condición es falsa, la entidad esperará en la cola asociada hasta que la condición se vuelva verdadera. Se aplica solo cuando Type es Scan for Condition.• Queue Type: determina el tipo de cola en el que esperan las entidades. Si se selecciona Queue, se especifica el nombre de la cola. Si se selecciona Set, se especifican la cola seleccionada y los miembros. Si se selecciona Internal, se emplea una cola interna para la espera de todas las entidades. Attribute y Expression son métodos adicionales para definir la utilización de la cola.• Queue Name: es visible si Queue Type es Queue y define el nombre simbólico de la cola.• Set Name: este campo solo es visible si Queue Type is Set y define
--	---



	<p>la cola seleccionada que contiene la cola que se referencia.</p> <ul style="list-style-type: none">• Set Index: este campo es visible solo si Queu Type es Set y define una referencia en la cola seleccionada.• Attribute: este campo es visible solo si Queue Type es Attribute. El atributo introducido en este campo será evaluado para indicar que cola se va a utilizar.• Expression: este campo solo es visible su Queue Type es expression. La expresión introducida en este campo será evaluada para indicar que cola se emplea.
<p>Módulo Failure</p>  <p>Failure</p>	<p>El módulo Failure se diseña para utilizarlo con recursos. Cuando ocurra un fallo, a pesar de su capacidad, el recurso completo falla. Los Failures se han diseñado para utilizarse con recursos de capacidad simple o múltiple cuyas unidades de recurso individual todas fallan al mismo tiempo.</p> <p><u>Posibles usos</u></p> <ul style="list-style-type: none">• Información de rotura de una máquina• Cambio de cinta de caja registradora cada x clientes. Mantenimiento.• Apagado o reiniciado aleatorio de un ordenador. <p><u>Parámetros</u></p> <ul style="list-style-type: none">• Name: el nombre del fallo asociado a uno o más recursos.• Type: determina si la ocurrencia del fallo está basada en el tiempo o en un contador.• Count: define el número de recursos liberados hasta que aparece el fallo. Válido cuando Type es Count.• Up time: define el tiempo entre fallos como el tiempo que transcurre entre ellos. Válido cuando Type es Time.• Up Time Units: unidades de tiempo entre fallos (Up time) para fallos definidos por tiempo.• Down Time: define la duración del fallo.• Down Time Units: unidades de duración del fallo (Down Time).• Uptime in this State only: define el estado que debería considerarse para el tiempo entre fallos. Solo para definición de fallos basada en tiempo.• Si el estado no se especifica, entonces se consideran todos los estados. Por ejemplo el tiempo especificado entre fallos no depende del tiempo que ocupa un estado específico, sino del tiempo total de la simulación.

Tabla 3. Módulos del panel Advanced Process



- Modelado de fallos en una máquina

Los fallos se pueden considerar un tipo de ociosidad en las líneas de producción. Normalmente un fallo en una estación de trabajo supone reparar la estación lo antes posible y continuar con la operación de la máquina, una vez se completa la reparación.

El impacto que producen los fallos de la máquina en el funcionamiento del sistema y la reducción del impacto mediante la colocación de buffers entre las máquinas, han sido ampliamente estudiados [51], [52], [53], [54]. Por otra parte, Whitt estudió el análisis en las redes de producción con buffers infinitos [55].

Cuando el sistema a modelar presentan fallos, la persona que lo modela normalmente se interesa en las probabilidades “long-run” de los estados de no funcionamiento de la máquina, las cuales trasladan en funcionamiento la fracción de tiempo que se emplea en estos estados de no funcionamiento. Es lógico que la operación eficiente requiera que se minimicen los tiempos de parada, ya que representan una pérdida de tiempo de producción.

Las probabilidades “long-run” de los estados de no funcionamiento, se pueden estimar mediante la opción Frequency en el módulo Statistic.

La simulación puede ayudar al diseñador del modelo a comprender la influencia de los fallos en el funcionamiento del sistema.

Un proceso estocástico que modela fallos se caracteriza por las propiedades estadísticas de los intervalos de tiempo que separan dos fallos consecutivos o mediante el proceso de recuento, del número de fallos ocurridos en un período de tiempo. Estas dos caracterizaciones matemáticas son equivalentes, aunque en simulación se suele emplear la primera forma.

Los periodos de operación y fallo se denominan Uptimes y Downtimes respectivamente.

Un fallo en una estación puede llegar a ser un cuello de botella, pues causa un bloqueo aguas arriba de la estación de trabajo y supone una falta de suministro en la estación aguas abajo.

Un escenario típico de fallo en una estación simple se presenta como sigue. Después de un periodo normal de operación (Uptime), si ocurre un fallo, la estación para su proceso y entonces atraviesa un período en el que está parada (Downtime), mientras se somete a una reparación.

Uptime es el tiempo entre una reparación completa y el siguiente fallo o tiempo de fallo, mientras Downtime es el tiempo de reparación.

Los fallos que ocurren mientras la máquina está procesando trabajos, se denominan Operation Dependant.

Se denomina Operation Independant a las paradas debidas al mal funcionamiento, limpieza, mantenimiento y otras paradas de un sistema particular.



Fundamentalmente cada máquina presenta dos estados: Idle u ocioso y Busy u ocupado.

Los estados de parada son tres: Down, Cleaning y Adjustment.

Arena admite cualquier número particular de estados definidos por el usuario, denominados auto-states. Los posibles estados para un recurso que propone Arena son: idle, busy, failed e inactive.

En cualquier punto en el tiempo un recurso estará en estos estados. La transición de un estado a otro la causa la ocurrencia de un evento. Por ejemplo la aparición de un fallo, automáticamente produce que la máquina experimente una transición al estado de fallo. Los estados definidos por el usuario y sus transiciones, pueden ser programadas a decisión del diseñador.

Las reglas de fallo son tres: preempt, wait e ignore.

Preempt, supone que comience el tiempo de parada suspendiendo el recurso inmediatamente que llega el fallo. El procesamiento del artículo actual continuará una vez el downtime termina.

Wait, permite terminar de procesar la entidad actual y posteriormente se suspende el recurso y se inicia el downtime. Se graba el downtime completo.

La regla ignore, supone que el downtime comienza después de que termine de procesarse la unidad del artículo que actualmente se está procesando. Sin embargo, solo se graba, aquella parte del tiempo de parada o downtime siguiente a la finalización de la unidad actual [56].

- Preparar para la simulación

Para preparar el modelo para la simulación, hay que especificar la información general del proyecto y la duración de la ejecución de la simulación. Mediante la opción Run>Setup en la pestaña Project Parameters, en la sección de Statistics Collection se seleccionan las estadísticas de los módulos que se quieren recoger (entities, queues, resources, processes...).

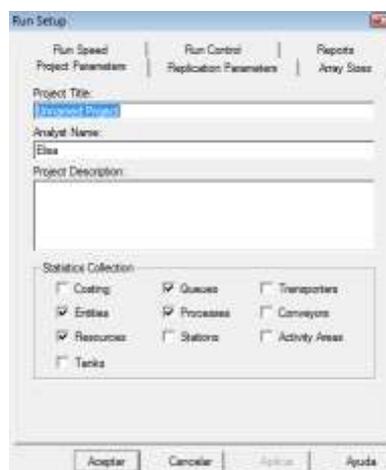


Figura 16. Ventana de opción Run Setup de Arena (Project Parameters)



En Run>Setup en la pestaña Replications parameters, se puede seleccionar la longitud de las réplicas que se desean realizar y las unidades de tiempo u horizonte de la simulación.



Figura 17. Ventana de opción Run Setup de Arena (Replication Parameters)

Para salvar la simulación en el menú File>Setup se selecciona Save o Save As y se selecciona el nombre del archivo. En la barra de herramientas Standard se puede encontrar un botón similar a otras aplicaciones de Windows, para salvar la simulación.

- Simular el proceso

Se comienza a simular el proceso a partir del botón play de la barra de herramientas Standard o a partir del menú Run >Go. Arena primero comprueba si se ha definido un modelo válido, si es así emprende la simulación. A medida que el proceso de simulación progresase pueden observar dibujos de entidades, asemejándose a páginas moviéndose entre las formas del diagrama de flujos. Además bajo los módulos aparecen valores cambiantes, que corresponden a las entidades que se crean y procesan.

Si la animación se mueve muy rápido se puede ajustar el factor de escala de la animación. Esto se puede hacer de dos formas, mediante Run>Speed>AnimationSpeedFactor, introduciendo un valor bajo. Otra forma es haciendo clic a la izquierda de la barra que se encuentra en el extremo izquierdo dentro de barra Standard.

