



UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Ingeniería, Arquitectura y Diseño

Maestría y Doctorado en Ciencias e Ingeniería

Algoritmo de Aproximaciones Estocásticas con
Perturbación Simultánea para la Optimización
Robusta de Procesos

TESIS

Como Requisito Parcial para Obtener el Grado de
Doctor en Ciencias de la Ingeniería Industrial

Presenta

Jesús Everardo Olguín Tiznado

Director de Tesis

Dr. Rafael García Martínez

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO
MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA

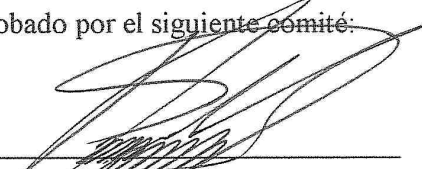
ALGORITMO DE APROXIMACIONES ESTOCÁSTICAS CON PERTURBACIÓN SIMULTÁNEA
PARA LA OPTIMIZACIÓN ROBUSTA DE PROCESOS


TESIS


QUE PARA OBTENER EL GRADO DE DOCTOR EN CIENCIAS PRESENTA:


JESÚS EVERARDO OLGUÍN TIZNADO

Y aprobado por el siguiente comité:


Dr. Rafael García Martínez
Director del comité


Dr. Jaime Sánchez Leal
Miembro del comité


Dr. Manuel Arnoldo Rodríguez Medina
Miembro del comité


Dr. Enrique Javier de la Vega Bustillos
Miembro del comité


Dr. Jorge Luis García Alcaraz
Miembro del comité

Enero de 2014

AGRADECIMIENTOS

A Dios, por darme la oportunidad de vivir y seguir luchando en la vida.

A mi amada esposa, compañera y amiga Claudia Camargo Wilson por su constante amor y apoyo.

A mis padres, Irma Tiznado y Lucio Olguín, por su formación y consejos para ser un hombre honesto y humilde.

A mis hermanos, Susana, †Gerardo, Rosa Isela, María Elsa, Mario Alberto, quienes siempre me han apoyado y han sido una fuente de motivación en mi vida.

A la Universidad Autónoma de Baja California, por el gran apoyo brindado para lograr este objetivo personal, al mismo tiempo por su tarea de forjar diariamente al recurso humano que México necesita. En especial al Dr. Oscar Roberto López Bonilla por creer en este proyecto de doctorado y su valioso apoyo en las facilidades brindadas para llevarlo a buen fin.

A la Dirección del Sistema de Institutos Tecnológicos, en especial a los Institutos Tecnológicos del Valle del Yaqui, de Hermosillo y de Ciudad Juárez, por su apoyo brindado al otorgar las facilidades para que el claustro doctoral se trasladara este tiempo de formación a la UABC, Campus Ensenada, para lograr culminar este gran proyecto.

A mi Director de tesis, Dr. Rafael García Martínez, por su valioso apoyo y la confianza recibida durante el desarrollo de esta tesis.

Al claustro doctoral: Dr. Jaime Sánchez Leal, Dr. Enrique Javier de la Vega Bustillos, Dr. Jorge Luis García Alcaraz y Dr. Manuel Arnoldo Rodríguez Medina, ya que con sus acertados comentarios y sugerencias enriquecieron la presente tesis.

Al equipo de trabajo de la Facultad de Ingeniería, Arquitectura y Diseño por su apoyo en la gestión de los trámites administrativos para la culminación y presentación de esta tesis.

A todos mis alumnos de licenciatura, que colaboraron en distintas formas en esta tesis: Luis Barbosa, Jose Luis Diaz, Angel Flores, Jorge Castañeda, Dennisse Perez, Edgar Cacho, Iván Cardenas, Francisco Valdes, Felipe Ruíz, Iris Cruz, por su valiosa ayuda.

A mi amada esposa Claudia

A mi Abuela Felipa[†]

A mi hermano Gerardo[†]

ALGORITMO DE APROXIMACIONES ESTOCÁSTICAS CON PERTURBACIÓN SIMULTÁNEA PARA LA OPTIMIZACIÓN ROBUSTA DE PROCESOS

RESUMEN

Todo proceso productivo tiene como meta que un “alto” porcentaje de los bienes o servicios generados cumplan con las especificaciones de diseño, dado que esta condición permite garantizar su permanencia en el mercado, al reducir desperdicios, retrabajos, entre otros. Este enfoque se concentra en agregar calidad dentro del diseño de productos y servicios a través de la determinación de las condiciones óptimas de operación, el desarrollo de procesos o productos más robustos, con el propósito de minimizar la variabilidad del rendimiento y la desviación del valor objetivo de interés.

Para los procesos productivos en los que se desconozca la relación funcional de la aptitud de los procesos para cumplir con las especificaciones, denotado por el índice de capacidad real de proceso C_{pk} , sus factores controlables e incontrolables (o ruido), existen métodos experimentales de optimización dual útiles para mejorar la capacidad o aptitud de los procesos. Sin embargo, cada uno de estos métodos presentan al menos una de las siguientes dos desventajas: primera, el número de observaciones de la variable de respuesta que requieren se incrementa conforme crece el número de factores controlables y factores de ruido; segunda, necesitan ajustar un modelo de regresión para representar dicha relación.

El objetivo de este trabajo de investigación plantea superar estas dos desventajas, para lo cual se propone modificar el Algoritmo de Aproximaciones Estocásticas con Perturbación Simultánea (AAEPSM), donde, dicho algoritmo se desarrolla para determinar las condiciones óptimas de operación de procesos conforme a la Metodología del Diseño de Parámetro Robusto (MDPR), tal que se observe el mejor valor del C_{pk} como una medida de razón señal a ruido.

Los resultados obtenidos se llevaron a cabo a través de simulación; los cuales permiten establecer que la efectividad del AAEPSM o que el valor máximo del C_{pk}

dependen del punto inicial X_0 , dado que es un parámetro que afecta el desempeño del algoritmo. Además, el valor del C_{pk} mejora conforme aumenta el valor del volumen del diseño factorial 2^k (considerado como el factor de ruido) con y sin un criterio de paro. Por último, la eficacia del AAEPSM para el caso de la variable Eficiencia en la Función Ganancia (EFG) en cada una de las funciones de prueba evaluadas tanto con criterio de paro como sin criterio de paro si es afectada. Sin embargo, para la variable Distancia al Óptimo (DO) y el Número de Iteraciones (NI) son afectadas parcialmente tanto con criterio de paro como sin criterio de paro del algoritmo.

Del AAEPSM, de acuerdo con lo expuesto anteriormente, se concluye que es un algoritmo viable para el Diseño de Parámetros Robusto (DPR) de procesos en los que se desconoce la relación funcional entre el C_{pk} , los factores controlables y los factores de ruido; procesos en los que se busca determinar las condiciones óptimas de operación.

Palabras claves: Diseño Robusto, Factor de Ruido, Optimización, Simulación, Diseño Factorial 2^k .

