



UNIVERSIDAD DE CARABOBO
ÁREA DE ESTUDIOS DE POSTGRADO
FACULTAD DE INGENIERÍA
DOCTORADO EN INGENIERÍA
ÁREA QUÍMICA



CONTRIBUCIÓN AL DISEÑO INTEGRADO DE COLUMNAS DE DESTILACIÓN BINARIAS

Autor: MSc. Ing. JESÚS ENRIQUE CEBALLOS JIMÉNEZ

Bárbula, Febrero de 2016.

Título: CONTRIBUCIÓN AL DISEÑO INTEGRADO DE COLUMNAS DE
 DESTILACIÓN BINARIAS.

Autor: JESÚS ENRIQUE CEBALLOS JIMÉNEZ

1 era. Edición, 2016.

Derechos Reservados:

© 2014, Jesús Enrique Ceballos. Universidad de Carabobo, Venezuela.

ISBN:

Depósito Legal:

Portada: Jesús Enrique Ceballos Jiménez

Montaje electrónico: Jesús Enrique Ceballos Jiménez

Impreso en Venezuela por: Centro de Reproducción PEQUIVEN Morón

Printed in Venezuela.



UNIVERSIDAD DE CARABOBO
ÁREA DE ESTUDIOS DE POSTGRADO
FACULTAD DE INGENIERÍA
DOCTORADO EN INGENIERÍA
ÁREA QUÍMICA



CONTRIBUCIÓN AL DISEÑO INTEGRADO DE COLUMNAS DE DESTILACIÓN BINARIAS

Tesis Doctoral presentada
ante la Dirección de
Postgrado de la Facultad
de Ingeniería de la
Universidad de Carabobo,
para optar al Grado de
Doctor en Ingeniería.

Autor: MSc. Ing. Jesús Enrique Ceballos Jiménez

Tutor:

Dr. Sergio Pérez

Bárbula, Enero de 2016

**CONTRIBUCIÓN AL DISEÑO INTEGRADO DE COLUMNAS DE
DESTILACIÓN BINARIAS**

DOCTORADO EN INGENIERÍA

Tesis Doctoral

CONTRIBUCIÓN AL DISEÑO INTEGRADO DE COLUMNAS DE DESTILACIÓN BINARIAS

Autor: **MSc. Ing. Jesús Enrique Ceballos Jiménez**

INSTITUCIONES FINANCIERAS:

UNIVERSIDAD DE CARABOBO



La Dirección de Postgrado de la Facultad de Ingeniería de la Universidad de Carabobo aprobó el Proyecto de Tesis Doctoral, que permitió la realización de este trabajo, según oficio S/N del 15-02-08.

CENIPRO, UNEXPO BARQUISIMETO



El Centro de Investigaciones de Procesos (CENIPRO) de la Universidad Nacional Experimental Politécnica de Barquisimeto (UNEXPO), aprobó la pasantía doctoral, la cual permitió cumplir con un requisito del plan doctoral y completar este trabajo de grado, según oficio N° S/N del 26-10-12.

PEQUIVEN



La Vice Presidencia de la Corporación Petroquímica de Venezuela, a través de la Gerencia Gerencia Corporativa de Investigación y Desarrollo Petroquímico, aprobó la subvención de este trabajo, lo cual permitió cumplir con un requisito del plan doctoral y completar este trabajo de grado, según oficio N° S/N del 05-13.

AGRADECIMIENTOS

A Dios todopoderoso por los momentos maravillosos que me ha dado en la vida y por mí existir

A mis padres, por estar siempre ahí en mis momentos más difíciles y por ser tan especiales.

A la Universidad de Carabobo, CENIPRO y todo su personal por el apoyo prestado.

A las profesoras Yris González, Marisol Verruschi y Alexander Peraza, por todo el apoyo prestado ya que sin ellos no hubiese sido posible la culminación de este trabajo.

Al Profesor Pablo por sus consejos y apoyo prestado.

Y muy especialmente al Profesor Sergio Perez, por el apoyo, consejos y dedicación en todas las fases de este trabajo y así llegar a su culminación.

DEDICATORIA

Este trabajo esta dedicado especialmente a cinco personas muy especiales y adoradas en mi vida las cuales son mis pilares, mis guías, orientación, mis amores y que con su dedicación han permitido llegar a puerto seguro y por ello se ha logrado la culminación de este proyecto de vida, ellos son: Mi Esposa Pamela Paredes, mi hijo y tesoro Maximiliano Ceballos, mis grandes padres Ana de Ceballos y Juan de Jesus Ceballos y mi querida, maravillosa hermana Rossana Ceballos.

Que Dios Todopoderoso me los Bendiga y Proteja.

Ceballos Jiménez Jesús Enrique. "Desing of Distillation Columns for Binary Systems, Traditional and Integrated Approach". Tesis Doctoral para optar al Grado de Doctor en Ingeniería Química, Universidad de Carabobo, Área de Estudios de Postgrado de la Facultad de Ingeniería, Carabobo, Venezuela, Enero de 2016.

ABSTRACT

The purpose of this thesis is to contribute methodologically through a systematic study of the application of different optimization techniques to integrated distillation columns in binary systems, to do optimization techniques are applied, the distillation column at steady state was optimized design. It was used; type a search taboo to solve the binary problem. For each choice of the whole search, continuous variables with the deterministic method for process design perturbation is applied to the mole fraction of the power is optimized. It develops and implements the calculation algorithm and the proposed methodology for the design of distillation columns of binary comparing the dynamic simulation system and uncontrolled designed with both approaches, and subjected to a disturbance 10% is validated. There can be seen as dynamic response. It is shown in every case that the integrated design has better performance than the traditional design, your answer is better with a faster time and lower settlement on impulse.

Key words: Integrated Design, Traditional Design and Optimization

Ceballos Jiménez Jesús Enrique. “Contribución al Diseño Integrado de Columnas de Destilación Binarias”. Tesis Doctoral para optar al Grado de Doctor en Ingeniería Química, Universidad de Carabobo, Área de Estudios de Postgrado de la Facultad de Ingeniería, Carabobo, Venezuela, Enero de 2016

RESUMEN

El propósito de esta tesis es contribuir metodológicamente, mediante un estudio sistemático de la aplicación de diferentes técnicas de optimización al diseño integrado de columnas de destilación en sistemas binarios, para ello se aplican técnicas de optimización, se optimizó la columna de destilación en estado estacionario. Se utilizó, una búsqueda tipo tabú para resolver el problema binario. Para cada escogencia de la búsqueda entera, se optimizó las variables continuas con el método determinístico, para el diseño de procesos se aplica la perturbación a la fracción molar de la alimentación. Se desarrolla e implementa el algoritmo de cálculo y se valida la metodología propuesta para el diseño de columnas de destilación binaria comparando la simulación dinámica del sistema con y sin control diseñado con ambos enfoques, y se sometió a una perturbación 10%. Allí se observa como es la respuesta dinámica. Se demuestra en todos los casos que el diseño integrado presenta mejor desempeño que el diseño tradicional, su respuesta es mejor con un tiempo de asentamiento más rápido y un sobre impulso menor.

Palabras claves: Diseño Integrado, Diseño Tradicional, Perturbaciones, Optimización

ÍNDICE GENERAL

	Paguinas
Índice General	I
Índice de Tablas	III
Índice de Figuras	IV
Índice de Gráficos	V
CAPÍTULO I. <i>EL PROBLEMA</i>	
1.1. Planteamiento del problema	30
1.2. Objetivo general	35
1.3. Objetivos específicos	39
1.4. Justificación	43
CAPÍTULO II. <i>MARCO TEÓRICO</i>	
2.1 Diseño de procesos	12
2.1.1 El diseño de procesos enfoque tradicional	13
2.1.2 Diseño integrado	16
2.2 Columnas de destilación binarias	18
2.3 Control de procesos	20
2.3.1 Control de Columnas de destilación binarias	21
2.4 Conceptos Asociados a los modelo matemático de optimización	24
2.4.1 Optimización local	24
2.4.2 Optimización global	24
2.4.3 Calsificación de los métodos de optimización	24
2.4.4 Búsqueda tabú	28
2.5 Problema de diseño integrado utilizando MATLAB R2012a	30
2.6 Búsqueda tabú o tabú search en MATLAB R2012a	32
2.7 La función fmincon	35

CAPÍTULO III. MARCO METODOLÓGICO

3.1 Tipo de investigación	43
3.2 Fase de la investigación	43
3.3 Desarrollo procedimental	44
3.3.1 Definición los aspectos técnicos para el caso a diseñar	43
3.3.1.1 Definición de variables clasificación y las técnicas de de optimización	49
3.3.1.2 Establecer cuales son los objetivos de optimización	51
3.3.2 Modelado matemático	52
3.3.3 Resolución del problema de diseño integrado	58
3.3.3.1 Estructura algorítmica para la obtención de los valores óptimos	58
3.3.3.2 Implementación de la estructura algorítmica	68
3.3.4 Validar los resultados	69

CAPÍTULO IV. DISCUSIÓN DE RESULTADOS

4.1 Metodo del gradiente	73
4.2 Algoritmos genéticos	83
4.3 Diseño integrado vs diseño tradicional.	90
CONCLUSIONES	95
RECOMENDACIONES	98
BIBLIOGRAFÍA	99
APENDICE A. PRODUCTIVIDAD DE LA TESIS DOCTORAL	103
ANEXOS	104

ÍNDICE DE TABLAS

	Página
3.3.1 Parámetros que se asumen constantes en el problema	48
3.3.2 Clasificación de las variables del proceso	49
3.3.3 Variables para la formulación del modelo matemático	50
4.1.1 Resultados optimización de variables de diseño en estado estacionario. Puntos de partida para el problema maestro	75
4.1.2 Reporte de resultados de la programación en MATLAB R2012a para el cálculo de las variables de control en estado estacionario	77
4.1.3 Reporte de resultados del diseño integrado de una columna de destilación binaria. Función “fmincon”	79
4.2.1 Reporte de resultados del diseño integrado de una columna de destilación binaria. algoritmos genéticos	85
4.2.2 Comparación de resultados del diseño integrado de una columna de destilación binaria obtenido mediante métodos de optimización determinístico y estocástico	88
4.3.1 Diseño Integrado vs diseño tradicional para una columna de destilación binaria	90
4.3.2 Resumen cualitativo. Diseño Tradicional vs Diseño Integrado de una columna de Destilación Binaria	93

ÍNDICE DE FIGURAS

	Página
2.1 Metodología del enfoque secuencial	15
2.2 Metodología del enfoque simultáneo	17
2.3 Columna de rectificación binaria	18
2.4 Columna de destilación binaria y sistema de control asociado	23
2.5 Representación de la estructura de una solución	28
3.1 Diagrama del proceso	45
4.6.1.1 Diagrama de bloques de la estructura algorítmica para la resolución del problema de diseño integrado de una columna de destilación binaria	60
4.6.1.2 Algoritmo para resolver el problema MINLP-DAE del diseño Integrado de columnas de destilación binarias. Módulo I: Optimización en estado estacionario	66
4.6.1.3 Algoritmo para resolver el problema MINLP-DAE del diseño integrado de columnas de destilación binarias. Módulo II: Optimización en estado transitorio	67

ÍNDICE DE GRÁFICAS

	Página
4.1.1 Respuesta dinámica del error cuadrático integral (ISE)	76
4.1.2 Relación entre el costo y la controlabilidad del sistema	78
4.1.3 Respuesta dinámica de la composición del destilado. Diseño integrado mediante método del gradiente	81
4.1.4 Respuesta Dinámica de la composición del residuo o fondo. Diseño integrado mediante método del gradient	81
4.2.1 Relación entre el costo y la controlabilidad del sistema	83
4.2.2 Respuesta Dinámica de la composición del destilado. Diseño integrado mediante algoritmos genéticos	86
4.2.3 Respuesta Dinámica de la composición del residuo o fondo. diseño integrado mediante algoritmos genéticos	86
4.3.1 Respuesta dinámica de la composición del destilado. Diseño tradicional	92
4.3.2 Respuesta dinámica de la composición del residuo o fondo. diseño tradicional	92

INTRODUCCIÓN

La integración entre diseño y controlabilidad ha tomado una relevancia en el diseño de procesos reactivos, debido a la necesidad de evaluar las características de operación dinámica de los procesos durante las fases de diseño en cuanto a sus aproximaciones cuantitativas. Cabe mencionar que en las últimas cinco décadas, se ha venido realizando un esfuerzo significativo dirigido a los estudios algorítmicos y teorías de aplicaciones que surgen del problema de diseño de procesos y control. Las primeras ideas de introducir la controlabilidad en el diseño fueron planteadas por Ichikawa (1975, 1976). Los trabajos de Nishida (1974), Nishida e Ichikawa (1975) y Nishida (1976) fueron de los primeros que estudiaron sistemáticamente el problema de incluir consideraciones dinámicas y de control dentro del problema de síntesis de procesos que contienen incertidumbres en los parámetros. Morari y Skogestad (Morari, 1983; Skogestad y Morari, 1987b); Morari y Zafiriou, (1989); Skogestad, (1991); Skogestad y Wolff (1992), han realizado contribuciones significantes en el análisis de la controlabilidad y en el estudio de la capacidad de adaptación dinámica de los sistemas. Ellos han introducido y analizado magnitudes de controlabilidad para la interacción de las variables y el rechazo de perturbaciones. También han investigado las características inherentes a un proceso que limitan su controlabilidad y han propuesto un procedimiento de síntesis de reguladores que consideran la estabilidad robusta en presencia de incertidumbres en los modelos y acciones en las perturbaciones. Sheffield (1992) proporcionó una perspectiva industrial sobre la necesidad de integrar el diseño y el control de sistemas. Tanto Morari (1992) como Perkins (1989), han realizado una recopilación de algunos de los resultados importantes y esfuerzos previos, concernientes a la interacción entre el diseño y el control. Bansal y colaboradores (2000) y Salkizli y colaboradores (2004), han

centrado esfuerzos en el estudio y aplicación del diseño integrado y control de sistemas de destilación, utilizando programación dinámica bajo incertidumbres.

Luyben y Floudas (1994) usan un enfoque multiobjetivo que considera simultáneamente en el diseño tanto los aspectos económicos como los de controlabilidad. En ésta estructura puede observarse la interacción entre varias medidas económicas y de controlabilidad de lazo abierto del proceso. Este enfoque incorpora tanto los aspectos de control como los de diseño dentro del proceso de la estructura de síntesis y es el único enfoque que aborda los temas de síntesis. A través de la aplicación de técnicas multiobjetivo se determina un diseño del proceso que es económicamente óptimo y controlable. Walsh y Perkins (1996) resumen el uso de la optimización como una herramienta para el problema de diseño y control. Ellos hacen notar que los adelantos en hardware computacionales y las herramientas de optimización han hecho posible resolver la complejidad de los problemas que se alcanzan en el diseño y control. Estas afirmaciones se enfocan en el problema de la selección de la estructura de control donde el costo económico de una perturbación se balancea contra el rendimiento del controlador. El problema de Diseño Integrado de procesos del sistema descrito por Schweiger y Floudas (1997) fue resuelto mediante métodos de optimización local. Los algoritmos desarrollados y la simulación implementada arrojan resultados que ponen de manifiesto las ventajas del enfoque simultáneo de diseño sobre el enfoque secuencial. Surge así el enfoque sobre el cual se basa el presente trabajo, el Diseño Integrado, denominado así por Gutiérrez y Prada en el año 2003.

Se establece el enfoque en la tarea de diseño de procesos químicos de dos formas: el enfoque tradicional (enfoque secuencial) el cual ha sido llevada a cabo tradicionalmente de manera manual e intuitiva, una vez que se proponen diferentes esquemas de diagramas de flujo, los ingenieros se centran en elegir el esquema de la planta económicamente óptimo desde el punto de vista de ingeniería civil, pero sin tener en cuenta la secuencia e interconexión óptima entre las unidades que lo componen, y sin considerar las características de controlabilidad en lazo abierto o

en lazo cerrado que tendría la planta una vez diseñada. Esta estrategia implica evolucionar hacia la solución a través de sucesivos refinamientos de una solución inicial aproximada. Se propone una estrategia de diseño desarrollando en principio soluciones muy simples para luego agregar capas sucesivas de detalles.

Y el diseño integrado (enfoque simultáneo), es “ la técnica de diseño de procesos mediante la cual se obtienen los parámetros físicos de la planta que minimicen el coste de operación y construcción, al mismo tiempo que cumplen con las características de controlabilidad que se imponen al proceso". La interacción entre el diseño de procesos y el control de procesos surge porque el diseño de procesos químicos determina de un modo inherente su controlabilidad, que cualitativamente significa lo bien que un proceso es capaz de rechazar perturbaciones, cuán severamente interactúan las múltiples variables y cuán fácilmente el sistema se mueve de un punto de operación a otro.

Por esta razón, se presenta este trabajo que consiste en realizar un estudio sistemático de la aplicación de diferentes técnicas de optimización, clásicas y heurísticas. Así como la implementación de un algoritmo de resolución matemática para obtener el máximo rendimiento de los métodos de optimización tanto locales como globales. De esta manera se realiza un aporte metodológico al problema de diseño integrado en columnas de destilación binarias

Las diferentes estrategias metodológicas presentadas servirán para seleccionar la vía más útil para diseñar de forma integrada una columna de destilación binaria, de acuerdo a las necesidades particulares y serán la base para futuras investigaciones en otros procesos y/o unidades de procesos.

CAPÍTULO I

EL PROBLEMA

1.1 PLANTEAMIENTO DEL PROBLEMA

El proceso de separación de los componentes de una corriente proveniente de una operación unitaria específica, representa un paso sumamente relevante al momento de obtener compuestos de una pureza deseada.

Tal es el caso de la destilación, ampliamente empleada y difundida dentro de la industria de los procesos químicos. Dado su carácter, generalmente imprescindible, son significativamente útiles todos los esfuerzos tendientes a implementar mejoras al proceso de destilación.

Se ha observado que comúnmente la manera tradicional de diseñar los procesos químicos es realizar el diseño de la planta y el diseño del sistema de control de manera secuencial. Es decir, se determina la estructura del diagrama de flujo, las condiciones de operación en estado estacionario que se requieren para alcanzar los objetivos de producción y el cálculo de los parámetros físicos de la planta, colocando en primer lugar y algunas veces como único criterio de diseño aquel que sea más económico tanto en gastos de mantenimiento como de construcción de las mismas.

Hasta este punto puede considerarse que el enfoque para diseñar procesos de separación es completamente razonable, pero cuando se toma en consideración que se le ha brindado una atención menor a la controlabilidad dinámica del proceso, es cuando se empieza a cuestionar la efectividad del método de diseño. Y esto no meramente por una observación intuitiva o en aras de buscar defectos en el enfoque aplicado, sino más bien por la misma aplicación del método científico a los procesos industriales ya existentes y en operación, en los cuales se reportan problemas causados por la ausencia de controlabilidad.

Si este enfoque genera problemas de operación, es entonces una prioridad explorar nuevas técnicas de optimización que conlleven a la eficiencia de los procesos y muy específicamente al proceso de destilación, que es la operación unitaria abordada en el presente estudio. La problemática es mucho más preocupante de lo que se puede apreciar hasta este nivel, y más aún, tan evidente que amerita el estudio exhaustivo de la misma.

A lo largo del desarrollo del trabajo el lector podrá asegurarse con bases científicas e históricas el gran dilema que se ha venido arrastrando hasta el presente en cuanto al diseño de procesos por parte de la Ingeniería Química y su desvinculación con la Ingeniería en Control de Procesos. La elevada capacidad de cómputo de los microprocesadores ya no permite colocar más excusas para integrar ambas disciplinas. Solamente permanece una justificación obsoleta para aplicar enfoques secuenciales de diseño de procesos.

En el desarrollo de la presente investigación se exploran dos métodos diferentes para efectuar el diseño integrado de procesos de una columna de destilación binaria, en el cual se aplica de manera simultánea el diseño del proceso y el diseño del control del proceso. Los resultados podrán contribuir a soportar, explorar, ponderar y decidir acerca de la mejor manera de abordar el diseño de una columna de destilación binaria.

Académicamente se aporta una contribución metodológica al diseño integrado de columnas de destilación binarias mediante un estudio sistemático de la aplicación de diferentes técnicas de optimización, con el objetivo de presentar las ventajas y limitaciones de los diferentes métodos, así como diferentes esquemas e implementaciones de un algoritmo de resolución matemática, para en último término, obtener el máximo rendimiento de los métodos de optimización tanto locales como globales al problema específico.

Para el desarrollo de la tesis doctoral se establece el problema como uno Multiobjetivo del tipo Programación No Lineal Entera Mixta con ecuaciones Algebraico Diferenciales (MINLP/DAE). Se implementa el algoritmo de solución matemática en una herramienta computacional accesible, la cual incluye resolvedores de ecuaciones algebraico diferenciales (DAE) y de problemas de optimización no lineal (MINLP).

Permitiendo tener las variables de diseño y los parámetros de control que resuelve los problemas de optimización de enteros mixto antes planteado. Que optimicen tanto los criterios económicos como la controlabilidad del proceso y a la vez garantice una operación factible

Con este trabajo se pretende resolver dado el sistema particular, el diseño integrado de una columna de destilación binaria mediante un estudio sistemático de la aplicación de diferentes técnicas de optimización, clásicas y heurísticas.

1.2 OBJETIVO GENERAL

Estudiar el diseño integrado columnas de destilación en sistemas binarios mediante la aplicación de diferentes técnicas de optimización.

1.3 OBJETIVOS ESPECÍFICOS

- Analizar los aspectos técnicos necesarios para describir el proceso o caso específico a diseñar.

- Diseñar un algoritmo para la solución del problema de diseño integrado de columnas de destilación en sistemas binarios y selección de la ley de control.
- Implementar el algoritmo en una herramienta computacional adecuada.
- Validar los resultados encontrados comparando la respuesta dinámica del sistema obtenida mediante diseño integral y diseño tradicional.

1.4 JUSTIFICACIÓN

La complejidad de los procesos se ha incrementado debido a las exigencias del mercado, por la necesidad de mejorar la calidad de los productos, regulaciones ambientales y el aumento de la productividad, avances tecnológicos y la competitividad.

Debido a esto surge el control automático con el objeto de mantener en determinado valor de operación las variables del proceso tales como: composiciones, temperaturas, presiones, diámetro de las columnas de destilación, entre otros.

Todo eso ha llevado no sólo a diseñar procesos de manera óptima sino también a mantener operando el proceso de manera estable a pesar de las perturbaciones, por lo que el control de procesos se hace cada vez más indispensable.

Ahora surge la necesidad de integrar ambas disciplinas. Mantener aislado o diseñar de forma separadas el proceso y su arquitectura de control, es algo que ha traído preocupaciones debido a que esto a la larga no refleja el impacto del sistema de control en las operaciones económicas, así como tampoco sobre la relación entre

el diseño y la controlabilidad, lo que niega el hecho de que la controlabilidad dinámica del proceso es una característica inherente a su diseño.

Surge así el enfoque sobre el cual se basa el presente trabajo, el diseño integrado, denominado así por Gutiérrez y Prada en el año 2003. El diseño integrado es la técnica de diseño de procesos mediante la cual se obtienen los parámetros físicos de la planta que minimicen el coste de operación y construcción, al mismo tiempo que cumplen con las características de controlabilidad que se imponen al proceso, obteniendo de esta manera diseños de planta flexibles capaces de mantener sus condiciones de controlabilidad frente a futuras perturbaciones que se puedan presentar.

Todo estudio relativo a desarrollar ampliamente el diseño integrado de procesos constituye un avance más hacia la eficiencia en la producción. Ensayar distintos métodos de resolución del problema hasta obtener el que más se adapte a un proceso específico contribuirá a la extrapolación del empleo de estas técnicas de diseño a nivel industrial y de esta manera poder salvar todas las dificultades generadas por el diseño secuencial del proceso y de los sistemas de control.

Es pues así que surge la importancia de la presente investigación, que se justifica primeramente desde el punto de vista académico, al generar los antecedentes y la base científica de conocimiento que permitan dar un respaldo a la implementación del diseño integrado de procesos y por supuesto a nivel industrial, ya que con este tipo de investigaciones se estudian posibles vías para garantizar que una vez diseñado el proceso se generen en la práctica procesos de producción más eficientes. La promoción y masificación de estos trabajos permitirá ensayar todas las posibles técnicas de optimización para resolver el problema mencionado.

Por lo expuesto, en el presente trabajo se propone diseñar el prototipo de un equipo específico de una columna de destilación binaria para validar los datos obtenidos a través de simulación utilizando programación dinámica bajo una perturbación tipo escalón.

Con esta investigación se pretende realizar un aporte metodológico al diseño integrado de columnas de destilación binarias.

CAPÍTULO II

MARCO TEÓRICO

Según Balestrini (2004), el marco teórico es: "...el resultado de la selección de aquellos aspectos más relacionados del cuerpo teórico epistemológico que se asumen, referidos al tema específico elegido para su estudio. De allí pues, que su racionalidad, estructura lógica y consistencia interna, va a permitir el análisis de los hechos conocidos, así, orientar la búsqueda de otros datos relevantes. En consecuencia, cualquiera que sea el punto de partida para la delimitación y el tratamiento del problema se requiere de la definición conceptual y la ubicación del contexto teórico que orienta el sentido de la investigación."

2.1 DISEÑO DE PROCESOS

Scenna (1999) denomina procesos "a la unidad o sistema estructural de transformación por medio del cual los materiales que ingresan se transforman en los productos deseados." El cual está compuesto por módulos (equipos u operaciones unitarias), encargados de realizar tareas específicas (separación, calentamiento, reacción química). Los equipos están conectados entre sí por medio de las corrientes que los vinculan, con el objeto de satisfacer la(s) tarea(s) especificada(s) de la mejor manera posible.

Por tal razón el diseño lleva implícita la necesidad de seleccionar dentro de un espacio de posibilidades, aquellas alternativas que en base a un criterio predeterminado cumplan con los objetivos deseados. En la búsqueda de lograr un planteamiento formal del problema de diseño, conviene establecer una función objetivo adecuadamente a ser optimizada. En este punto, conviene explorar los objetivos que se esperan de un proceso químico en general.

El marco económico constituye la fuente cuyo veredicto resulta inevitable de contemplar en toda actividad de ingeniería. Por lo tanto, costos de operación,

inversión y tasa de retorno, son criterios que generalmente se tienen en cuenta en la función objetivo a optimizar.

En efecto, existen una serie de aspectos que necesariamente deben ser contemplados, como por ejemplo, elasticidad que se refiere a la habilidad del proceso para tolerar condiciones adversas tales como perturbaciones o variación en los parámetros, o flexibilidad que se refiere a la capacidad estructural y operativa del proceso para mantenerse funcionando con el mejor desempeño cuando las condiciones operativas corresponden a un rango de condiciones de diseño, o controlabilidad, confiabilidad e impacto ambiental el cual es de suma importancia a la hora de la selección de tecnologías y de diseño de equipos.

Takama y Umeda (1980), el problema a resolver es multiobjetivo, dichos problemas se caracterizan por no poseer una única solución. Por lo tanto, decidir entre varias soluciones, lleva implícito subjetivamente la preferencia del diseño respecto de los objetivos.

No es posible resolver, modelar un problema complejo, basándose en una herramienta o algoritmo matemático que involucre todos los aspectos a considerar en forma simultánea, se plantean actualmente dos enfoques distintos: el enfoque tradicional de diseño de procesos o secuencial y el enfoque de diseño integrado.