Al final de la simulación, por defecto, aparece una ventana en la que el programa pregunta si se desean visualizar los resultados. En Run>Setup>Reports, se puede seleccionar si al final de la simulación se muestran los resultados siempre, nunca o preguntando al usuario.

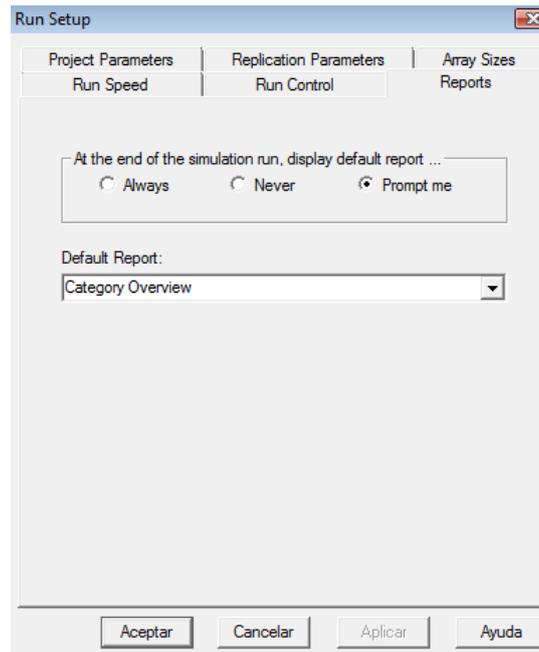


Figura 18. Ventana de opción Run Setup de Arena (Reports)

El informe resume los resultados a lo largo de las réplicas. Otro tipo de informes proveen detalles de cada réplica. Haciendo clic en las entradas en Category Overview, se pueden visualizar distintos tipos de información.

3.5. Optquest

Es un módulo del programa Arena, que amplía las capacidades de análisis de Arena, permitiendo buscar automáticamente las soluciones óptimas dentro de los modelos de simulación. El usuario describe el problema de optimización en Optquest, permitiendo buscar los valores de las variables de control que maximizan o minimizan un objetivo prefijado.

El interfaz de usuario se presenta en estructura de árbol, donde sus nodos son los componentes del modelo de optimización (controles, respuestas, restricciones, objetivos, soluciones propuestas y opciones). Cuando se pulsa cada nodo, aparece a la derecha un panel para seleccionar y dar valor a las variables que se van a considerar en el análisis.

En los apartados que se estudian a continuación, se explica el funcionamiento de este módulo de Arena y su aplicación para el caso de estudio que se ha desarrollado en el presente proyecto.

3.5.1. ¿Qué es un modelo optimizado?

Un modelo de simulación es una representación de un problema o sistema. Puede ayudar al análisis de diseños y a encontrar soluciones adecuadas.



En un modelo de simulación se seleccionan las características más importantes de un problema y los presenta de una forma fácil de interpretar.

Un modelo optimizado, lo que busca es maximizar o minimizar alguna cantidad, beneficio ó coste.

El modelo tiene tres elementos:

- Constraints (Restricciones)
- Controls (Controles)
- Objective (Objetivo)

Las restricciones son las relaciones entres controles y/o resultados. Por ejemplo, si se tiene un grupo de inversiones y una cantidad de dinero destinada a ellas, una restricción podría limitar la cantidad máxima que puede asignarse a estas inversiones.

Los controles son variables o recursos que pueden manipularse para afectar el comportamiento de un sistema que se simula. Por ejemplo: la cantidad de producto a fabricar, el tamaño de la flota de un sistema de transporte, el número de trabajadores a asignar a una actividad...

El objetivo, es la respuesta matemática o una expresión utilizada para representar el objetivo del modelo. Por ejemplo: minimizar colas, maximizar beneficios...

La solución de un modelo optimizado provee de un conjunto de valores para los controles que optimiza (maximiza o minimiza) el objetivo asociado.

Cuando los datos del modelo son inciertos y solo se pueden describir probabilísticamente, el objetivo se representa por una distribución de probabilidad que varía con la elección de un grupo de valores para los controles. Se puede encontrar una aproximación de la distribución de probabilidad simulando el modelo mediante Arena.

Un modelo optimizado con incertidumbre tiene bastantes elementos adicionales:

- Assumptions (Suposiciones)

Captura la incertidumbre de los datos del modelo utilizando las distribuciones de probabilidad. Las suposiciones del modelo se modelan inicialmente utilizando distribuciones de probabilidad para cada proceso estocástico.

- Responses (Respuestas)

Una salida del modelo de simulación. Ejemplo de respuestas son la utilización de un recurso, tiempo de ciclo, tamaño de cola...

- Response Statistics (Respuesta estadística)

Sumario o conjunto de valores de salidas como la media, desviación típica o la varianza.

Se puede controlar la optimización maximizando, minimizando y restringiendo las estadísticas de salida. Ejemplos: Tiempo medio de espera, máxima longitud de cola...



3.5.2. Metodología de Optquest

Optquest es un optimizador genérico que hace posible separar completamente el modelo que representa el sistema y el procedimiento que resuelve los problemas de optimización definidos en el modelo. Esta adaptación de diseño de métodos meta heurísticos, permite crear un modelo del sistema que incluye tantos elementos como sean necesarios para representar el caso real rigurosamente.

El procedimiento de optimización utiliza los outputs del modelo para evaluar los inputs. El procedimiento de optimización, analizando esta evaluación de los inputs y las anteriores, selecciona un nuevo conjunto de valores de inputs. Realiza una búsqueda no-monotónica especial, donde los inputs generados sucesivamente producen distintas soluciones, no cada vez mejores, pero que en un tiempo se encauzarán hacia las soluciones más eficientes o mejores soluciones. Normalmente este proceso continúa hasta que alcance algún criterio de finalización, usualmente un tiempo límite.

3.5.3. Elementos de un modelo de Optimización

Los elementos de un modelo de optimización son: controls, constraints y objective [49].

- Controls

Son variables o recursos en el modelo que hay que controlar. El modelo de optimización se formula en términos de los controladores seleccionados. Sus valores cambian antes de cada simulación, hasta que se encuentran los mejores valores en el tiempo predefinido. En un modelo de un sistema de control de la producción como Kanban o Conwip, los controls podrían ser entre otros las tarjetas o señales.

- Constraints

Son las restricciones. Definen la relación entre los controls y /o salidas. Optquest diferencia entre dos tipos de restricciones: lineales y no lineales.

Las restricciones lineales describen una relación lineal entre controls. Por ejemplo, la restricción de presupuesto para la compra de un equipo.

Una restricción no lineal contiene una expresión no lineal o una respuesta. Por ejemplo, el tiempo empleado en una cola.

Optquest puede evaluar las restricciones lineales sin correr una simulación en Arena. Las restricciones no lineales solo se pueden evaluar corriendo la simulación en Arena.

La solución que cumple todas las restricciones se considera una solución factible. Feasible es el término en inglés para referirse a factible.

Si una o más restricciones se incumplen, la solución es no factible o infeasible.

No todos los modelos de optimización necesitan restricciones; sin embargo, en todas ellas se distingue entre solución factible o no factible. Una solución factible es aquella que satisface todas las restricciones. La solución es no factible, cuando una



combinación de los valores de los controladores son puede satisfacer un conjunto de restricciones.

Resulta necesario añadir, que una solución puede ser no factible, por no satisfacer las restricciones del problema, y esto no implica que el problema o modelo sea no factible.

La prioridad de Optquest es encontrar una solución factible, cuyas restricciones sean factibles. Una vez encontrada, se centra en buscar soluciones que mejoren este valor.

- Objective

Cada modelo de optimización, tiene una función objetivo que representa matemáticamente el objetivo del modelo.

Un modelo de simulación es una representación de un problema o sistema que puede ayudar en el análisis de los diseños y a encontrar buenas soluciones.

Optquest encuentra el valor óptimo del objetivo, seleccionando valores mejorados de los diferentes controls. Cuando todos los datos del modelo son inciertos y se describen mediante distribuciones de probabilidad, el objetivo seguirá una distribución de probabilidad para cualquier conjunto de valores de los controls. Se puede encontrar una aproximación empírica de esta distribución de probabilidad mediante el análisis estadístico de una respuesta dada. Sin embargo, el modelo de optimización se define normalmente utilizando un estimador estadístico asociado con una respuesta como el objetivo de ser maximizado o minimizado.

3.5.4. Tipos de modelos de optimización

Los modelos de optimización pueden clasificar de acuerdo al tipo de las variables de control en:

Modelo	Tipo de control
Discreto	Solo variables de control discretas
Continuo	Solo variables de control continuas
Mixto	Variables de control discretas y continuas

Tabla 4. Tipos de modelos de optimización

Un modelo de optimización también se puede clasificar en función de las formas de las funciones que definen las restricciones y el objetivo. De este modo un modelo puede ser lineal o no lineal.