Índice

AGRADECIMIENTOS.....	vii
DEDICATORIA.....	vii
RESUMEN.....	vii
LISTA DE ACRÓNIMOS.....	vii
LISTA DE TABLAS.....	vii
LISTA DE FIGURAS.....	vii
1. INTRODUCCIÓN.....	1
1.1. Antecedentes.....	1
1.2. Planteamiento del Problema.....	3
1.3. Objetivos.....	6
1.4. Preguntas de Investigación.....	6
1.5 Hipótesis.....	7
1.6. Justificación.....	7
1.7. Limitaciones.....	8
1.8. Organización de la Tesis.....	9
2. LITERATURA REVISADA.....	10
2.1. Operación Evolutiva.....	10
2.2. Metodología de Superficie de Respuesta.....	11
2.3. Procedimientos de Aproximaciones Estocásticas.....	12
2.3.1. Método de Perturbación Simultánea.....	17
2.3.1.1. Análisis del Desempeño del Método de Perturbación Simultánea.....	18
2.4. Métodos Experimentales para el Mejoramiento del C_{pk}	19
2.4.1. Método de Respuesta Dual de Vining y Myers.....	20
2.4.2. Método de Respuesta Dual de Lin y Tu.....	22
2.4.3. Método de Respuesta Dual de Copeland y Nelson.....	23
2.4.4. Método de Respuesta Dual de Kwang y Lin.....	23
2.5. Algoritmo de Aproximaciones Estocásticas con Perturbación Simultánea Modificado.....	24
2.5.1. Procedimiento del Algoritmo de Aproximaciones Estocásticas con Perturbación Simultánea Modificado (AAEPSM).....	25
2.5.2. Criterio de Terminación.....	27
2.5.3. Algoritmo Propuesto.....	28
3. MATERIALES Y MÉTODOS.....	34
3.1. Materiales.....	34
3.2. Métodos.....	34
3.2.1. Descripción del proceso de simulación.....	35
3.3. Funciones de Prueba.....	36
3.4. Verificación de Hipótesis.....	40
3.4.1. Verificación que la efectividad del AAEPSM es independiente de las condiciones iniciales de operación (X_0) en las “primeras” iteraciones.....	40

3.4.2. Verificación que el AAEPSM logra mejorar el C_{pk} con factores de ruido.....	41
3.4.3. Verificación que la eficacia del AAEPSM es afectada por el volumen del diseño factorial 2^k , considerado como factor de ruido.....	42
4. ANÁLISIS Y DISCUSIÓN DE RESULTADOS.....	45
4.1. Verificar que la efectividad del AAEPSM es independiente de las condiciones iniciales de operación (X_0) en las “primeras” iteraciones.....	45
4.2. Verificar que el AAEPSM logra mejorar el C_{pk} con factores de ruido.....	46
4.2.1. Análisis de Resultados para cada Función de Prueba.....	47
4.2.1.1. Análisis de Resultados para la Función Valle Parabólico de Rosenbrock (FVPDR).....	47
4.2.1.2. Análisis de Resultados para la Función Cúbica de Rosenbrock (FCDR).....	49
4.2.1.3. Análisis de Resultados para la Función de Beale (FB).....	53
4.2.1.4. Análisis de Resultados para la Función Suma de Funciones Cuadráticas (FSFC).....	55
4.2.1.5. Análisis Gráfico para la FVPDR.....	58
4.2.1.6. Análisis Gráfico para la FCDR.....	60
4.2.1.7. Análisis Gráfico para la FB.....	62
4.2.1.8. Análisis Gráfico para la FSFC.....	64
4.3. Verificar que la eficacia del AAEPSM es afectado por el volumen del diseño factorial 2^k , considerado como factor de ruido.....	66
4.4. Validación de los Resultados Obtenidos con el AAEPSM.....	85
4.4.1. Validación y Análisis de Resultados del AAEPSM con la FVPDR.....	86
4.4.2. Análisis Gráfico para la Validación del AAEPSM con la FVPDR.....	86
4.4.3. Validación y Análisis de Resultados del AAEPSM con la FCDR.....	88
4.4.4. Análisis Gráfico para la Validación del AAEPSM con la FCDR.....	89
4.4.5. Validación y Análisis de Resultados del AAEPSM con la FB.....	90
4.4.6. Análisis Gráfico para la Validación del AAEPSM con la FB.....	91
4.4.7. Validación y Análisis de Resultados del AAEPSM con la FSFC.....	92
4.4.8. Análisis Gráfico para la Validación del AAEPSM con la FSFC.....	93
4.5. Discusión de Resultados.....	94
4.5.1. Comprobación de las Hipótesis.....	94
4.5.2. Discusión de los Resultados de Simulación.....	95
5. CONCLUSIONES Y RECOMENDACIONES.....	100
5.1. Conclusiones.....	100
5.2. Recomendaciones.....	102
BIBLIOGRAFÍA.....	104
APÉNDICE	
1. Publicaciones Derivadas del Trabajo de Tesis Doctoral.....	111
2. Listado de Programas.....	112

LISTA DE ACRÓNIMOS

AAEPSM: Algoritmo de Aproximaciones Estocásticas con Perturbación Simultánea Modificado.

MDPR: Metodología del Diseño de Parámetros Robusto

DPR: Diseño de Parámetro Robusto

AAE: Algoritmo de Aproximaciones Estocásticas

MPS: Método de Perturbación Simultánea

FVPDR: Función Valle Parabólico de Rosenbrock

FCDR: Función Cúbica de Rosenbrock

FB: Función de Beale

FSFC: Función Suma de Funciones Cuadráticas

ANOVA: Análisis de Varianza

EFG: Eficiencia en la Función Ganancia

DO: Distancia al Óptimo

NI: Número de Iteraciones

LISTA DE TABLAS

Tabla 2.1	Matriz del Diseño Robusto para el AAEPSM.....	26
Tabla 3.1	Funciones de Prueba Implementadas en el Proceso de Simulación.....	42
Tabla 3.2	Formato utilizado en el análisis de varianza para la AAEPSM.....	44
Tabla 4.1	Valores Promedios del C_{pk} para Dos Puntos Iniciales Distintos, para la FSFC con un volumen del Diseño Factorial 2^k $v = 0.1$	46
Tabla 4.2	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FVPDR Con Criterio de Paro.....	48
Tabla 4.3	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FVPDR Sin Criterio de Paro.....	49
Tabla 4.4	Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FVPDR Sin Factores de Ruido Con y Sin Criterio de Paro.....	50
Tabla 4.5	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FCDR Con Criterio de Paro.....	51
Tabla 4.6	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FCDR Sin Criterio de Paro.....	52
Tabla 4.7	Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FCDR Sin Factores de Ruido Con y Sin Criterio de Paro.....	52
Tabla 4.8	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FB Con Criterio de Paro.....	54
Tabla 4.9	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FB Sin Criterio de Paro.....	54
Tabla 4.10	Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FB Sin Factores de Ruido Con y Sin Criterio de Paro.....	55
Tabla 4.11	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FSFC Con Criterio de Paro.....	56
Tabla 4.12	Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FSFC Sin Criterio de Paro.....	57
Tabla 4.13	Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FSFC Sin Factores de Ruido Con y Sin Criterio de Paro.....	57
Tabla 4.14	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Valle Parabólico de Rosenbrock Con Criterio de Paro.....	67
Tabla 4.15	Análisis de Varianza para la Distancia al Óptimo de la Función Valle Parabólico de Rosenbrock Con Criterio de Paro.....	68
Tabla 4.16	Análisis de Varianza para el Número de Iteraciones de la Función Valle Parabólico de Rosenbrock Con Criterio de Paro.....	69
Tabla 4.17	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Cúbica de Rosenbrock Con Criterio de Paro.....	69
Tabla 4.18	Análisis de Varianza para la Distancia al Óptimo de la Función Cúbica de Rosenbrock Con Criterio de Paro.....	70
	Análisis de Varianza para el Número de Iteraciones de la Función	