2.1.1 El diseño de procesos enfoque tradicional

El modo tradicional de abordar el diseño de procesos químicos, ha sido realizar el diseño de la planta y el diseño y análisis del sistema de control de manera secuencial. Se determina la estructura del diagrama de flujo, las condiciones de operación en estado estacionario que se requieren para alcanzar los objetivos de

producción y el cálculo de los parámetros físicos de la planta (tamaño de las unidades del proceso).

El objetivo principal en la realización de esta tarea ha sido el de optimizar un índice económico (minimizar el costo anual de mantenimiento, maximizar los beneficios o ganancias o minimizar el costo de construcción), considerando únicamente la operación de la planta en estado estacionario, evaluando las alternativas de diseños y parámetros que satisfagan los requerimientos de operación establecidos, seleccionando finalmente la más económica tanto en gastos de costos de mantenimiento como de construcción de las mismas. Diseñada la planta, se establece una estrategia de control que asegure un comportamiento dinámico estable y que satisfaga los requerimientos de calidad del producto.

La controlabilidad debe alcanzarse en la presencia de grandes perturbaciones de fallos de los equipos, variaciones de carga, entre otros. Es una tarea difícil ya que la controlabilidad no forma parte integral en las etapas de diseño, es decir no se tiene en cuenta previamente.

En efecto, cuando se enfrenta un problema complejo, la experiencia y los postulados de la teoría general de sistemas (Umeda 1982) aconsejan una división (partición) del mismo en subproblemas más sencillos, con una determinada estrategia, para luego componer la solución a partir de las soluciones parciales.

Existen muchas alternativas de descomposición para elegir y lograr una solución adecuada para un problema dado, pero se requiere encontrar la mejor para cada diseño específico. El enfoque jerárquico propuesto por Douglas (1988) que descompone al problema completo en una serie de subproblemas usando una jerarquía de decisiones, fijando una serie de prioridades relativas al fijar la secuencia de subproblemas a resolverse.

En la figura 2.1, se muestran las secuencias de diseño aconsejadas por Quintana (2003), para realizar la tarea de diseño, comprende:

- Definición del problema.
- Establecimiento de la función objetivo, es decir, la determinación de los criterios en función de los cuales deben seleccionarse alternativas.
- El dimensionamiento, evaluación de los costos por ingeniería civil, controlabilidad, implementación de las técnicas y control automático hasta la evaluación del costo de explotación de la planta.

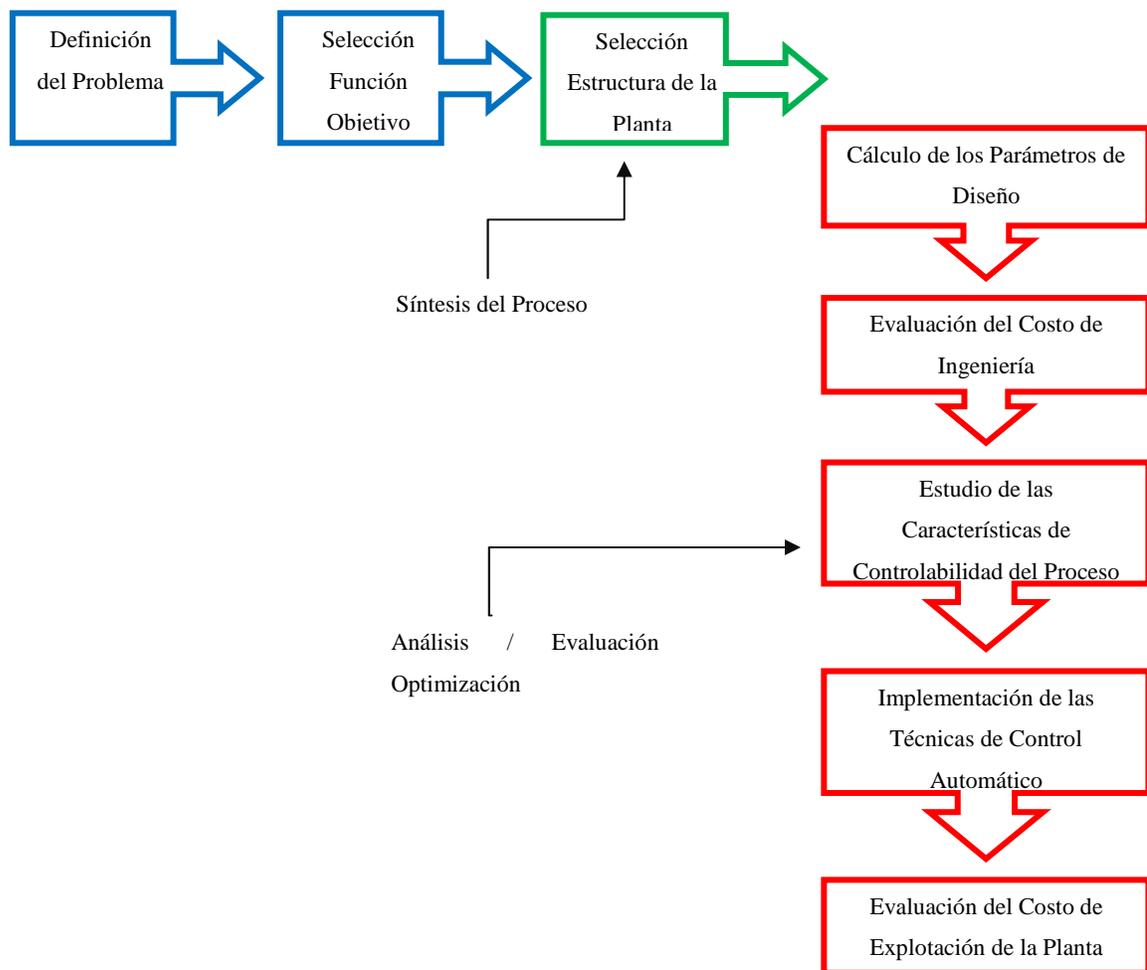


Figura 2.1. Metodología del enfoque secuencial. Fuente: Gutiérrez de Prada (2003) y Quintana (2003)

Debido a la elevada dimensión de los problemas asociados a la tarea de diseño, su elevado índice combinatorio y la gran cantidad de información que debe manejarse, la única herramienta viable para tal fin resultan ser los métodos computacionales. Además de no existir una herramienta computacional (programa o algoritmo) capaz de resolver el problema del ciclo completo de actividades en la tarea de diseño, tampoco existe algoritmo efectivo para la síntesis de procesos genéticos complejos. A medida que se simplifica (particiona) el problema, es decir, se adopta subsistemas del complejo a diseñar, los algoritmos propuestos se caracterizan por su mayor contenido formal y su robustez.

2.1.2 Diseño integrado

Es la posibilidad de diseñar procesos con criterios de optimización, incorporando condiciones que garanticen un buen desempeño dinámico del sistema en todas las primeras etapas del diseño, obteniéndose los parámetros de la planta como los del controlador en forma simultánea. La interacción entre el diseño de procesos y el control de procesos surge porque el diseño de procesos químicos determina de un modo inherente su controlabilidad, que cualitativamente significa lo bien que un proceso es capaz de rechazar perturbaciones, cuán severamente interactúan las múltiples variables y cuán fácilmente el sistema se mueve de un punto de operación a otro.

Es por ello que Gutiérrez y de Prada (2003) lo definen como una técnica o enfoque de diseño de plantas mediante el cual las características de controlabilidad son consideradas previamente en las etapas de diseño, lo que permite fijar especificaciones dinámicas del sistema que facilitará la operación y flexibilidad de las mismas que pueden ser gobernadas por un sistema de control automático.

La tarea de diseño de procesos está enfocada a la determinación de las condiciones de operación de cada una de las unidades del proceso y los dimensionamientos mientras que el control de procesos se centra en establecer las pautas de operación que aseguren un comportamiento estable del proceso satisfaciendo sus requerimientos en condiciones desfavorables. Los avances tecnológicos en hardware y software permiten realizar el diseño simultáneo de procesos y sistemas de control teniendo en cuenta restricciones de operación, restricciones de tipo físico y que consideren la controlabilidad del conjunto.

La Figura 2.2 se representa en forma esquemática el diseño integrado o enfoque simultáneo, el cual representa la forma metodológica o herramienta única que incorpora y crea metodologías eficientes y métodos de diseño automatizados o asistidos por computadora.

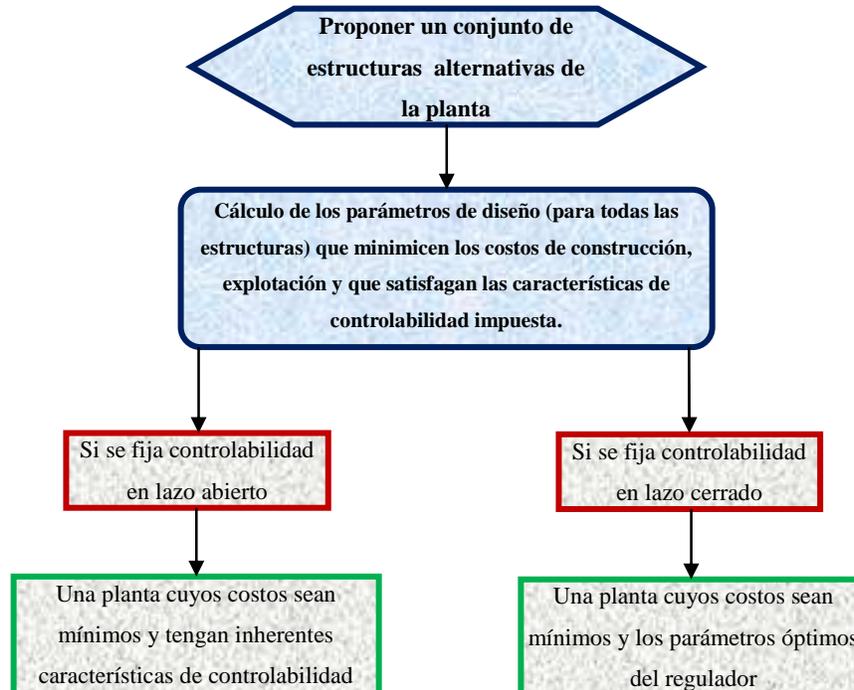


Figura 2.2. Metodología del enfoque simultáneo. Fuente: Gutiérrez y de Prada (2003)

2.2 COLUMNAS DE DESTILACIÓN BINARIAS

En la Figura 2.3 se muestra un esquema de una columna de rectificación. La columna consta de NS sectores, y dentro de cada uno de ellos los pisos se enumeran de arriba hacia abajo.

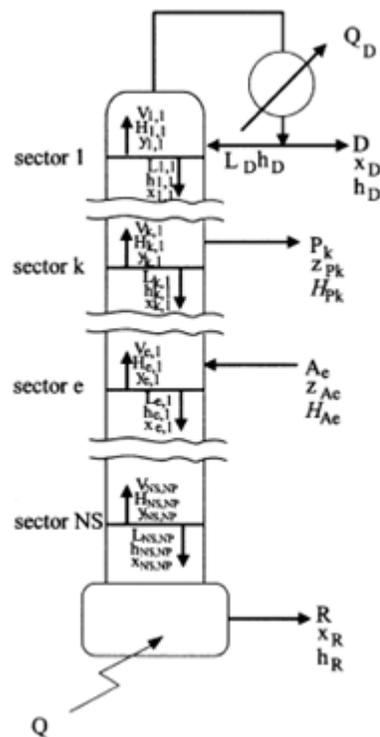


Figura 2.3 Columna de rectificación binaria. Fuente: Gomis, M. (1998).

Dónde:

R: caudal molar o másico del residuo.

D: caudal molar o másico del destilado

A_k : Caudal molar o másico de la alimentación introducido por encima del sector $k+1$.

P_k : Caudal molar o másico del producto por encima del sector $k+1$.

L_D : Caudal de reflujo.

y_{kj} : Composición del vapor que abandona el piso j del sector k (fracción molar o másica).

x_{kj} : Composición del líquido que abandona el piso j del sector k (fracción molar o másica).

z_{Ak} : Composición de la alimentación (vapor, líquido o mezcla de ambos).

V_{kj} : Caudal del vapor que abandona el piso j del sector k (kmol/s o kg/s).

L_{kj} : Caudal del líquido que abandona el piso j del sector k (kmol/s o kg/s).

H_{kj} : Caudal del vapor V_{kj} (kJ/kg o kJ/kmol).

h_{kj} : Caudal del líquido L_{kj} (kJ/kg o kJ/kmol).

H_{Ak} : Entalpía de la alimentación (vapor, líquido o mezcla de ambos).

NP_k : número de pisos del sector k .

NS : número de sectores de la columna.

h_D : Entalpía del destilado (líquido).

H_D : Entalpía del destilado (vapor).

h_R : Entalpía del residuo.

Q : Energía suministrada a la caldera por unidad de tiempo (kJ/s).

Q_D : Energía eliminada por el condensador por unidad de tiempo (kJ/s).

Cada una de las corrientes que circulan por el interior de la columna se identifica mediante dos subíndices: el primero hace referencia al sector en que se encuentra, y el segundo al piso del que procede.

Así, el primer subíndice k variará entre 1 y NS , y el segundo j entre 1 y NP_k , siendo NP_k el número de pisos del sector k . Puesto que se trata de mezclas binarias, se omite el subíndice correspondiente al componente, y se entiende que las fracciones molares o másicas son las correspondientes al componente más volátil.

El subíndice $k, 0$ hace referencia a una corriente líquida que entra en el primer piso del sector k , o a un vapor que llega al último piso (NP_k) de dicho sector. Con esta notación, el reflujo L_0 se representa por $L_{1,0}$.

Para Gomis, M. (1998), el diseño riguroso de una columna de rectificación supone el uso alternativo de las relaciones de equilibrio entre fases y de los balances de materia y de energía. En el caso de mezclas binarias y ternarias el problema se puede resolver mediante métodos gráficos. Mientras que para mezclas multicomponentes es necesario la utilización de métodos analíticos.

En el Capítulo IV se define el sistema objeto de estudio y se obtienen las ecuaciones de balance de materia y energía para efectuar el diseño de la columna.

2.3 CONTROL DE PROCESOS

El control de procesos es definido por Smith y Corripio (1991) como una manera de mantener la variable controlada en el punto de control a pesar de las

perturbaciones. El objetivo fundamental del control de procesos es mantener en determinado valor de operación las variables del proceso tales como: temperaturas, presiones, flujos y compuestos. Debido a que los procesos son de naturaleza dinámica, en ellos siempre ocurrirán cambios y si no se emprenden las acciones pertinentes, las variables importantes del proceso, es decir, aquellas que se relacionan con la seguridad, la calidad del producto y los índices de producción, no cumplirán con las condiciones de diseño.

2.3.1 Control de columnas de destilación binarias

La destilación es el método más común en la industria química. El control del proceso de destilación afecta la calidad de los productos, la producción y los consumos. Por lo tanto es muy importante disponer de técnicas prácticas para el desarrollo de sistemas de control que permitan la operación eficiente y segura de las columnas de destilación.

El control de las columnas de destilación constituye todavía un reto porque estos procesos suelen ser marcadamente multivariantes e inherentemente no lineales. Tienen muchas restricciones y están sujetos a grandes y frecuentes perturbaciones.

La mejora en el control de columnas de destilación puede tener un fuerte impacto económico en las unidades de proceso. Los potenciales beneficios se derivan del aumento de la capacidad de producción, del aumento de la recuperación de productos más valiosos y/o de la reducción de consumos energéticos.

Hergueta (2008) define las columnas de destilación desde el punto de vista de control, como sistemas 5x5, esto es, sistemas en los que se debe controlar cinco (5) variables y para ello se pueden manipular otras tantas.

Las variables que normalmente se pueden controlar se pueden observar en la Figura 2.4 para la columna de destilación binaria. Ellas son:

- La composición del producto de cabeza o destilado (y)
- Composición del producto de fondo (x)
- La presión de trabajo de la columna (P)
- El nivel del acumulador de cabeza (LC_c)
- El nivel del acumulador de fondo (LC_b)

Las dos primeras variables aseguran la calidad de los productos, mientras que las otras tres variables están relacionadas con la estabilidad de la operación, asegurando que se cumplen los balances de materia y energía.

Las variables que se pueden manipular en estas columnas son:

- El caudal de destilado (D).
- El caudal de reflujo (L).
- El caudal de fondo (B).
- El vaporizado (V) (normalmente lo que se manipula es el aporte de calor al rehervidor).
- La capacidad de condensación en el condensador (Q_c).

Las perturbaciones más habituales en este tipo de columnas son.

- El caudal de carga (F).
- Composición de la carga (z_F).

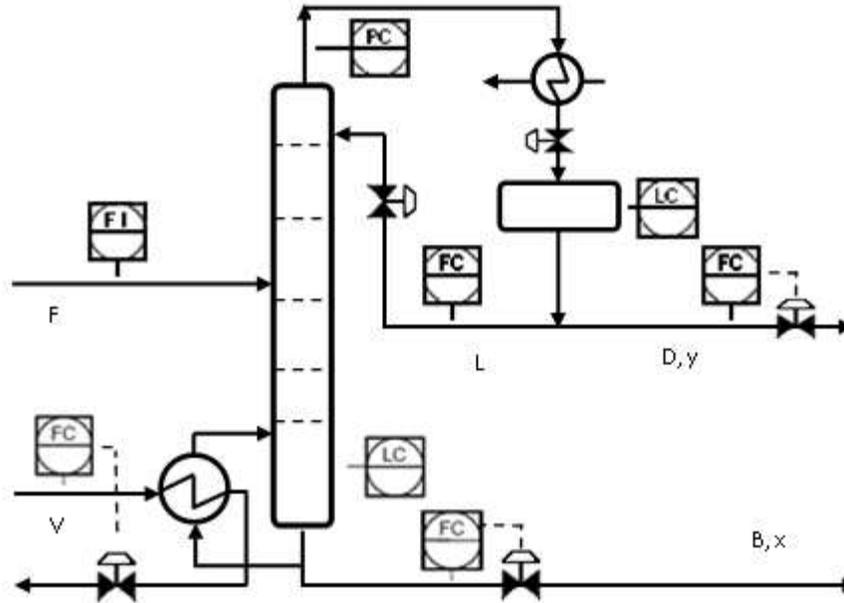


Figura 2.4 Columna de destilación binaria y sistema de control asociado.

Fuente: Hergueta, D. (2008).

La configuración del sistema de control básico de una columna se define de acuerdo con Hergueta (2008), como el emparejamiento de cada una de las variables controladas con una variable manipulada. El problema de la selección de la mejor configuración es un proceso complejo, puesto que existen $5!=120$ configuraciones (emparejamientos distintos) posibles. Sin embargo en la práctica, las configuraciones que deben evaluarse se reducen bastante. En casi todos los casos, lo primero que debe hacerse es estabilizar la columna cerrando los lazos de control de presión y de nivel, puesto que es imprescindible que estas variables estén bien reguladas para poder controlar la calidad de los productos.

2.4 CONCEPTOS ASOCIADOS A LOS MODELOS MATEMÁTICOS DE OPTIMIZACIÓN

2.4.1 Optimización local

Los métodos de optimización local se caracterizan por obtener la mejor solución en la inmediaciones del punto de inicio, con ello tiene una fuerte dependencia del punto de arranque del algoritmo. Estos utilizan la información del gradiente, requieren el cálculo de derivadas y, en definitiva, imponen sobre el espacio de búsqueda unas condiciones de diferenciabilidad y continuidad difíciles de garantizar y controlar en la práctica.

2.4.2 Optimización Global

Exhiben una gran independencia de la naturaleza del espacio de soluciones y, a diferencia de las técnicas de búsqueda local, son capaces de atravesar un espacio de búsqueda con múltiples mínimos o máximos locales y alcanzar una solución global al problema, entendiendo como tal la mejor solución posible o una solución en las inmediaciones de la región que contiene a la solución óptima.

2.4.3 Clasificación de los métodos de optimización

- ✚ **Estocásticos o heurísticos:** evalúan “f” sobre una muestra aleatoria sobre la región de interés, denominados también no confiables. Entre ellos está el método de recocido simulado (simulated annealing) y el de búsqueda aleatoria (random search). La eficiencia es una característica de estos

métodos y se emplean para resolver problemas de gran escala (+100 variables).

- ✚ **Determinísticos o clásicos:** Entre ellos el método de ramas y bordes (Branch and Bound) y métodos de intervalos (Interval methods). Estos no envuelve ningún elemento aleatorio, son los denominados métodos confiables.
- ✚ **Métodos de puntos (point methods):** estos métodos calculan valores de la función sobre puntos de muestras pero son incapaces para resolver confiablemente un problema de optimización global.
- ✚ **Métodos de bordes (Boundary Methods):** calculan cotas sobre conjuntos compactos. Si son implementados apropiadamente y considerando errores de redondeo, pueden producir soluciones globales rigurosas.

Riojas (2005), en su trabajo sobre el área Investigación de Operaciones específica que en los últimos años han aparecido una serie de métodos bajo el nombre de metaheurísticos con el propósito de obtener mejores resultados que los alcanzados por los métodos heurísticos tradicionales. El sufijo “meta” significa “más allá”, a un nivel superior, las metaheurísticas son estrategias para diseñar o mejorar los procedimientos heurísticos con miras a obtener un alto rendimiento.

A partir de 1986 han aparecido muchas propuestas de pautas o guías para diseñar mejores procedimientos de solución de problemas combinatorios. Los profesores Osman y Kelly (1996) indican que los procedimientos metaheurísticos son una clase de métodos aproximados que están diseñados para resolver problemas difíciles de optimización combinatoria, en los que los heurísticos clásicos no son efectivos. Los metaheurísticos proporcionan un marco general para crear nuevos

algoritmos híbridos combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos.

Riojas, A. (2005), las metaheurísticas se sitúan conceptualmente “por encima” de las heurísticas en el sentido que guían el diseño de estas, pueden estar compuestas por una combinación de algunas heurísticas. Por ejemplo una metaheurística puede usar una heurística constructiva para generar una solución inicial y luego usar otra heurística de búsqueda para encontrar una mejor solución.

El tipo de método metaheurístico está en función de qué tipo de heurística y se puede hacer una clasificación en:

- Metaheurísticas constructivas
- Metaheurísticas evolutivas
- Metaheurísticas de búsqueda

Los métodos metaheurísticos de búsqueda presuponen que existe una solución y realizan procedimientos de búsqueda, la diferencia con los métodos analíticos es que no necesariamente se encontrará la solución óptima. Uno de los riesgos al usar un algoritmo de búsqueda es el de alcanzar un óptimo local del que ya no sea posible salir.

Las principales metaheurísticas de búsqueda global surgen de las tres formas principales de escapar de los óptimos locales:

- a) Volver a comenzar la búsqueda desde otra solución inicial (Multi start).
- b) Modificar la estructura de entornos (metaheurística de entornos variables).

c) Permitir movimientos de empeoramiento de la solución actual.

En este tercer grupo de metaheurísticas está la búsqueda tabú (tabu search), empleada en éste trabajo para el problema de diseño integrado de columnas de destilación binarias. Otros autores (Pérez y Basterrechea, (2005)) encuentran que la búsqueda tabú, puede pertenecer a métodos heurísticos deterministas.

Lo que si se quiere es dar una visión general sobre los métodos más explorados actualmente y que para el caso específico del problema de Diseño Integrado de Columnas de Destilación Binarias, se han aplicado tres de ellos para determinar cual se adapta mejor y arroja la solución óptima. Y es así que se determina que para un proceso muy particular, algún método o combinación de ellos, será la herramienta adecuada.

Basado en los principios de la selección y evolución natural, el esquema clásico de los algoritmos genéticos hace evolucionar la población en base a la presión que ejercen los operadores de selección, cruce y mutación. Sin embargo, la amplia difusión del método de optimización ha hecho que los propios investigadores propongan continuas variantes y mejoras al esquema clásico, en un intento por incrementar el rendimiento del método.

Es importante mencionar el uso del método metaheurístico de optimización llamado “búsqueda tabú”, ya que este se emplea como herramienta previa a la resolución del problema MINLP-DAE, al ser empleado para abordar la característica binaria del diseño.

2.4.4 Búsqueda tabú (BT)

González y Cortés (2012), establecen que la búsqueda tabú (BT) o tabú search (TS) en inglés, es un procedimiento metaheurístico utilizado para guiar un algoritmo heurístico de búsqueda local para explorar el espacio de soluciones más allá de la simple optimalidad local. Esta técnica al igual que el recocido simulado está basada en la búsqueda de soluciones vecinas evitando óptimos locales, pero lo hace de forma determinística emulando los procesos de memoria del ser humano. La figura 2.5 se muestra como actúa la búsqueda tabú en un entorno de soluciones.

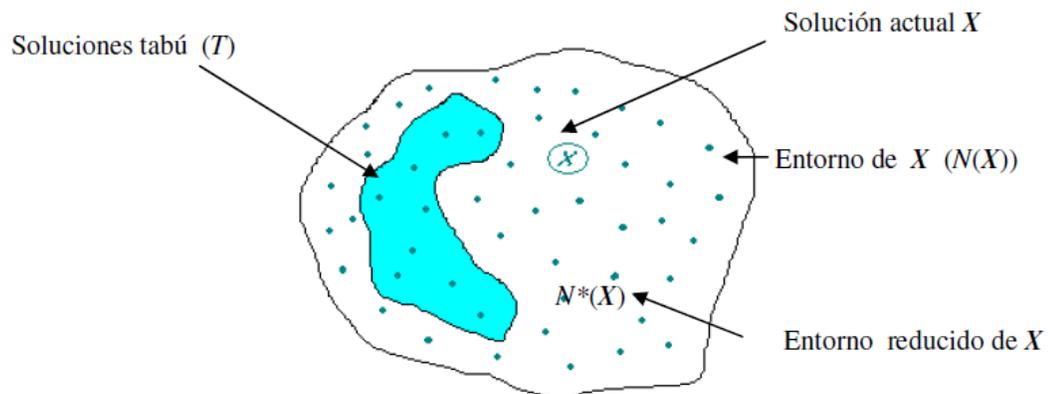


Figura 2.5 Representación de la estructura de entorno de una solución Fuente: Gonzalez y Cortes (2012)

Los elementos básicos de la búsqueda tabú tienen varias características importantes:

1. Memoria adaptativa:

- a. Selectividad, incluyendo olvido estratégico.
- b. Abstracción y descomposición.

c. Tiempo:

- i. Recencia de eventos.
- ii. Frecuencia de eventos.
- iii. Diferenciación entre corto y largo plazo.

d. Calidad e impacto:

- i. Atracción relativa de elecciones alternativas.
- ii. Magnitud de cambios en relaciones de estructura o restricciones.

e. Contexto:

- i. Interdependencia regional.
- ii. Interdependencia estructural.
- iii. Interdependencia secuencial.

2. Exploración sensible:

- a. Imposición estratégica de limitaciones e inducciones: condiciones tabú y niveles de aspiración.
- b. Enfoque concentrado en buenas regiones y buenas características de las soluciones: procesos de intensificación.
- c. Caracterización y exploración de nuevas regiones prometedoras: procesos de diversificación.
- d. Patrones de búsqueda no monótonos: oscilación estratégica.
- e. Integración y extensión de soluciones: reencadenamiento de trayectorias.

La adecuada combinación de estos aspectos conduce progresivamente a mejores soluciones e implementaciones prácticas.

El énfasis en la exploración sensible en búsqueda tabú, ya sea en una implementación determinística o probabilística, se deriva de la suposición de que una mala elección estratégica puede producir más información que una buena elección al azar. Esto es debido a que, al emplear un sistema de memoria, una mala elección basada en estrategia puede dar claves útiles acerca de cómo podrían hacerse modificaciones provechosas a la estrategia.