En un modelo lineal, todos los términos de las fórmulas consisten en una variable de control simple multiplicada por una constante. Ejemplo: $3 * x - 1.2 * y = 0$.

Cualquier modelo que contiene términos en las funciones objetivo y de restricciones que pueden considerarse como no lineales. Ejemplos: x^2 , $x * y$, $1/x$...



Una tercera clasificación, depende de la naturaleza de los datos. Distingue los modelos deterministas de los estocásticos. Un modelo será determinista, si todos los datos de las entradas son constantes o se asume que se conocen con certeza. Serán estocásticos, si existe incertidumbre en los datos y éstos se describen con distribuciones de probabilidad.

Los modelos estocásticos son mucho más difíciles de optimizar, porque requieren simulación para calcular la función objetivo. Optquest se ha diseñado para resolver modelos estocásticos utilizando Arena como el evaluador de la función objetivo, aunque también puede resolver problemas de modelos deterministas.

3.5.5. Optimizar el problema en Optquest

En la siguiente figura, se determinan los pasos que hay que seguir para optimizar un problema mediante Optquest [57]:

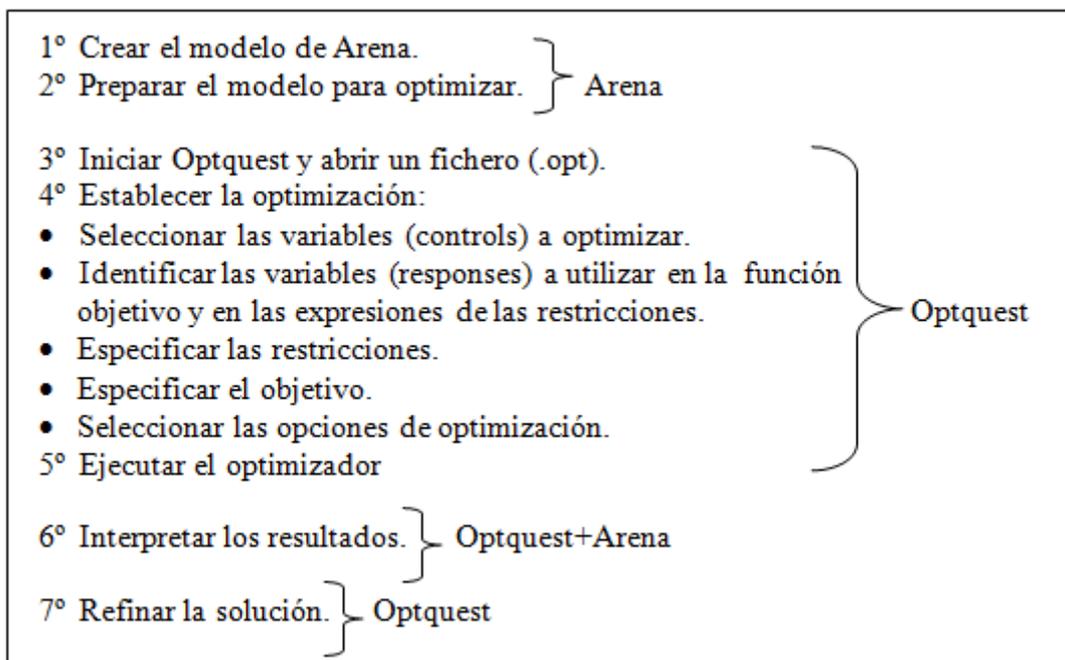


Figura 19. Esquema de optimización

A continuación se explican con más detalle cada uno de los puntos anteriores.

Los pasos primero y segundo, corresponden a la creación del modelo y su preparación para optimizar. Dependerán del caso concreto objeto de estudio. En primero de ellos, ya se estudió en el apartado anterior del presente proyecto.

2º Preparar el modelo en Arena

Antes de utilizar Optquest, se ha de desarrollar un modelo en Arena apropiado para el problema específico a tratar. Habrá que construir un modelo de simulación y comprobar que es adecuado, posteriormente habrá que definir los controles y respuestas que se



utilizarán en el modelo de optimización. Se ha de refinar el modelo Arena y realizar varias simulaciones para asegurar que el modelo trabaja correctamente y el resultado es el que se esperaba.

Una vez definidas las variables de control y las estadísticas de las respuestas en Arena se puede comenzar el proceso de Optimización en Optquest.

Lo primero es seleccionar los controls a optimizar. Como se ha comentado anteriormente, los valores de estas variables seleccionadas variarán con cada simulación, hasta que Optquest halle el mejor valor para el objetivo. Arena puede invalidar cualquiera de los valores de las variables de control que Optquest provee durante la simulación, con lo que se interferiría con la optimización. Por ello cualquier control lógico en el modelo arena deberá establecerse para trabajar con Optquest.

Para los recursos Optquest establece la capacidad inicial. Se ha de tener en cuenta que si el recurso sigue una programación en lugar de una capacidad fija y se quiere que sea una variable de control para la optimización, será necesaria una variable que identifique la máxima capacidad del recurso. La programación podría entonces seleccionar la capacidad del recurso multiplicando determinadas cantidades por esta variable. Por otra parte es necesario considerar que la lógica que ajusta la capacidad del recurso en el modelo debería ajustar la capacidad máxima del recurso. Y debería hacerlo, desde la optimización de diferentes valores de capacidad.

Por otra parte, se ha de considerar la exclusión de algunas variables. Cualquier variable en el modelo Arena puede utilizarse como una variable de control en el modelo de optimización. La lista de las variables y recursos aparece al desplegar en nodo Controls en el árbol de la ventana de la izquierda de la pantalla principal de Optquest. El modelo podrá contener variables que no se seleccionan para la optimización, porque son innecesarias.

Además la función objetivo y las restricciones pueden estar relacionadas con las salidas de la simulación y por tanto estarán basadas en las respuestas que se definen en el modelo de simulación de Arena.

Ciertas funciones del modelo se seleccionan en Optquest y pueden cambiar en el modelo Arena. Entre ellas:

- Batch Run (Animación)
- Run in Full- Screen Mode (Simular en pantalla completa)
- Quiet Mode Property (Propiedades de modo)
- Number of replications (Número de réplicas)

Cuando aparece un fallo en Arena, Arena se para después del fallo e interrumpe la optimización. Para evitarlo en Arena se selecciona en Arena en la pestaña Run de la barra superior de Arena, la opción de Setup y dentro de ella la de Run Control. Dentro de la ventana que aparece, se deselecciona la casilla de Pause After Warnings.



3° Iniciando Optquest

Para iniciar Optquest en la pestaña Tools del menú principal de Arena se escoge la opción de Optquest (Ver figura 20).

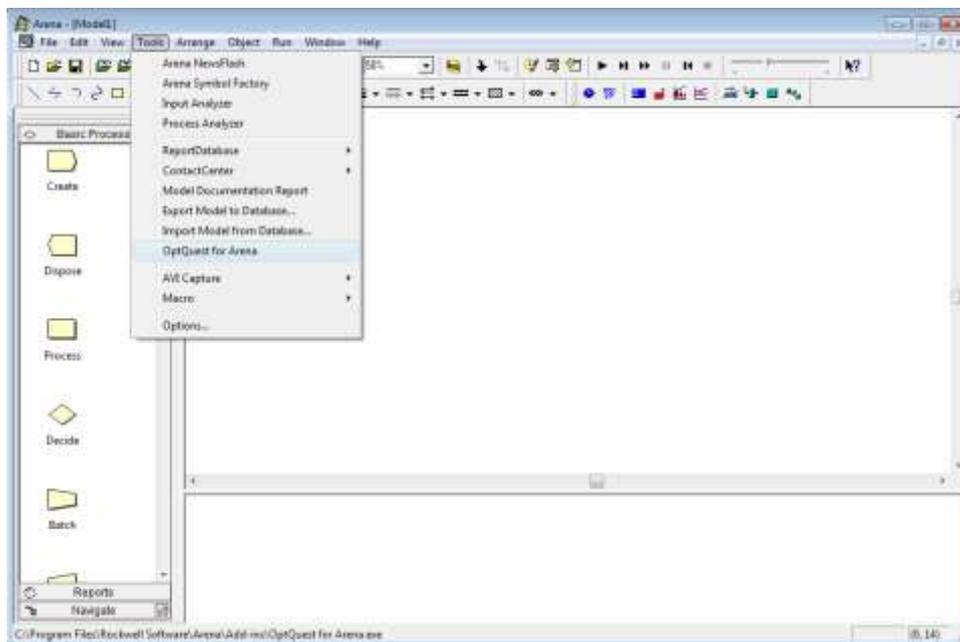


Figura 20. Abrir módulo de optimización Optquest

A continuación aparece la pantalla de inicio del optimizador. Véase figura 21.