Tabla 4.19	Cúbica de Rosenbrock Con Criterio de Paro.....	71
Tabla 4.20	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función de Beale Con Criterio de Paro.....	71
Tabla 4.21	Análisis de Varianza para la Distancia al Óptimo de la Función de Beale Con Criterio de Paro.....	72
Tabla 4.22	Análisis de Varianza para el Número de Iteraciones de la Función de Beale Con Criterio de Paro.....	73
Tabla 4.23	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Suma de Funciones Cuadráticas Con Criterio de Paro.....	74
Tabla 4.24	Análisis de Varianza para la Distancia al Óptimo de la Función Suma de Funciones Cuadráticas Con Criterio de Paro.....	74
Tabla 4.25	Análisis de Varianza para el Número de Iteraciones de la Función Suma de Funciones Cuadráticas Con Criterio de Paro.....	75
Tabla 4.26	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Valle Parabólico de Rosenbrock Sin Criterio de Paro.....	76
Tabla 4.27	Análisis de Varianza para la Distancia al Óptimo de la Función Valle Parabólico de Rosenbrock Sin Criterio de Paro.....	77
Tabla 4.28	Análisis de Varianza para el Número de Iteraciones de la Función Valle Parabólico de Rosenbrock Sin Criterio de Paro.....	77
Tabla 4.29	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Cúbica de Rosenbrock Sin Criterio de Paro.....	78
Tabla 4.30	Análisis de Varianza para la Distancia al Óptimo de la Función Cúbica de Rosenbrock Sin Criterio de Paro.....	79
Tabla 4.31	Análisis de Varianza para el Número de Iteraciones de la Función Cúbica de Rosenbrock Sin Criterio de Paro.....	80
Tabla 4.32	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función de Beale Sin Criterio de Paro.....	80
Tabla 4.33	Análisis de Varianza para la Distancia al Óptimo de la Función de Beale Sin Criterio de Paro.....	81
Tabla 4.34	Análisis de Varianza para el Número de Iteraciones de la Función de Beale Sin Criterio de Paro.....	82
Tabla 4.35	Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Suma de Funciones Cuadráticas Sin Criterio de Paro.....	83
Tabla 4.36	Análisis de Varianza para la Distancia al Óptimo de la Función Suma de Funciones Cuadráticas Sin Criterio de Paro.....	83
Tabla 4.37	Análisis de Varianza para el Número de Iteraciones de la Función Suma de Funciones Cuadráticas Sin Criterio de Paro.....	84
Tabla 4.38	Valores de Validación del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FVPDR Con Criterio de Paro...	87
Tabla 4.39	Valores de Validación del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FCDR Con Criterio de Paro.....	89
Tabla 4.40	Valores de Validación del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FB Con Criterio de Paro.....	91
Tabla 4.41	Valores de Validación del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FSFC Con Criterio de Paro.....	93

Tabla 4.42	Resultados de Significancia por Cada Función de Prueba Con Criterio de Paro para el AAEPSM.....	96
Tabla 4.43	Resultados de Significancia por Cada Función de Prueba Sin Criterio de Paro para el AAEPSM.....	97
Tabla 4.44	Valores Promedios de las Variables de Respuesta para cada Función de Prueba Con Criterio de Paro para el AAEPSM.....	98
Tabla 4.45	Valores Promedios de las Variables de Respuesta para cada Función de Prueba Sin Criterio de Paro para el AAEPSM.....	99

LISTA DE FIGURAS

Figura 3.1	FVPDR.....	37
Figura 3.2	FCDR.....	38
Figura 3.3	FSFC.....	39
Figura 3.4	FB.....	40
Figura 4.1	Gráfica de los \bar{C}_{pk} Generados para la FVPDR Con Factor de Ruido y Sin Factor de Ruido Con Criterio de Paro.....	58
Figura 4.2	Gráfica de los \bar{C}_{pk} Generados para la FVPDR Con Factor de Ruido y Sin Factor de Ruido Sin Criterio de Paro	59
Figura 4.3	Gráfica de los \bar{C}_{pk} Generados para la FCDR Con Factor de Ruido y Sin Factor de Ruido Con Criterio de Paro.....	61
Figura 4.4	Gráfica de los \bar{C}_{pk} Generados para la FCDR Con Factor de Ruido y Sin Factor de Ruido Sin Criterio de Paro	62
Figura 4.5	Gráfica de los \bar{C}_{pk} Generados para la FB Con Factor de Ruido y Sin Factor de Ruido Con Criterio de Paro.....	63
Figura 4.6	Gráfica de los \bar{C}_{pk} Generados para la FB Con Factor de Ruido y Sin Factor de Ruido Sin Criterio de Paro	64
Figura 4.7	Gráfica de los \bar{C}_{pk} Generados para la FSFC Con Factor de Ruido y Sin Factor de Ruido Con Criterio de Paro.....	65
Figura 4.8	Gráfica de los \bar{C}_{pk} Generados para la FSFC Con Factor de Ruido y Sin Factor de Ruido Sin Criterio de Paro	66
Figura 4.9	Gráfica de los \bar{C}_{pk} Generados para la Validación del AAEPSM con la FVPDR Con Criterio de Paro.....	88
Figura 4.10	Gráfica de los \bar{C}_{pk} Generados para la Validación del AAEPSM con la FCDR Con Criterio de Paro.....	90
Figura 4.11	Gráfica de los \bar{C}_{pk} Generados para la Validación del AAEPSM con la FB Con Criterio de Paro.....	92
Figura 4.12	Gráfica de los \bar{C}_{pk} Generados para la Validación del AAEPSM con la FSFC Con Criterio de Paro.....	94

Capítulo I. Introducción

1.1. Antecedentes.

La competitividad de una empresa y la satisfacción del cliente están determinadas por la calidad del producto, el precio, la calidad del servicio y la forma de cómo medir el desempeño de una organización. Dado que, la elección de lo que un negocio o un área mide y analiza, agrega valor y fija las prioridades de la organización (Gutiérrez y De la Vara, 2004).

Los métodos de mejoramiento de calidad son una herramienta útil en la medición y análisis del desempeño de los procesos productivos, dado que continuamente se optimiza el rendimiento de los niveles operativos de toda organización. Este mejoramiento se logra con los diferentes métodos o herramientas de calidad total existentes, entre las cuales se encuentran: Los Sistemas de Producción Justo a Tiempo, Kaizen, Mantenimiento Productivo Total, Poka Yoke, Control Estadístico de Procesos, Métodos de Optimización Experimental, entre otros.

La MDPR desarrollada por Taguchi en 1986 es importante para iniciativas de mejora de calidad industrial. Este enfoque se concentra en agregar calidad dentro del diseño de productos o servicios a través de la determinación de las condiciones óptimas de operación con el propósito de minimizar la variabilidad del rendimiento y la desviación del valor objetivo de interés. Varios de los modelos de optimización para el DPR relevantes están basados en el método de respuestas duales, que fue considerado por primera vez por (Myers y Carter, 1973). Aquí, la media y la varianza del proceso para una sola característica son modeladas separadamente utilizando superficies de respuesta. Estas funciones luego son optimizadas simultáneamente para determinar los parámetros óptimos de operación del sistema.

Por otro lado, existen diversos métodos de optimización experimental, algunos de ellos son: Algoritmo de Búsqueda Simplex, Operación Evolutiva, Metodología de Superficie de Respuesta (Vining y Myers, 1990), (Lin y Tu, 1995), (Copeland y Nelson, 1996), (Kwang y Lin, 1998), (Park y Cho, 2003), (Shin y Cho, 2005), (Cho y Park, 2005), (Kovach y Cho, 2006) y (Lee et al., 2007), los Métodos lagrangianos para restricciones con desigualdades (Kushner y Clark, 1978), Algoritmo de

Aproximaciones Estocásticas (Robbins y Monro, 1951), entre otros. Estos métodos implementan una serie de pruebas en las que se inducen cambios deliberados en las variables de entrada de un proceso o sistema, para observar e identificar las razones de los cambios en la respuesta de salida. Esto con el objetivo de determinar las condiciones óptimas de operación de los procesos productivos.

Dentro de la MDPR ha habido varios intentos para resolver problemas de parámetros robustos en optimización por medio de simulación. Por ejemplo: (Ittiwattana, 2002) propuso un algoritmo genético para encontrar aplicaciones en el proceso de ingeniería del diseño. (Wild y Pignatiello, 1991) presentaron una estrategia para diseñar sistemas robustos basados en arreglos cruzados. Enfatizaron la utilidad de su enfoque en ambientes dinámicos donde existe incertidumbre y su demanda de confiabilidad es elevada.