De esta manera les son atribuidos a las soluciones distintas características, conocidas en BT como atributos, que bajo ciertas condiciones pueden declararse "tabú", siendo obviadas entonces por el proceso de búsqueda pudiendo ser soluciones de calidad, que de esta manera posee la facultad de dirigir la exploración.

La búsqueda tabú trata de optimizar una función $f(x)$ en un conjunto. Para ello la búsqueda tabú comenzará como cualquier búsqueda local, procediendo iterativamente de un punto a otro hasta satisfacer un criterio de terminación. Cada $x \in X$, tiene un entorno (o vecindad) asociado $N(x) \subseteq X$, y cada solución $x' \in N(x)$ se puede alcanzar desde x mediante una operación llamada movimiento

2.4.5 Algoritmos Genéticos

El algoritmo genético es una técnica de búsqueda basada en la teoría de la evolución de Darwin, que ha cobrado tremenda popularidad en todo el mundo durante los últimos años. Esta técnica se basa en los mecanismos de selección que

utiliza la naturaleza, de acuerdo a los cuales los individuos más aptos de una población son los que sobreviven, al adaptarse más fácilmente a los cambios que se producen en su entorno. Hoy en día se sabe que estos cambios se efectúan en los genes de un individuo (unidad básica de codificación de cada uno de los atributos de un ser vivo), y que sus atributos más deseables (es decir, los que le permiten adaptarse mejor a su entorno) se transmiten a sus descendientes cuando éste se reproduce sexualmente.

Un investigador de la Universidad de Michigan llamado John Holland era consciente de la importancia de la selección natural, y a fines de los años 60 desarrolló una técnica que permitió incorporarla a un programa. Su objetivo era lograr que las computadoras aprendieran por sí mismas. A la técnica que inventó Holland se le llamó originalmente "planes reproductivos", pero se hizo popular bajo el nombre "algoritmo genético".

Versiones más complejas de algoritmos genéticos generan un ciclo iterativo que directamente toma a la especie (el total de los ejemplares) y crea una nueva generación que reemplaza a la antigua una cantidad de veces determinada por su propio diseño. Una de sus características principales es la de ir perfeccionando su propia heurística en el proceso de ejecución, por lo que no requiere largos períodos de entrenamiento especializado por parte del ser humano.

Los algoritmos genéticos usan una analogía directa con el comportamiento natural, ellos trabajan con una población de individuos, cada uno de los cuales representa una solución factible a un problema dado. A cada individuo se le asigna un valor ó puntuación, relacionado con la bondad de dicha solución. En la naturaleza esto equivaldría al grado de efectividad de un organismo para competir por unos determinados recursos. Cuanto mayor sea la adaptación de un individuo al

problema, mayor será la probabilidad de que el mismo sea seleccionado para reproducirse, cruzando su material genético con otro individuo seleccionado de igual forma. Este cruce producirá nuevos individuos descendientes de los anteriores, los cuales comparten algunas de las características de sus padres. Cuanto menor sea la adaptación de un individuo, menor será la probabilidad de que dicho individuo sea seleccionado para la reproducción, y por tanto de que su material genético se propague en sucesivas generaciones. De esta manera se produce una nueva población de posibles soluciones, la cual reemplaza a la anterior y verifica la interesante propiedad de que contiene una mayor proporción de buenas características en comparación con la población anterior. Así a lo largo de las generaciones las buenas características se propagan a través de la población.

Al favorecer el cruce de los individuos mejor adaptados, van siendo exploradas las áreas más prometedoras del espacio de búsqueda.

2.5 PROBLEMA DE DISEÑO INTEGRADO UTILIZANDO MATLAB R2012a

Es una herramienta de software matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M).

Este conjunto de funciones logran minimizar o maximizar funciones no lineales generales. Es decir, al usar la palabra “optimización” en MATLAB R2012a se refiere al proceso de búsqueda del mínimo o máximo de una función, denominada comúnmente “función objetivo” en las funciones de optimización que incluye el “Toolbox” de Optimización de MATLAB R2012a. Además, este “Toolbox”

también incluye funciones para la resolución de ecuaciones no lineales y problemas de mínimos cuadrados.

Cada una de las funciones resuelve un tipo optimización, para el caso de estudio, se sabe de acuerdo con el modelado matemático presentado más adelante en la “descripción del proceso o sistema”, que el problema es uno de programación no lineal entera mixta con ecuaciones algebraico-diferenciales (MINLP-DAE), requiriéndose la minimización de la función costo (no lineal), sujeta a restricciones.

Es importante tener en cuenta que los algoritmos de amplia escala no son aplicables a todo tipo de problemas que se puedan formular; por ejemplo, para una de las funciones empleadas en este trabajo “fmincon”, el algoritmo de amplia escala que usa MATLAB R2012a cubre los casos en los que solamente hay límites con restricciones. Para cada caso en el que se aplica un algoritmo de amplia escala se necesita una información adicional específica.

La función antes mencionada “fmincon” emplea uno de los siguientes algoritmos: active-set, interior-point, sqp, or trust-region-reflective.

2.6 BÚSQUEDA TABÚ (BT) Ó TABÚ SEARCH (TS) EN MATLAB R2012a

Configuración inicial. Una configuración es el conjunto de variables enteras del problema dispuesto en un arreglo (vector o matriz). La configuración inicial puede ser generada de manera aleatoria o puede ser obtenida utilizando un algoritmo constructivo que utilice factores de sensibilidad o cierta lógica heurística.

Para efectuar la configuración inicial de la búsqueda tabú en Matlab R2012a, se establecen las siguientes opciones con el respectivo valor asignado o función, de acuerdo al sistema estudiado.

- Máximo tamaño del paso de búsqueda: $\text{num2str}(p.\text{Step})$
- Valores iniciales: $\text{num2str}(p.y0)$
- Número máximo de iteraciones: $\text{num2str}(p.\text{Iter})$
- Tamaño máximo de la lista tabu: $\text{num2str}(p.N)$
- Contador de iteraciones: $s = 0$;
- Tamaño del paso inicial: $\text{Step} = 1$;
- Valor inicial de la función objetivo: $f_{\text{min}} = 1e10$;
- Valor inicial (pivote): $y_{\text{min}} = p.y0$;
- Valor inicial para las variables continuas: $x_{\text{min}} = p.x0$.
- Indicador de si se encontró o no un nuevo mínimo: $\text{Ind} = 0$.
- Numero de optimizaciones continuas: $\text{NeVal} = 0$
- Numero de total evaluación de función objetivo: $\text{NeFun} = 0$

Generación del vecindario. Un vecino de una configuración X es una configuración X' obtenida a partir de X . Para el problema estudiado un vecino se puede generar aumentando o disminuyendo el número de platos o bandejas de la columna en una cantidad específica.

- Contador de iteraciones: $s \leq p.\text{Iter}$, $\text{Step} \leq p.\text{Step}$
- Aumentar el contador de iteraciones: $s = s + 1$;
- Generación de alrededores (condiciones de fronteras periódicas): En el proceso se cambia $y0$ para que cumpla las condiciones de fronteras: $y0 = y_{\text{min}}$; $x0 = x_{\text{min}}$.

- Generación de alrededores: $L_aux = \text{surround}(y_0, L, U, \text{Step})$, donde L y U son los límites superiores e inferiores de búsqueda definidos previamente como $L = [10 \ 4]$, $U = [100 \ 96]$, respectivamente.

Selección del mejor vecino. Una vez que el vecindario es definido, cada vecino es evaluado para determinar el valor de su función objetivo y si cumple o no con las restricciones planteadas; de esta manera se determina la factibilidad de la configuración vecina. Los vecinos son clasificados en una lista de acuerdo al valor de la función objetivo y el proceso selecciona el mejor candidato. El primer candidato de la lista (de mejor función objetivo) es seleccionado si no es tabú (prohibido) y si es factible, de lo contrario se busca entre los siguientes vecinos. En cada iteración del proceso es almacenada la mejor solución (incumbente) es posible que en una iteración dada, el mejor candidato de la lista no tenga una mejor solución que la encontrada hasta el momento; no obstante el proceso de selección del mejor vecino es el mismo y el proceso continúa. En Matlab R2012a se escribe:

La evaluación de los puntos se denota de la siguiente manera, dependiendo del tipo de optimización:

- Estado estacionario: $[x, f, \text{exitf}, \text{op}] = \text{Opt_Con_SS}(L_act(:, i)', x_0, \text{param})$.
- Optimización estado estacionario solo control:
 $[xs, f, \text{exitf}, \text{op}] = \text{Opt_Con_SS}(L_act(:, i)', x_0(1:3), \text{param})$.
- Diseño Integrado: $[x, f, \text{exitf}, \text{op}] = \text{Opt_Con_Int}(L_act(:, i)', x_i, \text{param}, p.ty)$.
- Comparar cada termino con la lista tabú: $n = \text{size}(L_Tabu, 2)$.
- Comparar cada término con la lista tabú: $N_act = 0$, $n = \text{size}(L_Tabu, 2)$;
- Condición de no prohibición del punto:
 $f \leq f_min, \text{op.constrviolation} \leq \text{rest}$.

- Aceptación del punto: $\text{num2str}(L_aux(:,i))$.
- Aumento el numero de aceptados: $N_act = N_act + 1$; $L_act(:,N_act) = L_aux(:,i)$. Se guarda el punto a ensayar, se aumenta el paso a través de $\text{Step} = \text{Step} + 1$ para cada punto seleccionado y se aumenta el número de evaluaciones $\text{NeVal} = \text{NeVal} + 1$.
- Condición de prohibición del punto por violación de restricciones o por no observación de mejora de la función objetivo: $\text{op.constrviolation} > \text{rest}$.
- Adición del punto a la lista tabú: $L_Tabu = [L_act(:,i), L_Tabu]$.

Actualización de la estructura tabú. Se requiere de una estructura tabú que tiene la misma codificación de la configuración X y que almacena los sitios prohibidos o tabú. Una posición de la configuración X puede estar marcada como tabú durante un número específico de iteraciones. Este número es almacenado en la estructura tabú y es conocido como “Status Tabú”. Varias posiciones de la configuración X pueden estar prohibidas y el “Status Tabú” puede o no ser variable a lo largo del proceso (memoria adaptativa) para prevenir el ciclaje y el hacer el proceso de búsqueda sea inteligente.

Un vecino que sea generado a partir de una posición prohibida no puede ser seleccionado de la lista de candidatos a no ser que su función objetivo sea mejor que la incumbente encontrada hasta el momento durante el proceso (criterio de aspiración). Este criterio compensa, en parte, el hecho de que al establecer posiciones tabú se reduce el espacio de búsqueda y pueden ser ignoradas configuraciones de buena calidad. Finalmente cuando un mejor vecino es seleccionado, la posición de la configuración X a partir de la cual fue generado se debe prohibir durante k iteraciones y los sitios que están prohibidos deben

disminuir su estado tabú en una iteración. Esto corresponde al proceso de actualización de la estructura tabú. En Matlab R2012a se empleará:

- Redimensionamiento de la lista tabu: if (size (L_Tabu,2)>p.N), L_Tabu = L_Tabu(:,1:p.N).
- Adición del valor mínimo actual: L_Tabu = [L_Tabu, y_min'].

2.7 MÉTODO DEL GRADIENTE EN MATLAB R2012a. La función “fmincon”

Es una función de optimización empleada para minimización con restricciones, la cual encuentra el mínimo de una función no lineal de varias variables con restricciones, según Vizcaino (2006).

El propósito de esta función es encontrar el mínimo del valor escalar que devuelve una función de varias variables no lineal y sujeta a una serie de restricciones.

La función objetivo representa la función escalar de varias variables, se denota de la siguiente manera:

$$\begin{aligned}
 & \min_x f(x) \\
 & \text{sujeto a } c(x) \leq 0, \text{ ceq}(x) \leq 0, \\
 & A \cdot x = b, A_{\text{eq}} \cdot x = \text{beq}, \\
 & \text{lb} \leq x \leq \text{ub}
 \end{aligned} \tag{1}$$

A, Aeq son matrices. f, x, b, beq, lb, ub son vectores. c(x) y ceq(x) son funciones que devuelven un vector. c(x), ceq(x) y f(x) pueden ser funciones no lineales.

Para el caso que ocupa, una de las funciones que se emplea es la función “lsqnonlin” (resolvedor de problemas no lineales de mínimos cuadrados, incluyendo ajuste de datos) y “ode23” (resolvedor de ecuaciones diferenciales).

Los algoritmos de optimización internos que usan las rutinas del “Toolbox” para operar son seleccionados como opciones de la función.

Uno de los argumentos de entrada más interesantes de esta y otras funciones de minimización es la ya comentada función objetivo, *fun*, o función a ser minimizada. *fun* toma el vector *x* y devuelve un valor escalar, por ejemplo *f*, que es el valor que toma la función evaluada en *x*.

El argumento que define la función objetivo, se puede especificar de dos formas. *fun* puede ser el nombre de una función con extensión .m, realizada por el usuario, y que realiza los cálculos necesarios para evaluar $f(x)$ en *x*. La segunda forma de pasar el argumento *fun* a la función ‘fmincon’ es como un objeto en línea, es decir, en el lugar del parámetro *fun* se escribe directamente la expresión de la función $f(x)$.

Se puede además obtener el gradiente de la función objetivo, para ello habrá que activar un elemento de la estructura de opciones de optimización “options” que permite que la función objetivo compute el cálculo del gradiente.

Para activar dicho elemento en la estructura de optimización se usa ‘optimset’, que es una instrucción que permite crear o modificar los parámetros de optimización de la estructura. El uso de esta instrucción es como sigue en este caso:

```
options = optimset('GradObj', 'on')
```

El gradiente es la derivada parcial de f en el punto x . Es decir, si se llama g al gradiente, el i -ésimo componente de g sería la derivada parcial de la función $f(x)$ respecto del i -ésimo componente del vector x y evaluada en la solución x . En el caso de que se active la opción de cálculo del gradiente, la función `fun` devuelve dos parámetros de salida, es decir, tendríamos en este caso algo tal que así:

```
function [f,g] = myfun(x)
f = ...;
if nargin > 1 % fun se usa con dos argumentos de salida
g = ...;
end
```

También se puede calcular la matriz Hessiana o discriminante, activando esta opción en la estructura `options`. Es decir, se debería escribir antes de usar “`fmincon`”:

```
options = optimset('Hessian', 'on')
```

En este caso, también se puede chequear el valor de la variable “`nargout`” para evitar el computo de H si la función `fun` se llama con sólo con uno o dos argumentos de salida (es decir, en el caso en el que la optimización sólo necesita los valores de f y g , pero no H).

La matriz Hessiana es una matriz simétrica, donde cada elemento es una derivada parcial segunda. Esto es, el elemento (i,j) de la matriz H sería la derivada parcial segunda de f con respecto al elemento i -ésimo y j -ésimo de x .

La estructura de la función objetivo en el caso de que se compute la matriz Hessiana sería algo tal que así:

```

function [f,g,H] = myfun(x)
f = ...;
if nargout > 1 % fun se usa con dos argumentos de salida
    g = ...;
    if nargout > 2
        H = ...; % matriz Hessiana evaluada en x
    end
end
end

```

Otro argumento de entrada de gran importancia a la hora de usar tanto la función de minimización no lineal “fmincon”, como cualquiera de las demás funciones del “Toolbox” de Optimización de MATLAB R2012a, es la estructura de opciones de optimización, llamada “options”. Esta estructura contiene varios campos que permiten influir de manera notable en el proceso que lleva a cabo la rutina de optimización.

Hay dos argumentos de salida que merecen ser tenidos en cuenta. Estos dos parámetros son *exitflag* y *output*. El argumento de salida *exitflag* es una bandera que indica al usuario la razón por la que ha finalizado la función de optimización y puede tomar los siguientes valores:

- $\text{Exitflag} > 0$, indica que la función $f(x)$ converge en la solución x .
- $\text{Exitflag} = 0$, indica que se ha alcanzado el número máximo de evaluaciones permitidas a la función $f(x)$ o de iteraciones de la función *fmincon*.
- $\text{Exitflag} < 0$, indica que la función $f(x)$ no converge a una solución.

El argumento de salida `output` es una estructura, cuyos campos contienen información sobre la optimización que se ha llevado a cabo en la rutina. Los parámetros de esta estructura y el significado de ellos son:

- `output.Iterations`: iteraciones llevadas a cabo por `fmincon`.
- `output.funcCount`: número de evaluaciones de la función objetivo.
- `output.algorithm`: El algoritmo usado por la función `fmincon`.
- `output.cgiterations`: número de iteraciones de PCG (solo para el caso de algoritmos de escala amplia).
- `output.stepsize`: El tamaño del paso final tomado (sólo para algoritmos de media escala).
- `output.firstorderopt`: Es una media de optimicidad de primer orden (sólo para algoritmos de escala amplia).

CAPÍTULO III

MARCO METODOLÓGICO

3.1 Tipo de investigación

El trabajo que se desarrolló en cuanto al nivel de investigación del tipo *descriptiva* en el área de control de procesos debido al grado de profundidad con el cual se aborda el fenómeno de diseño integrado y por el enfoque de sus objetivos.

En cuanto a la estrategia o diseño de la investigación es de tipo *documental* debido a que se realizó una exhaustiva revisión bibliográfica de todos los estudios previos y actualizados sobre diseño integrado.

Las técnicas de recolección de datos consisten en el análisis documental y el análisis de contenido y la información obtenida se tabula y procesa empleando procedimientos propios de la investigación científica como balances de materia y energía en estado estacionario y transitorio. Se emplean además las distintas ecuaciones expuestas en el diseño integrado de columnas de destilación binarias, se codifican en un programa computacional y se registran los resultados obtenidos bien sea mediante tablas o gráficas los resultados obtenidos. Entre las técnicas lógicas se emplean tanto la síntesis como el análisis siendo los principales instrumentos el computador junto con una herramienta computacional adecuada.

3.2 Fases de la investigación

A continuación se muestran los pasos metodológicos que se llevan a cabo para el cumplimiento de los objetivos propuestos se desarrollará a lo largo de tres etapas:

- **Diagnóstico o selección:** Se realizó una búsqueda bibliográfica y documental de libros, publicaciones y trabajos asociados al diseño

integrado de columnas de destilación binarias, luego se organiza, clasifica y se selección de la información recopilada.

- **Factibilidad:** Se refiere a la disposición de recursos materiales, técnicos, económicos, humanos, tiempo y de información que permitan establecer que el proyecto es viable y que se puede realizar. Los recursos materiales y técnicos comprenden básicamente el computador y la herramienta computacional, mientras que los recursos económicos representan montos despreciables considerando que ya se dispone previamente de dichos recursos técnicos. El recurso más importante en el proyecto se refiere al tiempo que se necesita invertir en la investigación documental pasando por la programación o implementación del algoritmo hasta la obtención de resultados producto de la corrida del programa.
- **Diseño:** Para lograr efectuar el diseño integrado de procesos en el presente trabajo se establecieron 4 pasos básicos, lo cual comprende desde obtener la premisas, diseño del algoritmo, implementación en una herramienta computacional hasta la validación de resultados obtenidos.

3.3 Desarrollo procedimental:

Con el fin de cumplir con los objetivos planteados el desarrollo del trabajo se basa en el siguiente procedimiento metodológico.

3.3.1 Definir los aspectos técnicos necesarios para el caso específico a diseñar

El sistema está definido por una columna de destilación binaria. La alimentación a la columna es un líquido saturado de composición conocida y definida. Se debe determinar el número de bandejas, localización de la alimentación, diámetro de la columna, velocidades de flujo y composiciones de los productos de fondo y destilado. La pureza de éstos productos se encuentra establecida por lo tanto deberá controlarse. La superestructura considerada para el problema se muestra en la figura 3.1

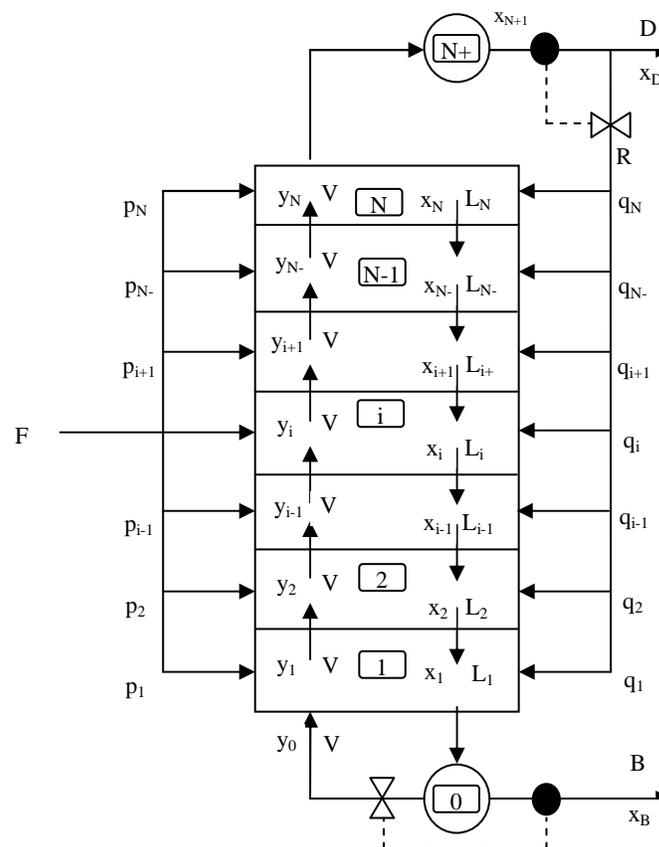


Figura 3.1 Diagrama del proceso. Fuente: Schweiger C., Floudas C. (1997)

Esclarecer los resultados que se deben obtener:

- **Finalidad primordial del sistema:** Separar una corriente de una mezcla líquida saturada conformada en sus dos componentes con una pureza requerida.
- **Tipo de sistema de control:** La superestructura considerada para el problema incluye un esquema de control del tipo Proporcional-Integral (PI), para el control de las composiciones de fondo y destilado. El flujo de vaporizado es usado como control de la composición del fondo y el caudal de reflujo se usa para controlar la composición del destilado. Las constantes asociadas a los controladores están representadas en ganancias y constantes de tiempo (K_v , K_R , τ_v , τ_R).
- **Objetivo de implementar el sistema de control:** Controlar las composiciones de los productos de fondo y destilado, a través del flujo de vaporizado o flujo molar de vapor y del caudal de reflujo.
- **Objetivo de efectuar el diseño integrado de proceso:** El objetivo primordial que se persigue con el diseño integrado de proceso es determinar las dimensiones físicas óptimas de la columna de destilación binaria (diámetro, número de platos o bandejas y ubicación de la alimentación) y los parámetros de entonación de los controladores asociados al control de las composiciones de fondo y destilado. El diseño que se propone obtener debe ser óptimo tanto a los criterios económicos como a la controlabilidad.

El criterio económico está representado por el costo total a través del costo anualizado y es una función del costo de utilidad por un factor de impuesto y el costo capital entre el período de recuperación de la inversión. El costo de capital está determinado por el valor resultado del diseño de la columna, como lo son el diámetro y el número de bandejas.

Por lo tanto se debe plantear la correlación matemática para establecer su relación, así como también para el costo de utilidad, que está relacionado con los calores latentes de vaporización y de condensación, el flujo o caudal de vaporizado y los coeficientes de costo de utilidad. La expresión para el costo total está documentada por Schweiger C., Floudas C. (1997).

Por su parte, el criterio de controlabilidad se incluye en el diseño por medio del “error cuadrático integral (ISE)” que es un criterio de desempeño que permite evaluar el comportamiento de los sistemas y representa la base para optimizarlos respecto del comportamiento de su error en el estado transitorio.

El ISE se representa a través de una ecuación diferencial que involucra las composiciones de fondo y destilado y sus respectivos puntos de consigna. Mientras más parecidos sean los valores reales de los puntos de consigna, más controlado estará el proceso y a su vez el ISE tenderá a ser cero, por lo cual el objetivo de la controlabilidad dinámica es la minimización del ISE y se denota con la letra “ μ ”. Las ventajas de su empleo radican en que proporciona importancia a los errores grandes y presenta una rápida respuesta.

Por lo tanto es necesario listar los parámetros constantes, en la Tabla 3.3.1 se muestran dichos parámetros.

Tabla 3.3.1 Parámetros que se asumen constantes en el problema

Descripción	Parámetro	Valor
Flujo molar de alimentación	F	1kmol/min
Fracción molar de F	z	0,45
Volatilidad relativa	α	2,5
Fracción molar destilado	x_D	$\geq 0,98$
Fracción molar residuo	x_B	$\leq 0,02$
Fugacidad	ff	0,8
Peso molecular	MW	92kg/kmol
Densidad del líquido	ρ_L	883 kg/m ³
Densidad del vapor	ρ_V	2,9 kg/m ³
Calor latente de vaporización	ΔH_{vap}	0,031 x 10 ⁶ kJ/kmol
Calor latente de condensación	ΔH_{cond}	0,032 x 10 ⁶ kJ/kmol
Coefficiente de costo de utilidad	c_{Ips}	6,1 x 10 ⁵ (\$/min)/(10 ⁶ kJ año)
Coefficiente de costo de utilidad	c_{cw}	1,5 x 10 ⁴ (\$/min)/(10 ⁶ kJ año)
Factor de impuesto	β_{tax}	0,4
Período de retorno de la inversión	β_{pay}	4 años
Altura sobre el vertedero	hGD	0,0254 m

Fuente: Luyben, M. Y Floudas C. (1996).

3.3.1.1 Definir el conjunto de variables su clasificación y las técnicas de optimización objeto de estudio.

- **Variables manipuladas:** Caudal de vapor (V) y reflujo (R).
- **Variables controladas:** Composiciones de los productos, como fracciones molares de destilado X_D y fondo o residuo X_B .
- **Variables de formulación matemática:** Según Schweiger C. y Floudas C., (1997), para el problema de diseño integrado del sistema definido anteriormente, se presentan diferentes tipos de variables y restricciones. Las variables continuas las representan las velocidades de flujo, composiciones, dimensiones de equipos, entre otras. Las variables continuas son clasificadas como variables de diseño o variables de decisión invariantes en el tiempo “ v ”, variables de estado dinámico “ $z(t)$ ” y variables de control o variables de decisión dependientes “ $u(t)$ ”. Las variables enteras “ y ” son usadas para representar la existencia de unidades de proceso.

Las variables listadas en la Tabla 3.3.2 están divididas las siguientes categorías:

Tabla 3.3.2. Clasificación de las variables del proceso

Notación	Descripción
V	VARIABLES DE DECISIÓN INVARIANTES EN EL TIEMPO
Y	VARIABLES ENTERAS
$z(t)$	VARIABLES DE ESTADO DINÁMICO
$u(t)$	VARIABLES DE CONTROL O VARIABLES DE DECISIÓN DEPENDIENTES
t	TIEMPO
t_i	INSTANTE DE TIEMPO

Fuente: Schweiger C., Floudas C. (1997)

Tabla 3.3.3. Variables para la formulación del modelo matemático

Descripción	Nomenclatura	Unidades	Tipo de Variable
Existencia de alimentación en la bandeja i	p_i	N/A	Entera
Existencia de reflujo en la bandeja i	q_i	N/A	Entera
Almacenamiento de líquido de la bandeja	M	kmol/min	Invariante en el tiempo
Constante de tiempo hidráulico de la bandejas	β	N/A	Invariante en el tiempo
Razón de reflujo en el estado estacionario	R_{ss}	kmol/min	Invariante en el tiempo
Vaporizado en el estado estacionario	V_{ss}	kmol/min	Invariante en el tiempo
Diámetro de la columna	D_c	m	Invariante en el tiempo
Ganancia controlador vaporizado	K_v	kmol/min	Invariante en el tiempo
Ganancia controlador reflujo	K_R	kmol/min	Invariante en el tiempo
Constante de tiempo controlador vaporizado	τ_v	min	Invariante en el tiempo
Constante de tiempo controlador reflujo	τ_R	min	Invariante en el tiempo
Composición del líquido,	X_i	N/A	Dinámica
Composición del vapor	y_i	N/A	Dinámica
Caudal de flujo del líquido en cada bandeja	L_i	kmol/min	Dinámica
Caudal de flujo de fondo	B	kmol/min	Dinámica
Caudal de flujo de destilado	D	kmol/min	Dinámica
Vaporizado o caudal de vapor	V	kmol/min)	Dinámica
Caudal de flujo del reflujo	R	kmol/min	Dinámica
Error cuadrático integral	μ	min^2	Dinámica
Número total de bandejas	N_t	N/A	Entera
Término integral controlador reflujo destilado	I_D	min^2	Dinámica
Término integral controlador residuo	I_R	min^2	Dinámica

Fuente: Schweiger C., Floudas C. (1997)

- **Técnicas de optimización para dar solución al planteamiento**

matemático: La finalidad es explorar algunas áreas en el campo de la optimización, como lo son:

- (1) **Métodos Clásicos:** Se estudiarán los métodos de búsqueda local o determinísticos, específicamente la técnica de optimización basada en búsqueda de gradientes.
- (2) **Métodos Heurísticos o Estocásticos:** Se utilizará el método probabilístico basado en población: Algoritmos Genéticos, siendo a su vez un método global de optimización.

