



EMULACIÓN DE UNA COLUMNA DE DESTILACIÓN BINARIA CONTÍNUA

DIEGO FERNANDO FRANCO OCAMPO



**UNIVERSIDAD DEL VALLE
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
PROGRAMA DE POSGRADO EN INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
-PPIEE-
MAESTRÍA EN INGENIERÍA CON ÉNFASIS EN AUTOMÁTICA
GRUPO DE INVESTIGACIÓN EN CONTROL INDUSTRIAL
-GICI-
SANTIAGO DE CALI
2014**

EMULACIÓN DE UNA COLUMNA DE DESTILACIÓN BINARIA CONTÍNUA

DIEGO FERNANDO FRANCO OCAMPO

**Trabajo de Investigación para optar al título de Magíster en Ingeniería con énfasis en
Automática**

**Director
Ing. Edinson Franco Mejía, PhD.**

**UNIVERSIDAD DEL VALLE
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
PROGRAMA DE POSGRADO EN INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
-PPIEE-
MAESTRÍA EN INGENIERÍA CON ÉNFASIS EN AUTOMÁTICA
GRUPO DE INVESTIGACIÓN EN CONTROL INDUSTRIAL
-GICI-
SANTIAGO DE CALI
2014**

A mi familia,
A Laurita y Héctor,
A mis sobrinos.

AGRADECIMIENTOS

- Al profesor Edinson Franco Mejía, por sus orientaciones, apoyo y dedicación en la realización de este proyecto.
- Agradecer a mis profesores de Posgrado por sus enseñanzas y valiosos aportes.
- Al profesor Jesús Alberto Gonzáles por su asesoría y buenos deseos.
- Al Programa de Posgrado en Ingeniería Eléctrica y Electrónica y su personal administrativo.
- A Luis Didier Arias Osorio por su colaboración técnica, valiosa en la culminación de este trabajo.
- A mi familia por su apoyo incondicional y paciencia, especialmente, mi madre, hijos y sobrinos por el tiempo que estuve ausente.
- A mis hermanos Nancy y Alexander por su amor, aliento y colaboración en todo momento.
- A Yolanda Domínguez Valverde, amiga y jefe, por brindarme su confianza y apoyo durante todo este proceso.
- Por último, mi gratitud a todas aquellas personas que de una u otra forma han colaborado en la realización de este trabajo.

CONTENIDO

	Pág.
1. INTRODUCCIÓN	10
2. MARCO CONCEPTUAL.....	12
2.1 MARCO TEÓRICO	12
2.1.1. Destilación.....	12
2.1.2. Simulación y emulación	15
2.2. REVISIÓN BIBLIOGRÁFICA	16
2.3. MODELADO DE COLUMNAS DE DESTILACIÓN BINARIA CONTÍNUA ..	20
3. METODOLOGÍA	30
4. DISEÑO E IMPLEMENTACIÓN.....	32
4.1. DEFINICIÓN DE LAS VARIABLES INVOLUCRADAS EN LA EMULACIÓN ...	32
4.2. DISEÑO E IMPLEMENTACIÓN.....	33
5. RESULTADOS Y VALIDACIÓN.....	39
6. CONCLUSIONES, RECOMENDACIONES Y FUTUROS TRABAJOS	52
6.1. CONCLUSIONES	52
APORTES	53
6.2. RECOMENDACIONES.....	53
6.3. FUTUROS TRABAJOS	54
REFERENCIAS BIBLIOGRÁFICAS.....	54
A.A. PLATAFORMA DE PROCESAMIENTO DISTRIBUIDO PARA LA EMULACIÓN DE PROCESOS EN INGENIERÍA – PERI.....	58
A.B. DESCRIPCIÓN DE PROYECTOS UTILIZADOS EN EL EMULADOR.....	59
A.C. MÉTODO PROPUESTO PARA LA EMULACIÓN DE COLUMNAS DE DESTILACIÓN BINARIAS CONTINUAS.....	61

LISTA DE TABLAS

	Pág.
Tabla 1. Variables internas columna de destilación binaria continua.....	32
Tabla 2. Entradas, columna de destilación binaria continua.....	33
Tabla 3. Perturbaciones, columna de destilación binaria continua.....	33
Tabla 4. Condiciones iniciales, columna de destilación binaria continua.....	33
Tabla 5. Parámetros de la columna.....	40
Tabla 6. Entradas y perturbaciones de la columna.....	40
Tabla 7. Composiciones y retenciones iniciales.....	40
Tabla 8. Parámetros de la columna.....	46
Tabla 9. Entradas y perturbaciones de la columna.....	47
Tabla 10. Composiciones y retenciones iniciales.....	47
 Tabla C-1. Variables internas columna de destilación binaria continua.....	 63
Tabla C-2. Entradas, columna de destilación binaria continua.....	64
Tabla C-3. Perturbaciones, columna de destilación binaria continua.....	64
Tabla C-4. Condiciones iniciales, columna de destilación binaria continua.....	64

LISTA DE FIGURAS

	Pág.
Figura 1. Configuración general de una columna de destilación.	13
Figura 2. Diagrama de bloques del emulador columna de destilación binaria.	35
Figura 3. Diagrama de flujo.	37
Figura 4. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	41
Figura 5. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.3$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	41
Figura 6. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.3$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	42
Figura 7. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.01$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	42
Figura 8. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.05$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	43
Figura 9. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.3$. jRTAI-Lab y MATLAB.....	43
Figura 10. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.7$. jRTAI-Lab y MATLAB.....	44
Figura 11. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.3$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	48
Figura 12. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.3$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	48
Figura 13. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.01$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	49
Figura 14. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.05$, $ZF = 0.5$. jRTAI-Lab y MATLAB.....	49
Figura 15. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.3$. jRTAI-Lab y MATLAB.....	50

Figura 16. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.7$. jRTAI-Lab y MATLAB.....	50
--	----

Figura C-1. Presentación de la carga de los módulos de RTAI-Lab en una consola de LINUX	62
Figura C-2. Contenido del archivo m_data.sce.....	65
Figura C-3. Habilidad del toolbox RTAI en la barra de herramientas.....	66
Figura C- 4. Cuadro de diálogo “ScicosLab dialog” correspondiente al C RTAI Block. ...	68
Figura C-5. Superbloque y cuadro de diálogo “Set Block Properties”	70
Figura C-6. Contenido del súper-bloque.....	71
Figura C-7. Cuadro de diálogo “Set Block Properties” después de elegir la opción “RTAI-CodeGen”.....	72
Figura C-8. Mensaje en la ventana de comandos de ScicosLab, finalizado el proceso correctamente.	73
Figura C-9. Ejecutable generado /home/realtimevm.	74
Figura C-10. SBT como script de configuración creado.....	75
Figura C-11. Líneas más importantes del archivo creado.....	76
Figura C-12. Respuesta gráfica generada en la emulación.	78
Figura C-13. Respuesta del proceso cuando se cambia zF (Value[7]) de 0.7 a 0.3.....	79

RESUMEN

En este documento se presentan los resultados del trabajo de investigación de la emulación de una columna de destilación binaria continua, importante en los procesos de enseñanza-aprendizaje o para el desarrollo de investigaciones en control de proceso, lo cual permite la experimentación controlada sin disponer del sistema real. El desarrollo del emulador es realizado a partir del modelo de la columna en una plataforma de aplicación en tiempo real basada en LINUX compuesta por los proyectos RTAI, RTAI-Lab y RTAI-XML. Los resultados de las emulaciones son validados con los resultados de las simulaciones obtenidas en MATLAB. Finalmente, se presenta una propuesta metodológica para la emulación de columnas de destilación binaria continua.

1. INTRODUCCIÓN

Para los procesos de enseñanza-aprendizaje o para el desarrollo de investigaciones en control de procesos tecnológicos industriales, como por ejemplo generadores de vapor y columnas de destilación, se presenta la dificultad de experimentación debido a la indisponibilidad total o parcial del sistema real; así por ejemplo, para el diseño del control, o para observar la respuesta del mismo ante cambios paramétricos o ante perturbaciones, o para operar el sistema en regiones de riesgo, en cada uno de estos casos resulta impráctico por los altos costos la experimentación sobre el sistema real y en otros casos puede incluso estar de por medio las vidas humanas.

La destilación es la operación de separar mediante vaporización y recondensación los diferentes componentes líquidos o licuados de una mezcla en dos o más fracciones que tienen diferentes puntos de ebullición [1]. Es una de las operaciones más ampliamente utilizadas y una de las más importantes y antiguas que se conoce; su aplicación va desde la destilación de alcohol hasta las refinerías de petróleo.

Considerando la importancia del proceso de destilación en las plantas industriales, el auge de la producción de alcohol carburante empleando columnas de destilación, la inexistencia total o baja disponibilidad del sistema real para experimentación (para realizar pruebas y experimentar cambios en los parámetros directamente sobre el proceso para el diseño de control o estudiar los diferentes comportamientos que se presentan), los altos costos en pruebas sobre el sistema real, la imposibilidad de operar el sistema en regiones de riesgo, tiempos de ensayo en el sistema real demasiado largos y para el desarrollo académico en instituciones de educación superior, surge la necesidad del desarrollo de una herramienta computacional o software que permita la emulación de una columna de destilación binaria a partir del modelo matemático del proceso.

Para dar respuesta a esta necesidad, pueden implementarse plantas prototipo o procesos reales en laboratorios de universidades o centros de investigación, pero estas opciones representan altos costos.

La emulación de un proceso permite la experimentación controlada y evita incurrir en los altos costos y riesgos mencionados. Por otro lado, para el desarrollo académico en instituciones de educación superior es importante que los estudiantes puedan validar sus trabajos y diseños relacionados con el tema.

Considerando lo anterior, el presente trabajo de investigación tuvo como objetivo diseñar e implementar un programa para la emulación de una columna de destilación binaria a partir del modelo matemático del proceso. La herramienta permite emular una columna de destilación binaria sobre una plataforma experimental, partiendo de un modelo matemático preestablecido (o definido de la revisión bibliográfica), con la posibilidad de experimentar cambios en las variables que permitan analizar el comportamiento en estado estable.

En el anexo C se presenta un método desarrollado para la emulación de una columna de destilación binaria sobre una plataforma experimental.

Los resultados obtenidos tendrán como usuarios de la herramienta y del ambiente de emulación a estudiantes, jóvenes investigadores, investigadores y personal de la industria que labore en áreas relacionadas y que trabajen en esta línea.

El documento está estructurado de la siguiente manera, en el capítulo dos se presenta brevemente un marco conceptual sobre el proceso de destilación binaria, una discusión de los conceptos de simulación y emulación, una síntesis de la revisión bibliográfica y el modelado matemático de la columna de destilación binaria con rectificación y agotamiento utilizada en la emulación; en el capítulo tres, se presenta brevemente la metodología de trabajo; el capítulo cuatro, trata el proceso de diseño del emulador en el cual se definen las variables involucradas, se analizan los requerimientos de implementación y se muestran los resultados obtenidos; en el capítulo cinco, se presentan las conclusiones, recomendaciones y trabajo futuro.

2. MARCO CONCEPTUAL

2.1 MARCO TEÓRICO

2.1.1. Destilación

La destilación [2] se define como la separación física de una mezcla en dos o más fracciones que tienen diferentes puntos de ebullición. Esta operación es la más empleada e importante en las refinerías e industrias químicas, lo que ocasiona que una gran parte de la energía usada en estas empresas sea consumida en los procesos de destilación

La destilación aprovecha la tendencia que presentan los compuestos volátiles en una mezcla líquida para concentrarse en la fase vapor y al mismo tiempo que los componentes más pesados de la mezcla tienden a enriquecer la fase líquida en equilibrio termodinámico con la fase vapor. Al calentar la mezcla líquida de dos materiales volátiles, el vapor que se separa tendrá una mayor fracción molar del material de menor punto de ebullición. Y de igual forma el líquido tendrá una fracción molar mayor del material de mayor punto de ebullición. Para llevar a cabo procesos de separación a escala industrial se utilizan las **columnas de destilación**, las cuales son carcasas metálicas en cuyo interior se colocan rellenos aleatorios u ordenados, o bandejas de platos, unas sobre otras.

Una columna de destilación se presenta en la figura 1, donde se observan las principales corrientes y los intercambiadores de calor para evaporar y condensar.

El flujo de alimentación entra en el plato *i* por medio de un conducto. El líquido empieza a deslizarse de platos superiores a inferiores dentro de la columna. Esta a su vez contiene un flujo de vapor en contra de la dirección del líquido teniendo así una transferencia de materia dando como resultado la evaporación del componente más ligero y la condensación del componente más pesado.

Según la ubicación del alimento en la columna, esta se divide en dos zonas principales: zona de despojamiento o agotamiento y zona de enriquecimiento o rectificación. En la primera zona, se requiere extraer el componente más volátil para evitar pérdidas en la corriente que se retira en el fondo de la columna conocida con el nombre de “fondos”. El

vapor de la cima de la columna se condensa y se divide generalmente en dos corrientes: reflujo y destilado.

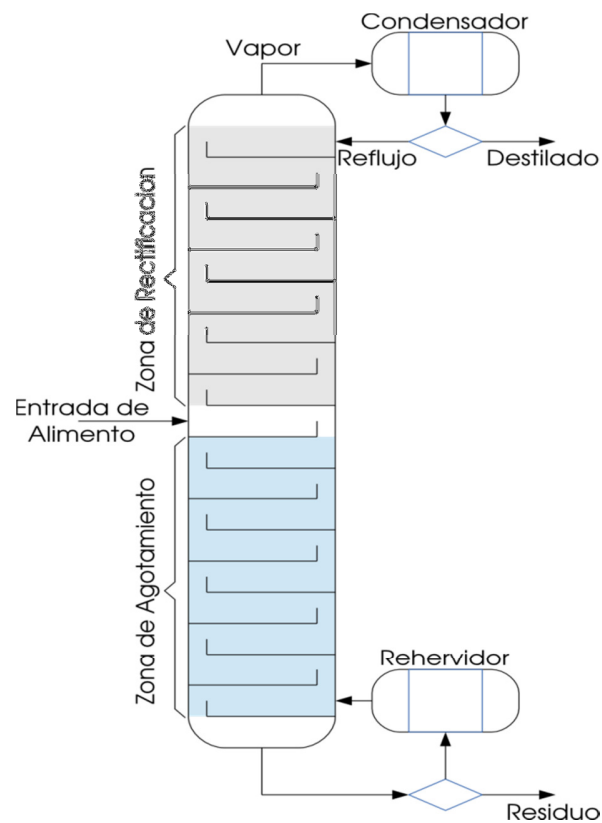


Figura 1. Configuración general de una columna de destilación.

El destilado o producto de cima se extrae de la columna para ser enviado una etapa del proceso “aguas abajo” o se enfría antes de ser almacenado según lo requiera el producto. A la relación entre el reflujo y la extracción de cima se le conoce como “relación de reflujo”. El reflujo es una de las principales variables de operación y control que permite regular la separación, el perfil de temperatura y garantizar la operación en forma eficiente. En ciertos procesos, para mejorar la calidad del compuesto de interés se requiere la extracción de corriente laterales en etapas intermedias que se conocen como “salidas laterales”. Estas extracciones se determinan con base en el perfil de composición y/o temperatura de la columna para purgar impurezas u obtener componentes con determinada concentración.

Cuando se tienen múltiples etapas de equilibrio sucesivas puede lograrse una separación muy significativa hasta obtener una corriente de cima muy rica en el componente más volátil (una fracción muy pura en la cima de la columna) y una corriente de fondo muy concentrada en el compuesto más pesado o menos volátil. En cada plato de la columna o etapa de equilibrio coexiste una fase líquida descendente en contacto íntimo con una fase vapor ascendente en continua transferencia de masa y calor.

Cada etapa o plato de equilibrio aporta una determinada separación de los componentes, de modo que para obtener el desempeño total de la columna debe considerarse la suma de todas las separaciones individuales. Durante la transferencia de masa, la fase vapor se enriquece con el componente de menor punto de ebullición y la fase líquida concentra el componente de mayor punto de ebullición, de modo que el vapor que abandona cada plato tiene mayor concentración de componente volátil que el vapor que entra al mismo plato y el líquido que rebosa un plato está descendiendo con mayor concentración del compuesto pesado. La transferencia de calor suministra la energía necesaria para que tenga lugar la transferencia de masa entre el líquido y el vapor de acuerdo con las condiciones de temperatura y presión de cada etapa de equilibrio.

Para mantener condiciones estables de operación se requiere que el vapor que sale por la cima de la columna se condense y se retorne a la columna para mantener el flujo de líquido descendente y de igual manera es necesario que se evapore parte de la fase líquida del fondo para garantizar un flujo de vapor ascendente. Estos flujos de operación se encuentran dentro de un rango muy definido que permite mantener la columna en forma estable y eficiente.

El objetivo del sistema de calentamiento o suministro de energía es generar el vapor que asciende a través de la columna. La fuente más común de energía es el vapor, y cuando el proceso permite el contacto entre el vapor y el fluido de operación, se utiliza vapor directo para calentar el líquido en el fondo de la columna. En caso contrario, el vapor se utiliza mediante contacto indirecto a través de un intercambiador conocido como rehervidor.

Por otro lado, el sistema de enfriamiento se utiliza para la condensación de los vapores de cima y generación de la fase líquida que se retorna a la columna y que en muchos casos se extrae como producto de cima. La condensación de los vapores puede ser total o parcial,

dependiendo de la cantidad de vapor que se condensa en el equipo. Si el condensador es total, todo el producto de cima es condensado, de manera que el destilado es recuperado en fase líquida y se puede asumir que el flujo de vapor que se genera en el condensador es igual a cero (no existe salida de vapor). En algunos sistemas se utiliza una corriente del proceso diferente a agua de torre para condensar estos vapores y generar una recuperación de energía con doble efecto simultáneo conveniente de condensación y precalentamiento. Una desventaja del proceso de destilación es su alto consumo de energía en términos de las utilidades requeridas como vapor y agua de torre para los respectivos procesos de calentamiento y enfriamiento. Su costo contribuye en más del 50% a los costos de operación.

2.1.2. Simulación y emulación

Es muy común encontrar que existe confusión acerca de estos dos términos, que aunque aparentemente pueden mostrar la misma intención y el mismo propósito, existen diferencias significativas en su uso y operación.

La diferencia principal entre simulación y emulación radica en el tiempo de ejecución del sistema simulado o emulado y el tiempo de ejecución real del proceso, de tal forma que en la simulación este tiempo depende de la capacidad de la máquina en la que se realiza la simulación, mientras que en la emulación este tiempo corresponde segundo a segundo con el tiempo de ejecución real del proceso.

Un simulador representa un sistema, como por ejemplo, una planta (de cualquier orden), la cual al someterse a una o varias entradas de referencia, entregan una respuesta que, de acuerdo al tipo de proceso, no siempre corresponde con la realidad.

El simulador suele presentar una respuesta muy ideal del sistema simulado.

Dado que todos los modelos son aproximaciones de sistemas reales, entonces existirán diferencias medibles entre el desempeño del sistema real y el modelo. Estas diferencias frecuentemente causan un grado de incertidumbre que la emulación persigue minimizar [4].

Las emulaciones de modelos son usadas de una forma mucho más precisa y definida, para evaluar la operación de sistemas bajo diferentes condiciones de carga del sistema. La emulación de un modelo refleja de forma más precisa el sistema que será implementado y como tal puede ser usado para llevar una serie de restricciones de procedimientos de verificación para asegurarse del desempeño o reacción del sistema de control [4].

En un emulador por ejemplo, se pueden introducir cambios en los parámetros, entradas y demás variables del sistema emulado, permitiendo observar inmediatamente la incidencia de estos cambios sobre las variables observadas de dicho sistema.

Los **emuladores** son una muy buena alternativa a la hora de querer experimentar cambios en los parámetros del proceso para realizar pruebas sin tener que utilizar el sistema real, pues buscan imitar las acciones que realizan en la industria los procesos a representar.

La palabra emulador tiene una relación muy estrecha con los desarrollos que se presentan en la informática de tal forma que el término se maneja como un software que permite ejecutar programas de computadora en una plataforma (arquitectura hardware o sistema operativo) diferente de la empleada para el desarrollo original.

2.2. REVISIÓN BIBLIOGRÁFICA

En las últimas dos décadas, la destilación ha sido uno de los procesos más investigados en el mundo, diseñando y mejorando algoritmos de control para optimizarlo.

En 1999 Klaus[3] en su tesis doctoral titulada “Modelado y Simulación de una Torre de Destilación Multicomponente”, hace una recopilación breve del estado del arte sobre columnas de destilación; presenta los comentarios y referencias que relacionan las fechas desde que se inició con el estudio sistemático sobre columnas de destilación (1893) e ilustra el impacto de la computadora digital en el desarrollo de algoritmos matemáticos y estudios más profundos (modelado dinámico, estabilidad, algoritmos de control) sobre la dinámica

de sistemas de destilación entre 1962 y 1975. Menciona además, algunos trabajos realizados en las décadas de los ochenta y noventa, relacionados con críticas a la literatura existente hasta ese momento, destacándose el modelado y control en columnas de destilación. Uno de estos trabajos, desarrollado en 1997 por Skogestad [5] plantea que aunque para ese momento la dinámica y control de columnas de destilación es considerado por muchos como un campo muy maduro y de pocas novedades, aparecieron nuevos enfoques como el análisis de plantas con termodinámica compleja, la facilidad de usar el destilado de fondo para control, la importancia de la dinámica de flujo para estudios de control, el entendimiento general del perfeccionamiento del comportamiento dinámico de las columnas de destilación, etc, aplicados a todo tipo de columnas tanto simples (con solo dos productos) como complejas, de tal forma que existe un buen número de áreas que requieren que se promueva la investigación.