(Kleijnen et al., 2003) demuestra la idea de robustez al simular 3 diferentes configuraciones de cadenas de suministro. Su metodología consiste en iniciar con una bifurcación secuencial para identificar factores importantes, seguido por clasificar los factores controlables y no controlables. (Sohn, 2002) uso una simulación Monte-Carlo para encontrar los niveles robustos de las características de un servidor para una cola M/M/1. Asumio que las tasas de llegada y de servicio son factores de ruido parcialmente aleatorios; y que la respuesta a optimizar es el ECM (Error Cuadrático Medio) del parámetro de intensidad de tránsito. (Mayers y Benjamin, 1992) usan metodologías de superficies de respuesta, pero con experimentos simulados en lugar de experimentos físicos para determinar la dirección de búsqueda que de un diseño de manufactura robusto.

Enfoques hacia la optimización de simulaciones, no necesariamente para optimización robusta han sido estudiados por diferentes autores, otros ejemplos son: (Ho et al., 2000); (Hurrion, 1997); (Yeomans, 2002). Finalmente, (Benjamin y Erraguntla, 1995) propusieron una manera heurística para resolver un problema robusto basado en una formulación con doble criterio. El AAEPS, sin embargo, no ha sido explorado completamente en problemas de DPR para optimización por medio de simulación, por lo que es nuestra meta principal en esta tesis, evaluando el Índice de

Capacidad Real del Proceso C_{pk} como una medida de razón señal a ruido dentro de este algoritmo.

La razón del índice real de capacidad del proceso C_{pk} presenta una correlación positiva con respecto a la aptitud o capacidad del proceso para cumplir con las especificaciones, es decir, a mayor magnitud del C_{pk} mayor es el porcentaje de la producción considerada como satisfactoria. Lo anterior se justifica por el hecho de que un crecimiento en el valor del C_{pk} equivale a un incremento en la probabilidad de generar productos que cumplan con las especificaciones. Por lo que, para mejorar la calidad en el producto, se deben buscar las condiciones óptimas de operación de los procesos en las que el C_{pk} alcance su valor máximo o exceda a un valor específico M ($M \in \mathfrak{R}^+ : \mathfrak{R}^+$ representa al conjunto de números reales positivos).

1.2. Planteamiento del Problema

Esta tesis presenta un algoritmo modificado combinando los AAEPS y la MDPR. Dado que, los AAEPS han sido reconocidos por (Beyer y Sendhorff, 2007) como un enfoque eficiente para la optimización de DPR. Sin embargo, aun no se ha encontrado literatura publicada que lo demuestre, esto referente ha su comportamiento con un número “pequeño” de iteraciones como lo mencionan (Miranda y Del Castillo, 2011); no se ha encontrado una teoría sobre su desempeño en las “primeras” iteraciones (García, 2000); no cuentan con un criterio teórico de terminación; es decir, no se establece ninguna condición que debe cumplirse para detener su proceso iterativo (Andradottir, 1999).

Por otro lado, la MDPR tiene como objetivo inmunizar a los procesos y productos de los factores de ruido, que resultan incontrolables en el uso, pero no durante la experimentación (Lagunes y Sánchez, 2011). (Taguchi, 1986) propone un arreglo ortogonal entre un diseño que contenga los factores controlables y otro diseño con los factores de ruido, el resultado es una matriz de las combinaciones entre los factores controlables y los factores de ruido, para la robustez de procesos.

El Algoritmo modificado en esta investigación, se implementa evaluando el índice de capacidad real de procesos como una medida de razón señal a ruido, la cual cuantifica la capacidad o aptitud para cumplir con las especificaciones y para determinar las condiciones óptimas de operación de los procesos productivos. El disponer de este indicador permite decidir entre, seguir operando en las condiciones actuales de operación o proceder a buscar, con algún método, nuevas condiciones de operación en las que se mejore dicho índice hasta alcanzar un valor específico.

El problema planteado es que el valor de la variable de respuesta asume que lo mayor es lo mejor, para lo cual se hace necesario conocer el valor del Límite de Especificación Superior LES . Suponga que el símbolo C_{pk} representa al Índice de la Capacidad de un Proceso productivo en el que cada uno de sus d factores controlables, se encuentran operando en los niveles o valores (condiciones de operación) denotados por el vector $X = (X_1, \dots, X_d) \in \mathfrak{R}^d$ y sus factores de ruido, denotados por el vector $Z = (Z_1, \dots, Z_l) \in \mathfrak{R}^l$ donde \mathfrak{R}^{d+l} representan al espacio cartesiano de dimensión $d+l$. Entonces se puede asegurar que existe una relación funcional entre:

$$C_{pk} = f(X, Z; LES) \quad (1.1)$$

Dado el valor del LES , conocer el valor del C_{pk} al evaluarlo en las condiciones de operación (X, Z) . Para lograr lo anterior, son necesarios los valores esperados de (μ, σ^2) , donde $\mu: \mathfrak{R}^{d+l} \rightarrow \mathfrak{R}$ y $\sigma^2: \mathfrak{R}^{d+l} \rightarrow \mathfrak{R}$ son funciones reales (\mathfrak{R} representa al conjunto de números reales).

En la mayoría de los casos prácticos no es posible determinar los valores de los parámetros μ y σ^2 de la variable de respuesta en las condiciones de operación (X, Z) (Garcia, 2000), lo que trae como consecuencia que el valor del C_{pk} tampoco se logre especificar. Sin embargo, este valor puede ser estimado a través del estadístico \hat{C}_{pk} , el cual representa el valor de una variable aleatoria con distribución y

varianza desconocida, pero condicionado a que su valor esperado sea igual a la función $f(X, Z; LES)$, es decir:

$$C_{pk} = E(\hat{C}_{pk} | X, Z; LES) \quad (1.2)$$

Lo que indica que él C_{pk} puede ser expresado como el valor esperado de la variable aleatoria \hat{C}_{pk} , dado el vector X, Z , y el valor del LES . Suponga, además, que el estadístico \hat{C}_{pk} puede ser expresado mediante la ecuación:

$$\hat{C}_{pk} = f(X, Z; LES) + \varepsilon \quad (1.3)$$

En la que f representa a una función real cuya expresión algebraica es desconocida y ε corresponde al valor de una variable aleatoria. Por lo tanto, Este trabajo plantea resolver el siguiente problema:

$$\text{Maximizar } C_{pk} = f(X, Z; LES) \quad (1.4)$$

Para determinar la solución al problema presentado en la ecuación (1.4) es necesario optimizar simultáneamente los valores de la media $\mu(X, Z)$ y de la desviación estándar $\sigma(X, Z)$ de la variable de respuesta, lo que se lleva a cabo dependiendo de lo que resulta mejor para el valor de la característica de calidad o variable de respuesta que puede resultar en el caso de lo mayor es lo mejor. En este caso se requiere que la media $\mu(X, Z)$ sea lo más grande posible, mientras que la desviación estándar $\sigma(X, Z)$ sea el mínimo.

El método que se implemente para resolver el problema planteado en la ecuación (1.4) deberá poseer las siguientes características deseables (Sánchez, 1991) y (García, 2000): 1) *ser iterativo y recursivo*, que cada nueva solución se genere a partir de la anterior; 2) *eficiente*, que requiera la menor cantidad posible de observaciones de la variable de respuesta para determinar la solución óptima.

De los métodos experimentales señalados al inicio de este capítulo, ninguno posee, simultáneamente, estas dos características deseables: 1) *ser iterativo y recursivo*; 2) *ser eficiente*. Por ejemplo, la Metodología de Taguchi no es un proceso iterativo; la

Metodología de Superficies de Respuesta Dual, el Método Lagrangiano para restricciones con desigualdades y el Método de Optimización de Superficies de Respuesta Dual mediante un Método de Modelación Difusa requieren ajustar modelos cuadráticos, de lo que puede resultar que su implantación sea impráctica si el número de factores controlables es mayor o igual que cuatro ($d \geq 4$).