3.3.1.2 Establecer cuáles son los objetivos de optimización.

El objetivo primordial que se persigue con el diseño del proceso objeto de estudio es determinar las dimensiones físicas óptimas (diámetro, número de bandejas y ubicación de la alimentación) de la columna de destilación binaria para el sistema descrito y los parámetros de entonación del controlador asociados al control de las composiciones de los productos de fondo y destilado de la columna para que se lleve a cabo efectivamente la separación de los componentes de la mezcla binaria. El diseño que se propone obtener deberá ser óptimo tanto a los criterios económicos como a la controlabilidad del proceso garantizando una operación factible.

Como ya se indicó, los objetivos están representados por “El Costo total” y “El error cuadrático integral” (ISE), denotado con la letra “ μ ”.

Al ser el diseño integrado, un problema de optimización, los objetivos se traducen en:

- Minimizar el costo total, que es el objetivo de diseño de procesos y simultáneamente,
- Minimizar el ISE, que es el objetivo de la controlabilidad dinámica.

Para resolver este problema multiobjetivo se convierte uno de los objetivos en restricción $(obj_j \leq \varepsilon)$, en este caso el error cuadrático integral pasa a formar parte de las restricciones no lineales del problema, lo que implica resolver el problema MINLP-DAE a diferentes valores de ε que van desde el óptimo controlable hasta el óptimo económico, en este punto el problema multiobjetivo se convierte en un problema de programación no lineal con ecuaciones algebraico diferenciales.

3.3.2 Modelado matemático

Usando la figura 3.1 se obtiene el modelado dinámico para la optimización del sistema de la columna de destilación binaria

Balance de materia global

$$D + B - F = 0 \quad (2)$$

Balance por componentes en la caldera o rehervidor

$$M_R \frac{d_{XB}}{dt} = L_1(x_1 - x_B) + V(x_B - y_B) \quad (3)$$

Balance por componente en la bandeja o plato

$$M_R \frac{dx_i}{dt} = L_{i+1}(x_{i+1} - x_i) + V(y_{i-1} - y_i) + p_i(z - x_i) + q_i r(x_D - x_i) \quad (4)$$

Balance por componente en el condensador

$$M_C \frac{dx_D}{dt} = V(y_N - x_D) \quad (5)$$

Balance total en la cadera o rehervidor

$$\beta \frac{dB}{dt} = L_1 - V - B \quad (6)$$

Balance total por bandeja

$$\beta \frac{dL_i}{dt} = L_{i+1} - L_i + p_i F + q_i R \quad (7)$$

Balance total en el condensador

$$\beta \frac{dD}{dt} = V - R - D \quad (8)$$

Expresión de equilibrio en el calderín

$$y_B = \frac{\alpha x_B}{1 + x_B(\alpha - 1)} \quad (9)$$

Expresión de equilibrio en la bandeja i

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

Retardos quinto orden en la medición de la composición del fondo y destilado

$$\frac{d^5 x_{B,m}}{dt^5} + 5 \frac{d^4 x_{B,m}}{dt^4} + 10 \frac{d^3 x_{B,m}}{dt^3} + 10 \frac{d^2 x_{B,m}}{dt^2} + 5 \frac{d x_{B,m}}{dt} + x_{B,m} = x_B \quad (11)$$

$$\frac{d^5 x_{D,m}}{dt^5} + 5 \frac{d^4 x_{D,m}}{dt^4} + 10 \frac{d^3 x_{D,m}}{dt^3} + 10 \frac{d^2 x_{D,m}}{dt^2} + 5 \frac{d x_{D,m}}{dt} + x_{D,m} = x_D \quad (12)$$

Expresiones matemáticas para los controladores PI

$$\frac{dI_B}{dt} = x_{B,m} - x_B^* \quad (13)$$

$$\frac{dI_D}{dt} = x_{D,m} - x_D^* \quad (14)$$

$$V_C = V_{SS} + K_V (x_{B,m} - x_B^*) - \frac{K_V}{\tau_V} I_B \quad (15)$$

$$R_C = R_{SS} + K_R (x_{D,m} - x_D^*) - \frac{K_R}{\tau_R} I_D \quad (16)$$

Retardos finales de control

$$0,9 \frac{dV}{dt} = V_C - V \quad (17)$$

$$0,5 \frac{dR}{dt} = R_C - R \quad (18)$$

El diámetro de la columna debe ser suficientemente grande como para evitar la inundación de la columna. La inundación está dada por la velocidad de flujo de vapor o caudal de vapor en la columna. Se establece por ello una restricción apropiada que asegure que el diámetro de la columna es suficientemente largo:

$$D_c \geq 0,6719 \sqrt{V_{SS}} \quad (19)$$

El número de bandejas de la columna está determinada por la posición del reflujo usando la siguiente restricción:

$$N_t = \sum_i (i)(q_i) \quad (20)$$

Las siguientes restricciones lógicas se incluyen para asegurar:

- La existencia de una sola alimentación,

$$\sum_{i=1}^N p_i = 1 \quad (21)$$

- La existencia de una sola corriente de reflujo,

$$\sum_{i=1}^N q_i = 1 \quad (22)$$

- Que la alimentación se introduzca en la bandeja número cuatro (4) o por encima,

$$\sum_{i=1}^N (i) p_i \geq 4 \quad (23)$$

- Que la corriente de reflujo se introduzca la bandeja número diez (10) o por encima de ella.

$$\sum_{i=1}^N (i) q_i \geq 10 \quad (24)$$

- Que la corriente de reflujo esté por lo menos cuatro (4) platos por encima de la alimentación.

$$\sum_{i=1}^N (i)q_i - (i)p_i \geq 4 \quad (25)$$

Las siguientes expresiones se tomaron de Schweiger y Floudas (1997), las restricciones para el proceso establecido, primero la función objetivo se relaciona con los costos de diseño y utilidades a través de la expresión

$$Cost = \beta_{tax} \text{cost}_{util} + \text{cost}_{cap} / \beta_{pay}$$

$$\text{cost} = 7756 \cdot V_{SS} + 3,075 \cdot (615 + 324D_C^2 + 486 \cdot (6 + 0,76N_t)D_C) + 61,25N_t(0,7 + 1,5D_C^2) \quad (27)$$

Y la expresión de la función objetivo que relaciona la controlabilidad del sistema es:

$$\frac{d\mu}{dt} = t(x_D - x_D^*)^2 + t(x_B - x_B^*)^2 \quad (28)$$

El estado dinámico del problema se induce por una perturbación tipo escalón en la composición de la alimentación, matemáticamente se expresa de la siguiente forma:

$$Z = 0,45 + \frac{0,9}{1 + e^{10(t-10)}} \quad (29)$$

El método de restricción ε se aplica al problema multiobjetivo mediante la imposición de la siguiente restricción del punto final:

$$\mu(t_f) \leq \varepsilon \quad (31)$$

3.3.3 Resolución del problema de diseño integrado con el uso de los métodos de optimización seleccionados, e implementación en la plataforma computacional MATLAB R2012a.

3.3.3.1 Estructura algorítmica para la obtención de los valores óptimos de diseño y parámetros del controlador.

Esta estructura algorítmica se descompone en tres partes: el problema entero mixto, la naturaleza multiobjetivo y el problema de control óptimo.

El problema entero mixto se presenta en el diseño integrado de columnas de destilación binarias debido a la necesidad de determinar el número de bandejas o platos de la misma y la ubicación de la alimentación. Estas incógnitas representan las llamadas “alternativas estructurales” del problema MINLP-DAE y se representan en este sistema por las variables binarias p_i y q_i las cuales indican la existencia de alimentación y reflujo en la bandeja i respectivamente (su suma representa el número total de bandejas N). Este es un problema de decisión, en el que no solo se involucran variables que pueden representarse por valores reales, sino decisiones de tipo discreto que de forma natural se encuentran establecidas mediante variables enteras o binarias. De esta forma se observa que p_i , por ejemplo, toma el valor discreto de cero (0) o uno (1), dependiendo si en el plato o bandeja i , se encuentra la alimentación a la columna. Otra variable de decisión de este problema es la localización del reflujo, pero esta se obtiene de manera simultánea cuando se determina el número de bandejas para la columna, ya que se asume que no hay flujo de líquido en las bandejas por encima de la localización del reflujo.

El problema entero mixto es abordado a través de la “Búsqueda Tabú” o “Tabu Search” (TS), ya que es un método combinatorial. Se emplea al inicio del algoritmo de resolución matemática tanto para el estado estacionario como para la optimización dinámica. Como ya se mencionó, para resolver el problema multiobjetivo se convierte uno de los objetivos en restricción ($obj_j \leq \varepsilon$), en éste caso el error cuadrático integral pasa a formar parte de las restricciones no lineales del problema, lo que implica resolver el problema MINLP-DAE a diferentes valores de ε que van desde el óptimo controlable hasta el óptimo económico, en este punto el problema multiobjetivo se convierte en un problema de un solo objetivo de programación no lineal con ecuaciones algebraico diferenciales NLP-DAE.

El problema de control óptimo entero mixto (MIOCP) se transforma hasta un problema de programación no-lineal con restricciones algebraicas y diferenciales (MINLP/DAE) por aplicación de una técnica de parametrización de control.

El algoritmo MINLP/DAE se implementa en una herramienta computacional para el análisis de la interacción entre el control y el diseño. La solución de los problemas MINLP/DAE son repetidos con valores diferentes de ε . Los resultados deberán mostrar cuantitativamente la interacción entre los objetivos económicos del estado estacionario y el objetivo de controlabilidad dinámica.

A continuación se presenta la estructura algorítmica en forma de diagrama de bloques.

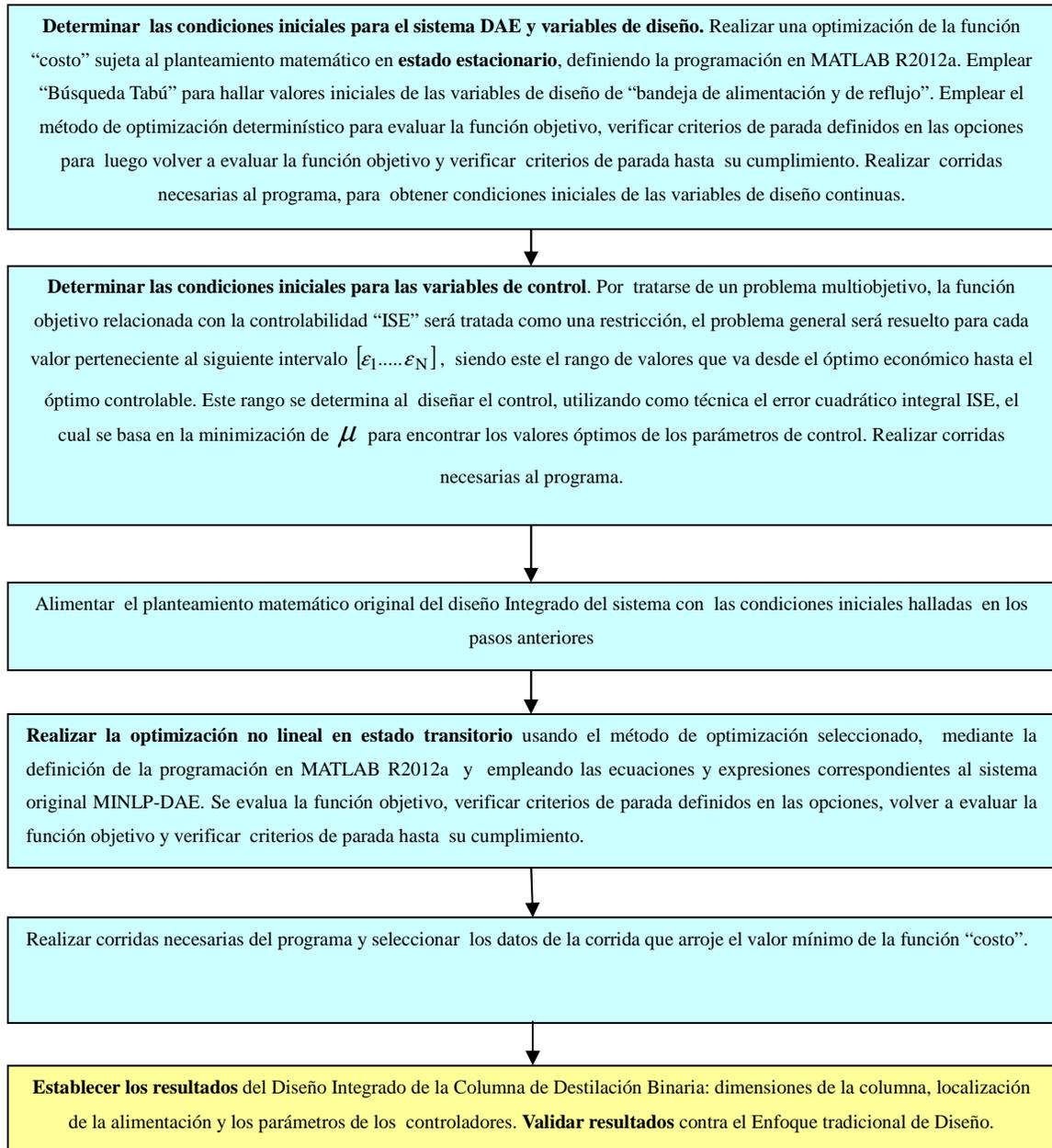


Figura 3.3.3.1 Diagrama de bloques de la estructura algorítmica para la resolución del problema de diseño integrado de una columna de destilación binaria.

Para formar la estructura algorítmica propiamente dicha, a partir de los pasos descritos en el diagrama de bloques de la figura anterior, se definen a continuación el conjunto de expresiones matemáticas que se deben usar en la programación.

Planteamiento matemático original (29)

Minimizar

$$\text{cost} = 7756 \cdot V_{SS} + 3,075 \cdot (615 + 324D_C^2 + 486 \cdot (6 + 0,76N_t)D_C) + 61,25N_t(0,7 + 1,5D_C^2)$$

Sujeto a:

$$D + B - F = 0$$

$$M_R \frac{dx_B}{dt} = L_1(x_1 - x_B) + V(x_B - y_B)$$

$$M_R \frac{dx_i}{dt} = L_{i+1}(x_{i+1} - x_i) + V(y_{i-1} - y_i) + p_i(z - x_i) + q_i r(x_D - x_i)$$

$$M_C \frac{dx_D}{dt} = V(y_N - x_D)$$

$$\beta \frac{dB}{dt} = L_1 - V - B$$

$$\beta \frac{dL_i}{dt} = L_{i+1} - L_i + p_i F + q_i R$$

$$\beta \frac{dD}{dt} = V - R - D$$

$$y_B = \frac{\alpha x_B}{1 + x_B(\alpha - 1)}$$

$$y_i = \frac{\alpha x_i}{1 + x_i(\alpha - 1)}$$

$$\frac{d^5 x_{B,m}}{dt^5} + 5 \frac{d^4 x_{B,m}}{dt^4} + 10 \frac{d^3 x_{B,m}}{dt^3} + 10 \frac{d^2 x_{B,m}}{dt^2} + 5 \frac{d x_{B,m}}{dt} + x_{B,m} = x_B$$

$$\frac{d^5 x_{D,m}}{dt^5} + 5 \frac{d^4 x_{D,m}}{dt^4} + 10 \frac{d^3 x_{D,m}}{dt^3} + 10 \frac{d^2 x_{D,m}}{dt^2} + 5 \frac{d x_{D,m}}{dt} + x_{D,m} = x_D$$

$$\frac{dI_B}{dt} = x_{B,m} - x_B^*$$

$$\frac{dI_D}{dt} = x_{D,m} - x_D^*$$

$$V_C = V_{SS} + K_V (x_{B,m} - x_B^*) - \frac{K_V}{\tau_V} I_B$$

$$R_C = R_{SS} + K_R (x_{D,m} - x_D^*) - \frac{K_R}{\tau_R} I_D$$

$$0,9 \frac{dV}{dt} = V_C - V$$

$$0,5 \frac{dR}{dt} = R_C - R$$

$$M = 7,538115 \left(\left(\left(0,0014134 / D_c \right)^{2/3} \right) + h_\omega \right) D_c^2$$

$$\beta = 0,05271 D_c^{1,3333}$$

Restricciones:

$$D_c \geq 0,6719 \sqrt{V_{ss}}$$

$$N_t = \sum_i (i)(q_i)$$

$$\sum_{i=1}^N p_i = 1$$

$$\sum_{i=1}^N q_i = 1$$

$$\sum_{i=1}^N (i) p_i \geq 4$$

$$\sum_{i=1}^N (i) q_i \geq 10$$

$$\sum_{i=1}^N (i) q_i - (i) p_i \geq 4$$

La restricción de la controlabilidad del sistema:

$$\mu \leq \varepsilon$$

Estructura de control para el diseño del sistema (30)

Minimizar

$$\mu$$

Sujeto a:

$$D + B - F = 0$$

$$M_R \frac{d_{XB}}{dt} = L_1(x_1 - x_B) + V(x_B - y_B)$$

$$M_R \frac{d_{Xi}}{dt} = L_{i+1}(x_{i+1} - x_i) + V(y_{i-1} - y_i) + p_i(z - x_i) + q_i r(x_D - x_i)$$

$$M_C \frac{d_{XD}}{dt} = V(y_N - x_D)$$

$$\beta \frac{dB}{dt} = L_1 - V - B$$

$$\beta \frac{dL_i}{dt} = L_{i+1} - L_i + p_i F + q_i R$$

$$\beta \frac{dD}{dt} = V - R - D$$

$$y_B = \frac{\alpha x_B}{1 + x_B(\alpha - 1)}$$

$$y_i = \frac{\alpha x_i}{1 + x_i(\alpha - 1)}$$

$$\frac{d^5 x_{B,m}}{dt^5} + 5 \frac{d^4 x_{B,m}}{dt^4} + 10 \frac{d^3 x_{B,m}}{dt^3} + 10 \frac{d^2 x_{B,m}}{dt^2} + 5 \frac{d x_{B,m}}{dt} + x_{B,m} = x_B$$

$$\frac{d^5 x_{D,m}}{dt^5} + 5 \frac{d^4 x_{D,m}}{dt^4} + 10 \frac{d^3 x_{D,m}}{dt^3} + 10 \frac{d^2 x_{D,m}}{dt^2} + 5 \frac{d x_{D,m}}{dt} + x_{D,m} = x_D$$

$$\frac{dI_B}{dt} = x_{B,m} - x_B^*$$

$$\frac{dI_D}{dt} = x_{D,m} - x_D^*$$

$$V_C = V_{SS} + K_V (x_{B,m} - x_B^*) - \frac{K_V}{\tau_V} I_B$$

$$R_C = R_{SS} + K_R (x_{D,m} - x_D^*) - \frac{K_R}{\tau_R} I_D$$

$$0,9 \frac{dV}{dt} = V_C - V$$

$$0,5 \frac{dR}{dt} = R_C - R$$

En las Figuras 3.3.3.2 y 3.3.3.3 se presenta la estructura algorítmica para la resolución del problema MINLP-DAE del diseño integrado de una columna de destilación binaria.

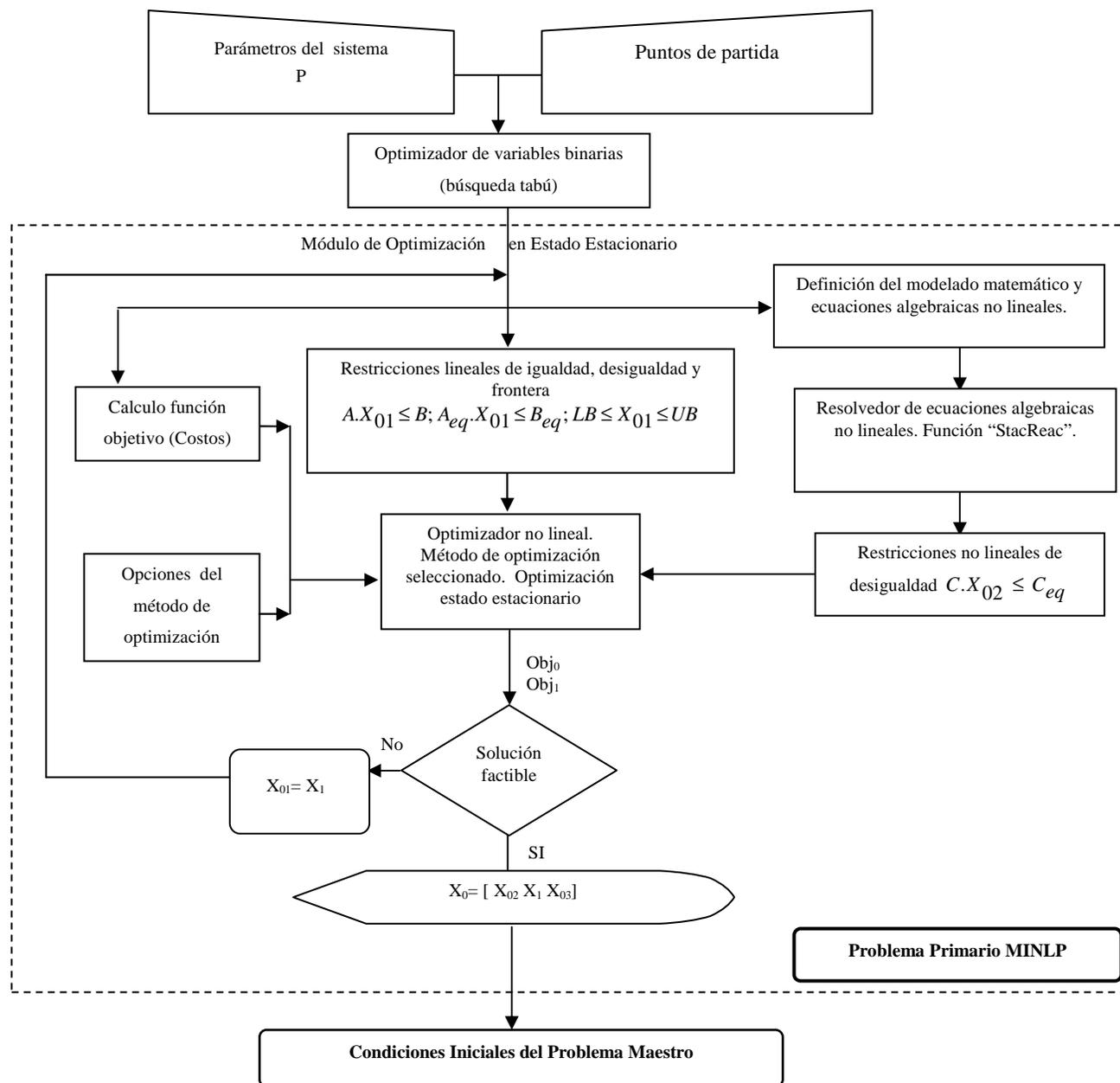


Figura 3.3.3.2. Algoritmo para la optimización en estado estacionario.

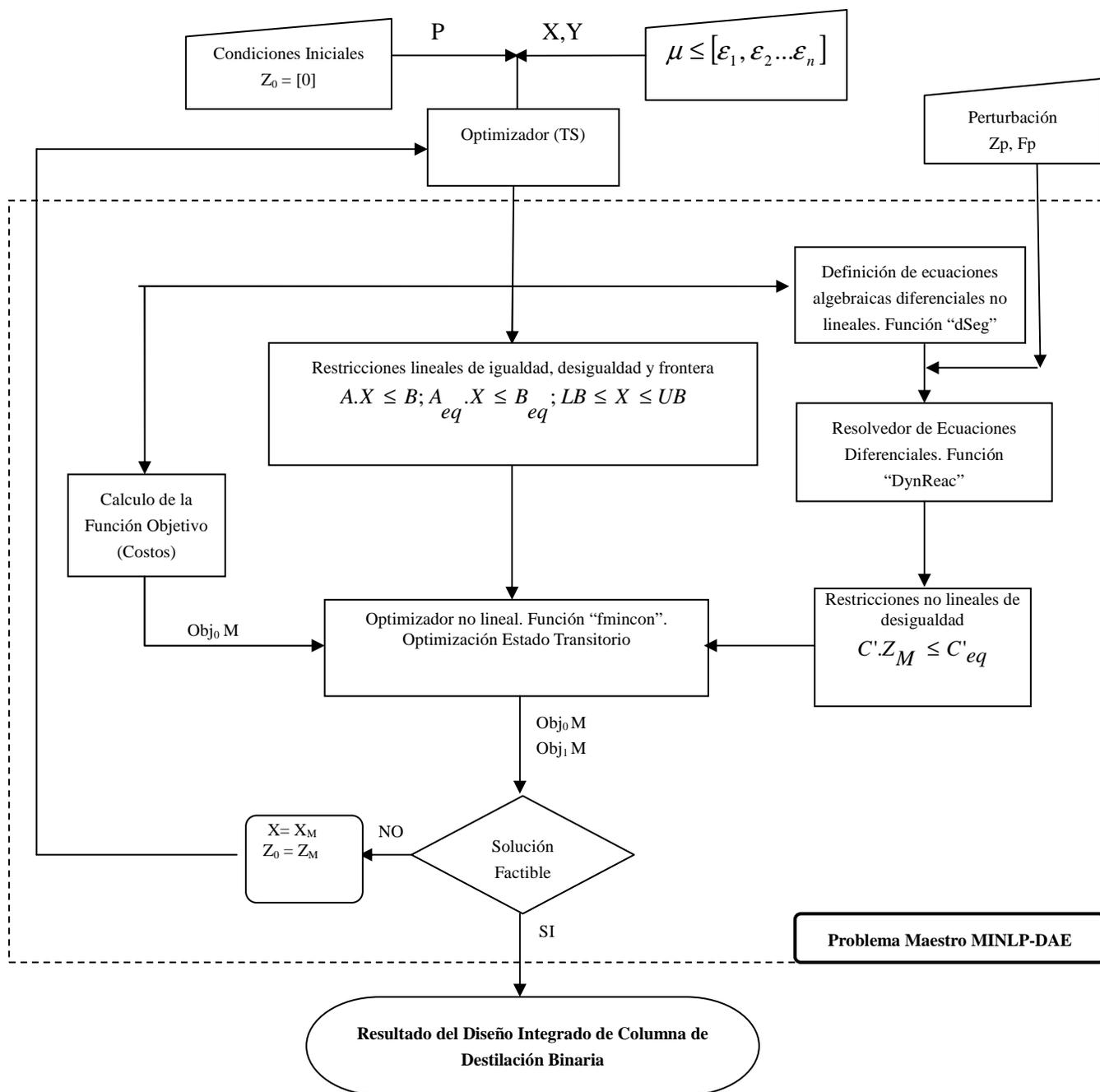


Figura 3.3.3.3 Algoritmo para la optimización en estado transitorio.