Figura 21. Pantalla inicio de Optquest



Como se aprecia en la anterior figura, en la parte inferior de la pantalla inicial se encuentran dos botones. Si se hace clic sobre el botón de New Optimization o nueva optimización, se inicia una nueva optimización en base al modelo Arena que actualmente está abierto. Otra manera de iniciar la optimización, será seleccionando la opción (New) de la pestaña (File) del menú superior de Optquest.

Por otra parte si lo que se desea es abrir una optimización ya existente se hace clic en el botón Browse o en la opción Open de la pestaña File del menú superior.

El formato de archivo específico que crea Optquest es el (.opt) y es allí donde se guardan los cambios en la configuración de la optimización.

4º Establecer la optimización

- Seleccionar las variables de control

Una vez definidos las variables de control en el modelo de simulación, se seleccionan las que se optimizarán en Optquest. Se ha de tener en cuenta que Optquest cambiará los valores de los Controls en cada simulación, hasta encontrar los valores que hacen alcanzar el mejor objetivo. En algunos análisis se fijan valores de ciertas variables y se optimiza el resto.

El editor de las variables de control, se presenta en la pantalla de sumario que está en la parte derecha de la ventana general de Optquest. Esta pantalla aparece al seleccionar el nodo Controls en el árbol del área izquierda de la pantalla, como muestra la siguiente figura.

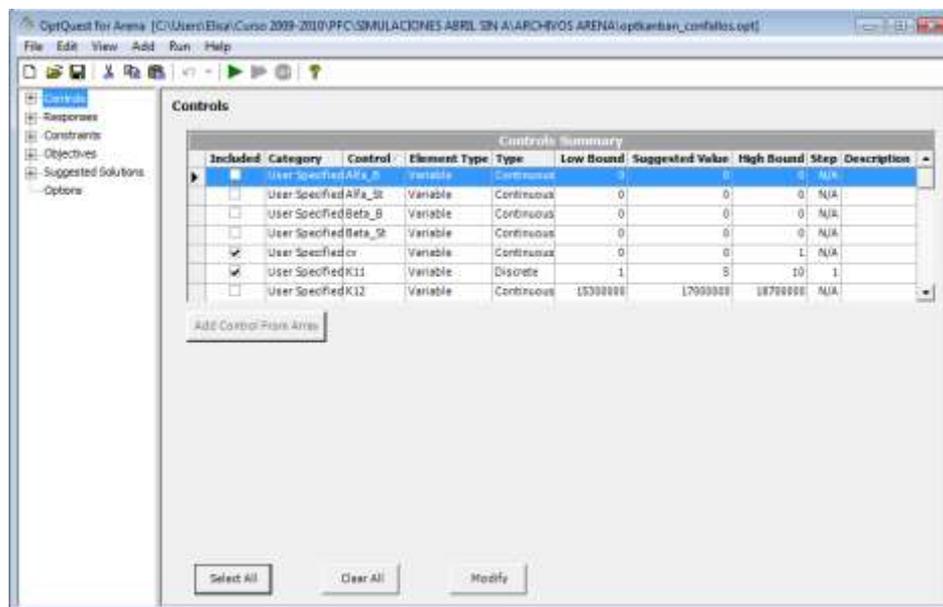


Figura 22. Ventana general Optquest (Editor variables de control)

Otra forma de visualizar la pantalla de Controls es mediante la selección de la opción de Controls, de la pestaña View del menú superior de la pantalla.



En la ventana Controls, aparece una tabla con varias columnas. En la tabla 5 se presentan las características de cada una de ellas.

Include (Incluido)	Presenta las variables del modelo en Arena. Si se selecciona la casilla en esta columna, Optquest optimizará la variable de control seleccionada. Se puede seleccionar haciendo clic sobre la casilla cuadrada y deseleccionar volviendo a hacer clic sobre ella. Si la casilla no está marcada Optquest no optimizará el control.
Category (Categoría)	Identifica si el nodo de control es un recurso ó una variable especificada por el usuario. Este campo es informativo, no se puede manipular.
Control (Control)	Muestra el nombre del Control definido en Arena. Este campo es informativo, no se puede manipular.
Element Type (Tipo de elemento)	Determina si es una variable o un recurso. Este campo es informativo, no se puede manipular.
Type (Clase)	Identifica si la variable control es continua, discreta, binaria o entera. Se puede cambiar en la ventana de edición.
Low Bound (Límite Inferior)	Es el límite inferior de la variable de control. Por defecto es un 10% menos que el valor especificado en el modelo Arena, pero se puede modificar. Tanto el límite inferior como el superior que especifica el usuario, limitan la búsqueda para que Optquest encuentre la solución óptima.
Suggested Value (Valor sugerido)	Es el valor inicial con el que Optquest inicia la optimización. Por defecto utiliza los valores del modelo Arena. Si los valores propuestos no se encuentran dentro de los límites específicos o no se encuentran en las restricciones, Optquest los ignora.
High Bound (Límite superior)	Es el límite superior de control. Por defecto es un 10% más que el valor de la variable de control especificada en el modelo. Se puede modificar el valor.
Step (Paso)	Refleja el tamaño del paso discreto. Se puede modificar en la ventana de edición.
Description (Descripción)	Permite introducir una descripción en forma de texto. Se puede modificar.

Tabla 5. Columnas de la tabla Controls Summary de Optquest



Para cambiar el valor del campo Type, Low Bound, High Bound ó Suggested value, de una de las variables de control, se despliega el árbol de la izquierda de la pantalla y se hace doble clic sobre el recurso o variable especificada por el usuario que se quiera modificar. Otra posibilidad es directamente dentro de la ventana Controls Summary, hacer doble clic sobre una variable de control. Aparece entonces la ventana de edición, la cual se muestra a continuación.

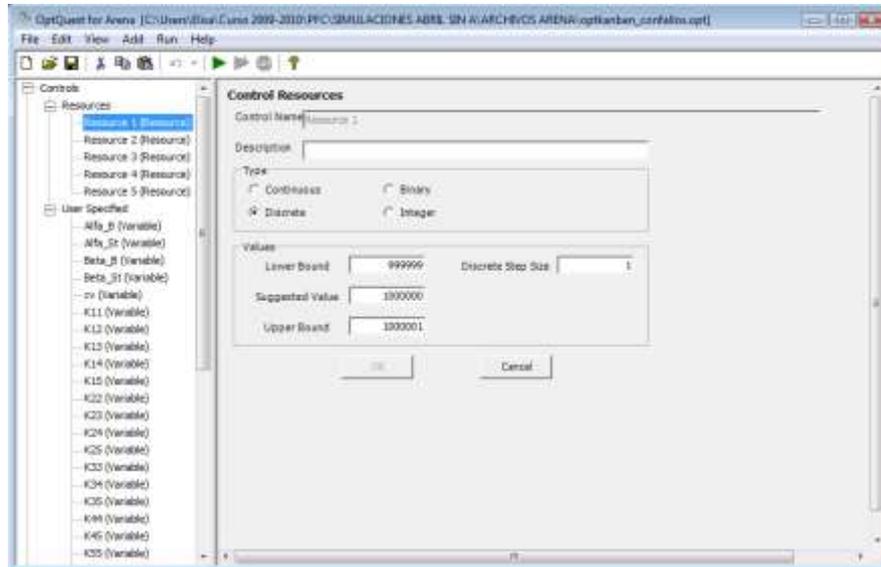


Figura 23. Ventana general Optquest (Editor variables de control)

Si se hace clic en la cabecera de una columna, las filas se reordenarán en orden ascendente o descendente en función del parámetro que se indica en el título de la columna. Esta opción es muy útil, cuando el número total de posibles variables de control, es muy grande en comparación con las seleccionadas para la optimización.