Revisando la literatura existente se encuentran trabajos sobre columnas de destilación que utilizan desde modelos simples hasta modelos rigurosos de las columnas de destilación dependiendo del tipo de estudio o desarrollo a realizar; sin embargo, aunque estos modelos describen muy bien las columnas, se requiere de herramientas de cómputo para resolver las ecuaciones que representan estos modelos y que permitan la simulación del proceso [5].

Con el auge de los computadores, la simulación es una herramienta que ha permitido simplificar el análisis, estudio y modelado de diferentes tipos de proceso, así como reducir el tiempo y costo asociados a estas actividades.

En las últimas dos décadas, se encuentran diversos trabajos que utilizan la simulación convencional para analizar el comportamiento de columnas de destilación. Por ejemplo:

- Planteamiento de la importancia de la simulación como una herramienta académica en la enseñanza de los problemas de control real [6].
- La simulación dinámica y control difuso de columnas de destilación [7].
- Desarrollo de un modelo de simulación por computador integrando el diseño de la dinámica de fluidos con la descripción fisicoquímica clásica de la destilación [8].

- La simulación del proceso global de obtención de alcohol carburante fundamentada en principios termodinámicos [9].
- Simulación del proceso de destilación extractiva de etanol azeotrópico utilizando glicerol como agente de separación. Aunque no plantean el modelo matemático, aborda el problema de la deshidratación a través de la simulación utilizando Aspen Plus y muestran la eficiencia desde el punto de vista energético [10].
- El estudio de algunas técnicas de modelamiento y simulación en columnas de destilación binaria donde el punto álgido es el comportamiento dinámico de la composición del producto bajo disturbios en la alimentación [11].

Por otra parte, una revisión de la bibliografía existente sobre **emulación** de columnas de destilación - simulación en tiempo real- arroja información limitada; se encuentra más información referente a otros tipos de procesos, lo que muestra que es un área abierta para el desarrollo de investigaciones.

En 2008, Chang, G. W; Liu, Y. J; y Dinavahi, V. [12] exponen las ventajas de la simulación en tiempo real frente a la simulación convencional en una aplicación sobre el estudio de armónicos en sistemas de potencia industriales, entre las que se destacan: mejor eficiencia computacional, programación flexible y escalable, conservando una exactitud tan buena como la simulación convencional.

Ramos, C., Romero, A., Giral, R., Vidal L., Martinez, L., en 2009, presentan una técnica de modelado basado en Fuzzy para la emulación de sistemas de generación de energía eléctrica con celdas de combustible [33].

Asimismo, Ramos, C., Romero, A., Giral, R., Martinez, L., Sanchez, C., en 2011, presentan un emulador de celdas de combustible donde evalúan la conmutación y los estados de potencia, diseñado para probar dispositivos y cargas destinadas a interactuar con prototipos reales [34].

En 2012, Restrepo, C., Ramos-Paja, C.A., Giral, R., Calvente, J., Romero, A. [35] proponen un emulador de pila de combustible basado en el Target de Matlab para tiempo real bajo Windows y una fuente de alimentación controlada para estimar la proporción de exceso de oxígeno en aplicaciones de electrónica de potencia, el cual puede ser utilizado para diseñar y probar dispositivos antes de su conexión a las pilas de combustible, evitando fenómenos no deseados.

La simulación en tiempo real se ha convertido en una herramienta poderosa y ha ganado fuerza su implementación con fines académicos por lo que algunos trabajos plantean su importancia. Veenu Dixit, Mahesh B. Patil y Mukul C. Chandorkar en 2009 [13], establecen su importancia en el diseño de sistemas electrónicos de potencia, especialmente en el contexto de la simulación de hardware in the loop (HIL), pero direccionan el trabajo hacia la implementación en un laboratorio de educación, presentando en forma clara los componentes que consideran requiere en términos generales un ambiente de simulación en tiempo real.

Santana José [14], propone una plataforma orientada al ámbito académico, con el fin de complementar o reemplazar el uso de plantas a escala a un precio económico y de igual forma con una descripción detallada de los elementos constitutivos tanto de hardware como de software de un sistema de tiempo real y bajo la filosofía HIL en la implementación de la plataforma.

En el enfoque académico que algunos desarrollos plantean vale la pena destacar la propuesta de Adelson Carvalo, Dante Barone y Milton Zaro [15] en 2010, en la que presentan el desarrollo de una plataforma como un recurso para la enseñanza de ingeniería de control, la cual combina el potencial educativo de software de diseño y simulación de sistemas de control y la conexión de este software y una columna de destilación de laboratorio que, aunque realiza simulación y no emulación, diseñan un experimento basado en la teoría constructivista y el interaccionismo social para detectar el desarrollo del estudiante y capturar sus impresiones respecto del estudio con la plataforma. Esta propuesta se relaciona con el enfoque educativo del grupo de investigación GICI de la universidad del

Valle y que utiliza la Plataforma de Educación Remota en Ingeniería PERI [4]. En el anexo A se presenta una breve descripción de la plataforma PERI.

Algunos trabajos proponen emulaciones de procesos como la destilación de plantas petroquímicas [16], o para el entrenamiento de operarios de centrales y subestaciones [17], [13]. Los artículos que tratan sobre el diseño de plataformas para emular procesos utilizan una metodología similar aunque, existen diferencias en el tipo de proceso o en el tipo de software utilizado, [18], [19].

2.3. MODELADO DE COLUMNAS DE DESTILACIÓN BINARIA CONTÍNUA

Uno de los mayores problemas del modelamiento de columnas de destilación es la resolución del modelo debido a la complejidad y la no-linealidad del mismo; sin embargo, una de las ventajas de la representación de sistemas mediante modelos matemáticos es la posibilidad de simplificar dichos modelos, teniendo en cuenta solo los aspectos más relevantes en el comportamiento general del sistema. En el caso de las columnas de destilación se pueden hacer aproximaciones en aspectos como la termodinámica, la dinámica de flujo de líquido, la interacción entre vapor y líquido. Estudiar el caso binario permite estudiar con claridad la estructura básica de las ecuaciones y comprender los conceptos básicos que intervienen en el proceso de destilación. De esta manera se obtiene una aproximación general del comportamiento de la columna y que puede ser fácilmente resuelto con las herramientas de cómputo actuales [20].

Para control, generalmente en los documentos que tratan sobre columnas de destilación se han usado modelos simplificados.

El modelado de las columnas de destilación se clasifica a menudo en tres grupos: modelado fundamental, modelado empírico y modelado híbrido [22].

En el modelado fundamental, el modelo es construido sobre la base de las propiedades físicas del sistema, tales como la conservación de la masa, energía y momentum, y

dependiendo de los niveles de precisión de los supuestos vamos a tener diferentes modelos que van desde modelos simples a modelos rigurosos.

El modelo empírico utiliza los datos de entrada y salida del funcionamiento de la columna para construir la relación entre el de entrada y el de salida. Con este método no es necesario entender la dinámica interna de la columna, y el cálculo puede ser reducido. Pero en el uso de este método tenemos que llevar a cabo experimentos en la columna real, y los resultados no pueden ser aplicados a otra columna, incluso los resultados de una columna pueden ser diferentes si las condiciones de la columna son diferentes entre el experimento y el funcionamiento real de la columna.

El modelo híbrido combina la modelización fundamental y el modelado empírico. Este método utiliza las ventajas de los otros dos, pero se necesita un modelo bien estructurado en el que se tiene que decidir en qué parte del modelo utilizar la técnica fundamental y en qué parte utilizar datos empíricos.

Para el desarrollo del emulador se escogió el modelo de columna Tipo A planteada por Skogestad [5]. Este modelo presenta las ventajas de un modelo fundamental, el cual se caracteriza por presentar mayor precisión, una comprensión más completa del proceso (permite entender la dinámica de las columnas de destilación) y validez mundial [22]. Aunque el modelo empírico se utiliza predominantemente en la industria (debido a su simplicidad en comparación con el modelo fundamental), no puede ser utilizado para predecir el comportamiento del sistema bajo otras condiciones de operación.

La columna está conformada por un condensador total (todo el producto de tope se condensa), de manera que el destilado es recuperado en fase líquida; una zona de rectificación con un número de platos finitos donde se hace llegar un reflujo del condensador, un plato de alimentación donde se introduce una mezcla binaria a una temperatura igual al punto de ebullición del elemento más volátil, una zona de agotamiento y un rehervidor. El rehervidor se encarga de entregar la energía térmica al proceso para la vaporización de la mezcla, y de esta forma poder realizar la separación de los componentes.

1. Suposiciones y consideraciones

Considerando la complejidad que representa el modelado completo del proceso, se establecen algunas suposiciones y consideraciones. S. Skogestad [32] hizo una revisión crítica de la literatura sobre dinámica y control de columnas de destilación. El documento resume las simplificaciones del modelo riguroso ya que no se habían encontrado referencias en la solución de todas las ecuaciones del modelo riguroso. Las simplificaciones están dirigidas a la dinámica de vapor, para el balance de energía y para la dinámica de flujo de líquido. En el documento se recomienda no descuidar la dinámica del líquido (es decir, no asumiendo retención líquido constante) debido al hecho de que la respuesta inicial, un factor importante en el control de realimentación, se ve afectada en gran medida por la retención de líquido [21].

A continuación se presentan las suposiciones y consideraciones que se plantean en modelo de la columna de destilación binaria:

- Separación binaria: especificada cuando se requiere separar una mezcla que está formada por dos componentes, por ejemplo separación de etanol y agua.
- Flujos molares constantes: en una sección de la columna, podemos utilizar muy a menudo la suposición acerca de los flujos molares constantes. Es decir, se asume $L_i = L_{i+1} = L$ [mol/s] y $V_{i-1} = V_i = V$ [mol/s]. Esta suposición es razonable para mezclas ideales cuando los componentes tienen capacidad calorífica similar. Una implicación importante es que la línea de operación es entonces una línea recta para una sección determinada. Esto hace los cálculos mucho más simples ya que los flujos internos (L y V) no dependen de la composición [30].
- Volatilidad relativa constante: La volatilidad relativa es una herramienta usada para determinar la facilidad y viabilidad de la separación de una mezcla por destilación [32]. Es una medida numérica de la posibilidad de separación, por destilación, de los componentes de la mezcla binaria. A mayor volatilidad, mayor separación de los componentes.

$$\alpha = \frac{K_i}{K_j} \quad (1)$$

Donde, α es la volatilidad relativa del componente i con respecto al componente j en estado puro.

Entre mayor sea la volatilidad relativa entre los dos componentes, es más fácil y menos costosa la separación de estos compuestos. Si la volatilidad relativa entre los componentes es mayor que la unidad ($\alpha > 1$), la separación por destilación ordinaria de A y B es posible. Si la volatilidad relativa es unitaria ($\alpha = 1$), la separación de A y B es imposible.

- Presión constante a lo largo de la columna (presión en cada plato es constante): En una columna de destilación la presión de vapor es generada por el gas resultante de la ebullición del líquido de alimentación. Este vapor se encuentra en condición de saturación, típico en sistemas de evaporación, por lo cual existe una correspondencia entre presión y temperatura.

La presión en el interior de la columna modifica el punto de ebullición de la mezcla a destilar, por lo cual la escogencia de una presión de trabajo determina la temperatura de evaporación de los componentes de la mezcla, y también determina indirectamente el flujo y la presión de vapor de calentamiento necesario a través del rehervidor.

- Todos los flujos de entrada y salida de la torre se encuentran en estado líquido: indicando que no entran o salen flujos en forma de vapor.

- La alimentación se hace a través de un solo plato.

- Equilibrio vapor-líquido y perfectamente mezclado en todas las etapas: el concepto de etapas en equilibrio es importante en el proceso de destilación.

Aquí se asume equilibrio vapor-líquido (VLE) en cada etapa, donde el líquido se envía a la etapa de abajo y el vapor a la etapa precedente. Se establece que los cálculos basados en el concepto etapa de equilibrio (con el número de etapas ajustada adecuadamente) se ajusta muy bien a los datos de la mayoría de columnas reales.

Para los cálculos prácticos, el caso crítico por lo general no es el modelado de las etapas, sino obtener una buena descripción del equilibrio vapor-líquido. En esta parte ha habido

avances significativos, sobre todo después de la introducción de las ecuaciones de estado para la predicción VLE. Sin embargo, aquí se utiliza un modelo más simple (VLE volatilidad relativa constante) aplicables a mezclas relativamente ideales [30].

En equilibrio, la fase de vapor y la fase líquida poseen diferentes composiciones. La relación de las concentraciones de equilibrio para un componente i entre las fases es descrita por una relación de equilibrio definida como [32]:

$$K_i = \frac{Y_i}{x_i} \quad (2)$$

y

$$K_j = \frac{Y_j}{x_j} \quad (3)$$

Donde, K_i es la relación de equilibrio para el componente i , Y_i es la fracción molar del componente i en el vapor en equilibrio y x_i es la fracción molar de i en el líquido.

Las relaciones de equilibrio de dos componentes en el sistema pueden ser comparados a las mismas condiciones por medio de la volatilidad relativa, definida como,

$$\alpha = \frac{K_i}{K_j} \quad (4)$$

Donde, α es la volatilidad relativa del componente i con respecto al componente j .

Generalmente la volatilidad relativa es expresada como la relación del componente más volátil al menos volátil, de modo que el valor numérico de α sea mayor que la unidad.

Para una mezcla binaria, $x_j = 1 - x_i$ y $Y_j = 1 - Y_i$,

$$\alpha = \frac{Y_i/x_i}{(1-Y_i)/(1-x_i)} \quad (5)$$

o

$$Y_i = \frac{\alpha x_i}{(1+(\alpha-1)x_i)} \quad (6)$$

x_i es la fracción molar del componente ligero en el líquido en la etapa i (composición molar).

Y_i es la fracción molar del componente ligero en el vapor en la etapa i (composición molar).

- No hay retención de vapor (es decir, respuesta inmediata del vapor): se considera nula la acumulación de materia bajo la forma de vapor (capacidad molar del vapor igual a cero).

- No se asume retención de líquido M_i constante en las etapas, es decir, que se incluye la dinámica de flujo de líquido. Esto significa que se necesita algún tiempo (aproximadamente $NT - 2 * \tau$) desde que cambia el líquido en la parte superior de la columna hasta que cambia el flujo de líquido en el rehervidor. Esto es bueno cuando se hace control, lo que significa que se pueden evitar algunas de las interacciones fuertes que existen en el estado de equilibrio entre las composiciones en la parte superior y el fondo de la columna [31].

- Condensador total (no hay retención de vapor en el condensador): la composición del vapor que abandona la columna por la cima será la misma que la de la corriente de reflujo y destilado.

Estas suposiciones pueden parecer restrictivas pero en realidad satisfacen muchos sistemas reales y en todo caso las expresiones resultantes producen aportes invaluable [30].

El modelo está compuesto por las siguientes ecuaciones:

2. Balance total de materia en la etapa i

Es la primera ecuación que se describe, en la cual, cuando no hay reacciones químicas, puede expresarse en unidades molares. Si se supone que a causa de su baja densidad, la acumulación de masa de la fase vapor, en relación con la de la fase líquida, es despreciable, el balance total de masa en el plato i es:

$$\frac{dM_i}{dt} = L_{i+1} - L_i + V_{i-1} - V_i \quad (7)$$

Donde:

M_i es la retención de líquido en la plato i (líquido residual) en Kmol.

L_i es el flujo de líquido que sale del plato i en Kmol/min.

V_i es el flujo de vapor que sale del plato i en Kmol/min.

L_{i+1} es el flujo de líquido procedente del plato superior en Kmol/min.

V_{i-1} es el flujo de vapor procedente del plato inferior en Kmol/min.

3. Balance de materia para el componente ligero en el plato i

La ecuación que representa el balance del componente i es:

$$\frac{dM_i x_i}{dt} = L_{i+1} x_{i+1} + V_{i-1} Y_{i-1} - L_i x_i - V_i Y_i \quad (8)$$

Donde:

x_i es la fracción molar del componente ligero en el líquido en la etapa i (composición molar).

Y_i es la fracción molar del componente ligero en el vapor en la etapa i (composición molar).

Lo cual da la siguiente expresión para la fracción molar de líquido:

$$\frac{dx_i}{dt} = \frac{\left(\frac{dM_i x_i}{dt} - x_i \frac{dM_i}{dt} \right)}{M_i} \quad (9)$$

4. Ecuaciones algebraicas.

Asumiendo que el vapor y el líquido que abandonan cada etapa se encuentran en equilibrio, la relación de equilibrio que existe entre las concentraciones de fase líquida y vapor (composición de vapor y_i que está relacionada con la composición de líquido x_i en la misma etapa) se calcula a partir de la ecuación algebraica de equilibrio vapor-líquido:

$$Y_i = \frac{\alpha x_i}{(1 + (\alpha - 1)x_i)} \quad (10)$$

Donde α es la volatilidad relativa. Asumiendo flujos molares constantes y la no existencia de dinámica de vapor tenemos la siguiente expresión para los flujos de vapor (excepto en la etapa de alimentación si la alimentación está parcialmente vaporizada), donde

$$V_{NF} = V_{(NF-1)} + (1 - q_F)F: V_i = V_{(i-1)} \quad (11)$$

Donde:

NF es el plato de alimentación

q_F es la fracción de líquido en la alimentación

Los flujos de líquido dependen de la retención de líquido en la etapa superior y el flujo de vapor bajo la siguiente relación:

$$L_i = L_{01} + \frac{(M_i - M_{0i})}{\tau_l} + (V - V_0)_{(i-1)} * \lambda \quad (12)$$

Donde:

L_{0i} [Kmol/min] es el valor nominal de reflujo.

M_{0i} [Kmol] es el valor nominal de la retención en la etapa i .

τ_l es la constante de tiempo para la dinámica de flujo de líquido en cada etapa (excepto el rehervidor y el condensador).

λ es la constante por efecto del flujo de vapor sobre el flujo de líquido. Se asume que el flujo de vapor no afecta la retención de líquido (Efecto K2)

El flujo de vapor en la etapa puede afectar también la retención; λ puede ser positiva debido a que debido a que más vapor puede dar más burbujas y así por lo tanto puede empujar líquido fuera de la etapa. Si λ es grande (mayor que 0.5) entonces la retención en el rehervidor “se nivela” por algún tiempo en respuesta a un incremento en la evaporación, y si $\lambda > 1$ se obtiene una respuesta inversa [23].

λ también puede ser negativo si se incrementa la caída de presión causada por un V grande dando como resultado una retención grande en las bajantes – en general, es difícil estimar λ para columnas de platos. Para columnas empacadas λ es usualmente cercana a cero.

Las ecuaciones anteriores se aplican a todas las etapas excepto en el tope (condensador), etapa de alimentación y fondos (rehervidor).

Etapla de alimentación, $i=NF$ (se asume que la alimentación es mezclada directamente en el líquido de la etapa de alimentación).

El balance de masa global y por componente de la etapa de alimentación es:

Balance de masa global

$$\frac{dM_i}{dt} = L_{i+1} - L_i + V_{i-1} - V_i + F \quad (13)$$

Donde:

M_i es la masa retenida en el plato i

V es el flujo molar de vapor

L es el flujo molar de líquido

F es el flujo molar de alimentación (tasa de flujo de alimentación)

Balance de masa por componente

$$\frac{d(M_i x_i)}{dt} = L_{(i+1)} x_{(i+1)} + V_{(i-1)} Y_{(i-1)} - L_i x_i - V_i Y_i + F z_F \quad (14)$$

Donde:

M_i es la masa retenida en el plato i

V es el flujo molar de vapor

L es el flujo molar de líquido

F es el flujo molar de alimentación (tasa de flujo de alimentación)

Z_F la composición molar de alimentación (concentración del componente ligero en la alimentación)

x es la composición molar de líquido

Y es la composición molar de vapor

Condensador total,

$$i = NT (M_{NT} = M_D, L_{NT} = L_T) \quad (15)$$

Balance de masa global

$$\frac{dM_i}{dt} = V_{(i-1)} - L_i - D \quad (16)$$

Donde:

$M_i = M_D$, es la masa molar retenida en el condensador

$V_{i-1} = V_{NT-1}$, el flujo de vapor que entra al condensador

$L_i = L_{NT}$, el flujo molar de líquido que recircula a la columna

D es el flujo molar de destilado

Balance de masa por componente

$$\frac{d(M_i x_i)}{dt} = V_{(i-1)} Y_{(i-1)} - L_i x_i - D x_i \quad (17)$$

Donde:

$x_i = x_D$ es la composición molar de destilado

$Y_{i-1} = Y_{NT-1}$ es la composición molar de vapor

Rehervidor.

$$i = 1 (M_i = M_B, V_i = V_B = V) \quad (18)$$

Balance de masa global

$$\frac{dM_i}{dt} = L_{(i+1)} - V_i - B \quad (19)$$

Donde:

$M_i = M_B$ masa molar retenida en el rehervidor

B flujo molar de fondos

L_{i+1} flujo molar de líquido

V_i flujo molar de vapor

Balance de masa por componente

$$\frac{d(M_i x_i)}{dt} = L_{(i+1)} x_{(i+1)} - V_i Y_i - B x_i \quad (20)$$

Donde:

$M_i = M_B$ masa molar retenida en el rehervidor

B flujo molar de fondos

L_{i+1} flujo molar de líquido

V_i flujo molar de vapor

$x_i = x_B$ composición molar de fondos

$Y_i = Y_B$ composición molar de vapor

3. METODOLOGÍA

Para llevar a cabo el desarrollo del proyecto se realizaron las siguientes tareas:

Estado del arte

En esta etapa se recolectó, sistematizó y clasificó la información relativa a simulación (especialmente simulación en tiempo real) y modelos de columnas de destilación.