La definición de las variables utilizadas en el algoritmo se muestra a continuación:

Parámetros constantes para el problema primario y maestro

$$P = [F, \alpha, ff, MW, \rho_L, \rho_V, \Delta H_{\text{vap}}, \Delta H_{\text{cond}}, c_{\text{Ips}}, c_{\text{cw}}, \beta_{\text{tax}}, \beta_{\text{pay}}, h_{\text{CD}}]$$

Optimización no lineal estado estacionario: corresponden a las variables de diseño (puntos de partida del problema primario): $X_0 = [T_R, T_f, D_C, R_{SS}, V_{SS}, x_{\text{Bsp}}, x_{\text{Dsp}}, q, p]$, donde x_{Bsp} y x_{Dsp} son los puntos de consigna del fondo y destilado respectivamente.

Resultados optimización no lineal (estado estacionario): $X_0 = [T_R, T_f, D_C, R_{SS}, V_{SS}, D, V, L, x_D, y_B, Nt]$.

Optimización no lineal estado transitorio: variables de diseño (condiciones iniciales problema maestro): $X = [T_R, T_f, D_c, R_{SS}, V_{SS}, x_{\text{Bsp}}, x_{\text{Dsp}}, K_v, K_r, \tau_v, \tau_R]$.

Resultados del diseño integrado columna de destilación binaria: $[Nt, T_f, D_c, D, V_{SS}, R_{SS}, x_D, x_B, k_v, \tau_v, k_R, \tau_R, I_R, I_D, \varepsilon, C_c, C_u, \text{Cost}]$.

3.3.3.2 Implementación de la estructura algorítmica

La implementación de la estructura algorítmica de las Figuras 3.3.3.2 y 3.3.3.3 se realiza definiendo la programación en MATLAB R2012a. Esta se presenta en el Anexo 1-A: “optimización diseño-control de procesos: columna de destilación

binaria”, a través de la función definida como “OptIntg_Dest.m”. Este programa maneja varios programas internos o “scripts” basados en los problemas primario y maestro MINLP-DAE. Estos programas son:

Programa principal: Programa para simulación y optimización de columna de destilación binaria. Presenta 5 casos:

1. Evaluacion de la columna.

2. Optimizacion estado estacionario: Emplea la “búsqueda tabú” para la resolución de variables enteras y el método determinístico, representado por el método del gradiente. En MATLAB R2012a se emplea la función “fmincon”.

3. Optimizacion solo control: Se emplea el método determinístico para el problema de parametrización del control. Se usa “fmincon” de MATLAB R2012a.

4. Optimizacion integrada determinística: Se resuelve el diseño integrado por medio de la búsqueda tabú para las variables enteras y el método determinístico, representado por el método del gradiente (“fmincon”).

5. Optimizacion integrada estocástica: Se resuelve el diseño Integrado por medio de la búsqueda tabú para las variables binarias y el método estocástico, representado por algoritmos genéticos (AG) en MATLAB R2012a.

Programas internos:

1. Costo económico: presenta la función costo total, costo de utilidad y costo capital para la destilación binaria.

2. **Columna de destilación en estado estacionario:** Simula la columna de destilación en estado de operación estacionaria.
3. **Columna de destilación en estado dinámico:** Resuelve el problema de diseño de la columna de destilación en estado dinámico.
4. **Sistema de ecuaciones – derivadas:** Evalúa los valores de la derivada de las variables dinámica de acuerdo a los balances de masa y energía dinámicos.
5. **Busqueda tabú - variables enteras:** Programa que encuentra la solución a las variables enteras del problema de diseño integrado de la columna de destilación.
6. **Optimizador determinístico – variables continuas estado estacionario:** programa que encuentra la solución a las variables contínuas del problema de diseño integrado de la columna de destilación en estado de operación estacionaria.
7. **Restricciones no lineales estado estacionario:** Presenta las restricciones del problema en estado estacionario.
8. **Optimizador determinístico – solo control:** Programa que resuelve el problema de parametrización del control para distintos valores de ypsilon.
9. **Función costo control:** Presenta la función μ .

10. **Restricciones no lineales - optimización del control:** Presenta las restricciones no lineales del problema para la optimización del control.
11. **Optimizador determinístico – integrado:** Resuelve el problema de diseño integrado de la columna de destilación binaria usando los puntos de partida encontrados y la función “fmincon”.
12. **Optimizador estocástico – integrado:** Resuelve el problema de diseño integrado de la columna de destilación binaria usando los puntos de partida encontrados y la función “ga”.
13. **Restricciones no lineales – integrado:** presenta las restricciones no lineales del problema para el diseño integrado del sistema.
14. **Reporte de simulación estacionaria y dinámica de la columna:** Presenta las instrucciones para el reporte que debe arrojar el programa.

3.3.4 Validar los resultados encontrados comparando la respuesta dinámica del sistema obtenida mediante diseño integrado y diseño tradicional

Los resultados que se obtendrán en el trabajo serán cotejados para su validación con el diseño del mismo proceso, pero efectuado de forma secuencial o tradicional.

CAPÍTULO IV

DISCUSIÓN DE RESULTADOS

4.1 MÉTODO DEL GRADIENTE

Para lograr obtener las condiciones iniciales del problema matemático del diseño integrado se optimiza en estado estacionario primeramente las variables enteras, mediante la búsqueda tabú.

Se plantea la programación de “BUSQUEDA TABÚ – VARIABLES” dentro del programa principal, como se expone en la sección II. Se procede a la configuración inicial de la búsqueda definiendo los parámetros del modelo, que son: paso final de la búsqueda tabú (p.Step), valor inicial de las variables (p.y0), número máximo de iteraciones (p.Iter), tamaño de la lista tabú (memoria) (p.N) y valores iniciales variables continuas (p.x0) (ver Capítulo II). Cuando se especifica el caso de optimización (p.case) se establece el valor de cero (0) o uno (1), según sea optimización en estado estacionario (SS) o transitorio respectivamente, (ver Anexo 1-A). Al correr el programa se da como terminado cuando se alcanza el tamaño de paso máximo, el programa reporta el valor óptimo de la función objetivo en el estado estacionario y los valores óptimos para las variables binarias.

Luego se optimizan las variables continuas de diseño de la columna de destilación, es decir, aquellas que influyen en el costo capital. Y segundo, se optimizan también en el estado estacionario, las variables de control que se relacionan con los parámetros de los controladores.

Aplicando las dos primeras instrucciones del diagrama de bloques del Capítulo III y el algoritmo presentado en la figura 3.3.4.2, se realiza una optimización de la función costo sujeta al planteamiento matemático en estado estacionario mediante el programa principal en MATLAB R2012a (Anexo 1-A), usando el método del gradiente como método de optimización seleccionado, a través de la función “fmincon” de MATLAB R2012a (ver Capítulo II). La programación se define

como optimizador determinístico. Variables continuas estado estacionario. Se incluyen dentro de este programa las funciones “*fmincon*” para la optimización no lineal y “*lsqnonlin*” para resolver el problema algebraico no lineal, el resultado da el punto de partida y condiciones iniciales de las variables a ser utilizadas en el sistema de ecuaciones algebraico diferenciales, estos resultados se pueden apreciar en la Tabla 4.1.1. Se puede decir que los valores que se muestra en dicha tabla son el resultado de la búsqueda óptima en estado estacionario y de esta manera lograr determinar el intervalo de estudio para el error cuadrático, en donde la controlabilidad tiene un impacto económico apreciable sobre el diseño, se emplea la estructura de control al diseño del proceso reactivo encontrado mediante la minimización de la función costo sujeto al planteamiento en estado estacionario.

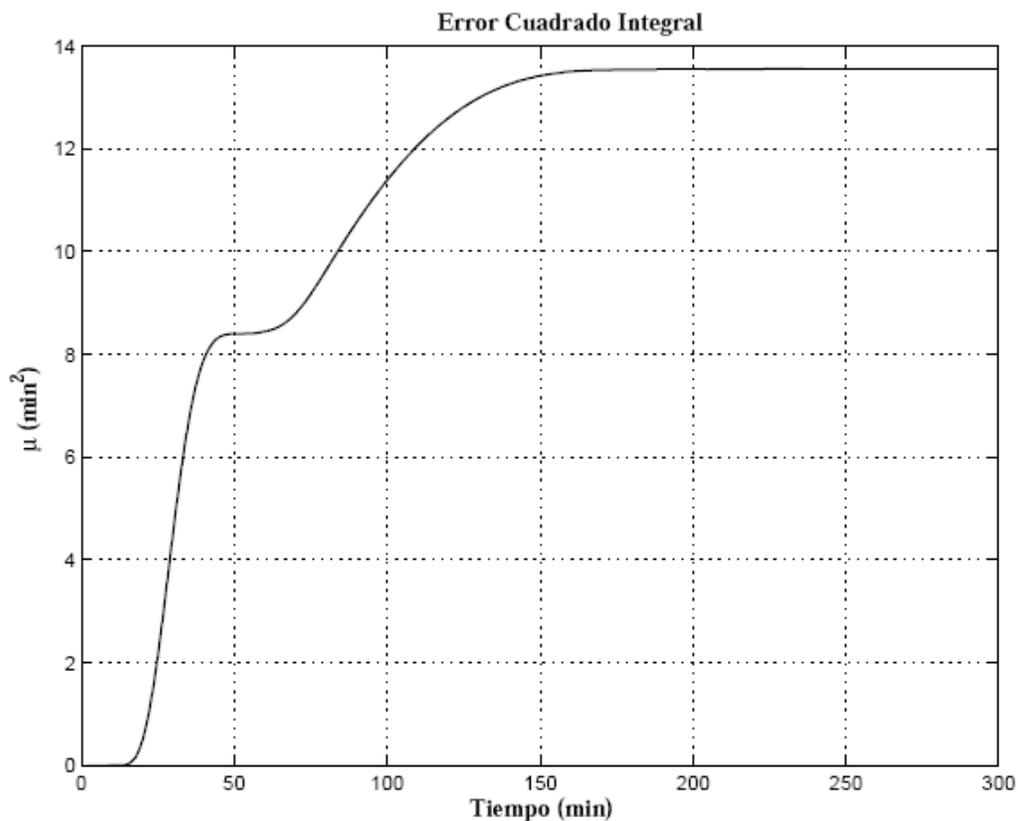
Se diseñan controladores tipo PI (proporcional-integral), utilizando como técnica el error cuadrático integral ISE, minimizando μ para encontrar los puntos de partida de los parámetros de cada controlador, ganancia proporcional y término integral (K y τ_i , respectivamente).

Tabla 4.1.1 Resultados optimización de variables de diseño en estado estacionario. Puntos de partida para el problema maestro.

Variables de Diseño			
Parámetro (unidades)	Nomenclatura	Unidades	Valor obtenido
Numero total de platos	Nt	N/A	16
Plato de alimentacion	Tf	N/A	8
Diametro de la columna	Dc	m	0,7535
Flujo de destilado	D	kmol/min	0,80971
Flujo de Vapor	Vss	kmol/min	1,2576
Variables de Control			
<i>Controlador flujo de vapor</i>			
Ganancia	k_v	kmol/min	-0,00000298
Constante de tiempo	τ_v	min	0,00060221
<i>Controlador flujo destilado</i>			
Ganancia	k_R	kmol/min	-0,15737
Constante de tiempo	τ_R	min	14,8966
Costos del Sistema			
Costo Capital	Cc	\$	24.426,6419
Costo Utilidad	Cu	\$	9.754,1488
Costo Total	Cost	\$	34.180,7906
Simulacion en Estado Estacionario			
Flujo de Destilado	D	kmol/min	0,44792
Flujo de Residuo	Rss	kmol/min	0,55208
Fraccion de A destilado	x_D	N/A	0,98
Fraccion de A Residuo	x_B	N/A	0,02
Simulacion en Estado Transitorio			
Flujo de Destilado	D	kmol/min	0,55193
Flujo de Residuo	Rss	kmol/min	0,44807
Fraccion de A destilado (liquido)	x_D	N/A	0,98005
Fraccion de A Residuo (liquido)	x_B	N/A	0,019967
Término-integral Controlador residuo	I_R	min ²	1,7872
Término-integral Controlador-reflujo dest.	I_D	min ²	-1,4742
ISE	\mathcal{E}	min ²	13,555

La programación en MATLAB R2012a es la correspondiente al “optimizador determinístico – solo control” (ver anexo A-1). En este se emplea la función *fmincon*.

El comportamiento dinámico del error cuadrático integral (ISE) se muestra en la Gráfica 4.1.1, donde se observa que la variación de la controlabilidad a lo largo del periodo de estudio ocurre en el rango que va de [0,0001- 13,555], los valores de la controlabilidad tiene un impacto apreciable sobre los costos al momento de realizar el diseño integrado.



Gráfica 4.1.1 Respuesta dinámica del error cuadrático integral (ISE).

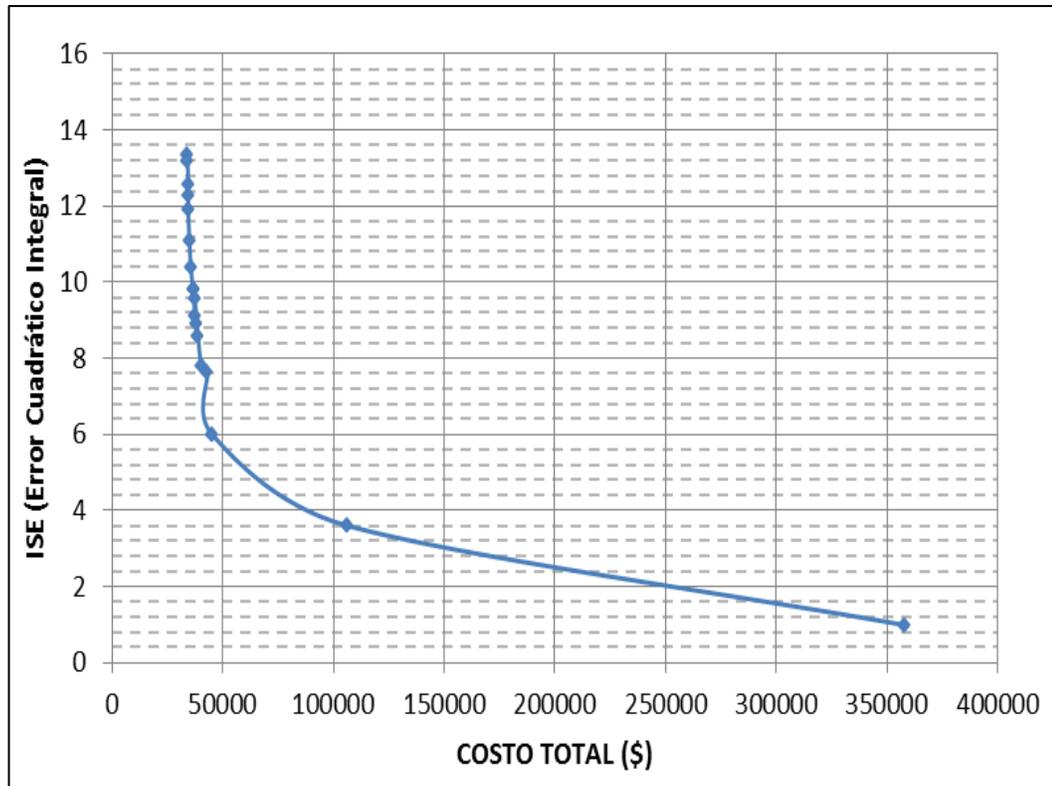
La Tabla 4.1.2 muestran los resultados obtenidos de la optimización de las variables de control en estado estacionario que se emplearán como condiciones iniciales para el problema de diseño integrado.

Tabla 4.1.2 Reporte de resultados de la programación en MATLAB R2012a para el cálculo de las variables de control en estado estacionario

Variables de Control			
<i>Controlador de flujo de vapor</i>	<i>Nomenclatura</i>	<i>Unidades</i>	<i>Valor</i>
Ganancia	k_v	kmol/min	-0,0001088
Constante de tiempo	τ_v	min	0,0016964
<i>Controlador de flujo de destilado</i>			
Ganancia	k_R	kmol/min	-0,0000048
Constante de tiempo	τ_R	min	0,004565

Se evalúan los distintos valores de ε (ypsilon) de controlabilidad dentro del rango ya encontrado y se obtiene la Gráfica 4.1.2 que relaciona el costo y la controlabilidad del sistema.

La curva indica que el costo total se mantiene sin muchas variaciones desde el valor superior evaluado $\varepsilon=13,555$, pero comienza a elevarse considerablemente cuando $\varepsilon=5,9808$, siendo el valor óptimo $\varepsilon=5,9473$. De esta forma se obtiene que el valor que proporciona un equilibrio tanto en los objetivos económicos como en los objetivos de control es $\varepsilon=5,9473$. Es obvio que a menor error cuadrático integral el costo total del diseño se incrementa, porque ejercer un mejor control sobre el sistema involucra aumento en los costos de utilidad que lógicamente repercuten en el costo total.



Gráfica 4.1.2 Relación entre el costo y la controlabilidad del sistema.

En este punto, el problema multiobjetivo se convierte en un problema de programación entero mixto no lineal con ecuaciones algebraico diferenciales, en el que se convierte uno de los objetivos en restricción ($\mu \leq \varepsilon$), es decir, el error cuadrático integral pasa a formar parte de las restricciones no lineales del problema.

De acuerdo con el diagrama de bloques de la Figura 3.3.4.1, al usar los puntos de partida de las Tablas 4.1.1 y 4.1.2 en el planteamiento matemático original del diseño integrado del sistema, se logra realizar la optimización no lineal en estado

transitorio usando el método de optimización determinístico, mediante la programación en MATLAB R2012a: “optimizador determinístico – integrado”, que resuelve el problema de diseño integrado de la columna de destilación binaria usando la función *fmincon*.

Se emplean las ecuaciones y expresiones correspondientes al sistema original MINLP-DAE. Se evalúa la función objetivo, y verifican criterios de parada definidos en las opciones, evaluando reiterativamente la función objetivo hasta su cumplimiento.

Se efectúan las corridas necesarias del programa y se seleccionan los datos de la corrida que arrojen el valor mínimo de la función “costo”. Véase Tabla 4.1.3 que presenta el reporte generado por el programa (Anexo A-1) para el valor de $\varepsilon = 5,9473$. **Estos datos representan el resultado del diseño integrado del sistema objeto de estudio.**

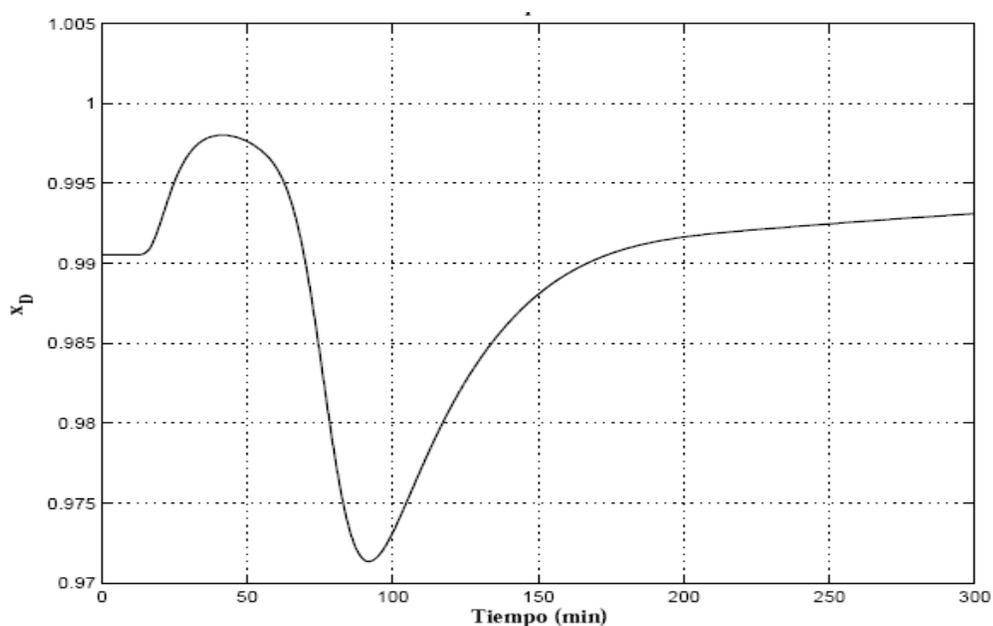
Tabla 4.1.3 Reporte de resultados del diseño integrado de una columna de destilación binaria. Función “*fmincon*”

Variables de Diseño			
Parámetro (unidades)	Nomenclatura	Unidades	Valor obtenido
Número total de platos	Nt	(N/A)	31
Plato de alimentacion	Tf	(N/A)	10
Diámetro de la columna	Dc	(m)	0,7142
Flujo de destilado	D	(kmol/min)	0,68401
Flujo de vapor	Vss	(kmol/min)	1,1299
Variables de Control			
<i>Controlador de flujo de vapor</i>			
Ganancia	k_v	(kmol/min)	-0,0002163
Constante de tiempo	τ_v	min	0,0031379
<i>Controlador de flujo de destilado</i>			

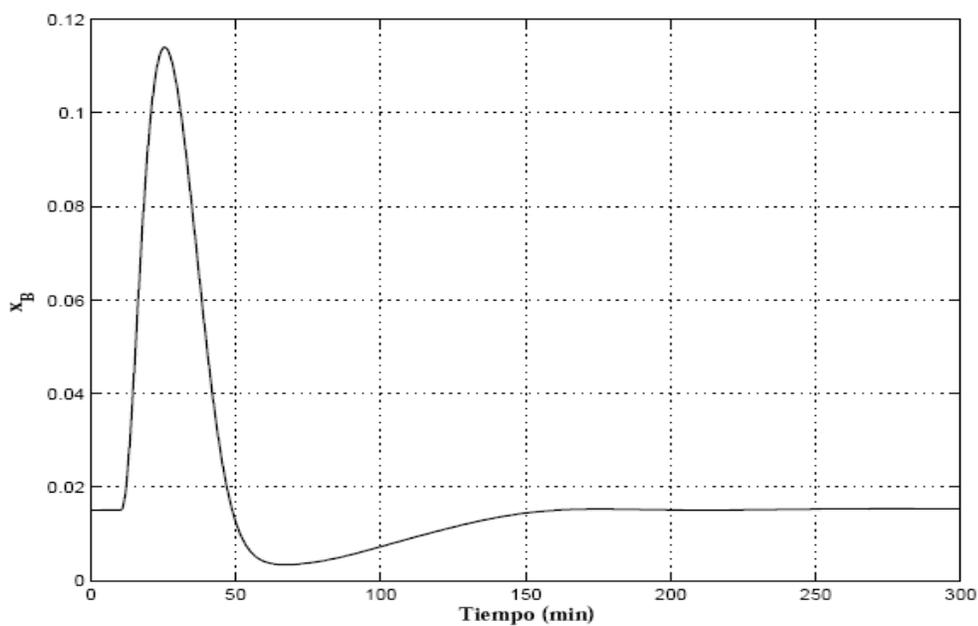
Ganancia	k_R	(kmol/min)	-0,19109
Constante de tiempo	τ_R	min	14,5211
Costos del Sistema			
Costo capital	C_c	(\$)	36.731,7645
Costo utilidad	C_u	(\$)	8.763,3532
Costo total	$Cost$	(\$)	45.495,1177
Simulacion en Estado Estacionario			
Flujo de destilado	D	(kmol/min)	0,44122
Flujo de residuo	R_{ss}	(kmol/min)	0,55878
Fraccion de A destilado	x_D	(N/A)	0,99494
Fraccion de A residuo	x_B	(N/A)	0,01972
Simulacion en Estado Transitorio			
Flujo de destilado	D	(kmol/min)	0,44122
Flujo de residuo	R_{ss}	(kmol/min)	0,55878
Fraccion de A destilado (liquido)	x_D	(N/A)	0,99494
Fraccion de A residuo (liquido)	x_B	(N/A)	0,01972
Término integral - Controlador residuo	I_R	(min ²)	1,376
Término integral - Controlador reflujo destilado	I_D	(min ²)	-0,39569
ISE	\mathcal{E}	(min ²)	5,9473

Dentro del proceso de destilación binaria, es importante mantener las composiciones de destilado y residuo dentro de un rango óptimo, ya que al garantizar el mismo, se obtiene de esta operación unitaria el objetivo deseado, el cual es la separación de los dos componentes de la mezcla. En el sistema que se estudia, se establece desde el inicio que la fracción molar de destilado no debe ser menor a 0,98 y la del residuo menor o igual a 0,02.

El comportamiento dinámico de las variables controladas se puede observar en las Gráficas 4.1.3 y 4.1.4



Gráfica 4.1.3 Respuesta dinámica de la composición del destilado. Diseño integrado mediante método del gradiente



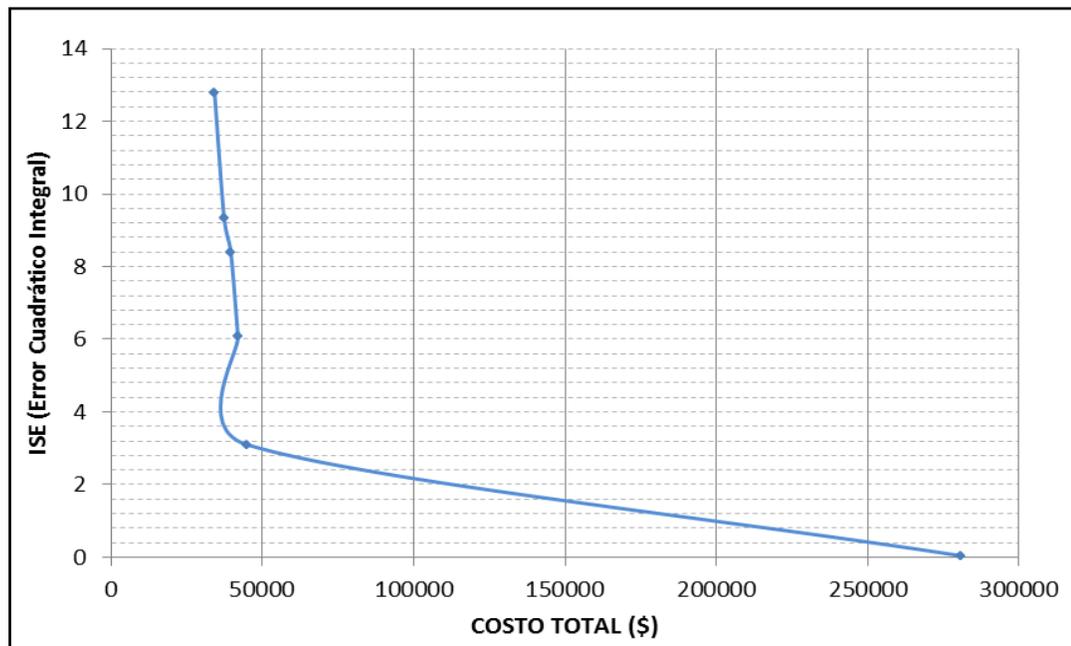
Gráfica 4.1.4. Respuesta dinámica de la composición del residuo o fondo. Diseño integrado mediante método del gradiente

La gráfica 4.1.3 muestra a lo largo del período de simulación de aproximadamente cinco (5) horas, como se va estabilizando la composición del destilado cuando el sistema es sometido a una perturbación tanto del flujo como de la composición de la alimentación. El valor máximo de desviación de la restricción de la pureza de destilado se observa que alcanza un sobreimpulso a un valor de 0,97. Sin embargo el control basado en la respuesta total del sistema indica que luego de 1,8 horas la composición se eleva a 0,98 y de allí en adelante tiende a estabilizarse a una fracción molar mayor a 0,99. Similarmente ocurre en la curva de fracción molar de destilado mostrada en la gráfica 4.1.4. El período de asentamiento se alcanza a aproximadamente 2,5 horas de operación, que es cuando la composición del residuo se mantiene a menos de 0,02.