Includ	Category	Control	Element Type	Type	Low Bound	Suggested Value	High Bound	Step	Description
<input checked="" type="checkbox"/>	User Specified	cv	Variable	Continuous	0	0	1	N/A	
<input checked="" type="checkbox"/>	User Specified	K11	Variable	Discrete	1	5	10	1	
<input checked="" type="checkbox"/>	User Specified	K22	Variable	Discrete	1	10	10	1	
<input checked="" type="checkbox"/>	User Specified	K33	Variable	Discrete	1	15	20	1	
<input checked="" type="checkbox"/>	User Specified	K44	Variable	Discrete	1	20	20	1	
<input checked="" type="checkbox"/>	User Specified	K55	Variable	Discrete	1	25	25	1	
<input checked="" type="checkbox"/>	User Specified	p_R	Variable	Continuous	0	0	100	N/A	
<input checked="" type="checkbox"/>	User Specified	Sistema	Variable	Integer	0	0	5	1	
<input checked="" type="checkbox"/>	User Specified	t_Fallas	Variable	Continuous	0	0	1000	N/A	
<input checked="" type="checkbox"/>	User Specified	t_repara	Variable	Continuous	0	0	111,11	N/A	
<input type="checkbox"/>	User Specified	Alfa_B	Variable	Continuous	0	0	0	N/A	
<input type="checkbox"/>	User Specified	Alfa_St	Variable	Continuous	0	0	0	N/A	
<input type="checkbox"/>	User Specified	Beta_B	Variable	Continuous	0	0	0	N/A	
<input type="checkbox"/>	User Specified	Beta_St	Variable	Continuous	0	0	0	N/A	
<input type="checkbox"/>	User Specified	K12	Variable	Continuous	15300000	17000000	18700000	N/A	
<input type="checkbox"/>	User Specified	K13	Variable	Continuous	15300000	17000000	18700000	N/A	
<input type="checkbox"/>	User Specified	K14	Variable	Continuous	23400000	26000000	28600000	N/A	
<input type="checkbox"/>	User Specified	K15	Variable	Discrete	1	5	40	1	
<input type="checkbox"/>	User Specified	K23	Variable	Continuous	21600000	24000000	26400000	N/A	
<input type="checkbox"/>	User Specified	K24	Variable	Continuous	17100000	19000000	20900000	N/A	
<input type="checkbox"/>	User Specified	K25	Variable	Continuous	20700000	23000000	25300000	N/A	
<input type="checkbox"/>	User Specified	K34	Variable	Continuous	16200000	18000000	19800000	N/A	
<input type="checkbox"/>	User Specified	K35	Variable	Continuous	16200000	18000000	19800000	N/A	
<input type="checkbox"/>	User Specified	K45	Variable	Continuous	14400000	16000000	17600000	N/A	
<input type="checkbox"/>	User Specified	N	Variable	Continuous	13500000	15000000	16500000	N/A	

Figura 24. Ventana resumen de Variables de Control



Para seleccionar las variables de control en primer lugar, se marcan las casillas de la columna Included en la tabla de Control Summary, de las variables de control que se quieren optimizar. Por defecto no están seleccionadas.

Para modificar una variable de control seleccionar la fila de la tabla Controls Summary donde se encuentra y hacer doble clic sobre ella, seleccionar el nodo del árbol y hacer doble clic o seleccionar la fila y hacer clic en el botón Modify. Se pueden realizar cambios en: los límites superior o inferior, el campo de valor sugerido, tipo y descripción. Cuando se completan los cambios de la fila seleccionada, para guardar los cambios, se hace clic sobre el botón OK, situado en la parte inferior de la pantalla. Para salir de la pantalla Editar sin guardar los cambios hacer clic sobre el botón Cancel.

- Identificar las variables salidas de la función objetivo (responses) y en las expresiones de las restricciones.

Las respuestas son las salidas de la simulación y no pueden modificarse en Optquest. El nodo Responses del árbol de Optquest muestra todas las respuestas disponibles en el modelo Arena. Cuando se selecciona el nodo de respuesta en el árbol, aparece una cuadrícula a la derecha, que muestra todas las respuestas posibles en el modelo Arena. Las respuestas se pueden utilizar para crear expresiones de objetivo y restricciones. Para incluir las respuestas en el modelo de optimización, se selecciona la casilla correspondiente en la columna de Included en la tabla de Response Summary.

- Especificar las restricciones.

Definiendo las relaciones entre controls y respuestas se puede incrementar la eficiencia para encontrar la solución óptima. El editor de restricciones permite añadir restricciones lineales y no lineales, representadas en términos de las variables de control que se han seleccionado para la optimización. Como se estudio anteriormente, existen dos tipos de restricciones: lineales y no lineales.

Las restricciones lineales, definen relaciones lineales entre variables de control. Las restricciones lineales, son expresiones matemáticas de términos lineales que se suman o se restan. Éstas se pueden evaluar sin realizar la simulación. Optquest descarta las soluciones que violan las restricciones lineales.

Las restricciones no lineales son expresiones matemáticas que contienen una respuesta o término no lineal. Estas restricciones requieren que se realice la simulación para determinar la factibilidad de la restricción.

El editor de restricciones te permite construir expresiones matemáticas de restricciones para emplearlas en la optimización.

Cuando se extiende el nodo Constraints del árbol de Optquest, se visualiza en cada subnodo, cada una de las restricciones que se han definido en el modelo. Seleccionando el nodo correspondiente se puede modificarla restricción en la ventana Constraints Summary que aparece a la derecha de la pantalla.



Para añadir una nueva restricción en primer lugar, se hace clic derecho en el nodo de Constraints y se selecciona Add New o se hace clic en el botón Add situado en la parte inferior de la pantalla. El editor de restricciones mostrará los Controls y Responses que se han incluido en la optimización.

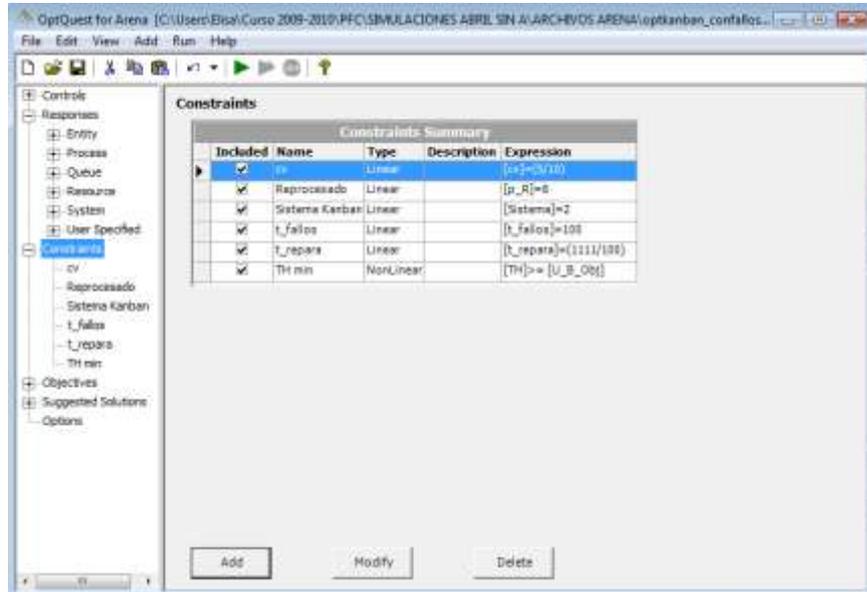


Figura 25. Ventana general Optquest (Resumen de restricciones)

En segundo lugar, en los campos en la parte inferior de la ventana, introducir el nombre de la restricción (Name), una descripción (Description) y una expresión matemática (Expression), que combinará las variables de control y las respuestas que hay en el árbol.

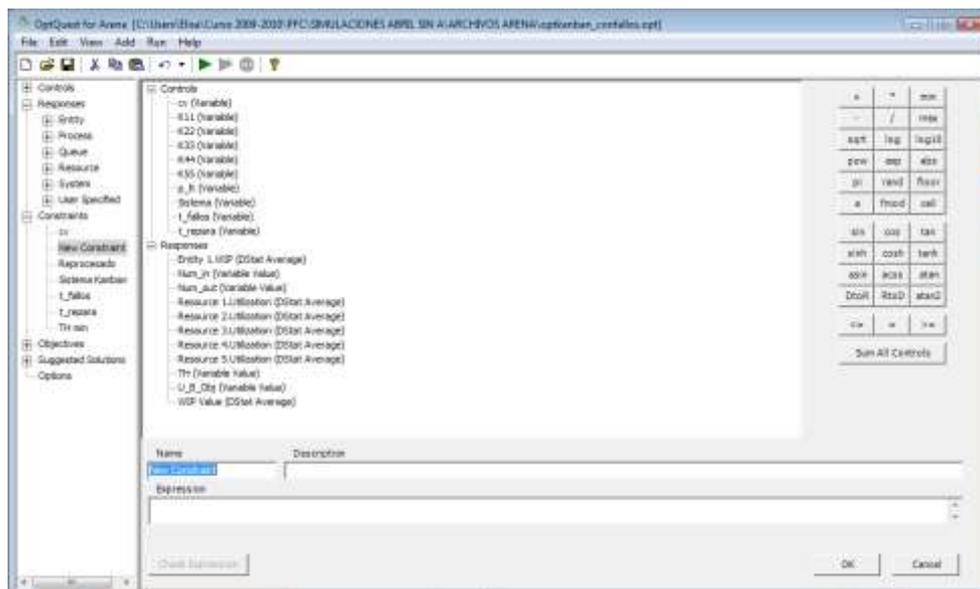


Figura 26. Ventana general Optquest (Editor de restricciones)



Cuando se hace clic en una variable de control o respuesta del árbol de la ventana de la derecha, automáticamente se añade a la expresión que se está elaborando. En la parte superior derecha, se muestran todas las funciones que se pueden emplear para crear las expresiones de las restricciones. Simplemente se ha de hacer clic sobre una de ellas para utilizarla. Si un control o una respuesta no aparecen en el árbol superior, habrá que volver atrás y seleccionar la casilla Included de la ventana Response Summary o Controls Summary. El botón Sum All Controls, crea automáticamente una expresión que es suma de todas las variables de control. Además se puede emplear el operador lógico “or” para combinar dos o más expresiones en una restricción simple.