Definición del modelo matemático

Se realizó la revisión de desarrollos de investigación reconocidos que plantean modelos de columnas de destilación, se estudió su funcionamiento y se seleccionó el modelo descrito en el punto 2.3 del presente documento.

Definición de las variables involucradas en la emulación

Se estudió la información clasificada y relacionada principalmente con el modelamiento de columnas de destilación y se identificaron cada uno de los subsistemas que componen una columna de destilación para así definir las variables con base en el análisis funcional del sistema.

Diseño e implementación del emulador

Una vez definidas, se describen las variables involucradas en la emulación y se realiza el diseño del programa que permitió emular la columna de destilación binaria considerando los requerimientos de implementación.

Diseño de experimentos de simulación

Mediante un procedimiento definido, se llevaron a cabo las emulaciones que permitieron validar los resultados obtenidos frente a las respuestas y resultados de las simulaciones en MATLAB.

Se han elegido los siguientes casos para la emulación de la columna de destilación binaria continua:

- Dos casos de emulación de estado estacionario para las perturbaciones F y zF, y un caso de emulación para las entradas D y B en la columna tipo A (41 etapas), propuesta por Skogestad [5].
- El mismo conjunto de pruebas para un número menor de etapas (22 etapas).

Método propuesto para la emulación de columnas de destilación binarias continuas

Se estableció una metodología que permite al usuario emular una columna de destilación en la plataforma RTAI (ANEXO C).

4. DISEÑO E IMPLEMENTACIÓN

4.1. DEFINICIÓN DE LAS VARIABLES INVOLUCRADAS EN LA EMULACIÓN

En las tablas 1 a 4 se describen las variables involucradas en el modelo y en el diseño del emulador para realizar con base en estos datos las simulaciones necesarias.

Variable	Descripción	Valor	Unidad
NT	Número de platos incluidos el rehervidor y el condensador total	Entre 5 y 50	UN
NF	Plato de alimentación	Entre 3 y NT-2	UN
α	Volatilidad relativa	1.5	
$M0_i$	Retención nominal	0.5	kmol/min
τ_l	Constante de tiempo para la dinámica de flujo de líquido en cada etapa (excepto el rehervidor y el condensador).	0.063	min
λ	Constante por efecto del flujo de vapor sobre el flujo de líquido. Se asume que el flujo de vapor no afecta la retención de líquido (Efecto K2)	0	
L0	Flujo nominal de reflujo (de los datos de estado estable)	2.706	kmol/min
$L0_b$	Flujo nominal de líquido (debajo del plato de alimentación)	En función de $L0_b$	kmol/min
V0	Flujo de vapor nominal (sólo se requiere si λ es diferente de cero)	3.20629	kmol/min
V0t	Flujo de vapor nominal (encima del plato de alimentación)	En función de V0	kmol/min

Tabla 1. Variables internas columna de destilación binaria continua.

Variable	Descripción	Valor	Unidad
L_T	Reflujo	2.706	kmol/min
V_B	Flujo de vapor en el rehervidor	3.206	kmol/min
D	Flujo de producto del destilado (tope)	< 0.5	kmol/min
B	Flujo de producto en el fondo	< 0.5	kmol/min

Tabla 2. Entradas, columna de destilación binaria continua.

Variable	Descripción	Valor	Unidad
F	Flujo de alimentación	1	kmol/min
Z_F	Composición de la alimentación	Entre 0.1 y 0.9	Fracción molar
q_F	Fracción de líquido en la alimentación	1	Fracción molar

Tabla 3. Perturbaciones, columna de destilación binaria continua.

Variable	Descripción	Valor	Unidad
x_i	Composición inicial en cada etapa	Entre 0.5 y 1	Fracción molar
M_i	Retención inicial en cada etapa	Entre 0.5 y 1	kmol/min

Tabla 4. Condiciones iniciales, columna de destilación binaria continua.

4.2. DISEÑO E IMPLEMENTACIÓN

Para el desarrollo de la emulación de una columna de destilación binaria se ha elegido trabajar con la Plataforma RTAI (Real Time Application Interface), interface de aplicaciones en tiempo real para Linux. Esta plataforma está compuesta por tres sistemas:

- Un **Kernel Linux** modificado con los módulos RTAI.
- El Proyecto **RTAI-Lab**, que es una cadena de herramientas para convertir diagramas de bloques en ejecutables RTAI (compilar y ejecutar en tiempo real). Los diagramas de bloques se pueden desarrollar utilizando Scilab / Scicos (Open

Source) o Matlab / Simulink / RTW (comercial). Proporciona la paleta de bloques para Simulink y Scicos.

- El Proyecto **RTAI-XML** que permite monitorear y comunicarse con el proceso que se encuentra en ejecución en la plataforma, a través de la aplicación **jRTAILab**. Se puede encontrar más información en el ANEXO B.

Para realizar el diseño del emulador de la columna de destilación binaria se requiere el diseño de un diagrama de bloques que puede ser desarrollado en **SCICOS** o **SIMULINK**, a partir del cual se genera un código fuente en lenguaje C compatible con la plataforma RTAI. Para el emulador, el diagrama se realizó en SCICOS.

En forma general, el procedimiento a seguir para generar una aplicación ejecutable en la plataforma RTAI a partir de SCICOS está compuesto por las siguientes etapas:

Diseño del diagrama de bloques

En esta etapa se construyó el diagrama de bloques que representa el proceso a emular (figura 2). Para construirlo, se pueden usar, además de los bloques de uso general de SCICOS, los bloques especiales proporcionados por la paleta RTAI-Lib para SCICOS, listados a continuación:

- **Sine**: genera una señal senoidal de amplitud y frecuencia variables.
- **Scope**: funciones de osciloscopio.
- **Square**: permite generar una señal cuadrada.
- **Step**: genera una señal escalón.
- **C RTAI Block**: permite ingresar código en lenguaje C definido por el usuario.
- **Bloques COMEDI**: permiten comunicarse con hardware externo a través de los drivers COMEDI.

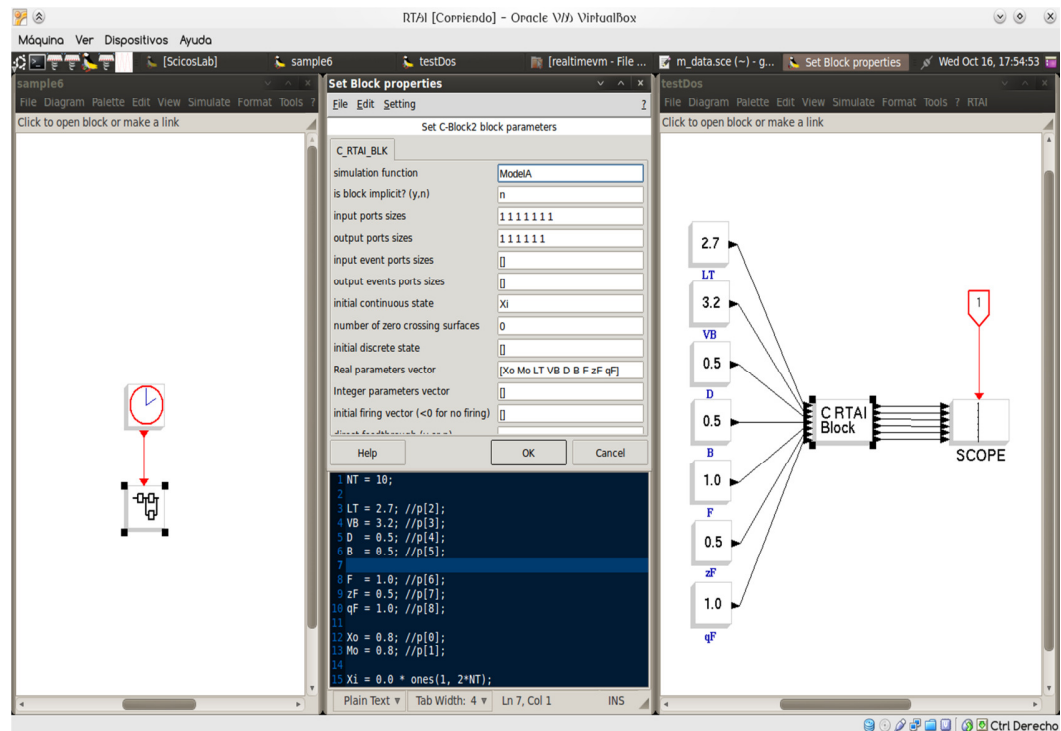


Figura 2. Diagrama de bloques del emulador columna de destilación binaria.

Debe tenerse en cuenta, que para la columna de destilación, por tratarse de un proceso altamente no lineal y que no puede representarse mediante bloques de función de transferencia, requiere el uso de un bloque que permita el ingreso de código definido por el usuario (user-defined), para tal efecto en la paleta RTAI-Lib se encuentra el bloque “C RTAI Block”. Como su nombre lo indica, este bloque permite usar código escrito en lenguaje C, este código debe cumplir con el estándar ANSI C, ajustarse al diagrama de flujo mostrado en la figura 3 y a la siguiente estructura:

```
#include <math.h>
#include <stdlib.h>

#ifdef MODEL
#include <scicos/scicos_block4.h>
#endif

void toto(scicos_block *block, int flag) {
```

```

//1. Declaración de variables adicionales requeridas en
//el programa (opcional)

//2. Declaración de punteros para acceder a las funciones
// de scicos_block4.h:
// - GetRparPtrs(block): Accede al vector de parámetros
// de tipo real (decimales)
// - GetIparPtrs(block): Accede al vector de parámetros
// de tipo entero
// - GetRealInPortPtrs(block, n): Accede a las entradas
// del bloque, n indica el número de la entrada
// - GetState(block): "Accede a las variables de estado"
// - GetDerState(block): "Accede a las derivadas de
// las variables de estado"
// - GetOutPortPtrs(block, n): Accede a las salidas del
// bloque, n indica el número de la salida

//3. Inicialización de las variables adicionales definidas
//en < 1. >

//4.
if (flag == 4) {
    //Inicialización de las variables de estado "y otras
    //variables internas del programa" (ejecutado una sola
    //vez al iniciar la emulación)
}

if (flag == 0) {
    //Cálculo de las derivadas de las variables de estado
    //(ejecutado periódicamente durante el tiempo de emulación)
}

if (flag == 1) {
    //Actualización de las salidas (ejecutado "periódicamente?
    //o una sola vez al finalizar el tiempo de emulación?"
}
}

```

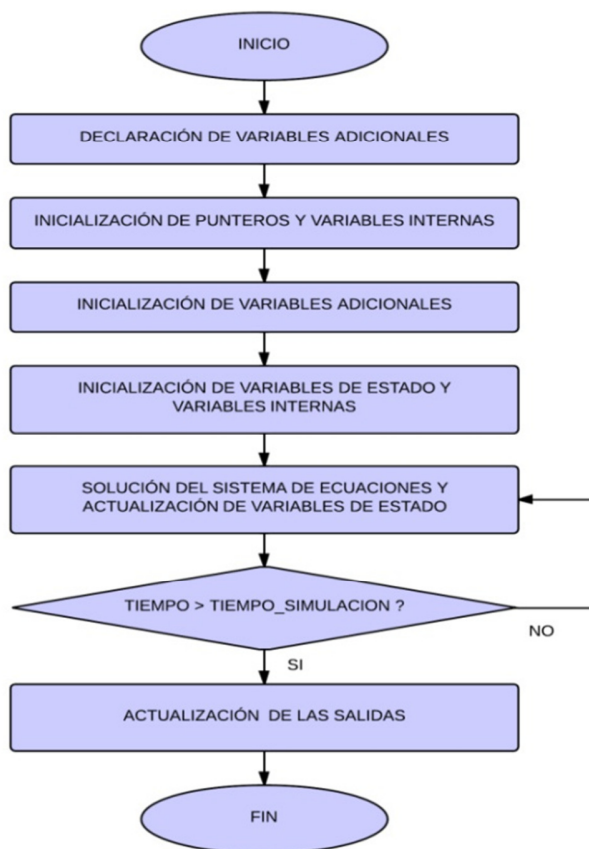


Figura 3. Diagrama de flujo.

Parametrización del modelo

Se puede realizar de tres formas:

- Definiendo las variables requeridas dentro del código fuente si se trata de un modelo definido en un “C RTAI Block”.
- Enlazando un archivo de ScicosLab (con extensión *.sce) a través de la función “Context” de Scicos y las propiedades del bloque C RTAI Block.
- Ubicando bloques de tipo “Constant”.

Los parámetros internos de la columna son definidos en el código. Las entradas y perturbaciones (las cuales pueden modificarse en el transcurso de la emulación) se ingresan a través de un archivo de ScicosLab en el caso de las emulaciones de

estado estacionario o a través de bloques “Constant” y “Step” en las emulaciones de estrategias de control.

Configuración del target

Etapas en la cual se ajusta el “solver” o rutina de solución del sistema de ecuaciones. En ScicosLab / Scicos se dispone de las siguientes rutinas:

- ode1: método numérico de Euler (Runge Kutta de primer orden)
- ode2: método de Heun o método de Euler modificado (Runge Kutta de segundo orden)
- ode4: Fórmula Runge Kutta de cuarto orden.

Además, en la etapa de configuración se ajusta el tiempo de muestreo, este tiempo es un valor crítico para la ejecución del proceso en la plataforma RTAI, puesto que un tiempo de muestreo muy alto puede llevar a respuestas erróneas e indeterminaciones por parte de la rutina de solución del sistema de ecuaciones. Por otro lado, cuanto más bajo sea el tiempo de muestreo, mayor será la carga de trabajo y consumo de recursos (memoria RAM y porcentaje de CPU) para la plataforma durante la ejecución del proceso, llevando incluso a la imposibilidad de ejecutar dicho proceso por falta de recursos en el computador.

Generación y compilación del código

Se genera y compila el código compatible con RTAI a partir del diagrama con la herramienta RTAI-CodeGen.

El ANEXO C presenta una propuesta metodológica para la emulación de una columna de destilación binaria.

5. RESULTADOS Y VALIDACIÓN

La columna de destilación binaria se emuló empleando una plataforma de emulación en tiempo real compuesta por los proyectos RTAI, RTAI-Lab y RTAI-XML. Validar el emulador de la columna de destilación obtenido en un proceso industrial o real genera algunos problemas que requieren a veces mucho tiempo. Uno de estos problemas es la adquisición del equipo de medición para poder realizar las pruebas necesarias. Otro de los problemas es el financiamiento de las instalaciones y de la mezcla a destilar. Por estos, y muchos otros problemas que se presentan, los resultados del emulador se validan con los resultados obtenidos de las simulaciones en Matlab obtenidas a partir del programa de autoría de Sigurd Skogestad [5],[31], una de las personas que más ha trabajado sobre columnas de destilación y existen varios trabajos sobre simulaciones basados en el trabajo de Skogestad.

Se realizan dos conjuntos de pruebas, las cuales difieren en el número de platos y la ubicación del plato de alimentación. En cada una de ellas se realiza la simulación para dos valores diferentes de la composición de la alimentación y del flujo de alimentación para así obtener la respuesta en la concentración de ligero o composición en el tope (Destilado), composición en el plato de alimentación y composición en el rehervidor.

Las gráficas de respuesta generadas por jRTAI-Lab están desplazadas 0.01 unidades hacia arriba con respecto a las de MATLAB, puesto que se superponen y no permitirían una observación clara.

Conjunto de pruebas 1.

Se realizó la emulación (en jRTAI-Lab) y simulación (en MATLAB) para obtener la concentración de ligero o composición en el tope (Destilado), composición en el plato de alimentación y composición en el rehervidor, variando ZF (Composición de la alimentación en fracción molar), en los siguientes tres casos: ZF=0.3 (inicial), ZF=0.5 y ZF=0.7 y bajo ciertas condiciones o valores de los parámetros dados en las tablas 5 a 7.

ZF: Composición de la alimentación (fracción molar del componente ligero).

Parámetro	Valor	Parámetro	Valor
NT	41	$qF0$	1
NF	21	$L0$	2.7
$M0$	0.5	$L0_b$	$L0 + qF0 * F0$
α	1.5	λ	0
τ_L	0.063	$V0$	3.2
$F0$	1	$V0_t$	$V0 + (1 - qF0) * F0$

Tabla 5. Parámetros de la columna.

Entrada	Valor	Perturbación	Valor
LT	2.7	F	1.0 – 1.01 – 1.05
V_B	3.2	zF	0.3 – 0.5 – 0.7
D	0.3 – 0.5	qF	1.0
B	0.5		

Tabla 6. Entradas y perturbaciones de la columna.

Composición	Valor	Retención	Valor
x_0	0.5	Mo	0.5

Tabla 7. Composiciones y retenciones iniciales.

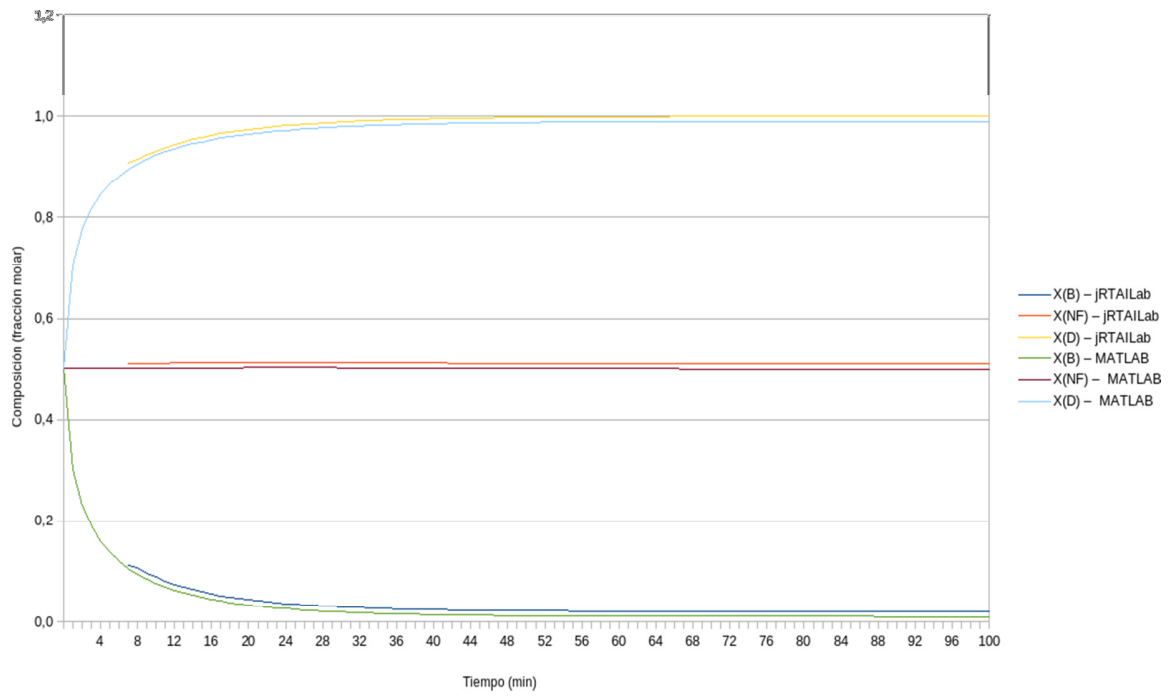


Figura 4. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

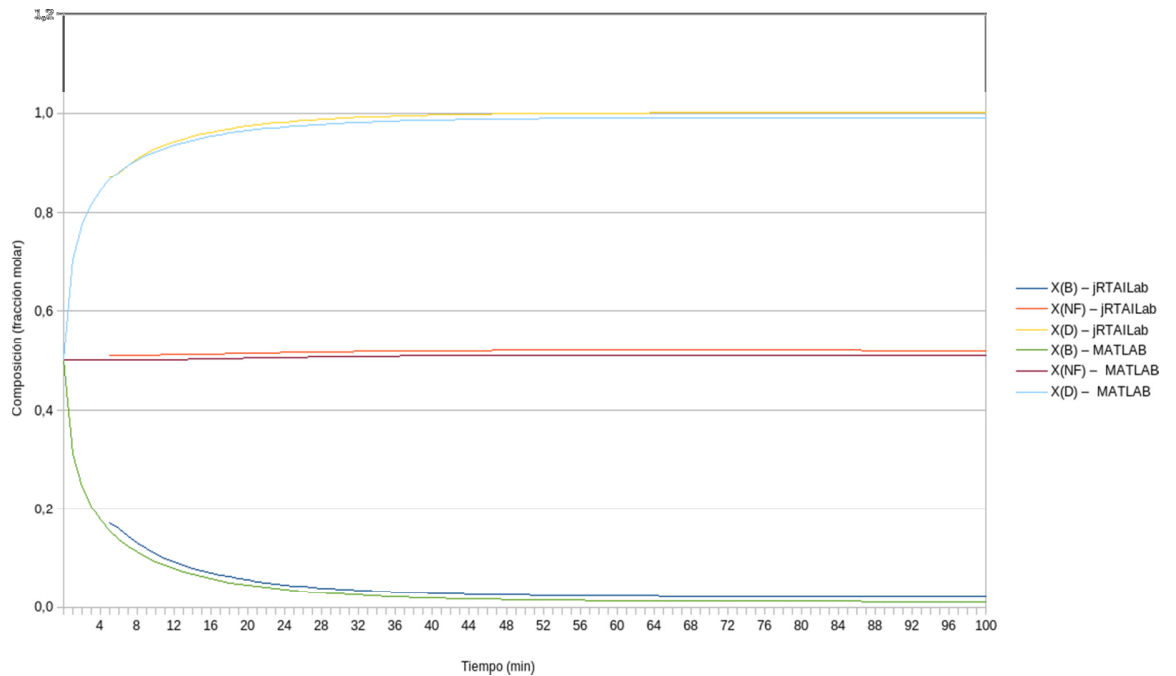


Figura 5. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.3$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