Todo esto indica que efectivamente el diseño integrado para la columna de destilación planteada, efectúa una separación idónea de los componentes y simultáneamente ejerce un control adecuado sobre la composición de los productos manteniendo la estabilidad y controlabilidad del sistema.

4.2 ALGORITMOS GENÉTICOS

A continuación se evalúan los distintos valores de ε (ypsilon) de controlabilidad dentro del rango ya encontrado y se obtiene la siguiente gráfica que relaciona el costo y la controlabilidad del sistema.



Gráfica 4.2.1 Relación entre el costo y la controlabilidad del sistema

La curva indica que el costo total se mantiene sin muchas variaciones desde el valor superior evaluado para ε , pero comienza a elevarse considerablemente cuando $\varepsilon=3,0996$, siendo el valor óptimo $\varepsilon=3,0624$, que proporciona un equilibrio tanto en los objetivos económicos como en los objetivos de control. A menor error cuadrático integral el costo total del diseño se incrementa, porque ejercer un mejor control sobre el sistema involucra aumento en los costos de utilidad que repercuten en el costo total. En este punto, el problema multiobjetivo se convierte en un problema de programación entero mixto no lineal con

ecuaciones algebraico diferenciales, en el que se convierte uno de los objetivos en restricción ($\mu \leq \varepsilon$), es decir, el error cuadrático integral pasa a formar parte de las restricciones no lineales del problema.

De acuerdo con el diagrama de bloques de la Figura 3.3.4.1, al usar los puntos de partida de las Tablas 4.1.1 y 4.1.2 en el planteamiento matemático original del diseño integrado del sistema, se realiza la optimización no lineal en estado transitorio usando el método de optimización estocástico, mediante la programación en MATLAB R2012a: “**Optimización Integrada Estocástica**”, que resuelve el problema de diseño integrado de la columna de destilación binaria usando la función “ga”.

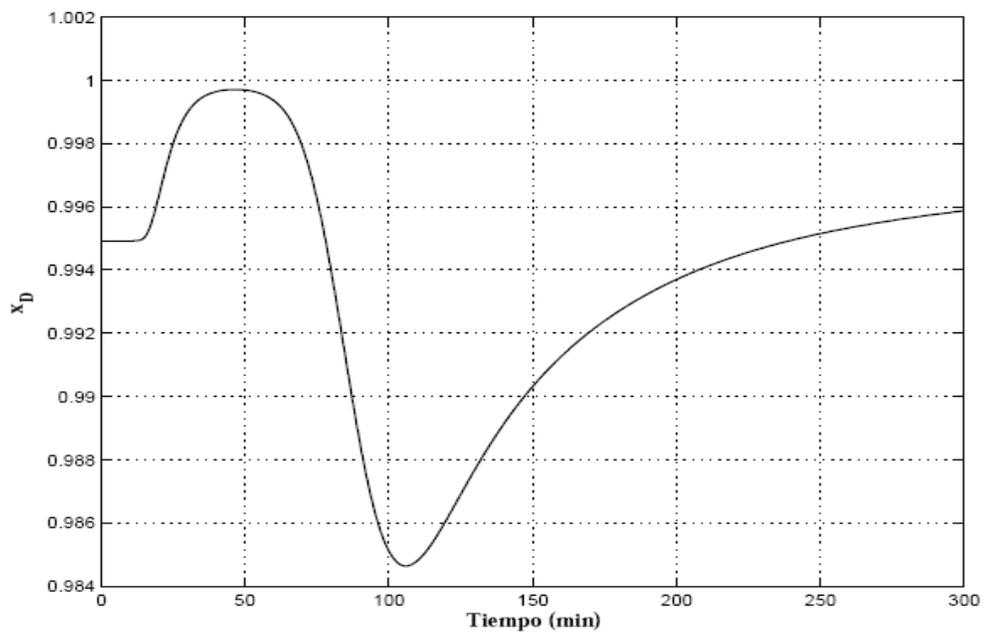
Se emplean las ecuaciones y expresiones correspondientes al sistema original MINLP-DAE. Se evalúa la función objetivo, y verifican criterios de parada definidos en las opciones, evaluando reiterativamente la función objetivo hasta su cumplimiento.

Se efectúan las corridas necesarias del programa y se seleccionan los datos de la corrida que arrojen el valor mínimo de la función “costo”. Véase Tabla 4.2.1 que presenta el reporte generado por el programa (Anexo A-1) para el valor de $\varepsilon = 3,0624$. Estos datos representan el resultado del diseño integrado del sistema objeto de estudio.

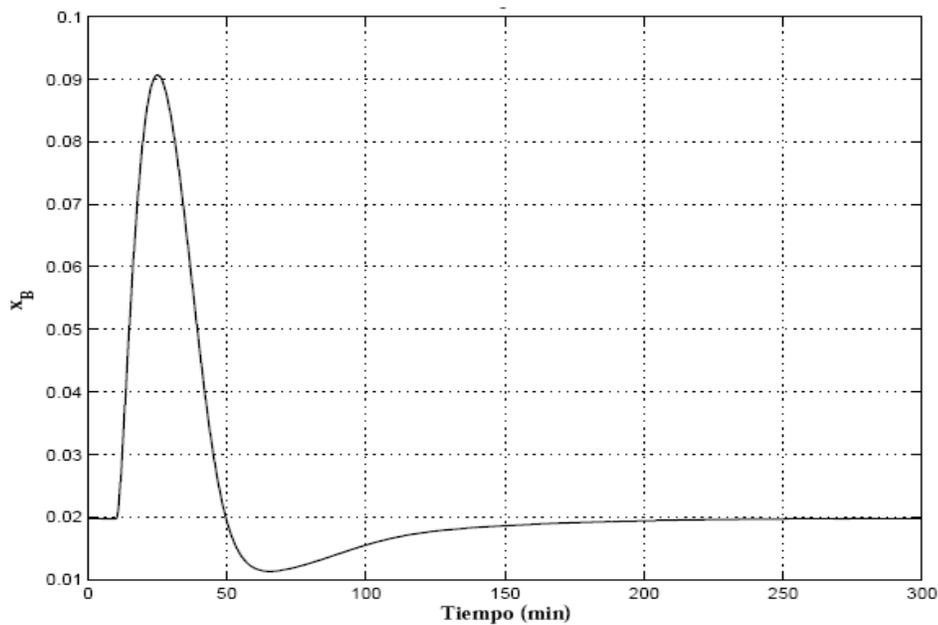
El comportamiento dinámico de las variables controladas se puede observar en las Gráficas 4.2.2 y 4.2.3.

Tabla 4.2.1. Reporte de resultados del diseño integrado de una columna de destilación binaria. Algoritmos genéticos.

VARIABLES DE DISEÑO			
Parámetro (unidades)	Nomenclatura	Unidades	Valor obtenido
Número total de platos	N_t	(N/A)	28
Plato de alimentación	T_f	(N/A)	7
Diametro de la columna	D_c	(m)	0,74393
Flujo de destilado	D	(kmol/min)	0,78467
Flujo de vapor	V_{ss}	(kmol/min)	1,2259
VARIABLES DE CONTROL			
<i>Controlador de flujo de vapor</i>			
Ganancia	k_v	(kmol/min)	-0,00056615
Constante de tiempo	τ_v	min	0,0064452
<i>Controlador de flujo de destilado</i>			
Ganancia	k_R	(kmol/min)	-0,1677
Constante de tiempo	τ_R	min	14,9725
COSTOS DEL SISTEMA			
Costo capital	C_c	(\$)	35.395,469
Costo utilidad	C_u	(\$)	9.507,9515
Costo total	$Cost$	(\$)	44.903,4205
SIMULACION EN ESTADO ESTACIONARIO			
Flujo de destilado	D	(kmol/min)	0,44122
Flujo de residuo	R_{ss}	(kmol/min)	0,55878
Fracción de A destilado	x_D	(N/A)	0,99494
Fracción de A residuo	x_B	(N/A)	0,01972
SIMULACION EN ESTADO TRANSITORIO			
Flujo de destilado	D	(kmol/min)	0,54258
Flujo de residuo	R_{ss}	(kmol/min)	0,45741
Fracción de A destilado (liquido)	x_D	(N/A)	0,99587
Fracción de A residuo (liquido)	x_B	(N/A)	0,01979
Término integral - controlador residuo	I_R		1,0983
Término integral - controlador reflujo destilado	I_D		-0,42275
ISE	\mathcal{E}	(min ²)	3,0624



Gráfica 4.2.2 Respuesta dinámica de la composición del destilado. Diseño integrado mediante algoritmos genéticos



Gráfica 4.2.3 Respuesta dinámica de la composición del residuo o fondo. Diseño integrado mediante algoritmos genéticos

La Gráfica 4.2.2 muestra a lo largo del período de simulación de aproximadamente 5 horas, como se va estabilizando la composición del destilado cuando el sistema es sometido a una perturbación tanto del flujo como de la composición de la alimentación. Se observa que el valor mínimo de la pureza de destilado alcanza un sobreimpulso al valor de 0,985, manteniéndose siempre por encima del valor de restricción. Similarmente ocurre en la curva de fracción molar de destilado mostrada en la gráfica 4.2.3. La composición del residuo se mantiene en 0,02 cuando para un tiempo de asentamiento de una (1) hora de operación.

El diseño integrado encontrado para la columna de destilación planteada efectúa una separación idónea de los componentes y simultáneamente ejerce un control adecuado sobre la composición de los productos manteniendo la estabilidad y controlabilidad del sistema.

Resulta propio en este punto del estudio resaltar las diferencias que surgen de efectuar el diseño integrado de la columna de destilación binaria con los dos métodos estudiados. La Tabla 4.2.2 muestra los resultados comparativos.

La primera diferencia en el empleo de ambos métodos se observa en los valores de las variables de diseño. Cuando se emplea “*fmincon*” como función de optimización el número de platos totales es de 31 y el diámetro es 0,7142 m, mientras que al usar “*ga*” se obtiene un diseño de una columna con menor número de bandejas (28), pero de diámetro mayor (0,74393 m). Diseñando con AG, el costo total disminuye en unos \$591,7, atribuido a un menor valor en la inversión inicial, por presentar número de bandejas menor a pesar de una diferencia ligeramente superior en el diámetro de la columna.

Los flujos de vapor y destilado tienden a ser ligeramente mayores cuando el diseño integrado se hace mediante algoritmos genéticos, mientras que las diferencias para la composición de destilado y residuo son prácticamente despreciables.

Cuando se analiza el aspecto de controlabilidad del sistema o en otras palabras, el comportamiento del sistema frente a las perturbaciones asociadas a cambios en el flujo de alimentación y su composición, se observa que el sobreimpulso del comportamiento dinámico de la composiciones de destilado y residuo, cuando se hace uso de algoritmos genéticos es menor, observándose en ambos casos que el sistema nunca se desvía de los parámetros superiores e inferiores de control. De esta manera, el diseño obtenido provee de un sistema de separación de productos de alta calidad. Ver Gráficas 4.2.2 a 4.2.3.

Tabla 4.2.2. Comparación de resultados del diseño integrado de una columna de destilación binaria obtenido mediante métodos de optimización determinístico y estocástico

Variables de diseño				
Parámetro (unidades)	Nomenclatura	Unidades	<i>fmincon</i>	<i>ga</i>
Numero total de platos	Nt	(N/A)	31	28
Plato de alimentacion	Tf	(N/A)	10	7
Diametro de la columna	Dc	(m)	0,7142	0,74393
Flujo de destilado	D	(kmol/min)	0,68401	0,78467
Flujo de Vapor	V _{ss}	(kmol/min)	1,1299	1,2259
Variables de control				
<i>Controlador flujo de vapor</i>				
Ganancia	k_v	(kmol/min)	-0,0002163	-0,00056615
Constante de tiempo	τ_v	min	0,0031379	0,0064452
<i>Controlador flujo de destilado</i>				
Ganancia	k_R	(kmol/min)	-0,19109	-0,1677
Constante de tiempo	τ_R	min	14,5211	14,9725
Costos del sistema				
Costo Capital	Cc	(\$)	36.731,7645	35.395,469
Costo Utilidad	Cu	(\$)	8.763,3532	9.507,9515
Costo Total	Cost	(\$)	45.495,1177	44.903,4205

Simulacion en estado estacionario				
Flujo de Destilado	D	(kmol/min)	0,44122	0,44122
Flujo de Residuo	R _{ss}	(kmol/min)	0,55878	0,55878
Fraccion de A destilado	x_D	(N/A)	0,99494	0,99494
Fraccion de A Residuo	x_B	(N/A)	0,01972	0,01972
Simulacion en estado transitorio				
Flujo de Destilado	D	(kmol/min)	0,44122	0,54258
Flujo de Residuo	R _{ss}	(kmol/min)	0,55878	0,45741
Fraccion de A destilado (liquido)	x_D	(N/A)	0,99494	0,99587
Fraccion de A Residuo (liquido)	x_B	(N/A)	0,01972	0,01979
Termino-integral controlador residuo	I _R	(min ²)	1,376	1,0983
Termino integral controlador reflujo destilado	I _D	(min ²)	-0,39569	-0,42275
ISE	\mathcal{E}	(min ²)	5,9473	3,0624

4.3 DISEÑO INTEGRADO VS. DISEÑO TRADICIONAL

Se presenta a modo comparativo el diseño tradicional o secuencial del sistema y el diseño integrado, en la Tabla 4.3.1

Tabla 4.3.1. Diseño integrado vs. diseño tradicional para una columna de destilación binaria

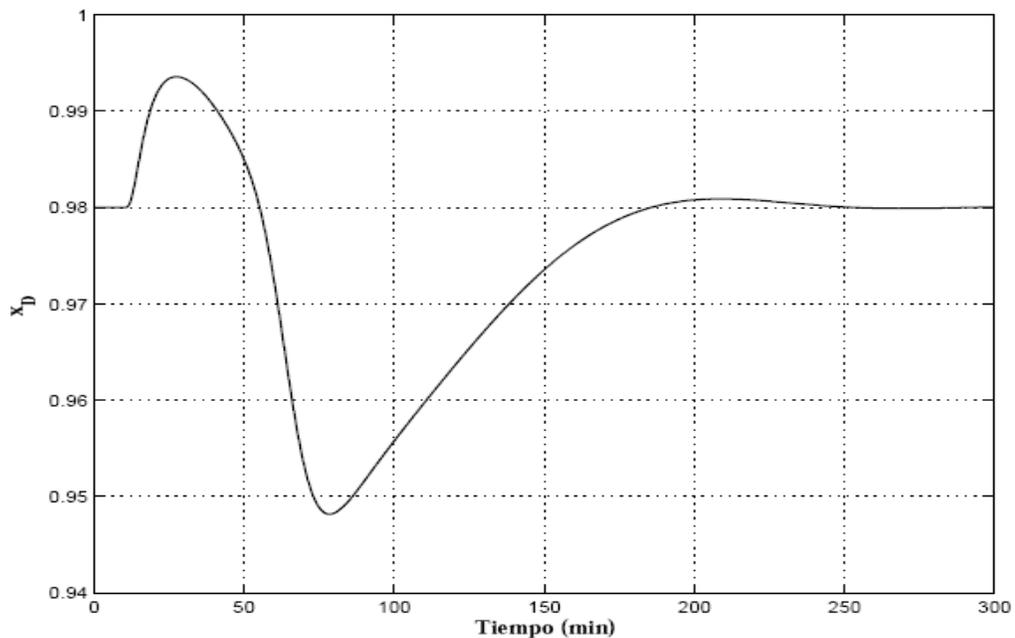
VARIABLES DE DISEÑO				
Parámetro (unidades)	Nomenclatura	Unidades	Tradicional	Integrado
Número total de platos	Nt	(N/A)	16	31
Plato de alimentación	Tf	(N/A)	8	10
Diametro de la columna	Dc	(m)	0,7535	0,7142
Flujo de destilado	D	(kmol/min)	0,80971	0,68401
Flujo de Vapor	V _{SS}	(kmol/min)	1,2576	1,1299
VARIABLES DE CONTROL				
Controlador de flujo de vapor				
Ganancia	k_v	(kmol/min)	-0,0000298	-0,0002163
Constante de tiempo	τ_v	min	0,00060221	0,0031379
Controlador de flujo de destilado				
Ganancia	k_R	(kmol/min)	-0,15737	-0,19109
Constante de tiempo	τ_R	min	14,8966	14,5211
COSTOS DEL SISTEMA				
Costo Capital	Cc	(\$)	24.426,6419	36.731,7645
Costo Utilidad	Cu	(\$)	9.754,1488	8.763,3532
Costo Total	Cost	(\$)	34.180,7906	45.495,1177
SIMULACION EN ESTADO ESTACIONARIO				
Flujo de Destilado	D	(kmol/min)	0,44792	0,44122
Flujo de Residuo	R _{SS}	(kmol/min)	0,55208	0,55878
Fraccion de A destilado	x_D	(N/A)	0,98	0,99494
Fraccion de A Residuo	x_B	(N/A)	0,02	0,01972
SIMULACION EN ESTADO TRANSITORIO				
Flujo de Destilado	D	(kmol/min)	0,55193	0,44122
Flujo de Residuo	R _{SS}	(kmol/min)	0,44807	0,55878
Fraccion de A destilado (liquido)	x_D	(N/A)	0,98005	0,99494
Fraccion de A Residuo (liquido)	x_B	(N/A)	0,019967	0,01972
Termino integral controlador residuo	I _R	(min ²)	1,7872	1,376
Termino integral controlador reflujo destilado	I _D	(min ²)	-1,4742	-0,39569
ISE	\mathcal{E}	(min ²)		13,555

Sobre los datos presentados en la tabla 4.3.1 es importante señalar las siguientes premisas:

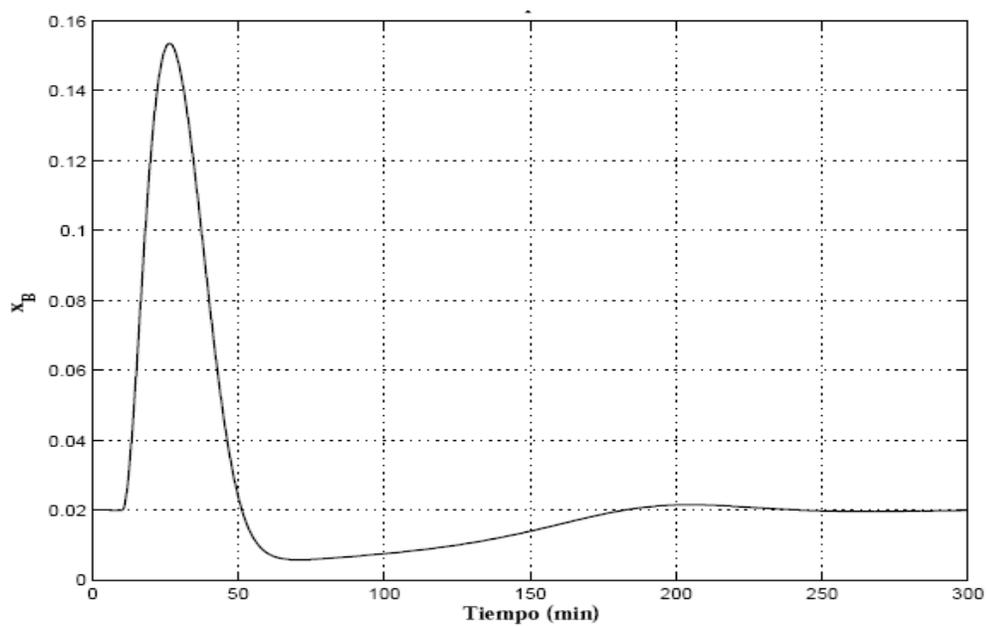
- El diseño tradicional se refiere al enfoque de procesos secuencial presentado en el Capítulo II.
- El diseño tradicional es efectuado a través del método del gradiente (función “*fmincon*”), correspondiente al diseño de las variables de diseño en estacionario de la Tabla 4.1.1 y al posterior diseño del control, de forma separada.
- Se comparan los datos de diseño para ambos casos usando el mismo método de optimización (gradiente).

Se puntualizan las observaciones más relevantes:

- Usando el diseño tradicional se obtiene una columna con 15 bandejas por debajo del número obtenido con el diseño integrado, lo que se traduce en una columna más corta. Esto repercute en el costo de inversión inicial, que implica un ahorro de \$ 12.305,12 como costo capital y de \$ 11.314,3271 en el costo total.
- Los flujos de destilado y de vapor son menores en el caso de diseño integrado en el orden de 0,12 kmol/min.
- La estabilidad del sistema frente a posibles perturbaciones es más pobre usando el diseño tradicional (Gráficas 4.3.1 y 4.3.2). El sobreimpulso en la curva del comportamiento dinámico de las variables controladas (fracciones molares de los productos) es de mayor magnitud, por lo que las fracciones molares salen del rango de control. Para el caso de la composición del destilado desciende a menos de 0,95 y para el residuo se eleva por encima de 0,14. Esto implica productos fuera de las especificaciones de calidad.



Gráfica 4.3.1. Respuesta dinámica de la composición del destilado. Diseño tradicional



Gráfica 4.3.2. Respuesta Dinámica de la composición del residuo o fondo. Diseño Tradicional.

Con el diseño tradicional se tardan más de cuatro (4) horas para alcanzar la estabilidad del sistema, mientras que con el diseño integrado el sistema tarda entre 1,8 y 2,5 horas para estabilizarse. Esto influye en la cantidad de productos defectuosos.

Se establece que el diseño integrado presenta un sistema con mejor estabilidad frente a las perturbaciones.

Estas observaciones se resumen en la tabla 4.3.2.

Tabla 4.3.2. Resumen cualitativo. Diseño tradicional vs diseño integrado de una columna de destilación binaria.

	Ventajas	Desventajas
Diseño Integrado	Control óptimo de la pureza de los productos.	Costo capital más elevado.
	Tiempos de asentamiento más cortos, menor cantidad de productos defectuosos.	
Diseño Tradicional	Costo capital menor.	Control deficiente de la pureza de los productos.
		Tiempos de asentamiento más largos, mayor cantidad de productos defectuosos.

La Tabla 4.3.2, exhibe claramente las ventajas y desventajas del diseño integrado. Es posible darse cuenta a éste nivel, del porque a lo largo de los años se ha trabajado con el diseño tradicional sin muchas objeciones: inicialmente es más económico.

El problema radica en las operaciones de planta debido a que los costos de operación son elevados al intentar mantener la estabilidad del sistema y de desechar o recircular productos fuera de especificación.

De esta manera, el diseño integrado optimiza tanto los costos de inversión inicial, los costos de operación, la estabilidad del sistema en el tiempo y rangos de control, así como productos de excelente calidad en el momento oportuno y en sintonía con la preservación del ambiente, mientras más control se tenga sobre el proceso menos serán las emisiones ambientales.

CONCLUSIONES

- Cuando se emplea “*fmincon*” como función de optimización el número de platos totales es de 31 y el diámetro es 0,7142 m, mientras que al usar “*ga*” se obtiene un diseño de una columna con menor número de bandejas (28), pero de diámetro mayor (0,74393 m). Diseñando con algoritmos genéticos el costo total disminuye en unos \$591,7, atribuido a un menor valor en la inversión inicial, por presentar un número de bandejas menor.
- Los flujos de vapor y destilado tienden a ser ligeramente mayores cuando el diseño integrado se hace mediante algoritmos genéticos, mientras que las diferencias para la composición de destilado y residuo son prácticamente despreciables.
- El sobreimpulso en el comportamiento dinámico de las composiciones de destilado y residuo cuando se efectúa el diseño integrado mediante algoritmos genéticos es menor, observándose en ambos casos que el sistema nunca se desvía de los parámetros superiores e inferiores de control. De esta manera, el diseño obtenido provee un sistema de separación de productos de alta calidad.
- El uso de algoritmos genéticos en el diseño integrado del sistema presenta el costo total menor y a su vez mayor estabilidad frente a posibles perturbaciones, que el caso del diseño efectuado mediante el método del gradiente.

- Para la estrategia metodológica más factible económicamente el comportamiento dinámico del error cuadrático integral (ISE) presentó una variación de la controlabilidad a lo largo del periodo de estudio en el rango que va de [0,77406-13,555]. El valor óptimo de controlabilidad \mathcal{E} (ypsilón), que proporciona un equilibrio tanto en los objetivos económicos como en los objetivos de control, es $\mathcal{E}=3,0624$.
- La estabilidad del sistema frente a posibles perturbaciones es más pobre usando el enfoque secuencial o tradicional de diseño de procesos. El sobreimpulso en la curva del comportamiento dinámico de las variables controladas (fracciones molares de los productos) es de mayor magnitud, por lo que las fracciones molares salen del rango de control. Para el caso de la composición del destilado desciende a menos de 0,95 y para el residuo se eleva por encima de 0,14. Esto implica productos fuera de las especificaciones de calidad.
- Usando la metodología de diseño tradicional de procesos para el sistema, se tardan más de cuatro (4) horas para alcanzar la estabilidad del mismo, mientras que con el Diseño Integrado el sistema tarda entre 1,8 y 2,5 horas para estabilizarse. Se obtiene en el primer caso, mayor cantidad de productos defectuosos mientras se espera que el sistema alcance el estado estable.
- El diseño integrado para el sistema estudiado presenta mejor estabilidad frente a las perturbaciones.

- Usando el Diseño Tradicional de procesos para el sistema estudiado, se obtiene una columna con 15 bandejas por debajo del número obtenido con el Diseño Integrado, lo que se traduce en una columna más corta. Esto repercute en el costo de inversión inicial, que implica un ahorro de \$ 12.305,12 como costo capital y de \$ 11.314,3271 en el costo total.
- El diseño integrado optimiza tanto los costos de inversión inicial, los costos de operación, la estabilidad del sistema en el tiempo y rangos de control, así como productos de excelente calidad en el momento oportuno y en sintonía con la preservación del ambiente.
- Desde la perspectiva del lector, ingeniero o ejecutor del diseño a un sistema real, se deberá seleccionar la alternativa que más se ajuste a sus necesidades, resultando el presente trabajo una contribución al diseño integrado de columnas de destilación binaria.
- Para resolver el diseño integrado de una columna de destilación binaria, hay que involucrar el uso de la búsqueda tabú (TS), para obtener las variables binarias.

RECOMENDACIONES

- Explorar nuevas estrategias metodológicas que abarquen nuevos métodos de optimización.
- Aplicar la metodología desarrollada del diseño integrado de columnas de destilación binarias a un sistema real.
- Aplicar la metodología del diseño integrado, para la construcción a escala piloto de un proceso similar al caso de estudio, para validar los resultados con datos reales.
- Se recomienda utilizar otras herramientas computacionales más especializadas en el área de optimización, como el GAMS y combinarlas con paquetes para la resolución de ecuaciones diferenciales.