Los Algoritmos de Aproximaciones Estocásticas, en particular los que usan el Método de Perturbación Simultánea, son los mejores métodos para resolver el problema de mejoramiento de la calidad en los sistemas productivos (Chin, 1997), dado que estos métodos experimentales cuentan con la mayoría de las características deseables, planteadas en el párrafo anterior.

1.3. Objetivos

1.3.1. Objetivo General

Desarrollar un algoritmo de aproximaciones estocásticas con perturbación simultánea robusto, el cual debe tener la característica de ser efectivo en el mejoramiento de los procesos, en los que el valor de la variable de respuesta es una función (desconocida) de un conjunto de factores controlables y factores de ruido.

1.3.2. Objetivos específicos:

1.3.2.1. Verificar que la efectividad del AAEPSM es independiente de las condiciones iniciales de operación X_0 en las “primeras” iteraciones;

1.3.2.2. Validar que el AAEPSM logra mejorar el valor del C_{pk} en un 50% de sus valores iniciales a sus valores óptimos con factores de ruido.

1.3.2.3. Verificar que la eficacia del AAEPSM es afectada por el valor del volumen del diseño factorial 2^k , considerado como factor de ruido. Donde, a los valores de la distancia que existe entre los puntos centrales y los vértices del diseño factorial 2^k , lo llamaremos lo que resta de la tesis como “volumen del diseño factorial 2^k ”.

1.4. Preguntas de investigación

¿Es posible que la efectividad del AAEPSM sea independiente de las condiciones iniciales de operación en las “primeras” iteraciones?

¿Es posible que el valor del C_{pk} mejore en un 50% con el AAEPSM de sus valores iniciales a sus valores óptimos?

¿Cómo se afecta la eficacia del AAEPSM por el volumen del diseño factorial 2^k , considerado como factor de ruido?

1.5. Hipótesis

Las hipótesis que dan respuesta a las preguntas de investigación, son:

1.5.1. La efectividad del AAEPSM es independiente de las condiciones iniciales de operación X_0 en las “primeras” diez iteraciones.

1.5.2. El AAEPSM logra mejorar el valor del C_{pk} en un 50% con factores de ruido.

1.5.3. La eficacia del AAEPSM es afectada por el volumen del diseño factorial 2^k , considerado como factor de ruido.

1.6. Justificación

El desarrollo de esta tesis proporciona una extensión de los AAE, en particular los que usan el MPS, referente a que no se ha encontrado trabajos en cuanto a su comportamiento con factores de ruido con un número “pequeño” de iteraciones (Miranda y Del Castillo, 2011); no se ha encontrado una teoría sobre su desempeño en las “primeras” iteraciones (García, 2000); y no cuentan con un criterio teórico de terminación; es decir, no se establece ninguna condición que debe cumplirse para detener su proceso iterativo (Andradóttir, 1996).

Este algoritmo presenta las siguientes ventajas:

1. *Es simple*, dado que se requiere de poco conocimiento sobre el proceso, la verdadera relación funcional entre la variable de respuesta y el conjunto de factores controlables y factores de ruido;

2. *Es eficiente*, requiere únicamente dos observaciones de la variable de respuesta por iteración;

Estas ventajas traen como consecuencia que se generen ahorros, al reducir el esfuerzo para determinar las condiciones óptimas de operación, en relación con otras

herramientas que pueden ser usadas para este mismo propósito. (Sánchez, 1991; Fu y Hill, 1997; García, 2000)

El AAEPSM es desarrollado como una herramienta económica para optimizar los procesos productivos, en relación con otras técnicas de optimización, puesto que requiere de menor cantidad de observaciones de la variable de respuesta en cada iteración; con lo que se evita el desperdicio de materiales originado por producir en condiciones de operación experimental, además se reduce el tiempo necesario para llevar a cabo la experimentación, lo que trae como consecuencia reducción de costos por concepto de implantación.

El AAEPSM evita el desperdicio de recursos, puesto que en caso de que éste no logre mejorar el valor del C_{pk} , es posible disponer de la información recopilada como el desarrollo o parte de un diseño de experimentos 2^d , cuyos resultados pueden ser analizados a través de un Análisis de Varianza (ANOVA) o usados para construir un Diseño Central Compuesto (DCC), mismo con el que se puede ajustar un modelo cuadrático útil en algún otro método de optimización dual.

1.7. Limitaciones

El algoritmo propuesto tiene la condición que en el proceso de simulación, la variable de respuesta, los factores controlables y de ruido son de naturaleza continua. Sin embargo, en la práctica puede ocurrir que estos factores sean de naturaleza discreta o que las escalas en los controles de los procesos estén expresadas en números enteros.

La verificación y validación de las hipótesis se lleva a cabo a través de simulación empleando un conjunto de funciones de prueba de dos variables, mismas que constituyen el universo en que son válidos los resultados que se obtienen en este trabajo.

El proceso de simulación que se lleva a cabo para efecto de verificar algunas de las hipótesis planteadas en esta tesis, es completamente diferente a la forma en que se lleva a cabo la implantación del AAEPSM en los casos prácticos, en el sentido de que en la simulación es posible, seleccionar arbitrariamente condiciones iniciales de

operación, ejecutar una cantidad “grande” de iteraciones o cambiar de valores en los parámetros.

1.8. Organización de la tesis

En el Capítulo 2, se presenta una breve revisión sobre los diferentes métodos que pueden ser utilizados para maximizar el C_{pk} y resolver el problema planteado en la expresión (1.4), así como, el AAEPSM. En el Capítulo 3 se presentan los materiales y métodos que son usados para lograr los objetivos planteados en esta tesis. En el capítulo 4 se presenta el análisis y discusión de los resultados que se obtienen de implementar el algoritmo modificado a través de simulación, para verificar y validar que éste logra mejorar el valor del C_{pk} en un 50% observado en las condiciones iniciales de operación. En el Capítulo 5 se presentan las conclusiones y futuras investigaciones generales de este trabajo.

Capítulo II. Literatura revisada

En este capítulo se presentan las definiciones, fundamentos teóricos implementados en los procesos que producen bienes o servicios y los métodos experimentales útiles para la mejora de éstos.

2.1. Operación Evolutiva.

El método de Operación Evolutiva (EVOP) es propuesta por Box en 1957, como un procedimiento de operación para el monitoreo y mejoramiento continuo de un proceso, cuyo objetivo consiste en introducir de manera sistemática pequeños cambios en los niveles de las factores controlables, con la aplicación secuencial de un diseño factorial de dos niveles (2^k).

Dentro del diseño factorial 2^k para el método EVOP, se deben recolectar datos de las variables de respuesta de interés en cada punto del diseño factorial, esto para calcular los efectos y las interacciones de las variables del proceso. Una vez recolectados los datos en cada punto del diseño, se dice que se ha completado un ciclo. Después de varios ciclos, el efecto de una o más variables del proceso o sus interacciones sobre la respuesta serán significativos.

Para probar la significación de las variables e interacciones del proceso, se necesita una estimación del error experimental. Ésta se calcula a partir de los datos del ciclo. Además, el diseño factorial 2^k se centra por lo general en torno a las mejores condiciones de operación actuales. Mediante la comparación de la respuesta en este punto con los puntos de la porción factorial, es posible verificar la curvatura o cambio en la media (CIM, por sus siglas en inglés); es decir, si el proceso en realidad se centra en el valor máximo de la variable de respuesta, por ejemplo, entonces la respuesta en el centro deberá ser significativamente mayor que las respuestas en los puntos periféricos del diseño factorial 2^k . En este punto se debe tomar una decisión para modificar las condiciones de operación iniciales a fin de mejorar la variable de respuesta. Una vez mejoradas las condiciones de operación, se dice que se ha terminado una fase. (Montgomery, 2004)

Las ventajas para el Box – OPEV que menciona (Lowe, 1974). Primero, el tipo de diseño es fácil de entender y fácil de aplicar. Segundo, la recolección de datos se puede completar en un periodo de tiempo. Tercero, los diseños factoriales de dos niveles se pueden utilizar para variables cualitativas o cuantitativas. Cuarto, permite la comparación con el proceso estándar para que el valor y costo del programa OPEV pueda ser estimado. Quinto, al comparar las magnitudes relativas de los efectos principales con el cambio de la media y los efectos de interacción es posible obtener una idea de la complejidad de la superficie de respuesta.