Posteriormente se hace clic en el botón inferior Check Expression para comprobar la validez de la expresión. En el caso de existir un error, aparece una pantalla con un mensaje de error y posteriormente un símbolo de exclamación a la derecha del cuadro Expression. Si se sitúa el cursor sobre el símbolo de error, aparece una aclaración del tipo de error que existe, en forma de cuadro de texto.

Y por último se guardan los datos de la edición de las restricciones, haciendo clic en OK. Las expresiones se deben validar antes de aceptar. Si la expresión no es válida aparecerá el fondo de la fila de la tabla Constraint Summary correspondiente, de color amarillo.

Si no se quiere añadir ninguna restricción al modelo de optimización, se deja vacío el editor de restricciones.

En algunas situaciones puede ser útil saber qué efecto tiene una restricción en la solución óptima y que ocurriría si se relajase o ajustase. Para ello, Optquest permite definir una restricción con límites variables. Los límites se colocan a la derecha de la expresión, separados por una coma y no es necesario que se incrementen. Se puede especificar el número de posibles límites que se quiera. Ejemplo: $x_1 * 3 * x_2 \geq 400, 500, 900$. La optimización comienza con el primer valor y se realizan un número de simulaciones y se determina una buena solución para este límite. Entonces, Optquest toma el siguiente límite y vuelve a realizar la operación anterior. Se continúa de este modo hasta que se comprueban todos los límites.

- Especificar el objetivo.

Arena te permite definir más de un objetivo, mientras Optquest sólo te permite definir un objetivo para la optimización, seleccionando la función objetivo. Los otros objetivos se ignoran y no afectan a la optimización. El nodo Objectives u objetivos en el árbol principal de Optquest contiene un subnodo para cada objetivo que se define.

Antes de seleccionar un objetivo, primero se ha de asegurar que se han seleccionado todas las casillas de cualquier control o respuesta deseada en la ventana Controls y Responses Edit respectivamente.

Posteriormente se ha de seleccionar Add>Objective del menú o hacer clic derecho en el nodo Objectivos y seleccionar AddNew o hacer clic en el botón Add de la parte derecha de la pantalla.



Una vez se abre el editor de Objetivo, se puede construir la función objetivo que se utilizará para la optimización.

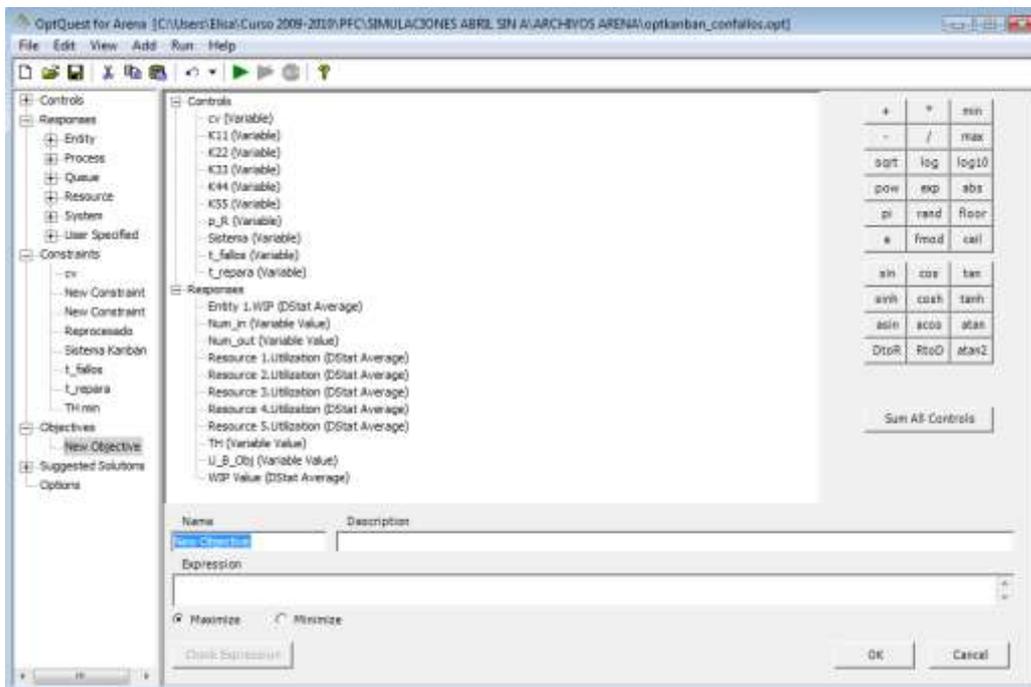


Figura 27. Ventana general Optquest (Editor de Objetivos)

Los campos de edición incluyen Name, para asignar el nombre principal del objetivo y Description, para añadir un comentario opcional que se guardará junto al objetivo. El árbol en la ventana de edición, muestra todos los controls y responses que se pueden incluir en la expresión del objetivo. Para añadir una variable, hacer un clic sobre el nodo del árbol de la ventana Edit que interese, e inmediatamente aparecerá como parte de la expresión. Los operadores que se pueden emplear están a la derecha de la ventana de edición. Al igual que con las restricciones, la opción Sum All Controls, crea automáticamente una expresión donde aparecen sumando todas las variables control.

Una vez definida la expresión, es necesario hacer clic en el botón CheckExpression para verificar la expresión. Los errores se presentan en una pantalla de error y un indicador del error aparecerá junto a la expresión.

Es conveniente asegurarse de seleccionar correctamente, Minimize (minimizar) o Maximize (maximizar) para definir la finalidad del objetivo. Por defecto aparece seleccionada la opción Maximize.

Por último, hacer clic en OK para aceptar los cambios. Si la expresión es inválida, aparece en amarillo en la tabla de Objective Summary de la ventana de la derecha.



- Seleccionar las opciones de optimización.

Se parte de la selección del nodo Options en el árbol principal. A la derecha aparece una ventana del OptionsEditor o editor de opciones. Ésta ventana está dividida en cuatro secciones: Stop Options, Tolerance, Replications per simulation y Solution Log.

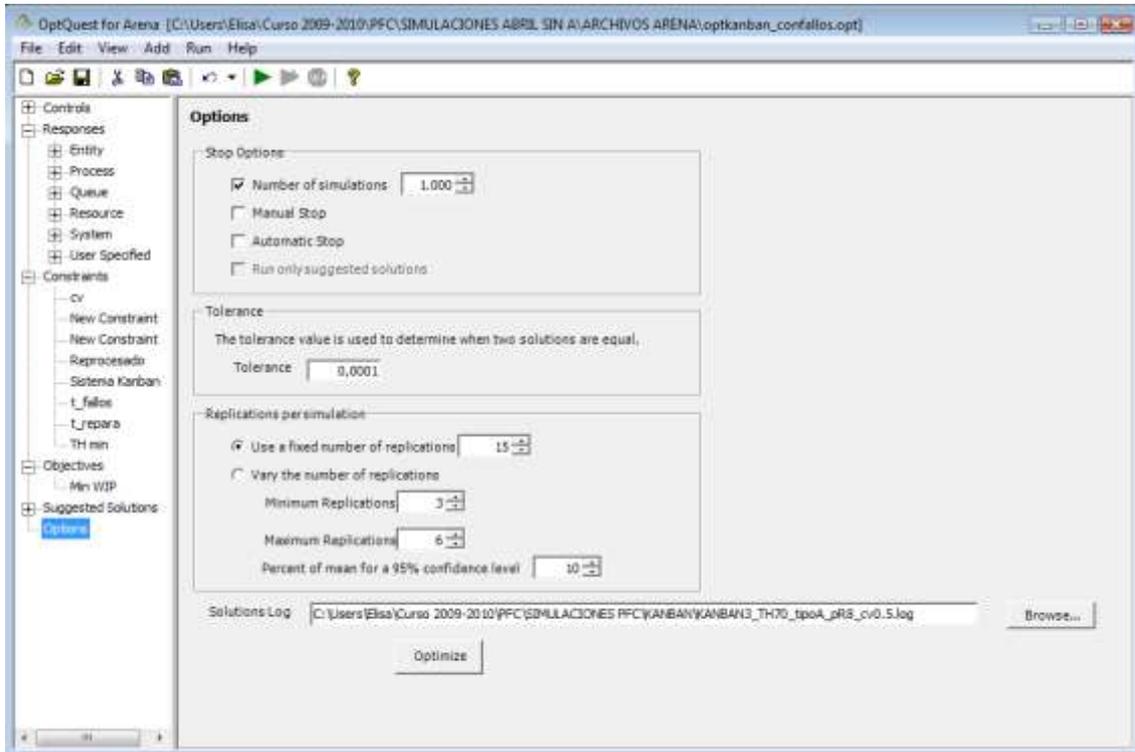


Figura 28. Ventana general Optquest (Opciones de optimización)

Optimization Stop Options, es la primera sección y corresponde a las opciones de parada de la optimización. Permite controlar cuando durará la simulación. Se puede seleccionar más de un modo de parada. La optimización se parará cuando se satisfaga una de las condiciones de parada. Al menos se deberá seleccionar una opción de parada. En la siguiente tabla, se recogen y explican los posibles criterios de parada de la optimización.