REFERENCIAS BIBLIOGRÁFICAS

- Balestrini A., M. (2004). Como se elabora el proyecto de investigación. Caracas : BL consultores asociados servicio editorial.
- BANSAL y colaboradores, (2000), Simultaneous design and control optimisation under uncertainty, *Computers and Chemical Engineering*, 24, 261-266, (2000).
- Gomis, M., (1998). Introducción a las Operaciones de Separación. Cálculo por etapas de equilibrio. Publicaciones de la Universidad de Alicante. Compobell, S.L. Murcia. España.
- Gonzalez E., y Cortés P. (2012). Un Algoritmo de Búsqueda Tabú para generar el calendario de exámenes de la nueva facultad de Ciencias de la Educación. Departamento de organización Industrial y Gestión de empresas. Escuela técnica Superior de Ingenieros. Universidad de Sevilla. Sevilla, España.
- Gutiérrez G, De Prada C (2003). Diseño Integrado y Síntesis de Procesos Químicos. Universidad de Valladolid. España. 1-6.
- Hergueta, D. (2008). Módulo procesos de fabricación y Sistemas Integrados. Control de Columnas de Destilación. Escuela de Negocios, EOI. España.
- Luyben ML, Floudas CA (1994) Analyzing the interaction of design and control. *Comp. Chem. Eng.* 18, Nro. 10, 933-969.
- Morari M (1983) Design of resilient processing plants III, A general framework for the assessment of dynamic resilience. *Ind. Chem. Eng. Sci.* 38: 1881-1891.

- Morari M. (1992). Effect of design on the controllability of chemical plants. In J. D. Perkins (Ed.). Preprints IFAC Workshop Interactions Between Process Design and Process Control. 3-16. Oxford: Pergamon Press.
- Morari M, Zafiriou E (1989) Robust Process Control, Prentice-Hall, Englewood Cliffs, NJ. 488.
- Nishida N, Ichikawa A, Tazoki E (1974) Synthesis of Optimal Process Systems with Uncertainty. Ind. Eng. Chem. Process. 13: 209-214.
- Nishida N, Ichikawa A (1975) Synthesis of Optimal Dynamic Process Systems by a Gradient Method. Ind. Eng. Chem. Process. 13: 236-242.
- Nishida N, Liu Y, Ichikawa A (1976) Studies in Chemical Process Design and Síntesis II. Optimal Synthesis of Dynamic Process Systems with Uncertainty. AIChE J. 22: 539-549.
- Osman I.H. y J.P Kelly. 1996. Meta-heuristics: an overview. En: Osman I.H. y J.P Kelly (Eds.), Meta-Heuristics: Theory and Applications, pp 1 – 22. Kluwer Academic Publishers. Boston.
- PÉREZ, J. y Basterrechea, J. (2005). Contribución a los métodos de optimización basados en procesos naturales y su aplicación a la medida de antenas en campo próximo. Universidad de Cantabria.
- Quintana F (2003) Síntesis de Procesos, (Optimización del diseño de Plantas Químicas). Universidad Politécnica de Madrid. España. 184.

- RIOJAS, A. (2005). Conceptos, algoritmo y aplicación al problema de las N – Reinas. Universidad Nacional Mayor de San Marcos Facultad Ciencias Matemáticas E.A.P. de Investigación Operativa. Lima, Perú.
- SAKIZLIS y colaboradores, (2004), “Recent advances in optimization-based simultaneous process and control design”, *Computers and Chemical Engineering*, 28, 2069-2086, (2004).
- Scenna NJ (1999) *Modelado, Simulación y Optimización de Procesos Químicos*. Buenos Aires, Argentina. 840 pp.
- Sheffield R (1992) Integrate process and control system design. *Ind. Chem. Eng. Prog.* 88: 30-35.
- Schweiger C, Floudas C (1997), Interaction of Design and Control. Optimization with Dynamic Models. En Hager WW, Pardalos PM (Eds) *Optimal Control Theory, Algorithms and Applications*. Kluwer Princeton, EEUU. 1-2
- Skogestad S (1991) Consistency of steady-state models using insight about extensive variables. *Ind. Eng. Chem. Res.* 30: 654-661.
- Skogestad S, Morari M (1987b) Design of resilient Processing Plants-IX. Effect of Model Uncertainty on Dynamic Resilience *Chem. Eng. Sci.* 42: 1765-1780.
- Skogestad S, Wolff E (1992), Controllability Measures for Disturbance Rejection. Preprints IFAC-workshop on interactions between design and control. London. 127-132. Edited by Perkins S, Perganom Press.

Smith C., Corripio, A., (1991), "Control Automático de Procesos", (Teoría y Práctica), Editorial LIMUSA S.A, Primera edición, Traductor: Manzanares, S. México.

TAKAMA N. y Umeda T. (1980) "Multi-level, Multi-objective Optimization in Process Engineering", Chem. Engng. Sci., 36, pp. 129-136. USA.

UMEDA, T. (1982). "Computer Aided Process Synthesis", Proceedings of Process Systems Engineering Symposium, p. 79, Kyoto, Japón.

VIZCAÍNO, G., (2006). "Automatización del proceso de diseño de un circuito electrónico con la herramienta MATLAB y el simulador de circuitos SPECTRE", Escuela Superior de Ingenieros Universidad de Sevilla. España.

Walsh S, Perkins JD (1996) Operability and control and process synthesis and design. *Adv. Chem. Eng.* 23: 301-402.

APENDICE A

En esta sección se presenta la productividad en la formación doctoral del MSc. Ing. Jesús Enrique Ceballos Jiménez en el Área de diseño integrado hasta la fecha de presentación de esta tesis doctoral, estas son:

Participación como ponente en Congresos:

- J.E. Ceballos J. / Diseño de Columnas de Destilación, Enfoque Integrado/ XXVII Congreso Interamericano y Colombiano de Ingeniería Química / CIIQ / 6 al 8 – 10 – 2014 / Cartagena de India / Colombia.
- J.E. Ceballos J. / Estudio Comparativo del Diseño de Columnas de Destilación Binarias, Enfoque Integrado Utilizando Metodos Deterministico y Algoritmo Genetico/ 2^{do} Congreso Venezolano de Ciencia Tecnología e Innovación en el Marco de la LOCTI y del PEII/ 7 al 10 – 11 – 2013/ Caracas / Venezuela.
- J.E. Ceballos J. / Contribution to the Desing of Integrate Distillation Columns / 40th International Conference of Slovak Society of Chemical Engineering / 27 al 31 – 05 – 2013 /Tatranske Matliare / Slovakia.

Revistas:

- J.E. Ceballos J. /Diseño de Columnas de Destilación para Sistemas Binarios. Enfoque Tradicional e Integrado / Volumen 39, N° 8 Revista Journal of Science and Tecnology of America (Interciencia) / Agosto 2014.
- J.E. Ceballos J. / Estudio Comparativo del Diseño de Columnas de Destilación Binarias. Enfoque Integrado (Deterministico y Algoritmo Genético) / Volumen 4, N° 1 Revista REDIP UNEXPO Barquisimeto Venezuela <http://redip.bqto.unexpo.edu.ve> / Octubre 2014
- J.E. Ceballos J. / Design Integrated Distillation Columns of Binary Systems/ Number: ISSN 1934 – 7375, Revista Journal of Chemistry and Chemical Engineering USA / Proceso de Publicación.

ANEXOS

ANEXO A-1. Programación Principal en MATLAB R2012a para el Diseño Integrado de una Columna de Destilación Binaria.

```

*****
*
%           OPTIMIZACION DISEÑO-CONTROL DE PROCESOS
%           >Columna de Destilación binaria<
%
%
%
%
%           (c) 2010
%*****
**
%% PROGRAMA PRINCIPAL.
function OptIntg_Dest()
%
%   Programa para simulación y optimización de columna de
destilación
%   binaria.
%
warning off
format('long','e')
clc
% ENTRADA DE DATOS
jobname = input(' Titulo del trabajo: ', 's');
disp(' ')
disp('   Casos disponibles')
disp('   1.- Evaluacion de la Columna ')
disp('       2.- Optimizacion Estado Estacionario - Tabu,
Deterministico')
disp('       3.- Optimizacion Solo Control -
Deterministico')
disp('       4.- Optimizacion Integrada - Tabu,
Deterministico')
disp('       5.- Optimizacion Integrada - Tabu, AG
')
disp(' ')
caso = input(' Caso a ejecutar : ');
ind = input(' Carpeta de salida: ','s');
disp(' ')
Vec_desig = input('Vector de Diseno (5): ');

```

```

Vec_contr = input('Vector de Control(4): ');
Perturb = input('% Perturbacion de z : ');
disp(' ')
if (size(Vec_desig)~=5)
    Vec_desig = [ 16 8 7.534950926333704e-001
8.097095265787664e-001 1.257626193245236e+000 ];
    disp('Advertencia: Tomando valores pre-determinados para')
    disp(' el Vector de Diseno')
end
if (size(Vec_contr)~=4)
    Vec_contr = [ -2.979309330736653e-05 -1.573687711428398e-01
6.022074153707791e-04 1.489660530393212e+01 ];
    disp('Advertencia: Tomando valores pre-determinados para')
    disp(' el Vector de Control')
end
if (size(Perturb)~=1)
    Perturb = 10;
    disp('Advertencia: Tomando valor pre-determinado para')
    disp(' el % Perturbacion z (10%)')
end
disp(' ')
t_fin = 300;
if (caso==1);
    t_fin0 = input('Tiempo final de Simulacion (min): ');
    if (numel(t_fin0)==1);t_fin=t_fin0;end
end

disp(' ')
Cont = ' ';
while and(Cont~='y',Cont~='n')
    Cont = input('Desea Continuar? (y/n): ', 's')
end

if (Cont=='y')
    clc
    try
        matlabpool close % Para calculo en
paralelo
    end
    matlabpool open % Para calculo en paralelo
    % PARAMETROS GENERALES DEL SISTEMA
    Param.eps = 1e-4; % Tolerancia numérica
    % PARAMETROS DEL PROBLEMA DE DESTILACION BINARIA
    Param.alpha = 2.50; % Relative volatility
    Param.h_w = 0.0254; % Height over the weir
    Param.Beta_pay = 4.0; % Payback period
    Param.Beta_tax = 0.40; % Tax factor
    Param.F = 1.00; % Flujo de
alimentacion

```

```

    Param.z          = 0.45;                % Composicion de la
alimentacion (compuesto interes)
    Param.xd_s       = 0.98;                % Especificacion para
la Composicion del destilado
    Param.xb_s       = 0.02;                % Especificacion para
la Composicion del residuo
    Param.a          = [Perturb/100,10,10]; % Parametros de
perturbacion fraccion molar
    Param.b          = [0.00,10,10];        % Parametros de
perturbacion Flujo de alimentacion
    Param.Time       = t_fin;               % Tiempo de
simulacion (min)
    Param.Dt         = t_fin/30000;         % Intervalo de
discretizacion del tiempo
    Param.proc       = 6000;                % tiempo maximo en la
ejecucion de un proceso (seconds)
    Param.monitor    = [ ];                 % Variables a
monitorear en la simulacion dinamica
    Param.epsilon    = 5.50;               % Valor de limite
inferior ISE
    % VARIABLES DE DISEÑO
    Design.T_R       = Vec_desig(1);        % Plato de reflujo
    Design.T_F       = Vec_desig(2);        % Plato de
alimentacion
    Design.Dc        = Vec_desig(3);        % Diametro de la
columna
    Design.Rss       = Vec_desig(4);        % Reflujo liquido
    Design.Vss       = Vec_desig(5);        % Reflujo Vapor
    % VARIABLES DE CONTROL
    Control.Kv       = Vec_contr(1);        % Ganancia
controlador del reflujo de vapor
    Control.Kr       = Vec_contr(2);        % Ganancia
controlador del reflujo de destilado
    Control.Tau_v    = Vec_contr(3);        % Constante de
tiempo controlador del reflujo de vapor
    Control.Tau_r    = Vec_contr(4);        % Constante de tiempo
controlador del reflujo de destilado

    tic                                     % Inicio del registro del
tiempo

switch caso
case 2
    p.Step = 15;
    p.y0   = Vec_desig(1:2);
    p.Iter = 1000;
    p.N    = 50;
    p.x0   = Vec_desig(3:5);
    p.case = 0;
    [y_min,x_min] = tabu(p,Param);
    Vec_desig     = [y_min, x_min];
    Vec_contr     = [0 0 1 1];

```

```

case 3
    x_min      = Opt_Con_OC(Design,Vec_contr, Param);
    Vec_contr  = x_min;
case {4,5}
    p.Step     = 4;
    p.y0       = Vec_desig(1:2);
    p.Iter     = 1000;
    p.N        = 50;
    p.x0       = [Vec_desig(3:5), Vec_contr];
    p.case     = 1;
p.ty         = caso - 4;
    [y_min,x_min] = tabu(p,Param);
    Vec_desig     = [y_min, x_min(1:3)];
    Vec_contr     = x_min(4:7);
    diary off
otherwise
    disp('Opcion no disponible')
end

% RESULTADOS DEL CASO EJECUTADO
Design.T_R      = Vec_desig(1);           % Plato de reflujo
Design.T_F      = Vec_desig(2);           % Plato de
alimentacion
Design.Dc       = Vec_desig(3);           % Diametro de la
columna
Design.Rss      = Vec_desig(4);           % Reflujo liquido
Design.Vss      = Vec_desig(5);           % Reflujo Vapor
Control.Kv      = Vec_contr(1);           % Ganancia controlador
del reflujo de vapor
Control.Kr      = Vec_contr(2);           % Ganancia
controlador del reflujo de destilado
Control.Tau_v   = Vec_contr(3);           % Constante de
tiempo controlador del reflujo de vapor
Control.Tau_r   = Vec_contr(4);           % Constante de tiempo
controlador del reflujo de destilado

report(Design, Control, Param, jobname,ind) % Reporte de
Resultados

matlabpool close % Cierre del entorno
paralelo
else
    clc
    disp(' ')
    disp(' Programa abortado por el usuario')
    disp(' ')
end % del if

end
%*****
**

```

```

%% COSTO ECONOMICO
function [Costo_t, Costo_u, Costo_c ]= Costo(k,y)
%
% Costo total, costo de utilidades y costo capital
% para la destilación binaria.
%
Dc = k(1);
Vss = k(3);
Nt = y(1);

Costo_u= 7756*Vss;

Costo_c= 3.075*(615+324*Dc.^2 + 486*(6 + 0.76*Nt).*Dc) + ...
        61.25*Nt.*(0.7+1.5*Dc.^2);

Costo_t= Costo_u+Costo_c;
end
%*****
**
%% COLUMNA DE DESTILACION EN ESTADO ESTACIONARIO
function Result = Dest_SS(Param,Design)
%
% Simula la columna de destilación en estado de operación
% estacionaria
%
% Opciones de convergencia de la columna
options =
optimset('MaxFunEvals',50000,'Display','off','TolX',...
        Param.eps/100, 'TolFun', Param.eps,
'MaxIter',100000,...
        'TolCon',Param.eps);

% Convergencia
try
    x=lsqnonlin(@(x)Eval_Dest_SS(x,Param,Design),0.6,0,1,options);
catch
    x = 0.00;
end

% Evalua la columna de destilacion en el punto de convergencia
[Error, Result] = Eval_Dest_SS(x,Param,Design);
end
%*****
**
%% CONVERGENCIA DE LA COLUMNA EN ESTADO ESTACIONARIO
function [Error, Result]=Eval_Dest_SS(x0,Param,Design)
%
% Calcula el error en la composición del vapor (grado de
convergencia)
%
% Parameters for the binary distillation problem

```

```

alpha          = Param.alpha;                % Relative
volatility     = Param.alpha;                % Relative
F              = Param.F;                    % Flujo de
alimentacion   = Param.F;                    % Flujo de
z              = Param.z;                    % Composición de
la alimentacion (compuesto interes)
% Asignacion de las variables de diseño
Tray_R        = Design.T_R;                  % Plato de reflujo
Tray_F        = Design.T_F;                  % Plato de
alimentacion
R              = Design.Rss;                 % Reflujo liquido
V              = Design.Vss;                 % Reflujo Vapor

q = zeros(1,Tray_R);                         % Indicador del
plato de reflujo
p = zeros(1,Tray_R);                         % Indicador del
plato de Alimentacion
q(Tray_R) = 1;                               % Asigna la
posición del reflujo
p(Tray_F) = 1;                               % Asigna la
posición de la alimentacion

x_D           = x0;                           % Composicion del
destilado
% Balance total del condensador determina flujo de destilado.
D = V - R;                                     %Flujo de Destilado

% Balance por componente en el condensador
y_n = x_D;                                     % Fraccion molar en
el plato N

% Balance Global: (*)
B = F - D;                                     % Flujo de salida en
el Bottom

% Balance Global por componentes (*)
x_B = (z*F - D*x_D)/B;

% Balance total en reboiler
L(1) = V + B;

% Equilibrio en el reboiler
y_B = alpha*x_B/(1 + x_B*(alpha-1));

% Balance por componente del Reboiler
x(1) = x_B - V*(x_B - y_B)/L(1);

% Equilibrio en el plato 1
y(1) = alpha*x(1)/(1 + x(1)*(alpha-1));

```

```

for i = 1:Tray_R-1
    % Decide la fraccion molar en el plato, si este es el primer
plato
    if i==1; yi = y_B; else yi = y(i-1); end
    % Balance Total del plato i + 1
    L(i+1) = L(i)-F*p(i) - R*q(i);
    % Balance por componente en el plato i+1
    x(i+1) = x(i)-(V*(yi-y(i))+p(i)*F*(z-x(i))+q(i)*R*(x_D-
x(i)))/L(i+1);
    % Equilibrios en el plato i+1
    y(i+1) = alpha*x(i+1)/(1 + x(i+1)*(alpha-1));
end

Error = 1000*(y(Tray_R)-y_n); % Error en el
calculo

if not(isfinite(Error)) % Verifica que error
no tome valor infinito.
    Error = 1000;
end

% Asignacion de resultados
Result.D = D; % Flujo de
Destilado
Result.B = B; % Flujo de
salida en el Bottom
Result.L = L; % Flujo de
Liquido en cada plato
Result.x_D = x_D; % Composicion
del destilado (liquido)
Result.x_B = x_B; % Composicion
en el bottom (liquido)
Result.y_B = y_B; % Composicion
en el bottom (Vapor)
Result.x = x; % Composicion
Liquido en cada plato
Result.y = y; % Composicion
vapor en cada plato
Result.Error = Error/1000; % Error en el
calculo

end
%*****
**
%% COLUMNA DE DESTILACION EN ESTADO DINAMICO
function [XYf, T] = Dest_DS(Param,Design,Control)
%
% Resuelve la columna de destilacion en estado estacionario
%
% Calculo para la columna en estado estacionario
Result = Dest_SS(Param,Design);

```

```

    if Result.Error > Param.eps % Si error es
grande asume cero la fraccion de destilado
        [Error, Result] = Eval_Dest_SS(0,Param,Design);
    end

    Design.x_Bsp = Result.x_B;
    Design.x_Dsp = Result.x_D;
    % Valores iniciales de las ecuaciones diferenciales (valores del
estado estacionario)
    Tray_R = Design.T_R;
    XY(1) = Result.D; % Flujo de
Destilado
    XY(2) = Result.B; % Flujo de salida
en el Bottom
    XY(3) = Result.x_D; % Composiciï¿½n
del destilado (liquido)
    XY(4) = Result.x_B; % Composiciï¿½n
en el bottom (liquido)
    XY(5) = Design.Vss; % Flujo de Vapor
    XY(6) = Design.Rss; % Reflujo de
destilado
    XY(7) = 0.00; % ISE
    XY(8:12) = [Result.x_B 0 0 0 0]; %
    XY(13:17) = [Result.x_D 0 0 0 0]; %
    XY(18) = 0.00; %
    XY(19) = 0.00; %
    XY(20:19+Tray_R) = Result.L; % Flujo de Liquido
en cada plato
    XY(20+Tray_R:19+2*Tray_R) = Result.x; % Composicion
Liquido en cada plato

    if Param.Dt <= 0.001 % Vector de tiempo
        Te = [0 Param.Time];
    else
        Te = 0:Param.Dt:Param.Time;
    end

    % Vector de indices de Variables definidas positivas
    posit = [1:8,13,20:19+2*Tray_R];

    XY(posit) = abs(XY(posit)); % Asegura que las
variables sean positivas

    % Resuelve el sistema de ecuaciones diferenciales
    % Options de acuerdo a si se va a monitorear gráficamente
    if numel(Param.monitor) == 0
        opt = odeset('RelTol' , Param.eps/100, 'AbsTol',
Param.eps/100, 'NormControl', 'on', 'NonNegative', posit);
    else
        opt = odeset('RelTol' , Param.eps/100, 'AbsTol',
Param.eps/100, 'NormControl', 'on', ...

```

```

'OutputFcn',@odeplot,'OutputSel',Param.monitor,'NonNegative',posit
);
end

% RESUELVE EL SISTEMA DE ECUACIONES DIFERENCIALES
try
    [T,XYf] = ode23t(@Dmodel,Te,XY,opt,Design, Param, Control);
catch
    T = Te;
    XYf = repmat(XY,numel(T),1);
    XYf(:,7) = 100000;
end

end

%*****
**
%% SISTEMA DE ECUACIONES - DERIVADAS
function DXY = Dmodel(T, XY, Design, Param, Control)
%
% Evalua los valores de la derivada de las variables dinamica
% de acuerdo a los balance de masa y energia dinamicos.
%
% T      : Valor del tiempo.
% XY     : Valores de las variables dinamicas.
% DXY    : Derivada de las variables dinamicas para T.
%
% Orden de las variables dinamicas en el vector XY
% XY(1)      = D      Flujo de Destilado
% XY(2)      = B      Flujo de salida en
el Bottom
% XY(3)      = x_D     Composicion del
destilado (liquido)
% XY(4)      = x_B     Composicion en el
bottom (liquido)
% XY(5)      = V      Flujo de Vapor
% XY(6)      = R      Flujo de Reflujo
% XY(7)      = mu     ISE
% XY( 8:12)  = x_bm   y sus 4 derivadas de
x_bm
% XY(13:17)  = x_Dm   y sus 4 derivadas de
x_Dm
% XY(18)     = I_b
% XY(19)     = I_D
% XY(20:20+Tray_R) = L      Flujo de Liquido
en cada plato
% XY(21+Tray_R:21+2Tray_R) = x      Composicion
Liquido en cada plato

% Load Parametros del modelo
% Parameters for the binary distillation problem

```

```

alpha      = Param.alpha;           % Relative volatility
h_w        = Param.h_w;             % Height over the weir
Beta_pay   = Param.Beta_pay;        % Payback period
Beta_tax   = Param.Beta_tax;        % Tax factor

% Perturbacion
F           = f_perturb(T,Param.F,Param.b);           % Flujo de
alimentacion
z           = z_perturb(T,Param.z,Param.a);           % Composicion
de la alimentacion (compuesto interes)

% Asignacion de las variables de diseño
Tray_R     = Design.T_R;             % Plato de reflujo
Tray_F     = Design.T_F;             % Plato de
alimentacion
Dc         = Design.Dc ;             % Diametro de la
columna
Vss       = Design.Vss;             % steady state vapor
boilup
Rss       = Design.Rss;             % steady state reflux
ratio
x_Bsp     = Design.x_Bsp;           % set point for outlet
bottom fraction
x_Dsp     = Design.x_Dsp;           % set point for outlet
destillate fraction

q = zeros(1,Tray_R);                % Indicador del plato
de reflujo
p = zeros(1,Tray_R);                % Indicador del plato
de Alimentacion
q(Tray_R) = 1;                       % Asigna la posicion
del reflujo
p(Tray_F) = 1;                       % Asigna la posicion
de la alimentacion

% Asignacion de los parametros de Control
Kv        = Control.Kv;
Kr        = Control.Kr;
Tau_v     = Control.Tau_v;
Tau_r     = Control.Tau_r;

% Asignacion de las variables dinamicas
D         = XY(1);                    % Flujo de
Destilado
B         = XY(2);                    % Flujo de salida
en el Bottom
x_D       = XY(3);                    % Composicion del
destilado (liquido)
x_B       = XY(4);                    % Composiciï¿½n
en el bottom (liquido)
V         = XY(5);                    % Flujo de Vapor

```

```

R      = XY(6);           % Flujo de Reflujo
mu     = XY(7);           % ISE
x_Bm   = XY(8:12)';      %
x_Dm   = XY(13:17)';    %
I_B    = XY(18);         %
I_D    = XY(19);         %
L      = XY(20:19+Tray_R)'; % Flujo de Liquido
en cada plato
x      = XY(20+Tray_R:19+2*Tray_R)'; % Composición
Liquido en cada plato

% Calculo de los holdups
M      = 7.538115*(((0.0014134/Dc)^(2/3))+h_w)*Dc^2; % Holdup
Molar - plato
Mr     = 10*M;           % Holdup
Molar - reboiler
Mc     = 10*M;           % Holdup
Molar - condenser
Beta   = 0.05271*Dc^(4/3); %
Constante de tiempo plato
Betr   = 100*Beta;      %
Constante de tiempo reboiler
Betc   = 100*Beta;      %
Constante de tiempo condenser

% Inicializacion de DXY
DXY = zeros(1,19+2*Tray_R);

% Tray equilibrium
y = alpha*x./(1+x*(alpha-1));
y_N = y(Tray_R);
y_B = alpha*x_B./(1+x_B*(alpha-1));

% Condenser total balance
DXY(1) = (V-R-D)/Betc;

% Reboiler total balance
DXY(2) = (L(1)-V-B)/Betr;

% Condenser component balance
DXY(3) = V*(y_N-x_D)/Mc;

% Reboiler Component Balance
DXY(4) = (L(1)*(x(1)-x_B) + V*(x_B-y_B))/Mr;

% Final control lags
Vc = Vss + Kv*(x_Bm(1)-x_Bsp)-Kv*I_B/Tau_v;
Rc = Rss + Kr*(x_Dm(1)-x_Dsp)-Kr*I_D/Tau_r;

DXY(5) = (Vc-V)/0.90;

```

```

DXY(6) = (Rc-R)/0.50;

% ISE
DXY(7) = T*(x_D-x_Dsp).^2+T*(x_B-x_Bsp).^2;

% Measurement lags for the bottom composition
DXY(8:11) = x_Bm(2:5);
DXY(12) = x_B - sum([1,5,10,10,5].*x_Bm);

% Measurement lags for the distillates composition
DXY(13:16) = x_Dm(2:5);
DXY(17) = x_D - sum([1,5,10,10,5].*x_Dm);

% PI controllers
DXY(18) = x_Bm(1) - x_Bsp;
DXY(19) = x_Dm(1) - x_Dsp;

% Tray total balance
DXY(20:19+Tray_R)=( [L(2:Tray_R) 0]-L + F*p + R*q)/Beta;

% Tray component balances
DXY(20+Tray_R:19+2*Tray_R)= ([L(2:Tray_R) 0].*([x(2:Tray_R) 0]-
x) + ...
V *([y_B y(1:Tray_R-1)]-y) + F*p.*(z-x) + R*q.*(x_D-
x))/M;

DXY = DXY';
Tim = toc; % Registro del tiempo

if Tim > Param.proc
    DXY = 1000*ones(size(DXY));
end

end

%*****
**
%% Funcion de perturbacion Fraccion molar
function zp = z_perturb(t,z,a)
    zp = z + a(1)./(1+exp(-a(2).*(t-a(3))));
end
%*****
**
%% Funcion de perturbacion Fraccion molar
function fp = f_perturb(t,f,b)
    fp = f + b(1)./(1+exp(-b(3).*(t-b(2))));
end
%*****
**
%% BUSQUEDA TABU - VARIABLES ENTERAS