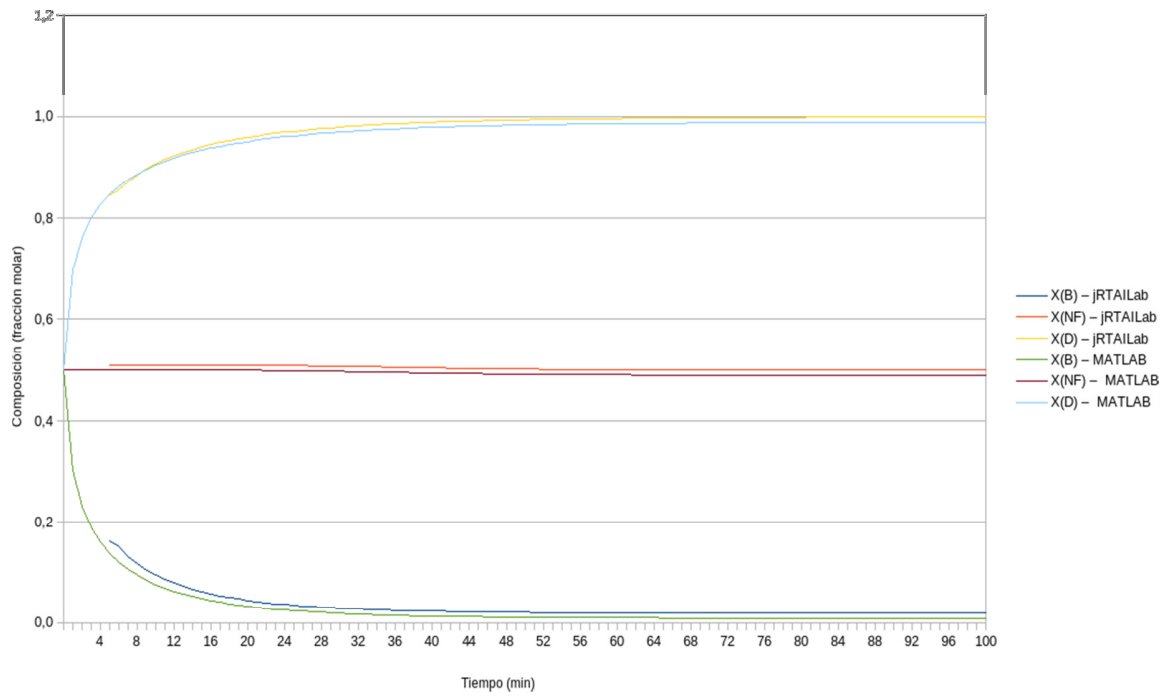


Figura 6. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.3$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

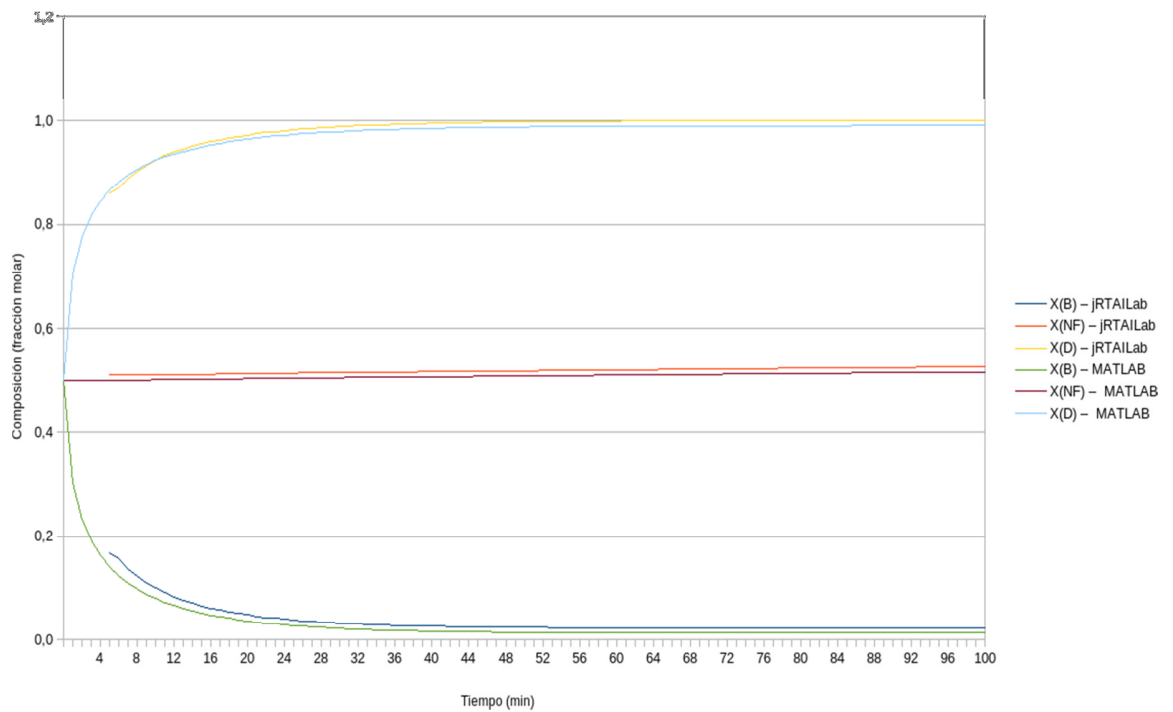


Figura 7. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.01$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

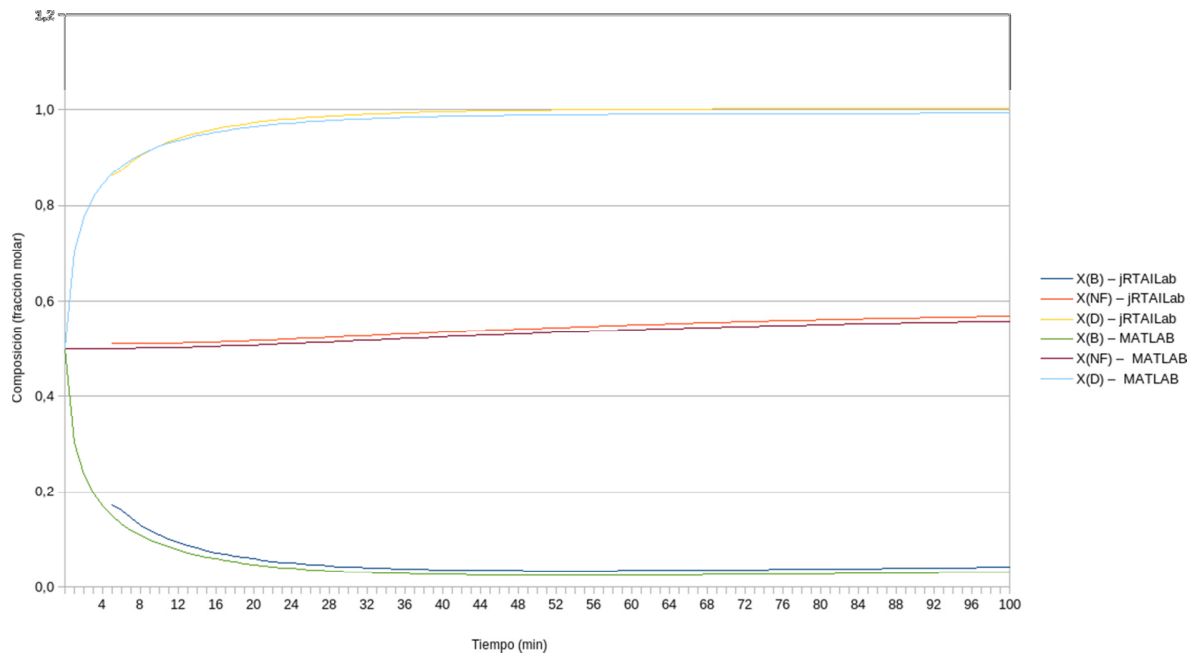


Figura 8. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.05$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

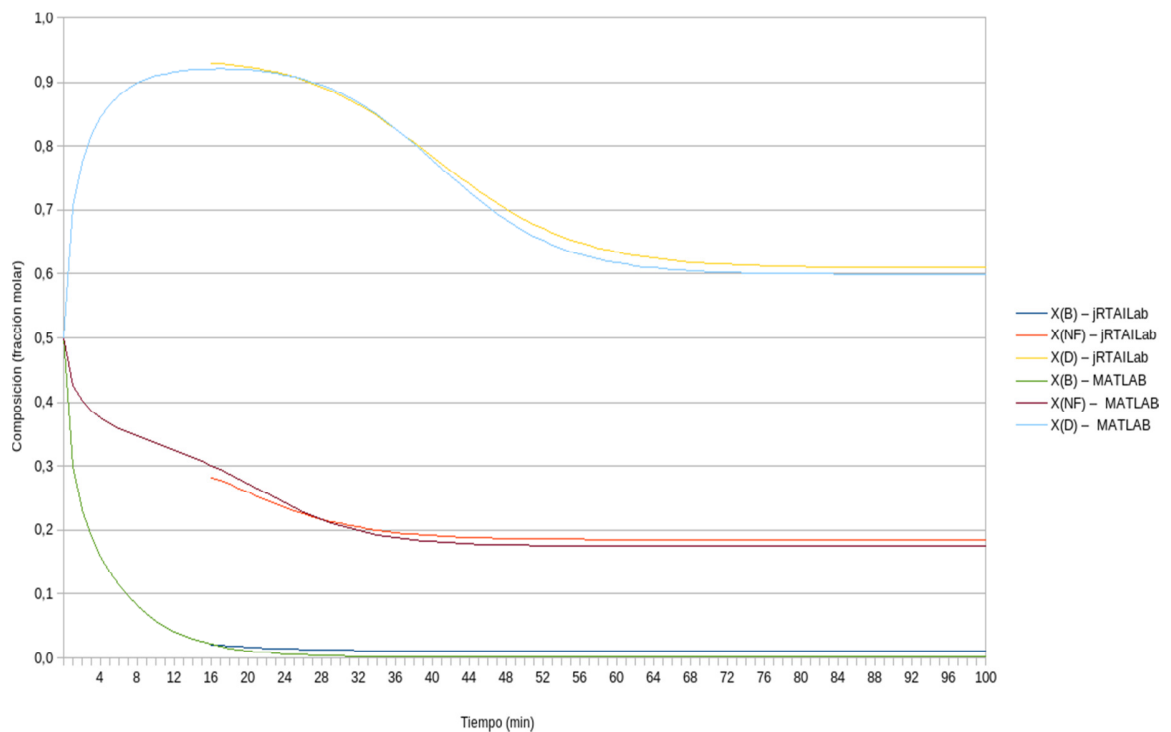


Figura 9. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.3$. jRTAI-Lab y MATLAB.

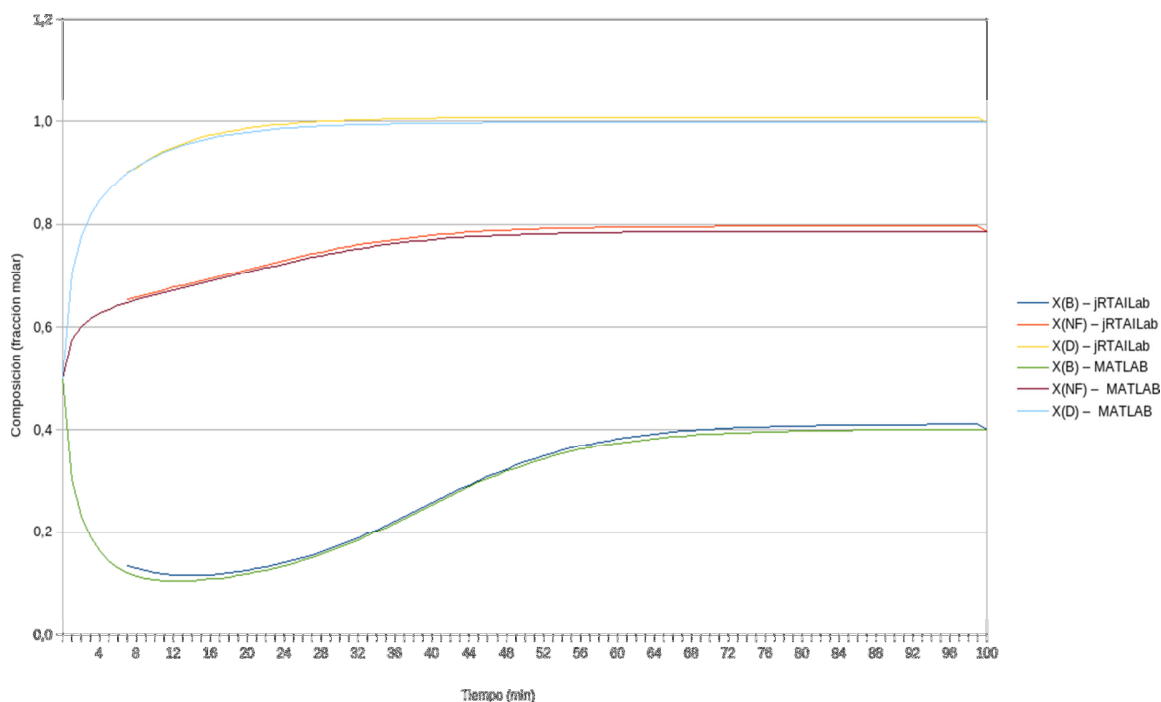


Figura 10. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.7$. jRTAI-Lab y MATLAB.

Observando los resultados obtenidos en la figura 4, se encuentra que la respuesta de la composición en el condensador, rehervidor y el plato de alimentación tiene la misma tendencia en jRTAI-Lab y en MATLAB, con un error porcentual máximo del 3% durante los primeros 10 minutos en la respuesta de la composición en el rehervidor, siendo cada vez menor dicho error (obteniendo error del 0.02%) en la medida que la respuesta se estabiliza. Además, se observa que el error es menor en las respuestas de la composición en el plato de alimentación y en el destilado frente a las respuestas correspondientes en MATLAB.

Desde el punto de vista físico – químico, presenta el mismo comportamiento de la respuesta que se espera en un proceso real, con una composición de destilado por encima de 0.9 (fracción molar) una vez se ha estabilizado y un producto de fondos por debajo de 0.03 (fracción molar).

Al cambiar el flujo de fondos $B = 0.3$ (figura 5), la tendencia de ambas respuestas sigue conservándose, sin embargo, al inicio de la gráfica, aproximadamente hasta los 25 minutos, se presenta una diferencia en los datos de la composición de fondos con un error porcentual

promedio de 4.5%. A partir de allí, la diferencia se reduce sustancialmente presentándose un error porcentual de 0.4%. En la composición de tope los datos son más cercanos entre si desde el inicio de la simulación con un error máximo de 2%.

Cuando el flujo de destilado cambia a $D = 0.3$ (figura 6), el comportamiento de la respuesta es similar al caso anterior, pero se observa que la diferencia que se presenta en los datos para la composición en el fondo es más pronunciada al comienzo (durante los primeros 25 minutos) con un error porcentual de 10%. Sin embargo, a partir de allí la diferencia es cada vez menor con un error porcentual promedio de 0.2%. De igual forma, es menor el error en las respuestas en la composición de la alimentación y destilado.

Si la variación se hace en el flujo de alimentación, con $F = 1.01$ y $F = 1.05$ (figuras 7 y 8), la respuesta mantiene la tendencia similar a los dos casos anteriores (Figuras 5 y 6) en cuanto a la variación que se presenta en la composición de fondos.

Si el cambio ahora se hace a $ZF = 0.3$ (figura 9), la evolución de las gráficas es la misma, observándose que las diferencias fuertes se presentan al inicio, con un desfase de aproximadamente 10 minutos y un error máximo del 10% (al inicio). Una vez se estabiliza, los resultados en jRTAI-Lab y MATLAB son muy cercanos presentando un error porcentual de 0.3% para la composición en el plato de alimentación y de 0.5 a 1% en el destilado.

La respuesta de la composición en este caso (figura 9), es diferente a los casos anteriores (figuras 4, 5, 6, 7 y 8), presentándose, por ejemplo, una caída de la composición en el destilado y en fondos debido a que inicialmente cada plato tiene una composición determinada y posteriormente se cambia la alimentación de la columna a un valor más bajo (ZF pasa de 0.5 a 0.3), lo cual significa que el componente ligero que hay inicialmente en la columna se evapora muy rápidamente y no se alcanza a compensar con la tasa de alimento (entrada), con lo cual la composición de salida también debe disminuir en el comportamiento físico-químico propio de la columna; cayendo de 0.9 a 0.6 (fracción molar) en el caso del destilado y, aunque normalmente el producto de fondos es bajo, al

disminuir la alimentación de la columna, éste debe caer aún más, llegando en este caso a 0.01 (fracción molar).

Para $ZF = 0.7$ (figura 10), se observa que al aumentar la alimentación, la composición en el tope se incrementa aún más porque hay mayor cantidad de componente ligero que se puede evaporar dentro de la columna. La respuesta del destilado crece de nuevo como ocurre en el proceso de una columna de destilación real. En este caso no se presenta el desfaseamiento que existe para $ZF = 0.3$ y la evolución es la esperada con un error porcentual máximo del 3.5% en la composición de fondos. El error es menor en la respuesta de la composición de destilado (0.01%) y en la del plato de alimentación desde 2% hasta 0.1% cuando alcanza el estado estable.

Conjunto de pruebas 2.

En este segundo conjunto de pruebas, se realizó la emulación cambiando el número de platos y el plato de alimentación para obtener la concentración de ligero o composición en el tope (Destilado), composición en el plato de alimentación y composición en el rehervidor, variando ZF (Composición de la alimentación en fracción molar), en los siguientes tres casos: $ZF=0.3$ (inicial), $ZF=0.5$ y $ZF=0.7$ y bajo ciertas condiciones o valores de los parámetros dados en las tablas 8 a 10.

Parámetro	Valor	Parámetro	Valor
NT	22	$qF0$	1
NF	11	$L0$	2.7
$M0$	0.5	$L0_b$	$L0 + qF0 * F0$
α	1.5	λ	0
τ_L	0.063	$V0$	3.2
$F0$	1	$V0_t$	$V0 + (1 - qF0) * F0$

Tabla 8. Parámetros de la columna.

Entrada	Valor	Perturbación	Valor
	2.7		1.0 – 1.01 – 1.05
	3.2		0.3 – 0.5 – 0.7
	0.5		1.0
	0.5		

Tabla 9. Entradas y perturbaciones de la columna.

Composición	Valor	Retención	Valor
	0.5		0.5

Tabla 10. Composiciones y retenciones iniciales.