(Kenworthy, 1967) y (Lowe, 1974), mencionan que la técnica Box-OPEV tiene varias desventajas. Primero, el número de factores estudiados no deben de exceder de tres, porque los cálculos con tres o más factores se vuelven complejos. La siguiente desventaja es que no hay indicación del número de corridas requeridas para el vértice de cada iteración. Las observaciones son tomadas hasta que la significancia es lograda o hasta cuando uno decide que la significancia no puede ser lograda. También, si la significancia es lograda, el procedimiento no ofrece una guía hacia donde moverse a la siguiente iteración. Finalmente, la determinación de los efectos del factor puede depender en la magnitud de la diferencia entre los niveles del factor. Por lo tanto, es difícil hacer que el procedimiento OPEV sea automatizado o computarizado.

2.2. Metodología de Superficie de Respuesta

La Metodología de Superficie de Respuesta (MSR) es una colección de técnicas matemáticas y estadísticas utilizadas para determinar los niveles óptimos de las variables independientes de un proceso productivo, la cual consiste en estimar un modelo de regresión de primer orden por medio del método de mínimos cuadrados como se muestra en la ecuación 2.1; con los coeficientes de este modelo se establece la dirección de búsqueda por el Método de Máxima/Mínima Pendiente Ascendente/Descendente (MMPA/D), posteriormente, se elige el tamaño del paso sobre la ruta de ascenso (Montgomery, 2005), hasta que no exista ningún incremento adicional en la respuesta, se detiene este método (kleijnen, 2008). Después, ajustamos un nuevo modelo de regresión lineal, se determina un nuevo camino de pendiente ascendente o descendente y el procedimiento continúa hasta que no ajusta el modelo de regresión de primer orden. Por último, se inicia en la región donde no fue posible

ajustar el modelo de regresión de primer orden, se plantea un diseño más detallado, como el Diseño Central Compuesto (DCC), el cual es el tipo de diseño clásico para ajustar modelos de segundo orden, como se muestra en la ecuación 2.2, y encontrar los valores óptimos de las variables independientes analizadas, empleando métodos de cálculo diferencial. (Box, 1999), (Khuri, 1996), (Khuri y Cornell, 1996), (Myers, 1999), (Myers et al, 2009), (Lagunes y Sánchez, 2011). La MSR también ha sido aplicada a modelos de simulación aleatorios, véase la literatura incluida en (Donohue et. al., 1993, 1995), (Hood y Welch, 1993), (Irizarry et. al., 2001), (Kleijnen, 1998, 2004, 2006), (Law y Kelton, 2000), (Neddermeijer et. al., 2000), y (Safizadeh, 2002).

$$\hat{y} = \beta_0 + \sum_{i=1}^k \beta_i x_i + \varepsilon \quad (2.1)$$

$$\hat{y} = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i=1}^k \beta_{ii} x_i^2 + \sum_{i < j}^k \sum_{j=1}^k \beta_{ij} x_i x_j + \varepsilon \quad (2.2)$$

Donde \hat{y} representa la característica de calidad a optimizar, β_i representan los coeficientes de regresión, x_{ij} representan las variables aleatorias y ε_i representa el error aleatorio.

2.3. Procedimientos de Aproximaciones Estocásticas.

El método de aproximaciones estocásticas presentado por (Robbins y Monro, 1951), es un método de búsqueda lineal de la raíz de la función desconocida $f : \mathfrak{R} \rightarrow \mathfrak{R}$ que representa al valor esperado de una variable aleatoria. (Kiefer y Wolfowitz, 1952) lo modifican para que pueda ser usado en la determinación del óptimo de f . (Blum, 1954) extiende los resultados de los autores anteriores a espacios cartesianos de dimensión mayor que 1.

A partir del trabajo presentado por (Blum, 1954) se da un incremento en la cantidad de métodos de aproximaciones estocásticas (Kushner y Clark, 1978; Polyak, 1991; Polyak y Juditsky, 1992; Andradóttir, 1995; Delyon, 1996; Kulkarni y Horn, 1996; Maeda, 1996; etc). Pero (Andradóttir, 1996), asegura que todos estos métodos son procedimientos sin un criterio teórico de terminación, que son usados para determinar X^* en \mathfrak{R}^d , de tal forma que $h(X^*) = \mathbf{0}$, donde $h : \mathfrak{R}^d \rightarrow \mathfrak{R}^d$ es la función que

corresponde al vector gradiente de la función f , de la cual se desconoce su expresión analítica, pero es posible cuantificar su valor para una combinación específica de valores o niveles de los factores controlables, medición que está sujeta a un error experimental, del que no se establece ningún supuesto en cuanto a su distribución de probabilidad.

(Chin, 1997) clasifica los procedimientos de aproximaciones estocásticas en dos tipos: *del tipo Robbins-Monro* y *del tipo Kiefer-Wolfowitz*. Los primeros se caracterizan por requerir las observaciones directas de h , dentro de los cuales se encuentran (los métodos de Robbins-Monro, pasos ascendentes/descendentes, Newton-Raphson, análisis de perturbación y tasa de verosimilitud), mientras que los segundos requieren estimaciones o aproximaciones de h como lo son (Kiefer-Wolfowitz, diferencias finitas, método de perturbación simultánea). Estos últimos los considera más útiles, dado que no requieren un conocimiento profundo del sistema a optimizar. Es decir, son aplicables en situaciones en las que se desconoce la relación funcional entre la variable de respuesta y los factores controlables, situación que se presenta con mayor frecuencia en la práctica.

(Andradóttir, 1996) generaliza los métodos de aproximaciones estocásticas de acuerdo al siguiente procedimiento: dada una sucesión $\{a_k\}$ de números reales, es posible generar una sucesión de vectores aleatorios $\{X_k\}$ de la que su límite converge en forma asintótica hacia X^* de acuerdo al procedimiento siguiente:

Paso 1. Seleccionar X_1 en \mathfrak{R}^d y considerar $k = 1$;

Paso 2. Dados X_1, X_2, \dots, X_k obtener una estimación Y_k de $h(X_k)$;

Paso 3. Calcular $X_{k+1} = X_k + a_k Y_k$;

Con a_k satisfaciendo las siguientes condiciones, mostradas en las ecuaciones (2.3):

$$a_k > 0, \quad \sum_{k=1}^{\infty} a_k = \infty \quad \text{y} \quad \sum_{k=1}^{\infty} a_k^2 < \infty ; \quad (2.3)$$

Paso 4. Considerar $k = k + 1$ y regresar al paso dos.

Los métodos o procedimientos de aproximaciones estocásticas pueden considerarse como la versión aleatoria del método de optimización de Newton-Raphson. Es decir, el primer método es usado cuando no es posible conocer la función entre la variable

de respuesta y los factores controlables del proceso a optimizar; mientras que con el segundo método, es necesario conocer el vector gradiente de dicha función. A continuación se presenta el segundo método con objeto de mostrar la similitud entre éste y el procedimiento descrito en esta sección.

El método de Newton-Raphson es propuesto para resolver el modelo presentado en la ecuación (2.4)

$$\text{Minimizar } f : \mathfrak{R}^n \rightarrow \mathfrak{R} \quad (2.4)$$

Donde f es una función real, de la cual se conoce su expresión analítica. Considere que el vector gradiente de f evaluado en el vector $X_k \in \mathfrak{R}^d$ está definido por $\nabla f(X_k) = [\partial_{x_i} f(X_k)]$, para $i = 1, 2, \dots, d$, y su matriz Hessiana dada por $H(X_k) = \left[\frac{\partial^2 f(X_k)}{\partial x_i \partial x_j} \right]$. $H(X_k)^{-1}$ denota la matriz inversa de $H(X_k)$.