<p>Number of simulations (Número de simulaciones)</p>	<p>Si se selecciona, es necesario determinar el número de simulaciones que se van a realizar. Este número se determina mediante la experiencia. No deberá ser muy elevado, para no alargar innecesariamente el tiempo de simulación. Por defecto está en 100 simulaciones.</p>
	<p>Permite parar la simulación manualmente</p>



Manual stop (Parada manual)	seleccionando Run>Stop. Se puede terminar siempre la optimización manualmente incluso si se ha seleccionado otro criterio de parada.
Automatic stop (Parada automática)	Para la simulación automáticamente cuando no encuentra mejora significativa en el mejor valor después de 100 simulaciones. Los valores de tolerancia definen el criterio para determinar cuando las soluciones se consideran iguales.
Run only suggested solutions (Realizar solo las soluciones propuestas)	Esta opción está permitida, si existen lo que se denomina Suggested Solutions o soluciones propuestas. Cuando se marca esta opción, sólo se comprueban las soluciones sugeridas. El número entre paréntesis indica el número de Suggested Solutions existentes. Cuando se selecciona, no hay ninguna otra opción de parada permitida. Y Optquest no evaluará otras soluciones que no sean las sugeridas.

Tabla 6. Criterios de parada empleados en Optquest

La segunda sección es la de Tolerance, en español Tolerancia. Este valor se emplea para determinar cuando dos soluciones son iguales. El valor por defecto es 0,0001.

La tercera sección, corresponde al número de réplicas por simulación. En ella se puede especificar el número de veces que el modelo se ejecutará en cada simulación. Existen dos posibilidades que se presentan en la tabla 7.

Use a fixed number of replications (Utilizar un número fijo de réplicas)	Cuando se selecciona esta opción, Optquest instruye a Arena para ejecutar el número indicado de réplicas por simulación.
Vary the number of replications (Variar el número de réplicas)	Cuando se selecciona esta opción, se utilizan los números dados como límites en el número de réplicas por simulación. Permite a Optquest comprobar la significancia entre el valor actual de la simulación y el mejor valor encontrado en las simulaciones. El objetivo, eliminar soluciones son utilizar mucho tiempo en ellas. La comprobación de la solución inferior “inferior solution” construye un intervalo de confianza del 95% alrededor del valor



	medio actual. Entonces, comprueba si el mejor valor está dentro de este intervalo. Si no es así, la solución deberá ser inferior y la simulación finaliza sin más réplicas. Por el contrario, si cae en el intervalo, Arena ejecuta otra réplica y lo comprueba de nuevo. Arena continúa corriendo réplicas a lo largo del intervalo para incluir el mejor valor, hasta el máximo número especificado.
--	--

Tabla 7. Opciones en Optquest del número de réplicas por simulación.

La cuarta y última sección corresponde al registro de los resultados obtenidos. Para ello es necesario, especificar la ruta, donde quedarán grabados los archivos de resultados tras la optimización. La información incluye cada solución como un intento, el valor de la función objetivo y de las restricciones. Se ha de seleccionar el botón Browse e identificar el nombre del archivo y la localización. El archivo por defecto es Optquest.log en el directorio temporal del programa.

Antes de pasar a ejecutar el optimizador, es necesario aclarar lo que significa y para que sirve el nodo de Suggested Solutions. Las suggested solutions, en español soluciones sugeridas, son soluciones que se cree que pueden estar cercanas a la óptima. Su uso, puede ayudar a disminuir el tiempo de encontrar la solución óptima. Estas soluciones propuestas, serán las primeras soluciones que deben ser evaluadas para encontrar la mejor solución. Se pueden introducir las soluciones que el analista considere o seleccionarlas de una optimización previa.

Para añadir una suggested solution, primero se hace clic derecho sobre el nodo Suggested Solution y se selecciona AddNew. El editor de Suggested Solutions mostrará las variables de control y los valores que han sido incluidos en la optimización. Una vez abierto el editor, se puede introducir un nombre a una solución y modificar los valores sugeridos en la tabla. Si el valor que se introduce viola los límites de control, la celda se volverá amarilla. Antes de guardar, hay que comprobar la solución mediante el botón Check Solution. De este modo se asegura, que todos los valores estén dentro de los límites de las variables de control y no se viole ninguna restricción. Cuando se ha comprobado, se salva la solución, pulsando OK y el programa vuelve a la ventana de Suggested Solutions Summary. Ésta ventana, muestra todas las soluciones propuestas que se han definido o seleccionado.

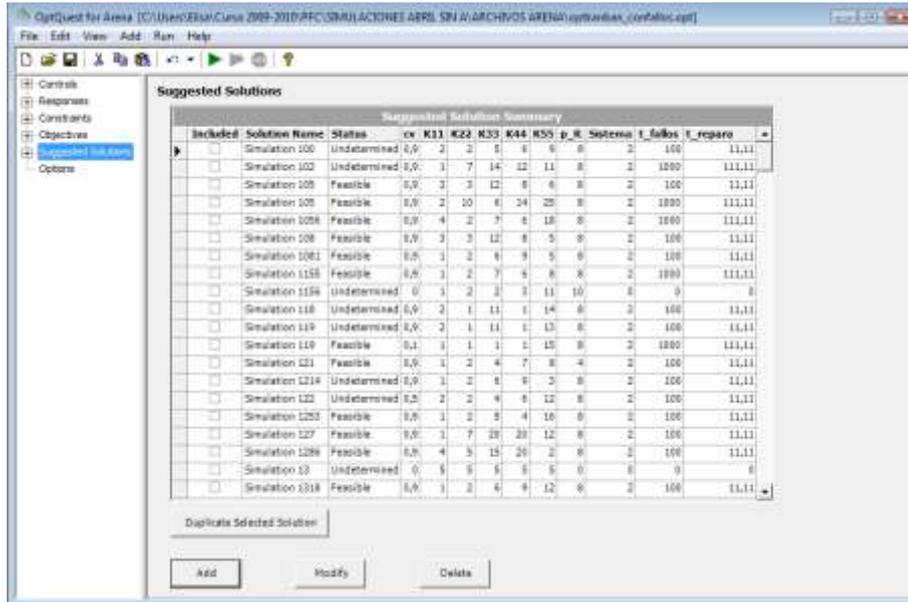


Figura 29. Ventana general Optquest (Soluciones sugeridas)

Las soluciones se pueden editar o borrar, haciendo clic en el nombre de la solución en el árbol o haciendo clic en los botones Modify o Delete respectivamente. Por otra parte, las soluciones se pueden incluir o excluir de la optimización seleccionando o deseleccionando la casilla de la columna de Included.

El botón de Duplicate Selected Solution, permite hacer una copia de la fila seleccionada. Entonces, se pueden editar uno o más de las variables de control para crear una determinada solución.

Después de ejecutar una optimización, aparece en el árbol el nodo de Best Solutions, que permite seleccionar una o más soluciones y salvarlas como soluciones sugeridas.

Es necesario saber, que Optquest comprueba estas soluciones antes de empezar la simulación. De este modo, la optimización no comenzará si hay soluciones propuestas seleccionadas, que son no factibles o con valores malos. Entonces, se puede corregir la solución o no incluirla en la solución del problema.

5° Ejecutar el optimizador

Una vez se completan las opciones de optimización se ha de poner en marcha la optimización. Los errores deberán ser comprobados y subsanados antes de comenzar la simulación. Si hay errores, aparecerán una o más ventanas de error y la simulación no comenzará hasta que no sean eliminados. Se puede buscar los errores examinando las filas para cada uno de los nodos. Los errores aparecerán como filas rojas o amarillas en las tablas de sumario.