```

```

function [y_min,x_min] = tabu(p,param)
% p  parametros del modelo
%   p.Step : Paso final de la busqueda tabu
%   p.y0   : Valor inicial de las variables
%   p.Iter : Numero maximo de iteraciones
%   p.N    : Tamaño de la lista Tabu (memoria)
%   p.x0   : Velores iniciales variables continuas
%   p.case : caso de optimizacion
%           0: Optimizacion SS           - deterministico
%           1: Optimizacion Integrada - deterministico
% Nota: se usa para la segunda variable la restriccion y(2) <=
y(1) - 4
%
delete 'Report_tabu.txt'
diary('Report_tabu.txt')
disp('%%%%%%%%%%          OPTIMIZACION          CON          BUSQUEDA          TABU
%%%%%%%%%%')
disp(' ')
disp('Numero de Variables Enteras: 2')
L = [10 4]; % Limite inferior de busqueda
U = [100 96]; % Limite superior de busqueda
disp(['Limite inferior: ', num2str(L)])
disp(['Limite superior: ', num2str(U)])
disp(' ')
disp('CONFIGURACION DE LA BUSQUEDA')
disp(' ')
disp(['Maximo tamaño del paso de búsqueda: ', num2str(p.Step)])
disp(['Valores iniciales : ', num2str(p.y0)])
disp(['Numero maximo de iteraciones : ', num2str(p.Iter)])
disp(['Tamaño máximo de la lista tabu : ', num2str(p.N)])
disp(' ')
disp('EJECUTANDO BUSQUEDA')
disp(' ')
L_Tabu = []; % Lista tabu (pares)
(2,p.N)
% Se generan los alrededores, condiciones de fronteras
periodicas
% En el proceso se cambia y0 para que cumpla las condicines de
fronteras
s = 0; % Contador de
iteraciones
Step = 1; % Tamaño del paso
inicial
f_min= 1e10; % Valor incial de la
funcion objetivo
y_min= p.y0; % valor incial
(pivote)
x_min = p.x0; % Valor incial para
las variables continuas
Ind = 0; % Indicador de si se
encontro o no un nuevo minimo

```

```

NeVal = 0; % Numero de
optimizaciones continuas
NeFun = 0; % Numero de total
evaluacion de funcion objetivo

while and(s<=p.Iter,Step<=p.Step) % aumenta el contador
    s = s + 1;
de iteraciones
    y0 = y_min;
    x0 = x_min;
    % Se generan los alrededores, condiciones de fronteras
    % periodicas
    % En el proceso se cambia y0 para que cumpla las condicines de
    % fronteras

    N_act = 0; %
    disp([num2str(s) ' : Calculando alrededores'])
    L_aux= surround(y0,L,U, Step);
    disp(' Lista Alrrededores a ensayar')
    disp([' ' sprintf('%5d',L_aux(1,:))])
    disp([' ' sprintf('%5d',L_aux(2,:))])

    L_act= [];

    % Compara cada termino con la lista tabu
    n = size(L_Tabu,2);
    disp(' Comparando lista Tabu')
    if (n ~= 0)
        disp(' Lista Tabu')
        disp([' ' sprintf('%5d',L_Tabu(1,:))])
        disp([' ' sprintf('%5d',L_Tabu(2,:))])
        for i = 1:size(L_aux,2)
            aux =
numel(find(sum(L_Tabu==L_aux(:,i))*ones(1,n))==2));
            if (aux == 0) % No se encuentra
en la lista Tabu
                disp([' Se Aceptó el punto: ',
num2str(L_aux(:,i))])
                N_act = N_act + 1; %
Aumenta el numero de aceptados
                L_act(:,N_act) = L_aux(:,i); %
Guarda el para a ensayar
            end
        end
    else
        disp(' Lista Tabu vacia')
        N_act = size(L_aux,2);
        L_act = L_aux;
    end
    disp(' Evaluando puntos')
    % evalua los puntos

```

```

        if N_act == 0                                % No hay puntos para
ensayar entonces disminuye el paso
            disp('          No existen puntos - Aumentando el paso')
            Step = Step + 1;                          % Aumenta el paso
            disp(['          Nuevo paso: ' num2str(Step)])
        else
            for i = 1: N_act                          % para cada punto
seleccionado
                NeVal = NeVal + 1;
                disp(['          Evaluando el punto: ',num2str(i),' ->
',num2str(L_act(:,i))'])
                switch p.case
                    case 0
                        [x,f,exitf,op] = Opt_Con_SS(L_act(:,i)',x0,param);
                        rest = 2*param.eps;
                    case 1
                        % Estimacion sin control
                        [xs,f,exitf,op] = Opt_Con_SS(L_act(:,i)',x0(1:3),param);
                        xc = x0(4:7);
                        xi = [xs, xc];                  % valor inicial

                        % Estimacion Integrada
                        [x,f,exitf,op] = Opt_Con_Int(L_act(:,i)',xi,param,p.ty);
                        rest = 2*param.eps;
                    end
                NeFun = NeFun + op.funcCount;
                if and(f <=f_min,op.constrviolation<=rest)
                    disp(['          Nuevo Minimo Encontrado: '
num2str(f)])
                    % Nuevo valor encontrado
                    disp('          Valor optimo')
                    disp(x)
                    y_min = L_act(:,i)'; f_min = f;
                    ops = op;
                    Ind = 1;
                    x_min = x; ops.Ni = NeVal; ops.NeFun = NeFun;
                    % Igualmente se agrega a la lista tabu
                else
                    if (op.constrviolation >rest)
                        disp('          Evaluacion Fallida: Violacion
de restricciones')
                    else
                        disp('          Evaluacion Fallida: No mejora
la funcion objetivo')
                    end
                end
                % Se añade a la lista tabu
                L_Tabu = [L_act(:,i), L_Tabu];
            end
            disp('          Redimensionando la lista tabu')
            % Redimensionamiento de la lista tabu

```

```

if (size(L_Tabu,2)>p.N)
    L_Tabu = L_Tabu(:,1:p.N);
    % Se agrega el valor minimo actual
    L_Tabu = [L_Tabu, y_min'];
else
    disp('                No se redimensionó')
end
if (Ind == 0 )
    Step = Step + 1;
    if Step <= p.Step
        disp(['                Nuevo paso - minimo NO
ENCONTRADO: ', num2str(Step)])

        end
    else
        Step = 1; Ind = 0;
        disp(['                Reinicio del paso - minimo
ENCONTRADO: ', num2str(Step)])
        end %
    end
end
op.Nt = NeVal;
if s>p.Iter
    disp(['Programa terminado: Numero de iteraciones alcanzado:
',num2str(p.Iter)])
else
    disp(['Programa terminado: Se Alcanzo el tamaño de paso
maximo: ', num2str(p.Step)])
end
disp('RESUMEN DE RESULTADO ')
disp(' ')
disp(['Funcion objetivo optimo: ',num2str(f_min)])
disp(['Variables binaria optima: ',num2str(y_min)])
disp(['Variables continua optima: ',num2str(x_min)])
disp(' ')
disp(['Numero de optimizaciones Continuas                :
',num2str(ops.Ni)])
disp(['Iteraciones Optimizacion Continua (Optima):
',num2str(ops.iterations)])
disp(['Evaluacion de funciones (optimo)                :
',num2str(ops.funcCount)])
disp(['Evaluacion Total de funciones                    :
',num2str(ops.NeFun)])
disp(['Violacion de restricciones                        :
',num2str(ops.constrviolation)])
disp('Algoritmo ')
disp(ops.algorithm)
disp('Mensaje de salida - optimizacion continua')
disp(ops.message)
disp(['Algoritmo: ',num2str(ops.algorithm)])
disp(' ')
disp(x_min)

```

```

diary off
disp('Detalle del proceso en Report_tabu.txt')
disp(' ')
end
%*****
**
%% DETERMINA ALREDEDORES EN BUSQUEDA TABU
function y_s = surround(y,L,U, Step)
% Se aplica condiciones de fronteras periodicas
y_s = [];
yfind = [];
yi = y'*ones(1,3);
yi = yi + [0 1 -1 ; 0 1 -1]*Step;
for i = 1:3
% Para la primera variable comprueba límites
if (yi(1,i)>U(1))
m = floor((yi(1,i)-U(1))/(U(1)-L(1)));
yi(1,i) = L(1)+(yi(1,i)-U(1))- m*(U(1)-L(1));
elseif (yi(1,i)<L(1))
m = floor((L(1) - yi(1,i))/(U(1)-L(1)));
yi(1,i) = U(1)-(L(1) - yi(1,i)) + m*(U(1)-L(1));
end
for j = 1:3
yaux(1) = yi(1,i);
yaux(2) = yi(2,j);
if (yi(2,j)>yaux(1)-4)
m = floor((yi(2,j)-yaux(1)+4)/(yaux(1)-4-L(2)));
yaux(2) = L(2)+(yi(2,j)-yaux(1)+4)-m*(yaux(1)-4-L(2));
elseif (yi(2,j)<L(2))
m = floor((L(2)-yi(2,j))/(yaux(1)-4-L(2)));
yaux(2) = yaux(1)-4 - (L(2)-yi(2,j)) + m*(yaux(1)-4-
L(2));
end
% Revisa si existen puntos repetidos
n = size(y_s,2);
if (n ~= 0)
aux = numel(find(sum(y_s==yaux'*ones(1,n))==2));
if (aux == 0) % No se encuentra repetido
y_s = [y_s, yaux'];
end
else
y_s = yaux';
end
yfind = [yfind, yaux'];
end
end
end
%*****
**
%% OPTIMIZADOR DETERMINISTICO. VARIABLES CONTINUAS ESTADO
ESTACIONARIO
function [x,f,exitf,op] = Opt_Con_SS(y,x0,p)

```

```

% y : Valores enteros
% x0 = Valores iniciales de las variables continuas
% Restricciones lineales de desigualdad
disp(' ')
disp('%%% OPTIMIZACION DETERMINISTICA - VARIABLES CONTINUAS
ESTADO ESTACIONARIO %%%')
disp(' ')
Aineq = [0 1 -1; 0 -1 1];
bineq = [0, p.F];
% Restricciones lineales de igualdad
Aeq = [];
beq = [];
% Restricciones de frontera
lb = [ 1e-3 1e-5 1e-5];
ub = [ 20      5  5 ];
% Opciones de optimizacion
Options = optimset('TolCon', p.eps, 'TolX', p.eps,
'TolFun',100*p.eps,...
'Diagnostics','off','Display',
'iter','UseParallel','always');
% Optimizacion en Estado estacionario - Caso Deterministico
[x,f,exitf,op] = fmincon(@(x)Costo(x,y), x0, Aineq, bineq, Aeq,
beq, ...
lb, ub, @(x)SS_nonlinear(x,y,p), Options);
disp(' ')
disp(['Funcion objetivo optimo: ',num2str(f)])
disp(['Violacion de restricciones :
',num2str(op.constrviolation)])
disp('Valores de los optimizadores: ')
disp(x)
disp(' ')

end
%*****
**
%% RESTRICCIONES NO-LINEALES ESTADO ESTACIONARIO
function [C,Ceq] = SS_nonlinear(x,y,p)
% C : restricciones no-lineales de desigualdad
% Ceq : restricciones no-lineales de igualdad
%
% Definicion de x
% T_R % Plato de reflujo
% T_F % Plato de alimentacion
% Dc % Diametro de la columna
% Rss % Reflujo liquido
% Vss % Reflujo Vapor
%
% The diameter of the column must be large enough to avoid
flooding within
% the column
C(1) = 0.6719*sqrt(x(3))-x(1);
% Resuelve la columna de destilacion en SS

```

```

% Variables de diseño
D.T_R = y(1); % Plato de reflujo
D.T_F = y(2); % Plato de
alimentacion
D.Dc = x(1); % Diametro de la
columna
D.Rss = x(2); % Reflujo liquido
D.Vss = x(3); % Reflujo Vapor
R = Dest_SS(p,D);
% Product specifications
C(2) = p.xd_s - R.x_D;
C(3) = R.x_B - p.xb_s;
% Restricciones para las composiciones
C(4) = -min(R.x); % Composiciones
positivas
C(5) = -min(R.y);
C(6) = max(R.x)-1; % Composiciones
menores a 1
C(7) = max(R.y)-1;
Ceq = [];
end
%*****
**
%% OPTIMIZADOR DETERMINISTICO - SOLO CONTROL
function [x,f,exitf,op] = Opt_Con_OC(x,k0, p)
% x : Variables de diseño (enteras y binarias)
% K_0 : Valores iniciales de las ganancias y constantes de
tiempo
% Restricciones lineales de desigualdad
delete 'Report_Opt_Con_OC.txt'
diary('Report_Opt_Con_OC.txt')
disp(' ')
disp('%% OPTIMIZACION DETERMINISTICA - SOLO CONTROL %%%')
disp(' ')
% Restricciones de frontera
lb = [ -1.0e+3 -1.0e+3 p.eps p.eps ];
ub = [ 1.0e+3 1.0e+3 1.0e+3 1.0e+3 ];
disp(['Limite inferior: ', num2str(lb)])
disp(['Limite superior: ', num2str(ub)])
disp(['Valor Inicial : ', num2str(k0)])
disp(' ')
disp('EJECUTANDO BUSQUEDA')
disp(' ')
Aineq = [];
bineq = [];
% Restricciones lineales de igualdad
Aeq = [];
beq = [];
% Opciones de optimizacion
Options = optimset('TolCon', p.eps, 'TolX', p.eps,
'TolFun',100*p.eps,...

```

```

                                'Diagnostics','off','Display',
'iter','UseParallel','always');

% Optimizacion en Estado estacionario - Caso Deterministico
[x,f,exitf,op] = fmincon(@(k)Control(k,x,p), k0, Aineq, bineq,
Aeq, beq, ...
                        lb, ub, @(k)OC_nonlinear(k,x,p), Options);

disp(' ')
disp('RESUMEN DE RESULTADO ')
disp(' ')
disp(['Funcion objetivo optimo: ',num2str(f)])
disp(['Variables control optima : ',num2str(x)])
disp(' ')
disp(['Iteraciones Optimizacion           :
',num2str(op.iterations)])
disp(['Evaluacion de funciones           (optimo):
',num2str(op.funcCount)])
disp(['Violacion de restricciones       :
',num2str(op.constrviolation)])
disp('Algoritmo ')
disp(op.algorithm)
disp('Mensaje de salida - optimizacion continua')
disp(op.message)
disp(['Algoritmo: ',num2str(op.algorithm)])
disp(' ')
disp('Valores de los optimizadores: ')
disp(x)
diary off
disp('Detalle del proceso en Report_Opt_Con_OC.txt')
disp(' ')
end
%*****
**
%% FUNCION COSTO CONTROL
function mu = Control(k,Design,Param)
    Ctr.Kv    = k(1);
    Ctr.Kr    = k(2);
    Ctr.Tau_v = k(3);
    Ctr.Tau_r = k(4);
    % Evalua la respuesta dinamica
    [XYf, T] = Dest_DS(Param,Design,Ctr);
    % Determina las restricciones no-lineales
    % Product specifications
    Nt = numel(T);
    mu = XYf(Nt,7);
end
%*****
**
%% RESTRICCIONES NO LINEALES - OPTIMIZACION DEL CONTROL
function [C,Ceq] = OC_nonlinear(k,Design,Param)
    % Asignacion de la estructura de control
    Ctr.Kv    = k(1);

```

```

Ctr.Kr      = k(2);
Ctr.Tau_v  = k(3);
Ctr.Tau_r  = k(4);
% Evalua la respuesta dinamica
[XYf, T] = Dest_DS(Param,Design,Ctr);
Nt = numel(T);
% Determina las restricciones no-lineales
% Product specifications
C(1) = (Param.xd_s - XYf(Nt,3));
C(2) = (XYf(Nt,4) - Param.xb_s);
% Restricciones para las composiciones
Tray_R = Design.T_R;
Rx = max(max([XYf(:,3:4), XYf(:,20+Tray_R:19+2*Tray_R)]));
C(3) = Rx-1; % Composiciones menores a 1
C(4) = (Param.Time - Param.Dt - T(Nt))/1e5;
% Restriccion en mu, derivada cercana a cero en el periodo de
control
% Restriccion en mu, derivada cercana a cero en el periodo de
control
%Nt2 = Nt - 1;
%if Nt == 1
%  Nt2 = Nt ;
%end
%
%DTau = (XYf(Nt,7)-XYf(Nt2,7))/(T(Nt)-T(Nt2));
%C(5) = abs(DTau)-Param.eps;
Ceq=[ ];
end
%*****
**
%% OPTIMIZADOR DETERMINISTICO - INTEGRADO
function [x,f,exitf,op] = Opt_Con_Int(y,x0,p,ty)
disp(' ')
disp('%%%% OPTIMIZACION DETERMINISTICA - VARIABLES CONTINUAS
INTEGRADO %%%%'')
disp(' ')
% y : Valores enteros
% x0 = Valores iniciales de las variables continuas
% Restricciones lineales de desigualdad
Aineq = [0 1 -1 0 0 0 0; 0 -1 1 0 0 0 0];
bineq = [0, p.F];
% Restricciones lineales de igualdad
Aeq = [];
beq = [];

% Restricciones de frontera
lb = [ 1e-3 1e-5 1e-5 -1.0e+3 -1.0e+3 p.eps p.eps ];
ub = [ 20      5      5      1.0e+3 1.0e+3 1.0e+3 1.0e+3 ];

% Opciones de optimizacion Deterministico
% Options = optimset('TolCon', p.eps,
'TolFun',1000*p.eps,'MaxFunEvals',5000,'MaxIter', Inf,...

```

```

%                                     'Diagnostics','off','Display',
'iter','UseParallel','always');
Options = optimset('TolCon',
p.eps,'TolX',p.eps,'DiffMaxChange',p.eps,'TolFun',100*p.eps,...
'Diagnostics','off','Display',
'iter','UseParallel','always');
if ty == 0
% Optimizacion en Estado dinamico - Caso Deterministico
tic % Inicializacion del Registro del tiempo
[x,f,exitf,op] = fmincon(@(x)Costo(x,y), x0, Aineq, bineq,
Aeq, beq, ...
lb, ub, @(x)Int_nonlinear(x,y,p), Options);
elseif ty == 1

Options =
gaoptimset('PopulationSize',100,'Generations',1000,...
'TolFun', 1000*p.eps,'TolCon',
10*p.eps,'InitialPopulation',x0,...
'Display','iter','UseParallel','always');

[x,f,exitf,op] = ga(@(x)Costo(x,y), numel(x0), Aineq, bineq,
Aeq, beq, ...
lb, ub, @(x)Int_nonlinear(x,y,p), Options);
op.funcCount = op.generations;
op.constrviolation = op.maxconstraint;
end

end

%*****
**
% RESTRICCIONES NO-LINEALES - INTEGRADO
function [C,Ceq] = Int_nonlinear(x,y,Param)
% C : restricciones no-lineales de desigualdad
% Ceq : restricciones no-lineales de igualdad
%
% Definicion de x
% T_R % Plato de reflujo
% T_F % Plato de alimentacion
% Dc % Diametro de la columna
% Rss % Reflujo liquido
% Vss % Reflujo Vapor
%
% Resuelve la columna de destilacion en estado estacionario
% Variables de diseño
D.T_R = y(1); % Plato de reflujo
D.T_F = y(2); % Plato de alimentacion
D.Dc = x(1); % Diametro de la columna
D.Rss = x(2); % Reflujo liquido
D.Vss = x(3); % Reflujo Vapor

% Asignacion de la estructura de control

```

```

Control.Kv      = x(4);
Control.Kr      = x(5);
Control.Tau_v   = x(6);
Control.Tau_r   = x(7);

[XYf, T] = Dest_DS(Param,D,Control);
Nt = numel(T);
% Determina las restricciones no-lineales

% The diameter of the column must be large enough to avoid
flooding within
% the column
C(1) = 0.6719*sqrt(x(3))-x(1);

% Product specifications
C(2) = (Param.xd_s - XYf(Nt,3));
C(3) = (XYf(Nt,4) - Param.xb_s);

% Restricciones de las variables dinamicas
Rx = max(max([XYf(:,3:4), XYf(:,20+D.T_R:19+2*D.T_R)]));
C(4) = Rx - 1;% Composiciones menores a 1
% Asegura que el tiempo es igual a Param.Time
C(5) = (Param.Time - Param.Dt - T(Nt))/1e5;
% Restriccion en mu, derivada cercana a cero en el periodo de
control
%Nt2 = Nt - 1;
%if Nt == 1
%  Nt2 = Nt;
%end
%DTau = (XYf(Nt,7)-XYf(Nt2,7))/(T(Nt)-T(Nt2));
%C(6) = abs(DTau)-Param.eps;
% Restriccion para mu
epsilon = XYf(Nt,7);
C(6) = (epsilon - Param.epsilon)/100;
C(7) = -(epsilon - 0.10*Param.epsilon)/100;
%Ceq = (epsilon - Param.epsilon)/100;

Ceq = [ ];

C = real(C);
Ceq = real(Ceq);

if sum(isnan(C)+isinf(C))>0
  C = 100*ones(size(C));
end

if sum(isnan(Ceq)+isinf(Ceq))>0
  Ceq = 100*ones(size(Ceq));
end

```

```

end
%*****
**
%% REPORTE DE SIMULACION ESTACIONARIA Y DINAMICA DE LA COLUMNA
function report(Design, Control, Param, jobname, ind)
    % Design : Parametros de diseño
    % Control: Parametros de control
    % Param  : Parametros del sistema
    % jobname: Descripción de la simulación
    % ind    : nombre indicador
    diary off
    clc
    % Se genera el nombre base de la simulación
    % base=[ind, '_', datestr(now, 'ddmmmmmyyyy_HH_MM')];
    base=ind;

    mkdir(base)
    diary([base, '/Report.txt'])
    % Encabezado
    disp(' ')

disp('*****
*****')
    disp('                OPTIMIZACION DISEÑO-CONTROL DE PROCESOS')
    disp(' Columna de Destilación binaria')
    disp(' ')
    disp(' (c) 2010')

disp('*****
*****')
    disp(' ')
    disp([' Titulo: ', jobname])
    disp(' ')
    disp([' Fecha: ', datestr(now, 'dddd dd mmmm yyyy HH:MM:PM')])

disp('*****
*****')
    disp(' ')
    disp(' Parametros del Sistema')
    disp(' ')
    disp([' Volatilidad Relativa      : ', num2str(Param.alpha)]);
    disp([' Altura de inundacion      : ', num2str(Param.h_w), ' m']);
    disp([' Tiempo de Recuperacion     : ', num2str(Param.Beta_pay), '
años']);
    disp([' Factor Fiscal              : ', num2str(Param.Beta_tax)]);
    disp([' Flujo de Alimentacion        : ', num2str(Param.F), '
kmol/min']);
    disp([' Fraccion de A alimentado: ', num2str(Param.z)]);
    disp([' Fraccion de A destilado  : ', num2str(Param.xd_s), '
(requerido)']);
    disp([' Fraccion de A residuo    : ', num2str(Param.xb_s), '
(requerido)']);

```

```

disp([' Tiempo de Simulacion      : ', num2str(Param.Time), '
min'])
disp([' Intervalo de Tiempo      : ', num2str(Param.Dt), ' min'])
disp(' ')
disp(' Parametros de Diseño')
disp(' ')
disp([' Numero total de platos : ', num2str(Design.T_R)]);
disp([' Plato de alimentacion   : ', num2str(Design.T_F)]);
disp([' Diametro de la columna  : ', num2str(Design.Dc), ' m']);
disp([' Reflujo destilado       : ', num2str(Design.Rss), '
kmol/min']);
disp([' Reflujo Vapor          : ', num2str(Design.Vss), '
kmol/min']);
disp(' ')
disp(' Parametros de Perturbacion')
disp(' ')
disp([' Fraccion molar          : ', num2str(Param.a)]);
disp([' Flujo de alimentacion   : ', num2str(Param.b)]);
disp(' ')
disp(' Parametros de Control')
disp(' ')
disp(' Controlador de Reflujo de vapor')
disp([' Ganancia                : ', num2str(Control.Kv), '
kmol/min']);
disp([' Constante de tiempo: ', num2str(Control.Tau_v), ' min']);
disp(' Controlador de Reflujo de destilado')
disp([' Ganancia                : ', num2str(Control.Kr), '
kmol/min']);
disp([' Constante de tiempo: ', num2str(Control.Tau_r), ' min']);
disp(' ')

disp('*****
*****')
disp(' ')
% Se calcula el costo
y(1) = Design.T_R; % Plato de reflujo
y(2) = Design.T_F; % Plato de alimentacion
x(1) = Design.Dc ; % Diametro de la columna
x(2) = Design.Rss; % Reflujo liquido
x(3) = Design.Vss; % Reflujo Vapor
[C_t, C_u, C_c ]= Costo(x,y);
disp([' Costo Capital          : ', num2str(C_c), ' $']);
disp([' Costo Utilidad          : ', num2str(C_u), ' $']);
disp([' Costo Total              : ', num2str(C_t), ' $']);
disp(' ')
disp(' Simulacion en Estado Estacionario')
R = Dest_SS(Param,Design);
disp(' ')
disp([' Flujo de Destilado      (kmol/min) : ',
num2str(R.D), ' kmol/min']);
disp([' Flujo de Residuo        (kmol/min) : ',
num2str(R.B), ' kmol/min']);

```

```

disp([' Fraccion de A destilado (liquido)           : ',
num2str(R.x_D)]);
disp([' Fraccion de A Residuo (liquido)           : ',
num2str(R.x_B)]);
disp([' Flujo Liquido en cada plato (kmol/min)     : ',
num2str(R.L), ' kmol/min']);
disp([' Fraccion de A Liquido en cada plato       : ',
num2str(R.x)]);
disp([' Fraccion de A vapor en cada plato        : ',
num2str(R.y)]);
disp([' Error en el calculo                       : ',
num2str(R.Error)]);
disp(' ')
disp([' Simulacion en estado dinamico - Tiempo    : ',
num2str(Param.Time), ' min']);
[XY, T] = Dest_DS(Param,Design,Control);
Tray_R = Design.T_R; % Plato de reflujo
nt = numel(T); % Numero de puntos
disp(' ')
disp([' Flujo de Destilado (kmol/min)           : ',
num2str(XY(nt,1))] );
disp([' Flujo de Residuo (kmol/min)            : ',
num2str(XY(nt,2))] );
disp([' Fraccion de A destilado (liquido)       : ',
num2str(XY(nt,3))] );
disp([' Fraccion de A Residuo (liquido)        : ',
num2str(XY(nt,4))] );
disp([' Flujo de Vapor (kmol/min)              : ',
num2str(XY(nt,5))] );
disp([' Flujo de destilado (kmol/min)          : ',
num2str(XY(nt,6))] );
disp([' ISE (min^2)                            : ',
num2str(XY(nt,7))] );
disp([' Fraccion de A Residuo - Derivadas(liq.): ',
num2str(XY(nt, 8:12))] );
disp([' Fraccion de A Destilado- Derivadas(liq.): ',
num2str(XY(nt,13:17))] );
disp([' Término integral - Controlador residuo : ',
num2str(XY(nt,18))] );
disp([' Término integral - Controlador destilado: ',
num2str(XY(nt,19))] );
disp([' Flujo de A en cada plato(liq.)(kmol/min): ',
num2str(XY(nt,20:19+Tray_R))] );
disp([' Fraccion de A en cada plato (liq)       : ',
num2str(XY(nt,20+Tray_R:19+2*Tray_R))] );

% Gráficas importantes
Gtit = ['          Flujo de destilado           '
'          Flujo de residuo                   '
'          Fraccion liquida de A destilado    '
'          Fraccion liquida de A residuo     '
'          Flujo de Vapor                      '

```