Suponga que f es doblemente derivable y que existe un $X^* \in \mathfrak{R}^d$ tal que $\nabla f(X^*) = \mathbf{0}$ y que $H(X_k)^{-1}$ existe, y que el punto X_1 está “suficientemente cerca” de X^* ; donde suficientemente cerca significa que existen números reales k_1 y k_2 mayores que cero con $k_1 k_2 \|X_1 - X^*\| \leq 1$, $\|H(X_k)^{-1}\| \leq k_1$.

Y que $\|\nabla f(X^*) - \nabla f(X_k) - H(X_k)(X^* - X_k)\| \leq k_2 \|X^* - X_k\|^2$, además para todo $X_k \in \mathfrak{R}^d$ se satisface que $\|X_k - X^*\| \leq \|X_1 - X^*\|$ ($\|X\|$ denota la norma euclidiana del vector X).

Si se cumplen las condiciones establecidas en los párrafos anteriores, entonces, el método de Newton es un proceso iterativo útil para minimizar f y éste opera de acuerdo a los siguientes pasos:

Paso 1. Tomar a X_1 suficientemente cerca del mínimo de f como el punto inicial.

Paso 2. Evaluar $\nabla f(X_k)$ y $H(X_k)^{-1}$, si $\|\nabla f(X_k)\| \leq \varepsilon$ o si $\|X_k - X_{k-1}\| \leq \varepsilon$ entonces detener el proceso ($\varepsilon \geq 0$). De otra forma ir al paso 3.

Paso 3. Evaluar $X_{k+1} = X_k - H(X_k)^{-1} \nabla f(X_k)$, sustituir X_{k+1} por X_k y regresar al paso 2. Garantizar la existencia del inverso de la matriz Hessiana de f . Para resolver las restricciones mencionadas, se desarrolla el método modificado de Newton, el cual

no establece condición alguna para el punto inicial, para lo que se propone la ecuación iterativa (2.5):

$$X_{k+1} = X_k + \hat{\lambda}d \quad (2.5)$$

$\hat{\lambda}$ es una solución óptima del problema minimizar $f(X_k + \lambda d)$ con $\lambda \geq 0$. Además $d = -B\nabla f(x_k)$, B representa una matriz simétrica definida por $B = (\alpha I + H)^{-1}$, donde I representa la matriz identidad del mismo orden que la matriz H , $H = H(X_k)$. El valor de α se determina de acuerdo al siguiente procedimiento: se fija un número real $\delta > 0$, y considere $\alpha \geq 0$ como el menor escalar que hace a todos los eigenvalores de la matriz $(\alpha I + H)$ mayores o iguales a δ , con lo cual se garantiza que B es una matriz definida positiva.

Para los métodos de aproximaciones estocásticas, (Chen, 1985) señala que si $\sum_{k=1}^{\infty} a_k < \infty$, entonces el procedimiento descrito en esta sección converge rápidamente, pero el límite puede diferir de X^* si el punto inicial X_0 está suficientemente lejos de X^* . Dado que si $Y_k(X_k)$ es acotado entonces la norma euclidiana $\|X_{k+1} - X_k\| < \omega$ ($\omega \in \mathfrak{R}^+$) es también acotada, lo que implica que el procedimiento puede no converger si el punto inicial X_0 está “suficientemente” lejos del óptimo X^* , es decir, a una distancia mayor que ω .

(Andradóttir, 1996) establece que los procedimientos o métodos de aproximaciones estocásticas son generalmente empleados para determinar el mínimo de la función f , para lo que precisa de un método que le permita estimar el vector gradiente Y_k . Lo anterior implica que estos procedimientos corresponden a métodos de pasos descendentes, de tal forma que si X_k es una estimación actual de la solución, entonces X_{k+1} se obtiene en la dirección descrita por $-Y_k$; es decir, sobre la ruta en la que se espera que el valor de f decrezca más rápidamente.

La **longitud de los pasos** o **distancia de separación hacia la nueva solución generada** por los algoritmos de aproximaciones estocásticas está determinada por $a_k \|Y_k\|$, donde $\|Y_k\|$ representa la norma euclidiana de Y_k , razón por la que la rapidez o tasa de convergencia del procedimiento depende del valor a_k . (Chung, 1954), (Sacks, 1958) y (Fabian, 1968) establecen las consideraciones sobre a_k de tal forma que $k^{x/2}(X_k - X^*)$ converja asintóticamente hacia una variable aleatoria con distribución normal, de la que es posible controlar su varianza asintótica, de la que Sacks (1958) señala que se minimiza tomando a_k como el valor inverso del Jacobiano de $h(X^*)$ multiplicado por $1/k$. Mientras (Delyon, 1996) establece que el valor que asume típicamente para a_k es $1/k$ o $1/k^{2/3}$ cuando el vector gradiente es igual a la media aritmética de m estimaciones.

Para garantizar su convergencia en los procedimientos de aproximaciones estocásticas del tipo Kiefer-Wolfowitz, además de la sucesión $\{a_k\}$, también se necesita definir la sucesión decreciente de números reales no negativos $\{c_k\}$ que junto con la sucesión anterior satisfacen las siguientes condiciones que se muestran en la ecuación (2.6):

$$\lim_{k \rightarrow \infty} c_k = 0; \quad \sum_{k=1}^{\infty} a_k c_k < \infty; \quad \sum_{k=1}^{\infty} \frac{a_k^2}{c_k^2} < \infty \quad \text{y} \quad \sum_{k=1}^{\infty} a_k = \infty \quad (2.6)$$

El desempeño del procedimiento de aproximaciones estocásticas depende de la forma en que se definan $\{a_k\}$ y $\{c_k\}$, de las cuales (Sadegh y Spall, 1998), proponen que éstas deben establecerse de acuerdo al siguiente criterio, mostrado en las ecuaciones (2.7) y (2.8):

$$a_k = \frac{a}{k^\alpha} \quad (2.7)$$

$$c_k = \frac{1}{k^\gamma} \quad (2.8)$$

Donde $a > 0$, $0 < \alpha \leq 1$, $\gamma > 0$, $\alpha - 2\gamma > 0$, y $3\gamma - \alpha/2 \geq 0$, con lo que se garantiza el cumplimiento de las condiciones establecidas en la ecuación (2.6). Para

ver ejemplos numéricos en los que se muestra la dependencia del desempeño de este método con respecto a combinaciones de diferentes tipos de estas sucesiones, consultar los siguientes trabajos: (Spall, 1988, 1998), (L'Ecuyer et al., 1994), (Andradóttir, 1996).

2.3.1. Método de Perturbación Simultánea.

(Fu y Hill, 1997) señalan que los métodos del tipo Kiefer-Wolfowitz que usan el método de perturbación simultánea presentan las siguientes ventajas sobre los del tipo Robbins-Monro:

1. **Generalidad**, es aplicable en cualquier sistema que pueda ser simulado;
2. **Eficiencia**, requiere de menos observaciones de la variable de respuesta para lograr los mismos o mejores resultados en comparación contra cualquier otro método para estimar el vector gradiente;
3. **Fácil de usar**, es tan sencillo de aplicar como lo es el método de diferencias finitas.

En ese mismo trabajo, muestran, mediante una comparación numérica, la viabilidad de su implantación en tres diferentes sistemas de eventos de teoría de colas, en los que se manifiesta la ventaja de este método sobre el método de diferencias finitas en relación con el número requerido de observaciones de la variable de respuesta.