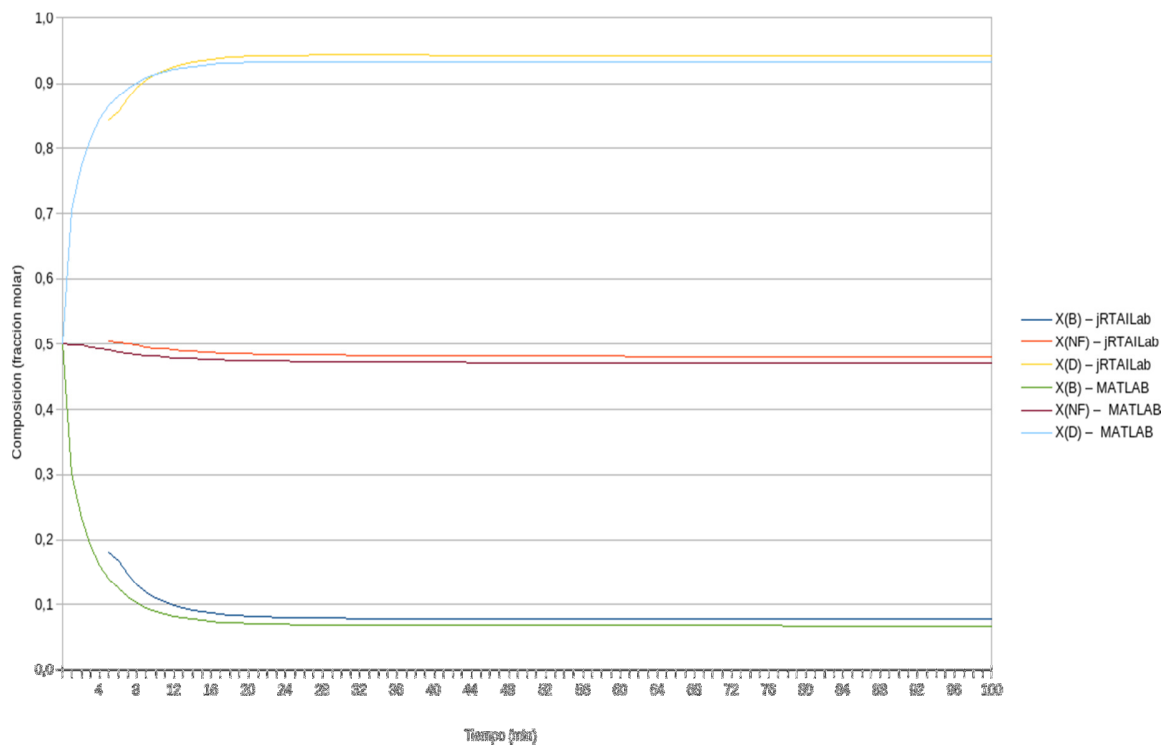


Figura 11. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

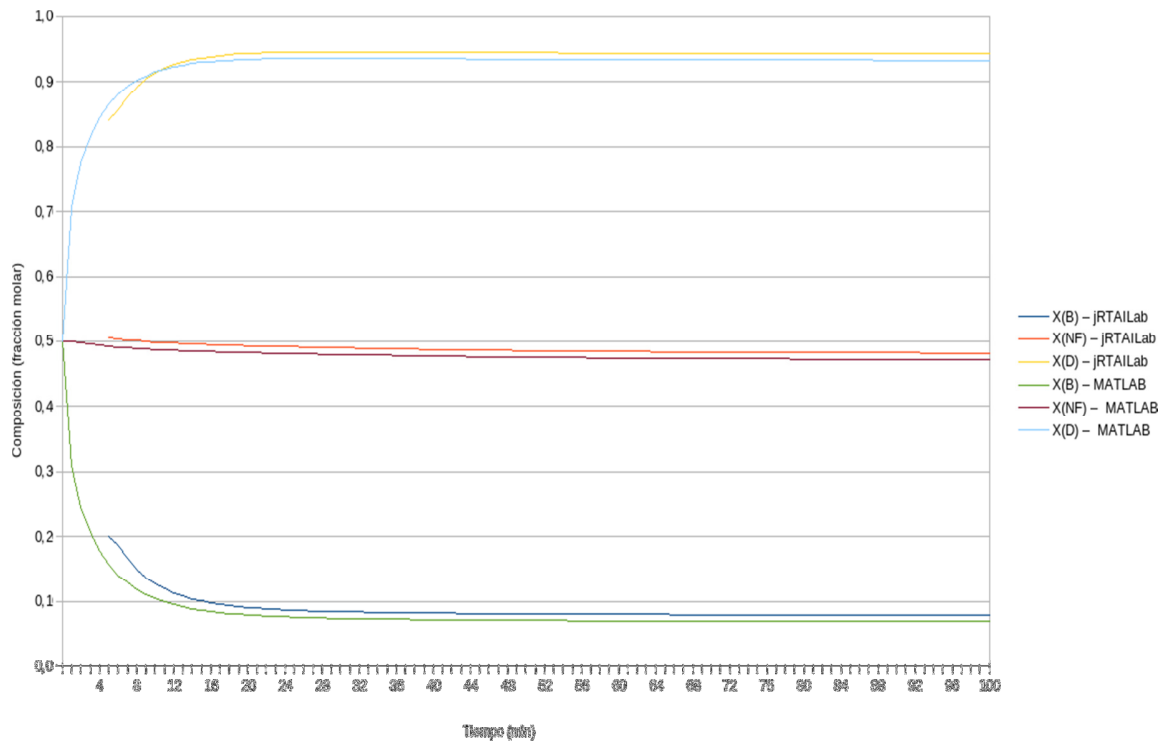


Figura 6. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.3$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

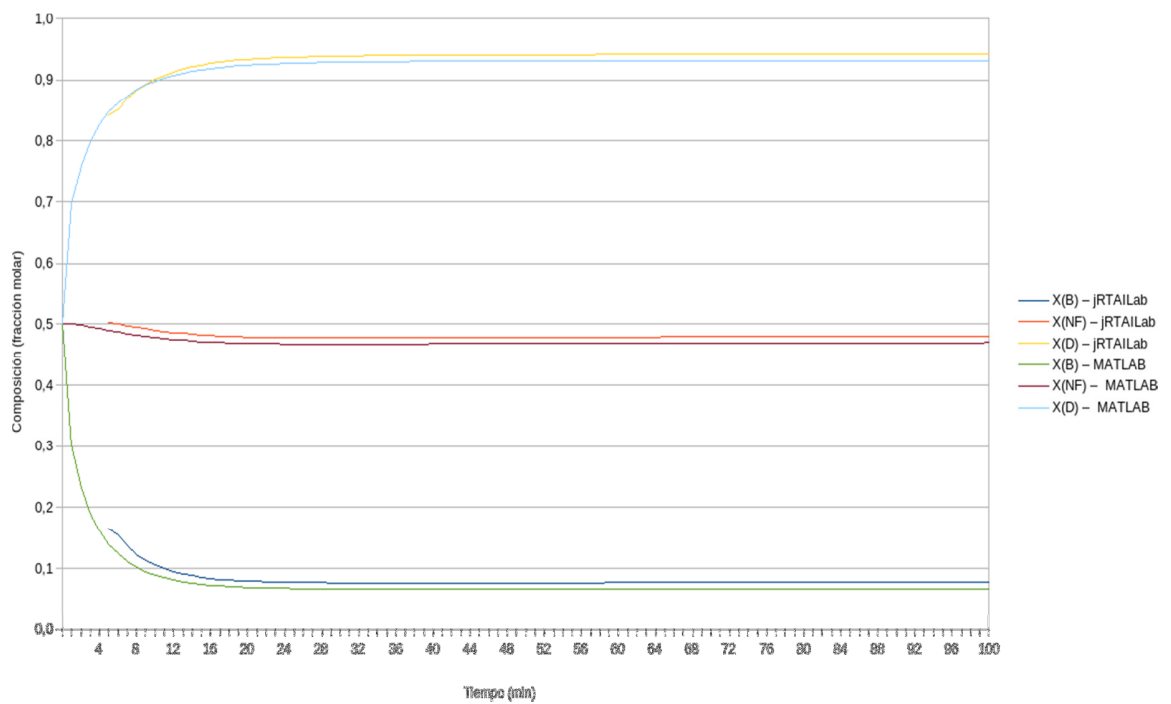


Figura 7. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.3$, $B = 0.5$, $F = 1$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

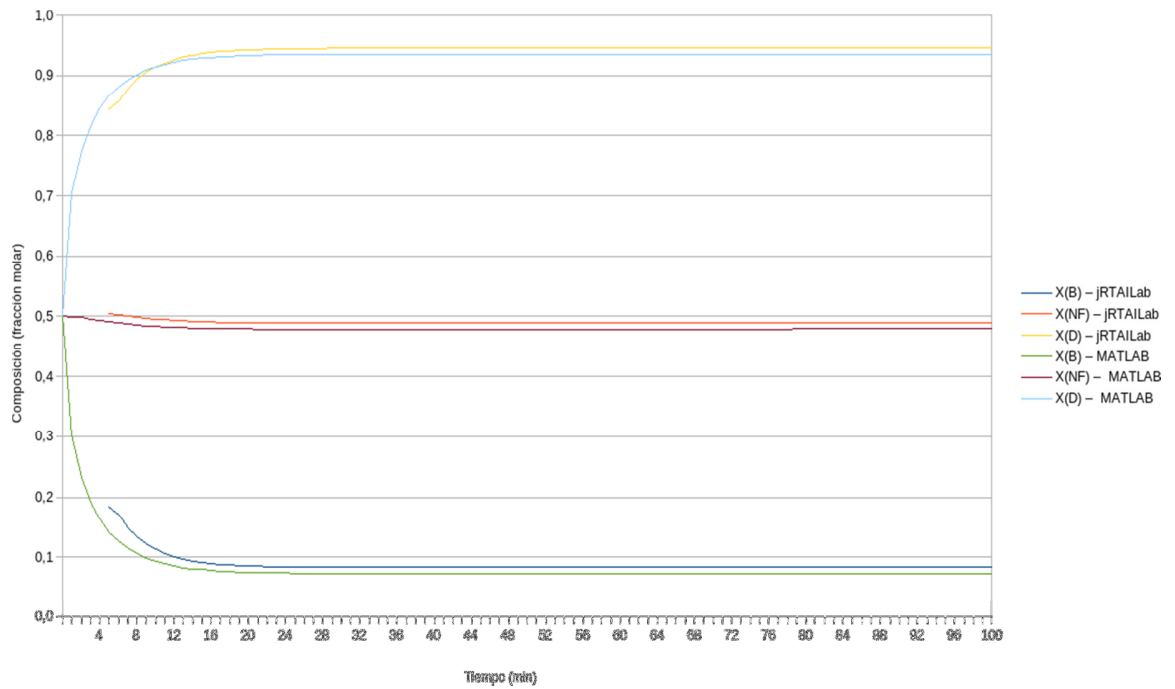


Figura 8. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.01$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

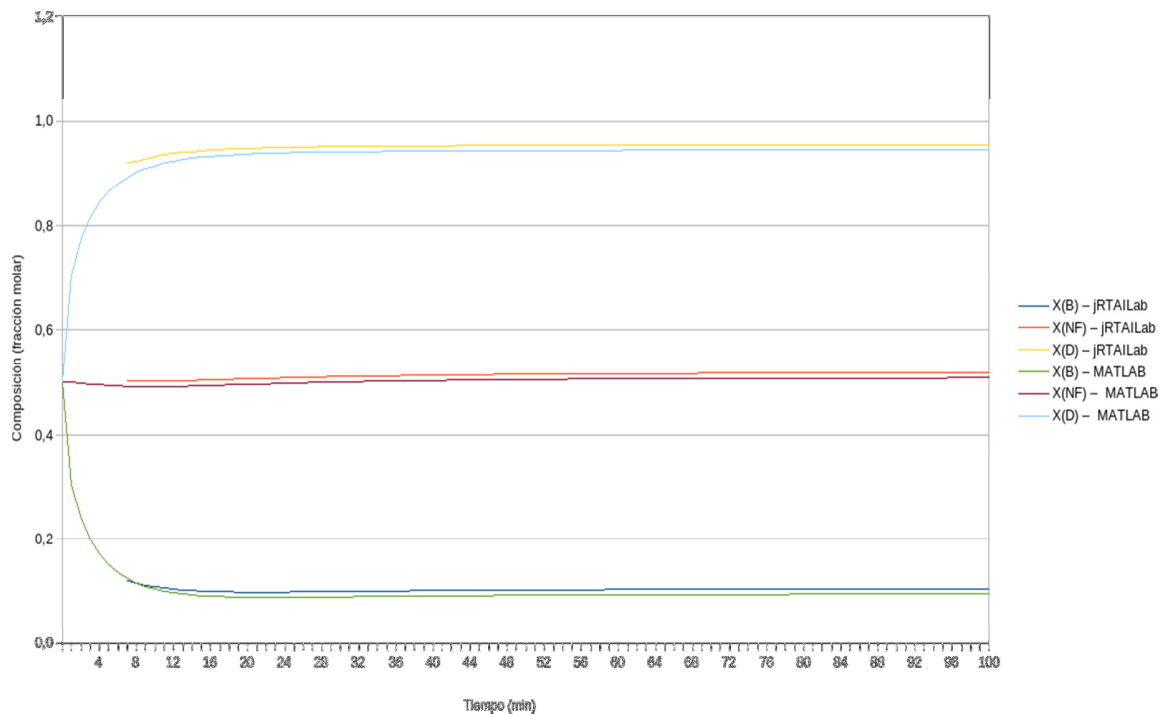


Figura 9. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1.05$, $ZF = 0.5$. jRTAI-Lab y MATLAB.

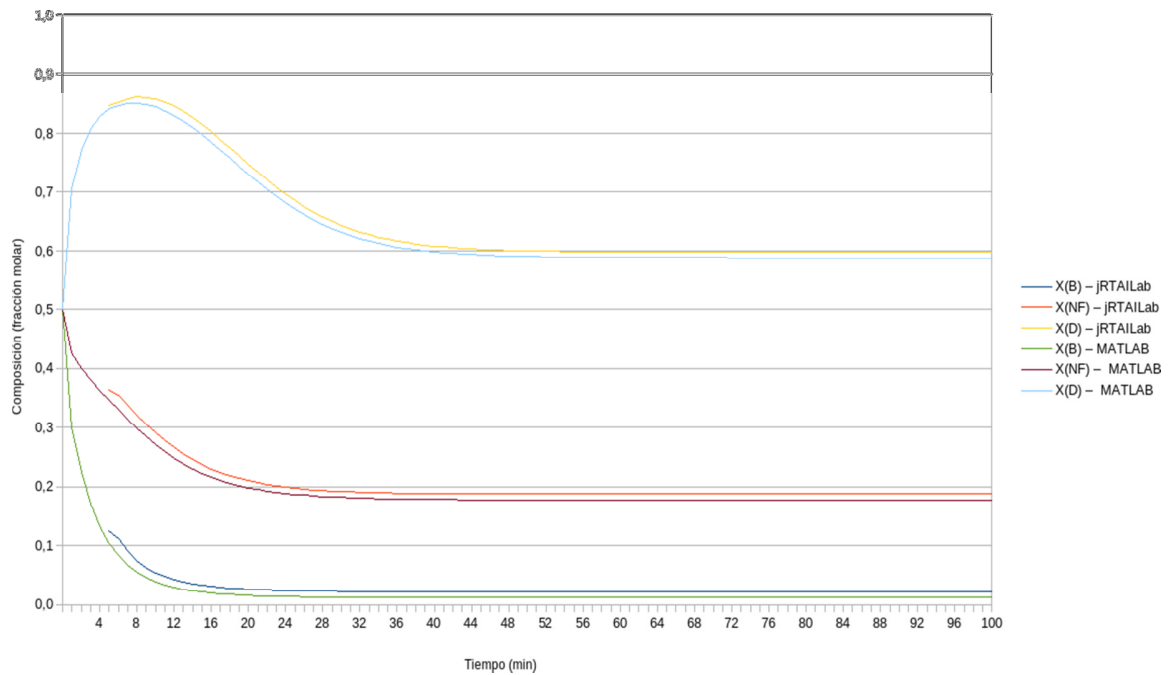


Figura 16. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.3$. jRTAI-Lab y MATLAB.

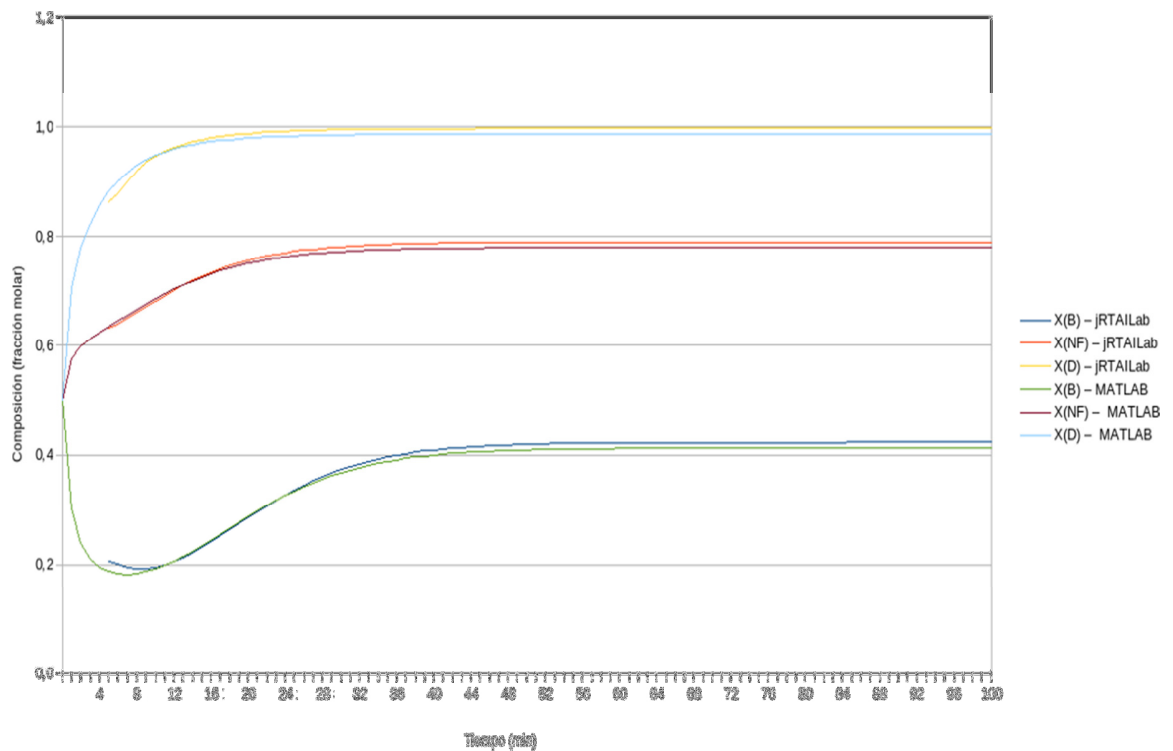


Figura 10. Respuesta de composiciones en el condensador, rehervidor y plato de alimentación. Para $D = 0.5$, $B = 0.5$, $F = 1$, $ZF = 0.7$. jRTAI-Lab y MATLAB.

Para un número menor de platos $NT = 22$, se puede observar que la tendencia de las respuestas obtenidas es similar al caso con $NT = 41$ etapas, sin embargo, las diferencias al inicio son mayores para $NT = 22$, aunque a medida que se va estabilizando el error porcentual es menor.

Para emulaciones adicionales se observó que el flujo de alimentación solo admite cambios muy pequeños (del orden del 1%), de lo contrario, la respuesta tiende a indeterminarse. Este fenómeno lo analiza en más detalle Troung [11].

Las diferencias que se presentan entre las respuestas del emulador y las obtenidas de la simulación en MATLAB, se deben principalmente a las características de la plataforma utilizada para realizar la emulación en cuanto a software se refiere especialmente, por ejemplo, en la precisión del método que utiliza la máquina para resolver el sistema de ecuaciones del modelo.

Se observa que el tiempo de muestreo es un valor crítico para la ejecución del emulador en la plataforma RTAI, de tal forma que con tiempos de muestreo muy altos se obtienen respuestas erróneas e indeterminaciones por parte de la rutina de solución del sistema de ecuaciones. Por otro lado, cuanto más bajo es el tiempo de muestreo, mayor es la carga de trabajo y consumo de recursos (memoria RAM y porcentaje de CPU) para la plataforma durante la ejecución del proceso y, dependiendo del valor, presenta incluso la imposibilidad de ejecutar dicho proceso por falta de recursos en el computador.

A diferencia de la simulación en Matlab, el emulador desarrollado permite introducir cambios en el proceso en tiempo ejecución. En una simulación (en MATLAB) no es posible hacerlo de esta forma. Además, se destaca el hecho de haber sido desarrollado usando herramientas de software libre ya que no se encuentran desarrollos sobre emulación de columnas de destilación utilizando este tipo de software.

La solución demostró una efectividad adecuada en la emulación del sistema abordado, generando una herramienta útil para evaluar sistemas de control y/o estrategias de optimización. Asimismo, la evaluación propuesta se validó con ejemplos claramente descritos”.

6. CONCLUSIONES, RECOMENDACIONES Y FUTUROS TRABAJOS

6.1. CONCLUSIONES

- La importancia del emulador desarrollado, es que permite introducir cambios en el proceso en tiempo ejecución. En una simulación no es posible hacerlo de esta forma. Además, se destaca el hecho de haber sido desarrollado usando herramientas de software libre ya que no se encuentran desarrollos sobre emulación de columnas de destilación utilizando este tipo de software. Se encuentran trabajos alrededor de otros tipos de aplicaciones, que proponen la emulación de procesos como la destilación de plantas petroquímicas; sin embargo, utilizando software de alto costo y muy específico.
- El comportamiento satisfactorio del emulador se contrastó a través de comparaciones de los datos emulados vs. datos simulados por autores muy respetados en el estudio y control de columnas de destilación como es el caso del Dr. S. Skogestad [32]. La solución demostró una efectividad adecuada en la emulación del sistema abordado, generando una herramienta útil para estudiar sistemas de control. Asimismo, la evaluación propuesta se validó con ejemplos claramente descritos.
- Los resultados de la emulación muestran que la respuesta de la composición a cambios en las entradas o cambios en las perturbaciones se acerca a la respuesta obtenida con la simulación en MATLAB, con un error porcentual en rangos aceptables; sin embargo, en algunos intervalos de tiempo para algunos casos emulados se presentan diferencias que aunque notorias, no son críticas para llevar a cabo el proceso de emulación.
- Dentro de las limitaciones del emulador, se encuentra que se debe limitar los valores que pueden tomar los parámetros de la columna, puesto que algunos de estos indeterminan el sistema de ecuaciones que representa la columna. Por ejemplo, para

un flujo de alimentación menor a uno, la respuesta de la composición en el condensador, rehervidor y plato de alimentación tiende a presentar asíntotas. Lo anterior permite concluir que algunas variables como el flujo de alimentación solo admiten cambios muy pequeños (del orden del 1%).