Los comandos para comenzar y parar el proceso de optimización se encuentran en el menú Run. Además se pueda acceder a estas opciones a partir de los iconos correspondientes de la barra de herramientas. Estos iconos se presentan en la siguiente tabla.



	Start (Iniciar)	Empieza una nueva optimización. No está disponible cuando una optimización se está ejecutando.
	Stop (Parar)	Para la optimización actual. Está disponible cuando está ejecutándose. Cuando se decide parar la optimización, no se puede continuar la optimización.

Tabla 8. Iconos inicio y parada de optimización en Optquest

Una vez se comienza la optimización, se añade un nuevo nodo al árbol y el progreso se optimización se muestra en la ventana de la derecha.

Mientras una optimización se está ejecutando no se puede trabajar en Arena o hacer cambios en Optquest, pero se puede trabajar en otros programas. Mientras se está ejecutando la optimización no se ha de cerrar Optquest.

Una vez iniciada la optimización, aparece la ventana de Optimization. La ventana de optimización se divide en tres partes. En la parte superior se muestran tres tablas correspondientes al objetivo, las variables de control y las restricciones. En la tabla 8 se detalla el contenido de cada una de ellas.

<p>Objective grid (Tabla de objetivo)</p>	<p>Esta tabla muestra el mejor valor y el valor actual para el objetivo. El título de la tabla muestra si el objetivo es maximizar o minimizar.</p> <p>El mejor valor de la fila muestra el mejor valor objetivo encontrado hasta el momento. El valor actual de la fila muestra el valor de la última solución comprobada.</p> <p>La columna Status indica si la solución satisface todas las restricciones o violan una o más restricciones.</p>
<p>Controls grid (Tabla de variables de control)</p>	<p>Esta tabla muestra el valor de cada variable de control.</p> <p>La columna Best value muestra los valores para la mejor solución encontrada.</p> <p>La columna de Current Value muestra la última solución que fue evaluada.</p>
<p>Constraints grid (Tabla de restricciones)</p>	<p>Cuando se definen las restricciones se mostrará esta tabla. Todos los valores en esta tabla pertenecen a la mejor solución.</p> <p>En la columna Type indica si es una restricción lineal o no lineal.</p> <p>La columna Status indica si la mejor solución era factible o no factible.</p>

Tabla 9. Tablas de la ventana Options



La parte inferior de la ventana de optimización muestra un gráfico que traza la trayectoria de la búsqueda de la mejor solución. El gráfico de valores del objetivo, muestra la velocidad a la que el mejor objetivo ha cambiado durante la búsqueda. Se muestra como se dibujan los mejores valores del objetivo, en función del número de soluciones que se han realizado. Las soluciones no factibles se dibujan como líneas rojas discontinuas. Las soluciones factibles se dibujan mediante líneas verdes continuas.

Cuando ha terminado la optimización o ha sido parada por el usuario, el valor actual de las celdas de todas las tablas se borran.

Cuando se completa la optimización, se añade al árbol el nodo de Best Solution y en la derecha se muestran las 25 mejores soluciones. Se puede seleccionar visualizar más soluciones cambiando el número en el grupo View Solutions y haciendo clic en el botón de Refresh. Otra manera de seleccionar el detalle de una solución es marcando la casilla Select y haciendo clic en el botón View. La primera fila de la tabla de Best Solutions corresponderá a la mejor solución, la segunda fila será la segunda mejor y así sucesivamente.

La columna Simulación identifica la simulación que generó la solución.

La columna Status indica si la solución cumple o no las restricciones. Optquest asigna siempre un mayor valor para las soluciones que tienen restricciones factibles, por tanto las tablas siempre muestran las soluciones factibles antes de las no factibles. Incluso si el valor del objetivo para una solución con restricciones no factibles es mejor que una solución con restricciones factibles, Optquest asigna un mayor valor a las últimas.

Si el usuario está ejecutando una variación del número de réplicas, la columna Confidence se muestra en la tabla. Esta columna indica si el intervalo de confianza definido en la forma Options se conoce o no. La columna replications muestra cuantas réplicas se han realizado. Las columnas siguientes muestran los valores de las variables de control para cada solución.

A continuación se explican las opciones de la ventana Best Solutions.

Select All and Clear All buttons	Las soluciones seleccionadas se pueden añadir como suggested solutions o escritas en un fichero. El botón Select All selecciona todas las soluciones que se muestran en la tabla. El botón Clear All deselectan todas las soluciones mostradas en la tabla.
View button	El botón View muestra información detallada para la solución destacada actual. La información detallada incluye los valores calculados para todas las restricciones, la factibilidad de cada restricción y los valores de las respuestas para Arena se han de incluir.



Advanced- More Solutions button	Si se requiere que Optquest genere más soluciones, hay que introducir el número de nuevas soluciones que se quieren generar y hacer clic en el botón More Solutions.
Advanced-Refine Solutions button	Si se quiere refinar las soluciones, se consigue mediante la ejecución de un mayor número de réplicas o mediante la ejecución de “Rank y Selection”, haciendo clic en el botón Refine Solutions. Aparece un formulario donde se puede seleccionar el tipo de refinamiento que se va a ejecutar.
Save Solutions-Add to Suggested button	El botón Add to Suggested añadirá cada solución seleccionada, marcada la casilla, al grupo de suggested solutions o soluciones sugeridas. Ésta solución se empleará en la siguiente vez, cuando se esté ejecutando la optimización. Para seleccionar una solución, se hace clic en la columna Select. Haciendo clic por segunda vez se deselecciona la casilla. Si se quiere seleccionar todas las soluciones, se hace clic en el botón Select All. Y si se quieren deseleccionar se hace clic en el botón Clear All.
Save Solutions-Write to File button	Si se hace clic en el botón Write to File, se guardan las soluciones seleccionadas en un fichero de texto.

Tabla 10. Opciones ventana Best Solutions

Desde Optquest no se puede alterar el modelo Arena, se puedan utilizar las soluciones en el fichero para hacer cambios manuales en el modelo Arena.

6º Interpretar los resultados.

Los resultados se interpretan en función de cada problema específico, teniendo en cuenta las mejores soluciones obtenidas por el optimizador Optquest.

7º Refinar la solución.

Optquest te permite refinar un grupo de soluciones, ejecutando réplicas adicionales o ejecutando el análisis “Rank and Selection”. Si se hace clic en el botón Refine Solutions



en la ventana de Best solutions, se crea el nodo Refine Solutions. En la ventana de edición se puede seleccionar las opciones para el refinamiento de la optimización.

Se pueden ejecutar réplicas adicionales a las n soluciones. De este modo, se puede realizar una optimización inicial especificando un bajo número de réplicas y posteriormente ejecutar más réplicas en las mejores soluciones. Las réplicas adicionales quizás cambien el valor objetivo, el cual quizás modificará la mejor solución. Para ejecutar más réplicas, se hace clic en el botón de opción Run more replications. Y se introduce un número adicional de réplicas que se quieren ejecutar. Si la optimización inicial tenía un número variable de réplicas, el límite superior y el inferior se verán incrementados por la cantidad introducida. Se especifica cuántas soluciones se quiere que se amplíen las réplicas, en el grupo de soluciones. Optquest tomará las primeras soluciones para evaluarlas de nuevo.



Figura 30. Ventana general Optquest (Refinar soluciones)

Cuando se está ejecutando la optimización con este refinamiento seleccionado, el progreso de las réplicas adicionales se muestra en el panel de Replication Summary que reemplaza la ventana original de progreso de la optimización. Mientras las réplicas adicionales están siendo ejecutadas, la fila para la solución actual se destaca y se ve como incrementa el contador de réplicas.

Rank and Select es un algoritmo sofisticado, que ejecuta más réplicas de soluciones que se han seleccionado para refinarlas, de la lista de las mejores soluciones. Este algoritmo elimina las soluciones de un grupo de soluciones, hasta que hay un porcentaje de posibilidad definido, de que las restantes soluciones estén la mayoría en el valor de la zona de indiferencia de la mejor de las soluciones candidatas.

Se pone en marcha haciendo clic en el botón Run Rank y Select. Se introduce el máximo número de réplicas a ejecutar mediante el algoritmo y se especifica cuántas soluciones se desean en el grupo de soluciones mediante el tecleo en una cuenta. Las soluciones que se encuentren en la parte superior de la lista que elabora Optquest, estarán en conjunto de soluciones de Rank and Select. Finalmente se selecciona la zona de indiferencia y probabilidad.



Cuando se está ejecutando la optimización con la opción Run Rank and Select escogida, el progreso del algoritmo es muestra en la ventana de Rank and Select Summary que reemplaza la ventana original de progreso de la optimización.