- El tiempo de muestreo es un valor crítico para la ejecución del emulador en la plataforma RTAI, puesto que un tiempo de muestreo muy alto puede llevar a respuestas erróneas e indeterminaciones en la rutina de solución del sistema de ecuaciones. Por otro lado, cuanto más bajo sea el tiempo de muestreo, mayor será la carga de trabajo y consumo de recursos (memoria RAM y porcentaje de CPU) para la plataforma durante la ejecución del proceso, llevando incluso a la imposibilidad de ejecutar dicho proceso por falta de recursos en el computador.

APORTES

El principal aporte de este trabajo fue el diseñar un emulador para columnas de destilación binaria que permite experimentar cambios en las variables en tiempo ejecución para analizar el comportamiento y la dinámica del sistema desarrollado usando herramientas de software libre.

La mayoría de los trabajos sobre columnas de destilación hacen únicamente simulación y los pocos que trabajan simulación en tiempo real, lo hacen sobre plataformas muy específicas y de alto costo.

El emulador propuesto puede ser utilizado principalmente para el desarrollo académico como recurso de experimentación que permite el estudio de columnas de destilación binaria.

6.2. RECOMENDACIONES

- En vista de que el número de etapas está limitado por las capacidades en hardware de la máquina en que se ejecuta el emulador, es conveniente disponer de equipos actualizados en cuanto a su capacidad en hardware, de tal forma que permita a futuro trabajar con columnas de mayor número de etapas.

- Es importante mantener el software que componen los proyectos RTAI, RTAI-Lab, RTAI-XML y jRTAI-Lab actualizado a las últimas versiones estables disponibles, ya que esto permite una mayor eficiencia y corrección de errores en dicho software.

6.3. FUTUROS TRABAJOS

- Un trabajo que puede ser importante es el desarrollo de una Interfaz con hardware externo para usar controladores físicos para una columna de destilación binaria, es decir, sistemas HIL (Hardware in the Loop).
- Un trabajo paralelo consistiría en el desarrollo de una interfaz gráfica que permita un mejor entendimiento del proceso emulado y trabajar en el modelamiento más riguroso de la columna de destilación.
- Después de haber obtenido el emulador de una columna de destilación, el trabajo futuro puede consistir en el desarrollo de emuladores para otros tipos de destilación que son muy comunes en la industria como la destilación azeotrópica y la destilación multicomponente.
- La validación frente al proceso real es un trabajo que puede realizarse a futuro considerando el compromiso desde el inicio de una empresa que cuente con una planta de destilación y que estén dispuestos a realizar los aportes tanto en tiempo como en materia prima para realizar las diferentes pruebas requeridas en el trabajo de investigación.

REFERENCIAS BIBLIOGRÁFICAS

- [1]. **Holland, Ch. D.** *Fundamentos de Destilación de Mezclas Multicomponentes*, Editorial Limusa, S.A. de C. V., 1988.
- [2]. **Holman, J. P.** "Thermodynamics", McGraw-Hill Book Company, 1988.
- [3]. **Klaus, Erich; Schmitz, Abe.** *Modelado y Simulación de Una Torre de destilación Multicomponente*. Submitted the Departamento de Mecánica del Instituto Tecnológico de Puebla in fulfillment of the requirement for the degree of maestro en Ciencias at the Instituto Politécnico Nacional. 1999.

- [4]. **Grupo GICI.** *Plataforma de Procesamiento Distribuido con Acceso Remoto Multi - Usuario y Emulación de Sistemas Dinámicos para investigación y Educación en Ingeniería. Plataforma PERI.* Escuela de Ingeniería Eléctrica y Electrónica. Universidad del Valle. Cali – Colombia. 2006 - 2008. <http://eieela.univalle.edu.co/>
- [5]. **Skogestad, Sigurd.** “*Dynamics and Control of Distillation Columns - a critical survey*”. Modelamiento, Identificación y Control. 1997. Vol. 18. No. 3. 177-217. Noruega.
- [6]. **Basualdo, Marta.** *Dynamic Simulation of Chemical Process as a Tool to Teach The Real Problem of Identification and Control.* Departamento de Electrónica. FCElyA. UNRE-mail basualdoQifir.edu.ar. Rosario. Argentina. 1995. 0-7803-3022-601995 IEEE.
- [7]. **Arbildo Lopez, A.; Lombira Echevarria, J.; Osorio López, I.** *Simulación dinámica y control difuso de una columna de destilación continua.* Unidad de Postgrado, Facultad de Química e Ingeniería Química. Universidad Nacional Mayor de San Marcos, Perú. 2002, Revista Peruana de Química e Ingeniería Química, Vol. 5, No 1, pags. 47-57.
- [8]. **Gorriño, Juan P.; Arias, Pedro L.; Legarreta, Juan A.** *Simulation Model to optimize distillation processes.* Department of Graphic Communications and Engineering, Engineering School of Bilbao, Universidad del País Vasco, España. 2002.
- [9]. **Cardona, Carlos A.; Sánchez, Oscar J.; Montoya, María I.; Quintero Julián A.** *Simulación de los procesos de obtención de etanol a partir de caña de azúcar y maíz.* Departamento de Ingeniería Química, Universidad Nacional de Colombia. 2005. Scientia et Technica Año XI No 28 Octubre UTP. ISSN 0122-1701.
- [10]. **Uyazán, A.M.; Gil, I.D.; Aguilar, J.L.; Rodríguez, G.; Caicedo, L.A.** *Produccion de alcohol carburante por destilación azeotrópica homogénea con glicerina.* Grupo de Procesos Químicos y Bioquímicos - Departamento de Ingeniería Química Universidad Nacional de Colombia – Bogotá, Colombia. Ciudad Universitaria Calle 45 con Carrera 30, Bogotá. 2003.
- [11]. **Truong, H. S.; Ismail, I.; Razali, R.** *Fundamental modeling and simulation of a binary continuous distillation column.* H. S. Electrical and Electronic Engineering Department Universiti Teknologi PETRONAS 31750 Tronoh, Perak, Malaysia. 2010.
- [12]. **Chang, G.W.; Liu, Y.J.; Dinavahi, V.; Ke, M. J.;** *Applications of Real-Time Simulation Techniques for Harmonics Study of Industrial Power System.* Department of Electrical Engineering, National Chung Cheng University, Min-Hsinug, Chia-Yi 621 Taiwan, R.O.C. Department of Electical and Computer Engineering, University of Alberta, Canada. 2008 IEEE.

- [13]. **Dixit, V.; Patil, M. B.; Chandorkar, M.C.** *Real Time Simulation of Power Electronic Systems on Multi-core Processor*. Department Of Electrical Engineering. Indian Institute of Technology Bombay. Mumbai-400076, India. PEDS 2009.
- [14]. **Santana Cobo, José David.** *Diseño e implementación de un emulador de plantas SISO analógicas*. Facultad de Ingeniería Electrónica, Escuela Politécnica del Ejército. Sangolquí, Ecuador.
- [15]. **Carvalho, Adelson; Barone, Dante; Zaro, Milton.** *A Technological platform for the teaching of control engineering*. Universidad Federal de Rio Grande do Sul UFRGS. Porto Alegre, Brasil. 2010. IEEE.
- [16]. **Bao, Guozhong.** *SIS Series Simulator and its Application in Petrochemical Plant Operations Training*. Automation Department. Fushun Institute of Petroleum. Fushun, 113001, Liaoning, P.R. China. 1997 IEEE.
- [17]. **Li, X.; Wu, C.; Hu, X.** *Full Scope Real Time Simulation of Hydropower Plant for a Training and Research Simulator*. College of Electrical Engineering and information Technology, China Three Gorges University, Yichang City, 443002, Hubei, China. 2005 IEEE.
- [18]. **Curti, M.R.; Magallán, G.A.; De Angelo, C.H.; García, C.O.** *Emulación Dinámica de una Turbina Eólica*. Grupo de Electrónica Aplicada. Facultad de Ingeniería. Universidad Nacional de Rio Cuarto, Córdoba, Argentina. AADECA 2006 – XX Congreso Argentino de Control Automático. Buenos Aires Argentina.
- [19]. **Zang, T.; Wang, A.** *Design and Application of the Hardware-in-The-Loop Simulation System for Distillation Columns*. School of Mathematics and Information Engineering, Taizhou University, Taizhou Zhejiang, China. 2009 IEEE.
- [20]. **Jacobsen, Elling.** *Studies on dynamics and control of DistillationColumns*. The Norwegian Institute of Technology. University of Trondheim. Trondheim.1991.
- [21]. **H. S. Truong, I. Ismail, R. Razali.** *Fundamental Modeling and Simulation of a Binary Continuous Distillation Column*. Electrical and Electronic Engineering Department Universiti Teknologi PETRONAS. 31750 Tronoh, Perak, Malaysia
- [22] **Z. Abdullah, N. Aziz and Z. Ahmad.** “Nonlinear Modelling Application in Distillation Column”. Chemical Product and Process Modelling: Vol. 2:Iss. 3, Artical 12. (2007). Available at:<http://www.bepress.com/cppm/vol2/iss3/12>.
- [23] **S. Skogestad y Morari M.,** “La comprensión del comportamiento dinámico de columnas de destilación”, Ind. Y el Ing.. Chem.. Investigación , 27 , 10, 1848-1862 (1988).

- [24]. **Dominguez, Hugo; Peña, Leonardo; Mulet, Mónica.** *Simulación en Tiempo Real de una Columna de Destilación para la Prueba de Controladores Programables.* Revista de Tecnología Química, Vol. XXII, No. 2, 2002. Universidad de Oriente.
- [25]. **Gómez, J. C.; Vaschetti, J. C.; Ibarlucea, C.; Coyos, C.; Modesti, M.** *Simulador de Entrenamiento y Certificación, para Operadores de Centro Control.* Universidad Nacional de Río Cuarto, Córdoba, Argentina. 2006.
- [26]. **Jack, A. G.; Atkinson, D. J.; Slater, H. J.** *Real-time emulation for power equipment development. Part 1: Real-time simulation.* Department of Electrical and Electronic Engineering, Merz Court, University of Newcastle upon Tyne. Marzo 1998. IEEE 19981753.
- [27]. **Modesti, Mario R.; Vaschetti, Jorge C.; Canali, Luis R.** *Plataforma de simulación de columna de destilación en tiempo real vía internet.* Universidad Tecnológica Nacional. Argentina 2003.
- [28]. **Organization for Economic Co-operation and Development (OECD) – Traducción por la Fundación Española de Ciencia y Tecnología (FECYT).** *Manual de Frascati. Propuesta de Norma Práctica para Encuestas de Investigación y Desarrollo Experimental,* 2002.
- [29]. **Ramirez, E. L.; Caicedo, H.; Pinedo, C.; Bacca, E.; Ramos, C.** *A platform for signals and systems internet-based education,* in Inted conference Valencia, España, 2008.
- [30] **Ivar J. Halvorsen, S. Skogestad.** “*Minimum Energy Requirements in Complex Distillation Arrangements*”. Department of Chemical Engineering, Norwegian University of Science and Technology. N-7491 Trondheim, Norway. Mayo 2001.
- [31]. **S. Skogestad.** “*Dynamics and Control of Distillation Columns – A Critical Review*”. Presented at IFAC-symposium DYCORS’92, Maryland, Apr. 27-29, 1992
- [32]. **A. Alzate.** “*Modelado y control de una columna de destilación binaria*”. Facultad de Ingeniería y Arquitectura. Departamento de Eléctrica, Electrónica y Computación. Universidad Nacional de Colombia - Sede Manizales. 2010.
- [33]. **C.A. Ramos Paja, A. Romero Nevado, R. Giral, E. Vidal-Idiarte, L. Martinez-Salamero.** “*Fuzzy based modelling Thechnique for PEMFC electrical power generation systems emulation*”. IET Power Electronics, 2 (3), pp. 241-255. (2009).
- [34]. **C.A. Ramos Paja, A. Romero Nevado, R. Giral Castillon, L. Martinez-Salamero, C.I. Sanchez Saenz.** “*Swtching and linear power stages evaluationfor PEM fuel cell emulation*”. International Journal Of Circuit Theory And Applications, 39 (5), pp. 475-479. (2011).

[35]. C. Restrepo,. C.A. Ramos-Paja, R. Giral, J. Calvente, A. Romero. “*Fuel cell emulator for oxygen excess ratio stimation on power electronics applications*”.Computers and Electrical Engineering, 38 (4), pp. 926-937. (2012).

ANEXO A.

PLATAFORMA DE PROCESAMIENTO DISTRIBUIDO PARA EMULACIÓN DE PROCESOS EN INGENIERÍA - PERI

La Plataforma de Procesamiento Distribuido para Emulación de Procesos en Ingeniería [4], fue concebida de acuerdo a especificaciones requeridas dentro de un proyecto en el cual se está trabajando en la Universidad del Valle con el apoyo de Colciencias llamado PERI (Plataforma para Educación Remota en Ingeniería - Colciencias), donde se requería la implementación y validación de una plataforma que permitiera emular sistemas físicos dinámicos de manera distribuida, este proyecto sería usado para educación en ingeniería.

Buscando cumplir con los objetivos con el que fue concebida la plataforma, sería necesario entonces implementar un cluster de computadores conectados a una red de alta velocidad mediante un “switch” como recurso físico para hacer el procesamiento distribuido de las emulaciones, para este proyecto se asignaron 4 máquinas encargadas de cumplir con este propósito. La Plataforma debía cumplir con ciertas especificaciones para poder simular procesos en tiempo real, por lo cual fue entonces necesario el uso de un sistema operativo que ejecutara tareas de este tipo.

También sería necesario el uso de tarjetas de adquisición de datos para interactuar con el medio físico y hacer implementaciones HIL para emular procesos. Las emulaciones debían interactuar con bloques de hardware y software, los cuales tendrían que implementarse mediante un programa de procesamiento matemático que contara con la opción de manejo de bloques implementando opciones en el que le permitieran interactuar con el sistema operativo de tiempo real.

Las Plataforma debía ser capaz de distribuir las emulaciones, de forma que el usuario estuviese habilitado para emular usando la planta o el controlador en forma de modelo matemático o en forma física, pudiendo manipular el intercambio de estos a su conveniencia.

La plataforma debía contar también con una interfaz de usuario que le permitiera acceder al sistema de manera amigable y poder ejecutar sus procesos desde allí.

Recomendación: Para tener una mejor experiencia favor descargue la última versión de Java JRE(Java Runtime Enviroment) de la página de Sun-Oracle, si desea un instalador póngase en contacto con los monitores de la plataforma.

ANEXO B

DESCRIPCIÓN DE PROYECTOS UTILIZADOS EN EL EMULADOR

RTAI

RTAI es una Interface de Aplicación en Tiempo Real. Estrictamente hablando, no es un sistema operativo de tiempo real, como VXWorks y QNX. Se basa en el kernel Linux, proporcionando a éste último algunas de las características de un sistema operativo en tiempo real.

Linux es un sistema operativo estándar de tiempo compartido que proporciona un buen rendimiento promedio y servicios altamente sofisticados. Al igual que otros sistemas operativos, ofrece a las aplicaciones al menos los siguientes servicios:

- Capa de abstracción de hardware HAL que permite tratar colas de eventos o interrupciones generadas por periféricos o por el procesador.
- Clases de planificación que se ocupan de la activación de procesos, prioridades, intervalos de tiempo
- Comunicación entre aplicaciones.

Linux carece de soporte de aplicaciones en tiempo real. Para obtener un comportamiento correcto en el tiempo, es necesario hacer algunos cambios en las fuentes del kernel, por ejemplo, en las políticas de planificación y manejo de interrupciones. De esta manera, usted puede tener una plataforma en tiempo real, con requerimientos de baja latencia y alta predicibilidad, en un ambiente Linux que no es en tiempo real Linux (acceso a TCP IP, servidor gráfico y los sistemas de ventanas, sistemas de bases de datos y archivos, etc.).

Consiste básicamente de un gestor de interrupciones: RTAI principalmente captura las interrupciones de periféricos y si es necesario las redirige a Linux. No se trata de una modificación intrusiva del kernel, sino que utiliza el concepto de HAL (capa de abstracción de hardware) para obtener información de Linux y para capturar algunas funciones fundamentales. Este HAL proporciona unas pocas dependencias al Kernel Linux. Esto lleva a una simple adaptación en el kernel Linux. RTAI considera a Linux como una tarea funcionando en segundo plano cuando no hay actividad en tiempo real.

RTAI-Lab

El proyecto RTAI-Lab provee una cadena de herramientas para desarrollar diagramas de bloques que se pueden compilar y ejecutar en una plataforma RTAI. RTAI-Lab está incluido en la distribución RTAI.

Los diagramas de bloques se pueden desarrollar utilizando Scilab / Scicos (Open Source) o Matlab / Simulink / RTW (comercial), [www.rtai.org].

RTAI-Lab presenta las siguientes características:

RTAI-Lab proporciona a Scicos una paleta de bloques llamada RTAI-Lib para desarrollar diagramas de bloques en tiempo real.

Permite la generación automática de código en tiempo real de Scilab / Scicos.

Permite portar proyectos de Matlab / Simulink / Real-Time Workshop a RTAI.

Proporciona una Interface con hardware de adquisición de señal y otros dispositivos compatibles con Comedi.

RTAI-XML

RTAI-XML es un componente de tipo servidor del proyecto RTAI (Interface de Aplicación en Tiempo Real), este implementa una forma orientada a servicios para diseñar y desarrollar aplicaciones de control en tiempo real (RT).

Consiste en un componente servidor que espera las llamadas entrantes en una caja donde se está ejecutando (o listo para ejecutarse) un proceso en tiempo real, denominado el “Target”. Un programa cliente genérico, el Host, puede conectarse al servidor a través de la red TCP/IP, utilizando un protocolo estándar basado en XML, y permitiendo interactuar

con el “Target”, con el fin de supervisar el estado del proceso en tiempo real RT, para ver las señales generadas por el sistema y también obtener y cambiar los parámetros del proceso (por ejemplo, las ganancias de un lazo de realimentación PID). En otras palabras, RTAI -XML ofrece un modo simple para controlar remotamente aplicaciones RT, añadiendo flexibilidad al proyecto RTAI, sin perder las características clave de ser una implementación abierta y estándar.

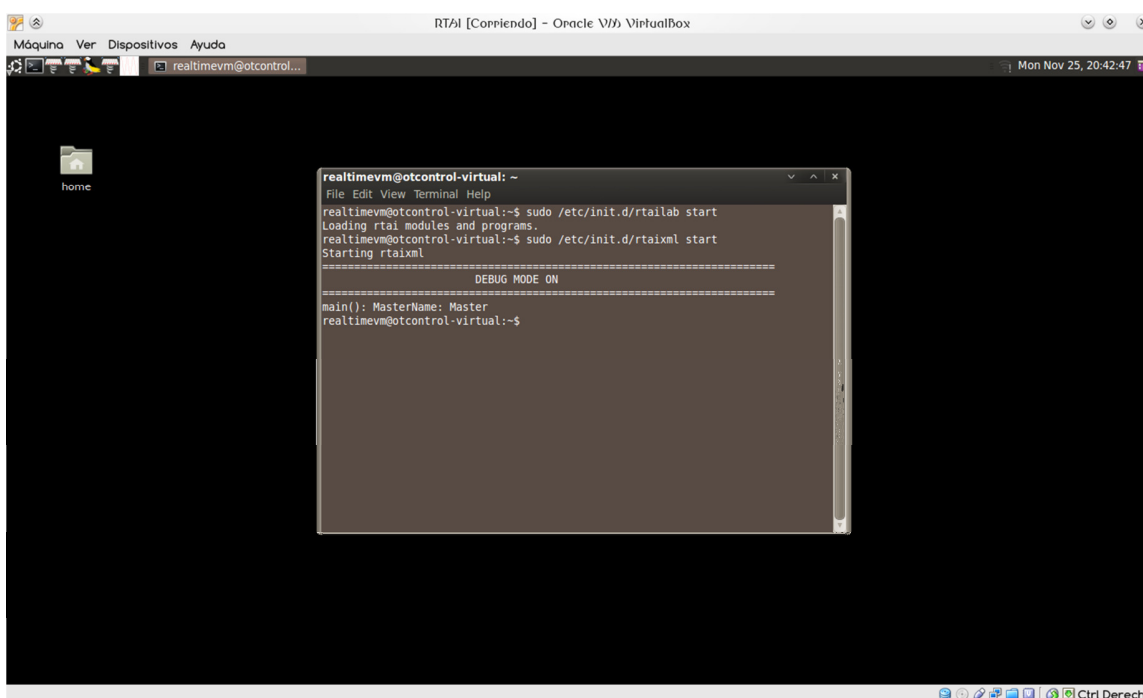
ANEXO C

METODO PROPUESTO PARA EMULACIÓN DE COLUMNAS DE DESTILACIÓN BINARIAS CONTINUAS

1. Disponer de un entorno de trabajo en el que se puedan construir los diagramas de bloques y generar los ejecutables. Para cumplir este requerimiento se tienen tres opciones:
 - La página en sourceforge del proyecto RTAI-XML (<http://sourceforge.net/projects/rtaixml/files>) proporciona una máquina virtual en el formato de *Oracle VM VirtualBox* con todas las herramientas para construir e interactuar con los procesos a emular.
 - Instalar todas las herramientas requeridas del proyecto RTAI en un equipo de cómputo. Se puede encontrar mayor información al respecto en la documentación del proyecto RTAI en la página www.rtai.org.
 - Disponer de una máquina con ScicosLab/Scicos y RTAI-Lab para construir o modificar los diagramas de bloques y generar los ejecutables para RTAI, y disponer del acceso a un servidor RTAI-RTAIXML (como el proyecto PERI de la Universidad del Valle) para ejecutar y supervisar el modelo emulado.

2. Cargar los módulos de RTAI-Lab (Figura C-1) utilizando el siguiente comando en una consola de LINUX:

sudo /etc/init.d/rtailab start



The screenshot shows a terminal window titled 'realtimevm@otcontrol-...' with a dark background. The terminal output is as follows:

```
realtimevm@otcontrol-virtual: ~  
File Edit View Terminal Help  
realtimevm@otcontrol-virtual:~$ sudo /etc/init.d/rtailab start  
Loading rtai modules and programs.  
realtimevm@otcontrol-virtual:~$ sudo /etc/init.d/rtaixml start  
Starting rtaixml  
=====  
DEBUG MODE ON  
=====  
main(): MasterName: Master  
realtimevm@otcontrol-virtual:~$
```

Figura C-1. Presentación de la carga de los módulos de RTAI-Lab en una consola de LINUX

3. Una vez se dispone del modelo de la columna de destilación binaria (*ColumnaTipoA.cos*) se procede a realizar las modificaciones de los parámetros de la columna: condiciones iniciales, entradas y perturbaciones. Todas las variables y los rangos numéricos permitidos para cada una se pueden observar en las tablas C.1 a C.4. Los datos de la tabla C.1 se encuentran dentro del código fuente del modelo de la columna y las entradas, perturbaciones y condiciones iniciales (tablas C.2 a C.4) se encuentran en el archivo *m_data.sce* que se puede abrir con un editor de texto plano (por ejemplo gedit), el contenido de este archivo se muestra en la Figura C-2.

En este archivo se hacen las modificaciones correspondientes al número de etapas (NT), entradas (LT, VB, D, B), perturbaciones (F, zF, qF) y condiciones iniciales (X_o , M_o).

Variable	Descripción	Valor	Unidad
NT	Número de platos incluidos el rehervidor y el condensador total	Entre 5 y 50	UN
NF	Plato de alimentación	Entre 3 y NT-2	UN
α	Volatilidad relativa	1.5	
$M0_i$	Retención nominal	0.5	Kmol/min
τ_l	Constante de tiempo para la dinámica de flujo de líquido en cada etapa (excepto el rehervidor y el condensador).	0.063	min
λ	Constante por efecto del flujo de vapor sobre el flujo de líquido. Se asume que el flujo de vapor no afecta la retención de líquido (Efecto K2)	0	
L0	Flujo nominal de reflujo (de los datos de estado estable)	2.706	Kmol/min
$L0_b$	Flujo nominal de líquido (debajo del plato de alimentación)	En función de $L0_b$	Kmol/min
V0	Flujo de vapor nominal (sólo se requiere si λ es diferente de cero)	3.20629	Kmol/min
V0t	Flujo de vapor nominal (encima del plato de alimentación)	En función de V0	Kmol/min

Tabla C-1. Variables internas columna de destilación binaria continua.

Variable	Descripción	Valor	Unidad
L_T	Reflujo	2.706	Kmol/min

V_B	Flujo de vapor en el rehervidor	3.206	Kmol/min
D	Flujo de producto del destilado (tope)	0.5	Kmol/min
B	Flujo de producto en el fondo	0.5	Kmol/min

Tabla C-2. Entradas, columna de destilación binaria continua.

Variable	Descripción	Valor	Unidad
F	Flujo de alimentación	> 0 , (1)	Kmol/min
Z_F	Composición de la alimentación	≤ 1 , (0.5)	Fracción molar
q_F	Fracción de líquido en la alimentación	≤ 1 ,	Fracción molar

Tabla C-3. Perturbaciones, columna de destilación binaria continua.

Variable	Descripción	Valor	Unidad
x_i	Composición inicial en cada etapa	Entre 0.5 y 1	Fracción molar
M_i	Retención inicial en cada etapa	Entre 0.5 y 1	Kmol/min

Tabla C-4. Condiciones iniciales, columna de destilación binaria continua.

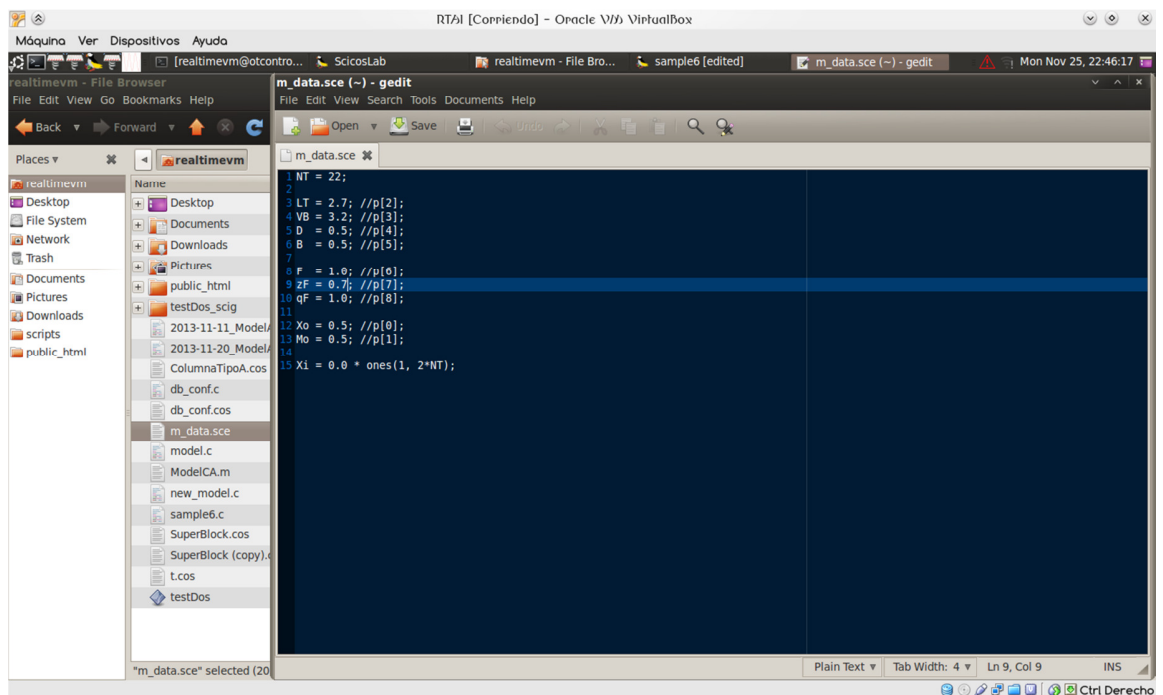


Figura C-2. Contenido del archivo m_data.sce.

4. Abrir ScicosLab y habilitar el toolbox RTAI en la barra de herramientas:

toolboxes → *RTAI*

Debe aparecer el mensaje *Scicos-RTAI Ready* en la ventana de comandos de ScicosLab, como se puede ver en la Figura C-3.

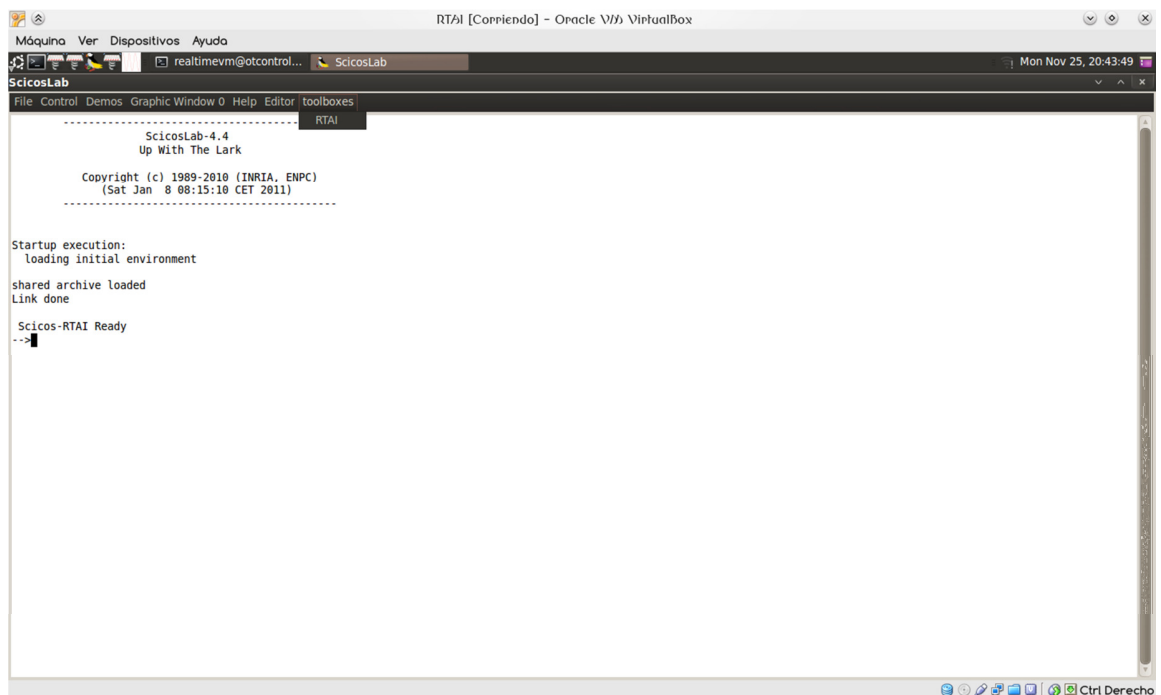


Figura C-3. Habilitación del toolbox RTAI en la barra de herramientas.

5. Si la ruta de trabajo actual de ScicosLab (se puede obtener con el comando `pwd`), difiere de la ruta en la que se encuentra el emulador, debe cambiarse con:

```
cd /ruta/del/emulador/
```

Una vez se tiene la ruta correcta, abrir el emulador ejecutando el siguiente comando en el entorno de ScicosLab:

```
scicos ColumnaTipoA.cos;
```

Este comando abre un cuadro de diálogo con el título “ScicosLab dialog” (Figura C-4) correspondiente al **C RTAI Block** que contiene el código fuente en lenguaje C de la columna de destilación. Las líneas 31 – 42 de este código se reproducen a continuación:

```

NT = 41;
NF = 21;
M0 = 0.5;
alpha = 1.5;
taul = 0.063;
F0 = 1;
qF0 = 1;
L0 = 2.7;
L0b = L0 + qF0 * F0;
lambda = 0;
V0 = 3.2;
V0t = V0 + (1 - qF0) * F0;

```

Estos son los parámetros internos de la columna descritos en la tabla C-1. Todas las variables se pueden modificar, excepto L0b y V0t que dependen de otras variables. Tenga en cuenta que el valor de NT debe ser igual al establecido en el archivo m_data.sce.

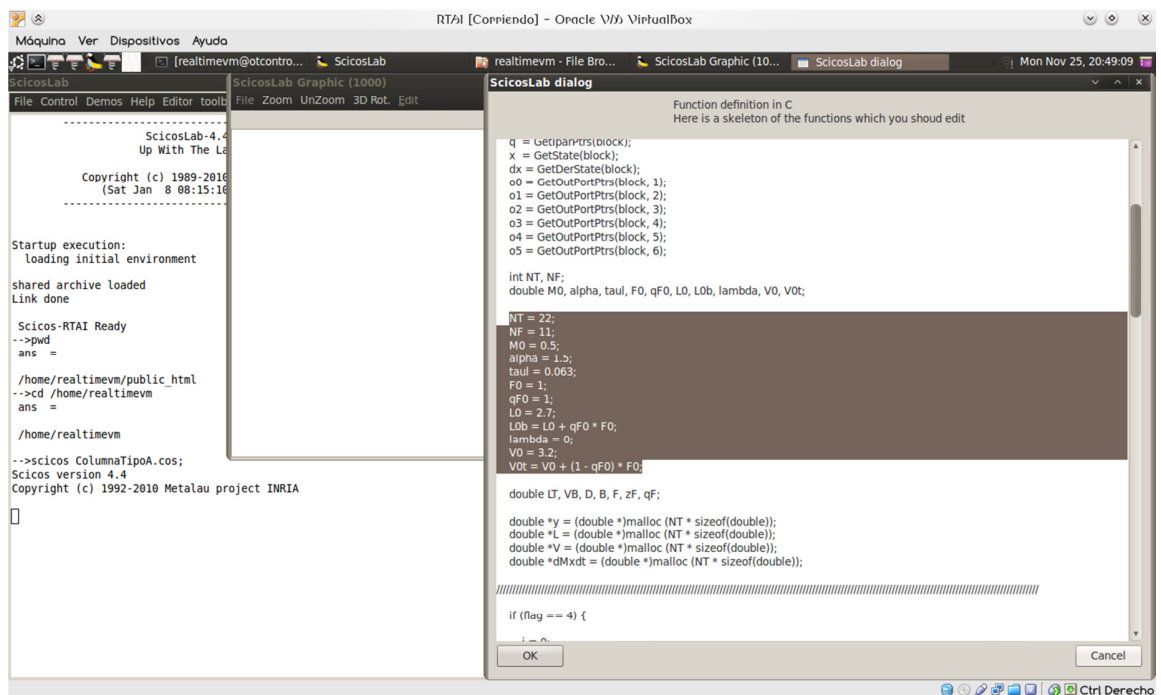


Figura C- 4. Cuadro de diálogo “ScicosLab dialog” correspondiente al C RTAI Block.

Nota: Al cerrar el cuadro de dialogo del “C RTAI Block” aparece el entorno de trabajo de Scicos. El emulador de la columna de destilación a primera vista parece estar compuesto por un reloj y un solo bloque, este bloque es en realidad un súper-bloque que contiene el modelo completo de la columna, debido a que la estructura de los modelos diseñados para ejecutarse en RTAI debe cumplir con esta condición.

- Una vez se han realizado todos los cambios planeados en el modelo, se procede a configurar el target para establecer la rutina de solución y el tiempo de muestreo para el sistema de ecuaciones. Para llevar a cabo este proceso, se selecciona el súper-bloque y la opción “Set Target” del menú “RTAI” en la barra de herramientas de Scicos:

RTAI → *Set target*

Esto despliega un cuadro de dialogo con el título “Set Block Properties” (Figura C-5) y los siguientes parámetros:

- **Target:** Indica para que tipo de sistema se va a generar el ejecutable, en este caso se conserva el valor “rtai”

- **Ode function:** Indica la rutina que usará la máquina para resolver el sistema de ecuaciones diferenciales (solver). Están disponibles las siguientes rutinas:

- ode1 (Euler o Runge-Kutta de primer orden)
- ode2 (Método de Heun, Euler modificado o Runge-Kutta de segundo orden)
- ode4 (Runge-Kutta de cuarto orden)

Aunque todos los *solver* arrojan la misma respuesta, se considera que ode4 tiene mejor rendimiento en el caso del emulador.

- **Step between sampling:** Tiempo de muestreo para resolver el sistema de ecuaciones (cada cuanto tiempo debe resolverse el sistema). Un valor muy bajo puede llevar a una carga de trabajo muy alta en la máquina, mientras que un valor muy alto puede llevar a soluciones erróneas e indeterminaciones en la respuesta. El valor más apropiado en este caso será entre 0.001 y 0.01.

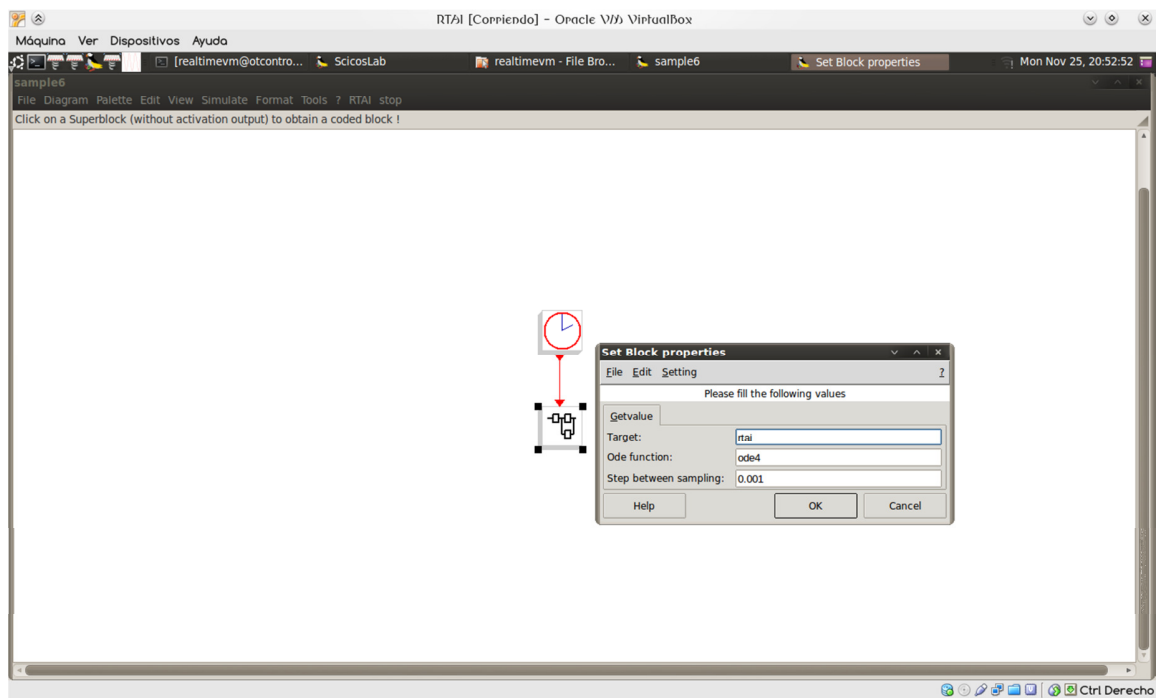


Figura C-5. Superbloque y cuadro de diálogo “Set Block Properties”.

Una vez configurado el target se desplegará nuevamente el diálogo descrito en el punto 5. Al cerrar este diálogo, se despliega una nueva ventana con el contenido del súper-bloque (Figura C-6).

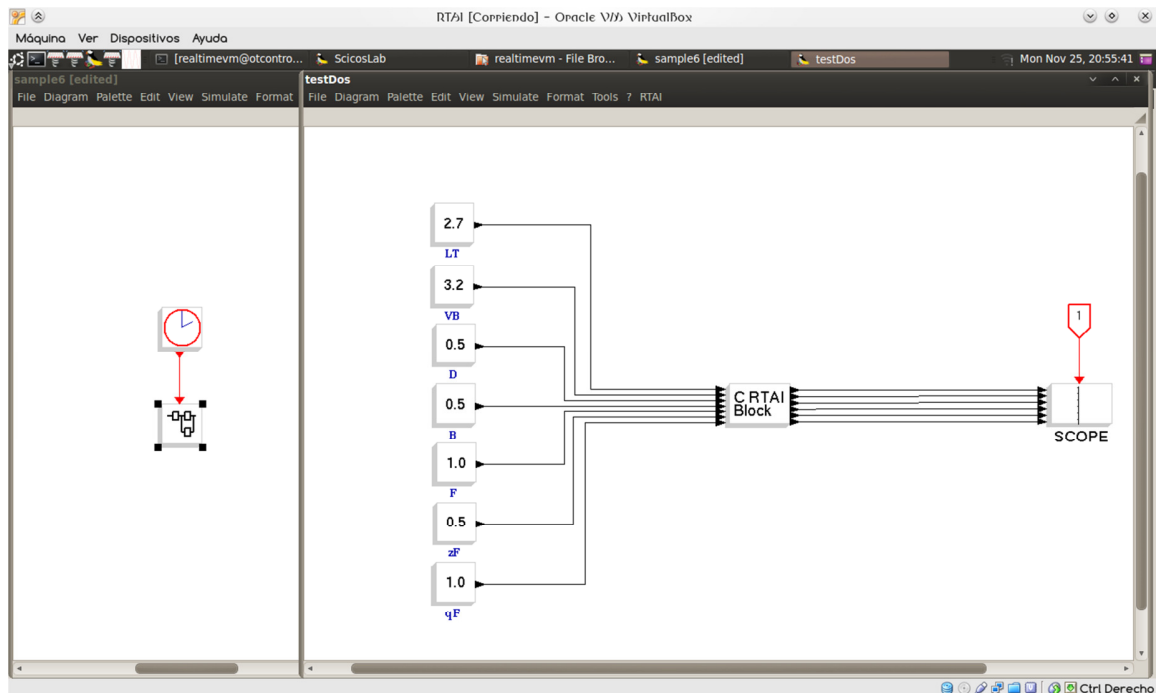


Figura C-6. Contenido del súper-bloque.

7. A continuación se procede a generar y compilar el código compatible con RTAI a partir del diagrama con la herramienta RTAI-CodeGen del menú RTAI. Para llevar a cabo este proceso se selecciona el súper-bloque y se elige la opción “RTAI-CodeGen” del menú “RTAI” en la barra de herramientas de Scicos:

RTAI → *RTAI-CodeGen*

Esto despliega un cuadro de diálogo con el título “Set Block Properties” (Figura C-7) y los siguientes parámetros:

- **New block's name:** indica el nombre que tendrá el ejecutable generado.
- **Created files path:** define la ruta en la que se van a almacenar todos los archivos generados por la herramienta RTAI-CodeGen durante el proceso.

- **Toolchain:** indica para que tipo de sistema se va a generar el ejecutable. Conserva el valor del parámetro **Target**, establecido en el punto 6.
- **Target board:** se mantiene en blanco.
- **ODE solver type:** conserva valor del parámetro **Target**, establecido en el punto 6
- **ODE solver steps betw. samples:** conserva el valor del parámetro **Target**, establecido en el punto 6.

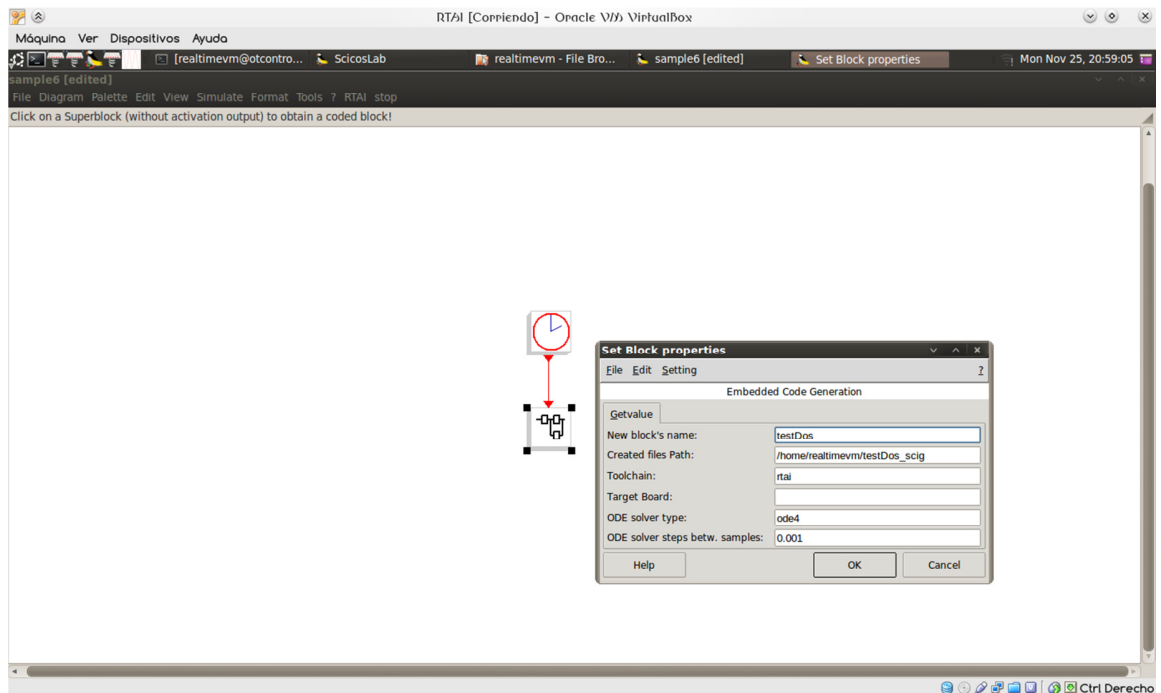


Figura C-7. Cuadro de diálogo “Set Block Properties” después de elegir la opción “RTAI-CodeGen”.

Si el proceso ha finalizado correctamente debe aparecer un mensaje como el siguiente en la ventana de comandos de ScicosLab (figura C-8):

----> *Target generation terminated!*

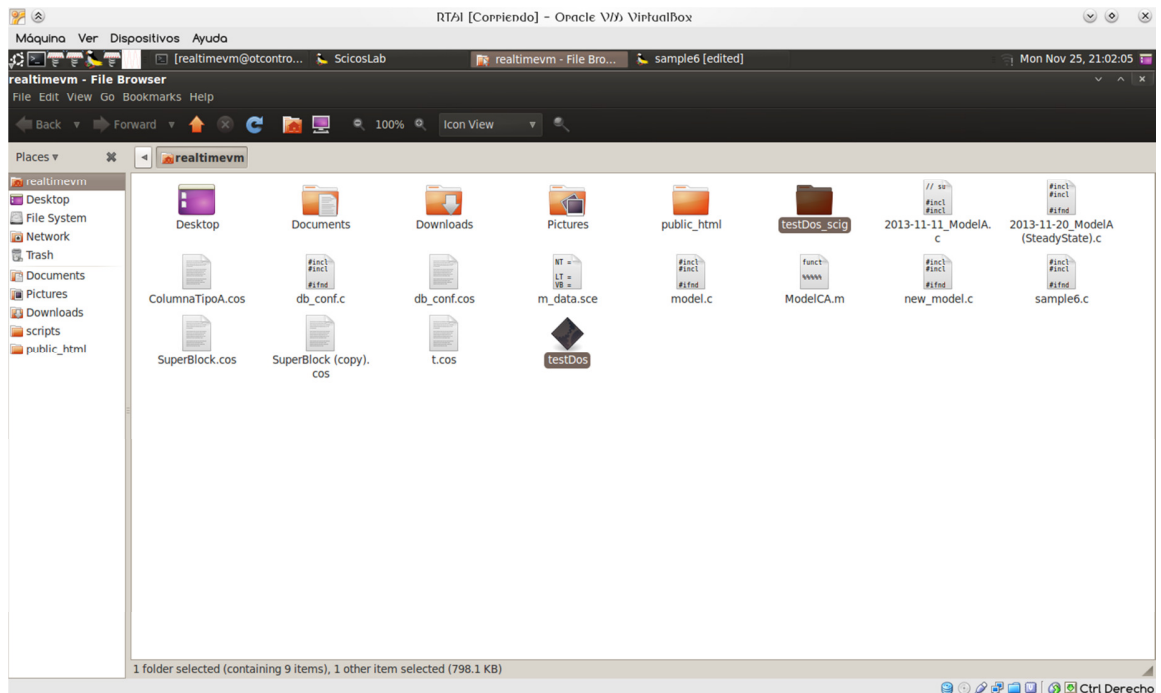


Figura C-9. Ejecutable generado /home/realtimevm.

Si por el contrario se han presentado errores durante el proceso y el ejecutable no se ha generado, los mensajes de error correspondientes aparecen en la ventana de comandos de ScicosLab.

8. Con el ejecutable generado correctamente, el siguiente paso es la ejecución e interacción con el proceso. Para este procedimiento, se hace uso del servidor rtaixml que debe iniciarse con el siguiente comando:

```
sudo /etc/init.d/rtaixml start
```

9. Para iniciar e interactuar con un proceso en la plataforma RTAI a través del servidor rtaixml, debe crearse un script de configuración en la ruta (Figura C-10):

```
/usr/realtime/rtaixml/
```

Las líneas más importantes de este archivo son las siguientes (Figura C-11):

target = testDos # El nombre del archivo ejecutable

dir = /home/realtimevm # La ruta o dirección absoluta en que se encuentra el ejecutable

time = 100 # Tiempo que va a durar la emulación (en segundos)

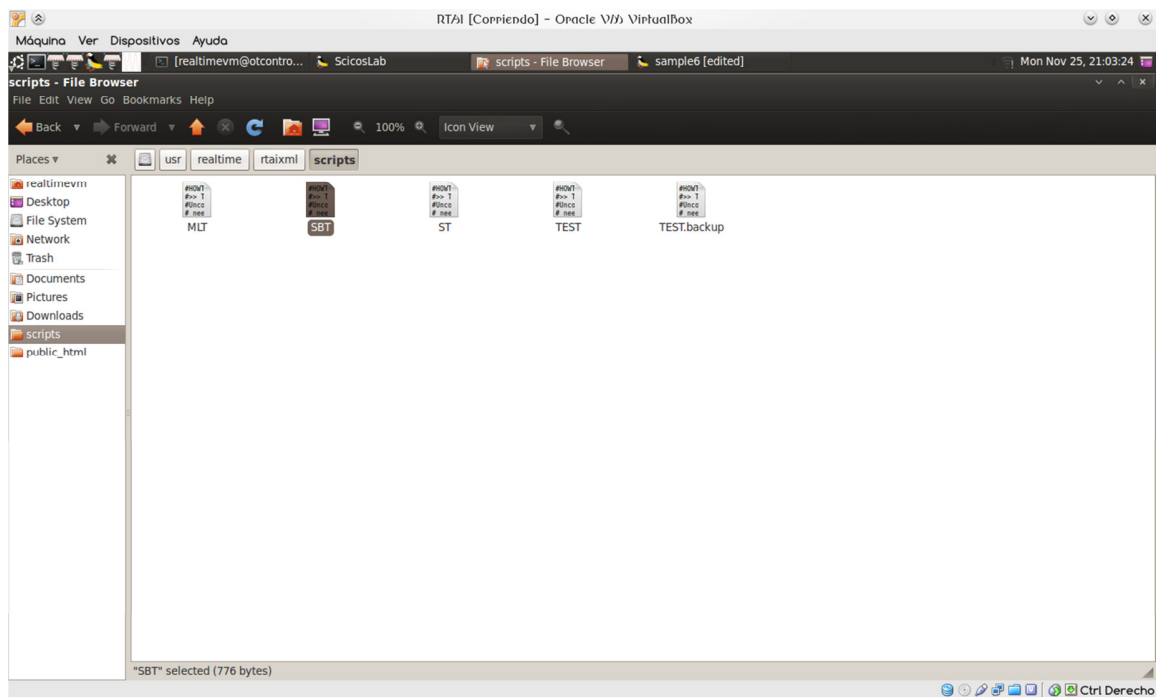
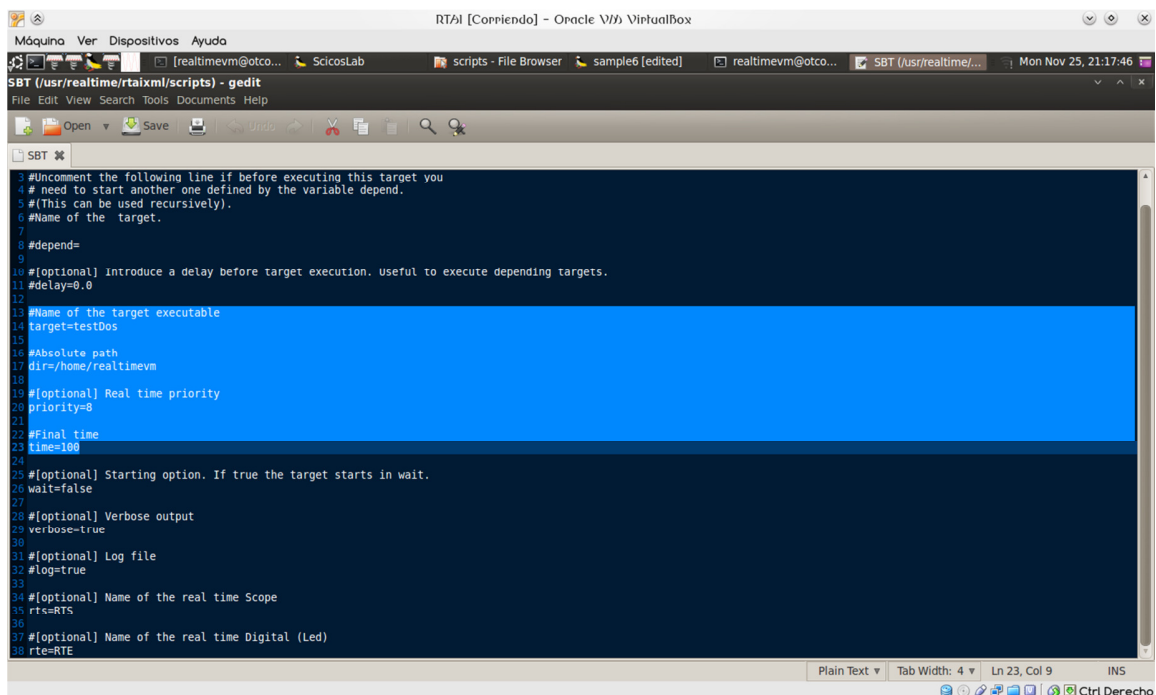


Figura C-10. SBT como script de configuración creado.



```
3 #Uncomment the following line if before executing this target you
4 # need to start another one defined by the variable depend.
5 # (This can be used recursively).
6 #Name of the target.
7
8 #depend=
9
10 #[optional] Introduce a delay before target execution. Useful to execute depending targets.
11 #delay=0.0
12
13 #Name of the target executable
14 target=testDos
15
16 #Absolute path
17 dir=/home/realtimevm
18
19 #[optional] Real time priority
20 priority=8
21
22 #Final time
23 time=100
24
25 #[optional] Starting option. If true the target starts in wait.
26 wait=false
27
28 #[optional] Verbose output
29 verbose=true
30
31 #[optional] Log file
32 log=true
33
34 #[optional] Name of the real time Scope
35 rts=RTS
36
37 #[optional] Name of the real time Digital (Led)
38 rte=RTE
```

Figura C-11. Líneas más importantes del archivo creado.

10. A continuación se abre la aplicación jRTAI-Lab, ésta tiene una barra de herramientas con 4 iconos de los cuales inicialmente solo el primero de izquierda a derecha está habilitado. Este icono es el de conexión al servidor rtai-xml, al pulsarlo, se despliega un cuadro de diálogo con el título “Connect”, y los siguientes parámetros:

- **IP/Hostname:** dirección IP o nombre del host en que se está ejecutando el servidor rtai-xml al que se quiere acceder.
- **Server Port:** puerto a través del cual se establece la conexión con el servidor rtai-xml, por defecto y a menos que se modifique en la configuración del servidor, este será el 29500.
- **Target ID:** corresponde al nombre del script creado en el punto 9.

- **Master ID:** es necesario para interactuar con el servidor con permisos de superusuario. Se puede dejar en blanco o con el valor “Master”.

Finalmente se pulsa el botón “Connect”. Si la conexión se establece correctamente, se habilitarán los otros tres botones de la barra de herramientas. De izquierda a derecha los botones de la barra de herramientas, son los siguientes:

- **Connect/Disconnect:** el botón descrito en el procedimiento anterior, sirve para establecer la conexión con el servidor rtai-xml.

- **Start/Stop:** detiene el proceso que se está ejecutando en la plataforma RTAI.

- **Signals:** gestiona los osciloscopios (SCOPES), permite graficar y guardar los datos generados por el proceso.

- **Parameters:** permite cambiar datos del proceso en ejecución, tales como amplitud y frecuencia de los generadores senoidales, el tiempo y la amplitud de las funciones escalón, los valores pasados en el vector de parámetros reales del “C RTAI Block”, entre otros.

11. Para graficar o guardar los datos generados por el proceso se pulsa el botón “Signals”. Esto desplegará el diálogo titulado Signal Manager, que se puede observar en la esquina superior izquierda de la Figura C-12. Al marcar la opción “Plot” se abre la ventana titulada SCOPE (Figura C-12, lado derecho), donde se muestra la gráfica generada. La opción “Save” guarda en un archivo de texto plano, ubicado en la misma ruta del archivo ejecutable que contiene el proceso emulado, los datos generados por este proceso.

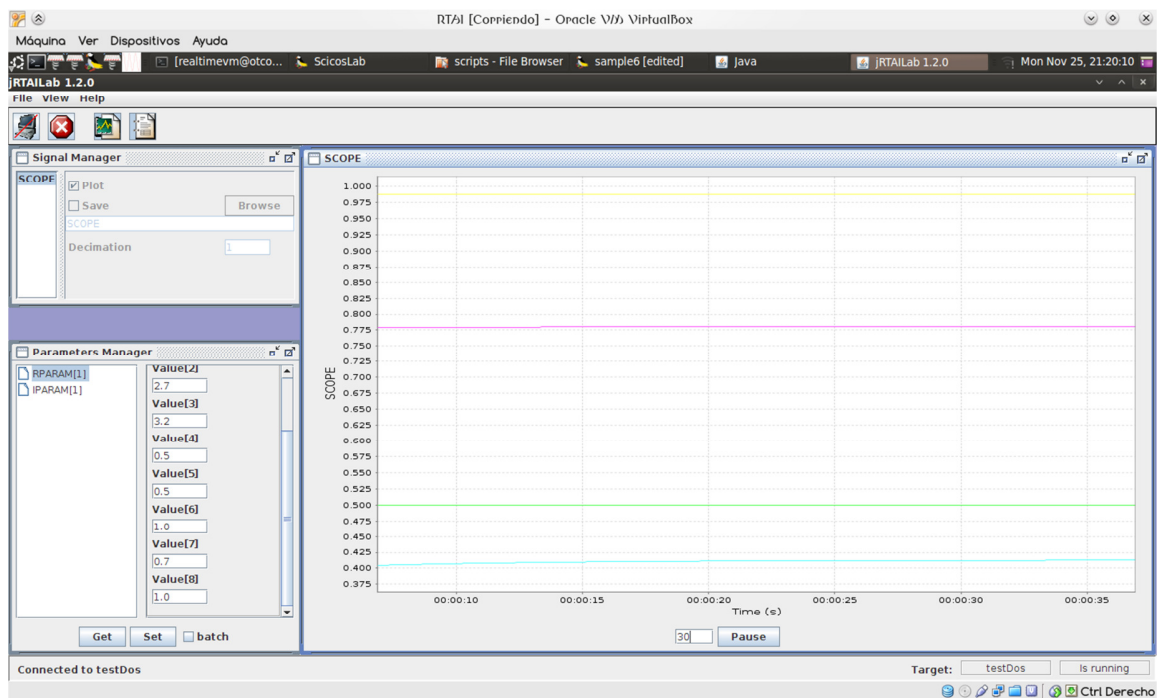


Figura C-12. Respuesta gráfica generada en la emulación.

12. Para cambiar parámetros del emulador, se pulsa el botón “Parameters”, esto despliega el diálogo titulado “Parameters Manager”, que se puede ver en la Figura C-12 en la esquina inferior izquierda. En la Figura C-12, se pueden observar dos nombres en la parte izquierda del diálogo “Parameters Manager”:

- **RPARAM[1]**: vector de parámetros reales del bloque “C RTAI Block”. A través de este vector se están modificando las entradas y perturbaciones del emulador.
- **IPARAM[1]**: vector de parámetros enteros del bloque “C RTAI Block”.

Los parámetros que se pueden cambiar se listan a continuación:

Value[2] = 2.7 # Corresponde al campo LT del archivo m_data.sce
 Value[3] = 3.2 # Corresponde al campo VB del archivo m_data.sce
 Value[4] = 0.5 # Corresponde al campo D del archivo m_data.sce

Value[5] = 0.5 # Corresponde al campo B del archivo m_data.sce
Value[6] = 1.0 # Corresponde al campo F del archivo m_data.sce
Value[7] = 0.7 # Corresponde al campo zF del archivo m_data.sce
Value[8] = 1.0 # Corresponde al campo qF del archivo m_data.sce

En la figura C-13 se puede observar la respuesta del proceso cuando se cambia zF (Value[7]) de 0.7 a 0.3.

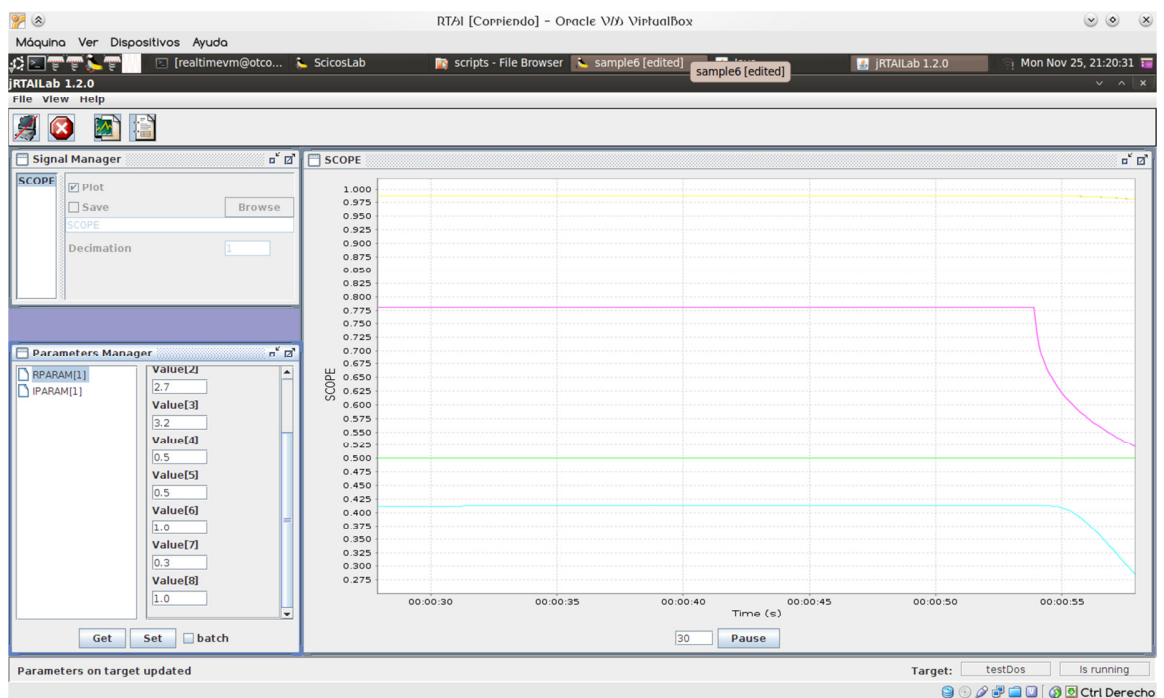


Figura C-13. Respuesta del proceso cuando se cambia zF (Value[7]) de 0.7 a 0.3.