El Método de Perturbación Simultánea (Spall, 1992) calcula \mathbf{Y}_k de acuerdo a los siguientes pasos:

Paso 1. Construir una sucesión de vectores $\{\Delta_k\}$ ($\Delta_k \in \mathfrak{R}^d$), de d variables aleatorias idénticamente distribuidas y mutuamente independientes $(\Delta_1, \dots, \Delta_d) = \Delta_k$, del que (Sadegh y Spall, 1998), establecen que la selección óptima de la distribución de probabilidad para cada uno de sus componentes es la distribución Bernoulli con valores simétricos con respecto al cero. Con; $|\Delta_{ik}| < \delta$ casi donde quiera y $E|\Delta_{ik}^{-1}| \leq \alpha_\theta$ casi donde quiera, ($i = 1, 2, \dots, d$); δ y α_θ representan números reales positivos;

Paso 2. Considere la sucesión $\{c_k\}$ de números reales positivos prevista en (2.8), la cual puede considerarse constante e igual a uno para toda k en la evaluación de las ecuaciones (2.9) y (2.10) (Sadegh y Spall, 1998), y obtenga

$$Y_k^+ = f(X_k + c_k \Delta_k) + \varepsilon^+ \quad (2.9)$$

$$Y_k^- = f(X_k - c_k \Delta_k) + \varepsilon^- \quad (2.10)$$

Donde ε^+ y ε^- representan medidas de ruido (error experimental) generados sucesivamente independientes, respectivamente, para las observaciones Y_k^+ y Y_k^- . La estimación $Y(X_k)$ del vector gradiente $h(X_k)$ está definida por la ecuación (2.11):

$$Y_k(X_k) = \begin{bmatrix} \frac{Y_k^+ - Y_k^-}{2c_k \Delta_1} \\ \cdot \\ \cdot \\ \cdot \\ \frac{Y_k^+ - Y_k^-}{2c_k \Delta_d} \end{bmatrix} \quad (2.11)$$

2.3.1.1. Análisis del Desempeño del Método de Perturbación Simultánea.

La nueva condición de operación X_{k+1} generada con el procedimiento descrito en esta sección y con el método de perturbación simultánea como método para estimar el vector gradiente de f , dada la condición de operación anterior X_k , es expresada como se muestra en la ecuación (2.12). Donde $|\Delta_i|$ representa el valor absoluto de la i -ésima coordenada del vector Δ_k . El signo de cada componente, respectiva, del vector $D \in \mathfrak{R}^d$ y del vector Δ_k son iguales entre sí. D es seleccionado del conjunto de los posibles 2^d vectores cuyas componentes se forman, respectivamente, con la selección aleatoria en d ocasiones de uno de los dos valores ± 1 .

$$X_{k+1} = X_k + a_k \begin{pmatrix} \frac{Y_k^+ - Y_k^-}{2|\Delta_1|c_k} \\ \cdot \\ \cdot \\ \cdot \\ \frac{Y_k^+ - Y_k^-}{2|\Delta_d|c_k} \end{pmatrix} D \quad (2.12)$$

Lo anterior implica que X_{k+1} corresponde a una de las 2^d posibles combinaciones de tratamientos de un diseño experimental factorial con dos niveles, lo que a su vez implica que si se consideran todas las posibles opciones de D para calcular X_{k+1} de acuerdo con (2.12), entonces se está ejecutando un diseño de experimentos de la forma 2^d . Además, dada la aleatoriedad de X_{k+1} también se puede concluir que no es posible garantizar que el valor de la variable de respuesta que se obtiene en X_{k+1} $f(X_{k+1})$ es siempre “mejor” que el valor de $f(X_k)$.

Si la sucesión de vectores $\{X_k\}$ generada con el procedimiento descrito en esta sección y con el método de perturbación simultánea como estimador del vector gradiente de f , converge con probabilidad 1 hacia el vector X^* , entonces se puede asegurar con la misma probabilidad, que para todo número real positivo δ existe un número natural $N(\delta)$ que está en función de δ , tal que para todo X_m y X_n con m y $n \geq N(\delta)$ ocurre que $\|X_m - X_n\| < \delta$.

2.4. Métodos experimentales para el mejoramiento del C_{pk} .

A continuación se presentan los métodos de naturaleza experimental útiles para mejorar el C_{pk} , es decir, métodos que no requieren conocer la relación funcional que determina el valor promedio o la varianza de la variable de respuesta con respecto a los factores controlables, función que es afectada por un error experimental que representa el valor de una variable aleatoria independiente con distribución normal con media igual a cero y varianza finita. En los siguientes párrafos se presentan dichos métodos.

2.4.1 Método de Respuesta Dual de Vining y Myers.

El método de respuesta dual (Vining y Myers, 1990) consiste, básicamente, en maximizar la capacidad de los procesos productivos para producir dentro de las especificaciones a través del uso de los métodos lagrangianos para superficies de respuesta dual (Myers y Carter, 1973), métodos que son adaptados para resolver los modelos listados a continuación.

(Vining y Myers, 1990) proponen los modelos,

$$\begin{aligned} &\text{Minimizar } \hat{W}_\sigma && (2.13) \\ &\text{Sujeta a : } \hat{W}_\mu = T \end{aligned}$$

$$\begin{aligned} &\text{Minimizar } \hat{W}_\mu && (2.14) \\ &\text{Sujeta a : } \hat{W}_\sigma \leq T \end{aligned}$$

Los cuales pueden ser usados para resolver el problema de maximizar los índices de capacidad de los procesos productivos. Estos modelos se construyen de acuerdo a la preferencia que se le asigna al valor de la variable de respuesta; así para el caso en el que el valor nominal es lo mejor, proponen el modelo (2.13), mientras que para los dos casos restantes en los que lo más grande o lo más pequeño es lo mejor recurren al modelo (2.14).

Los modelos representados en las ecuaciones (2.13) y (2.14), \hat{W}_σ y \hat{W}_μ representan modelos ajustados de segundo orden, con los que se estiman, respectivamente, las expresiones que representan la relación funcional entre el conjunto de factores controlables y la desviación estándar o entre el valor promedio de la variable de respuesta; mismas que son ajustadas (usualmente) a través del método de mínimos cuadrados. T corresponde en la ecuación (2.13) al valor nominal para el valor promedio de la variable de respuesta y en la ecuación (2.14) al máximo valor permitido para la desviación estándar.

El método de (Vining y Myers, 1990) que propone los modelos (2.13) y (2.14), resuelve el problema de la falta de sustento estadístico de la metodología de Taguchi; metodología que tiene como objetivo optimizar la capacidad de producción dentro de las especificaciones, para lo que propone la técnica de maximizar alguna de las

funciones señal a ruido que para el efecto se diseñan; funciones que adicionalmente presentan el problema de determinar cuál de las relaciones señal a ruido seleccionar para modelar a un proceso productivo particular.

A pesar de que el método de (Vining y Myers, 1990) es un método de optimización dual sustentado por la teoría estadística, éste presenta las siguientes desventajas para su implantación en casos prácticos:

1. Considera que las funciones del valor promedio y de la varianza de la variable de respuesta se ajustan a funciones cuadráticas, lo cual no necesariamente ocurre en todo proceso; por lo que las soluciones generadas para los modelos representados en las ecuaciones (2.13) y (2.14) pueden no coincidir con la solución para el caso práctico;
2. La solución generada puede no corresponder a la región factible o valores de los factores controlables X ($X \in \mathfrak{R}^d$), ya que aquí se pide que $\|X\| \leq \rho$, $\rho > 0$ ($\|X\|$ representa la norma euclidiana del vector X), mientras que en la práctica usualmente ocurre que cada componente X_i ($i = 1, 2, \dots, d$) del vector X está en el intervalo de números reales $[a_i, b_i]$; es decir, en este método se restringe el dominio a una región esférica de radio ρ , cuando en la práctica, por lo general, este dominio corresponde a una región rectangular;
3. Requiere de la utilización de los métodos de Lagrange para determinar la solución de los modelos de las ecuaciones (2.13) y (2.14), lo que resulta complicado si no se cuenta con un software de matemáticas adecuado para el efecto;
4. La cantidad necesaria de mediciones de la variable de respuesta en diferentes condiciones de operación de los factores controlables depende del número de dichos factores; luego, para una cantidad grande ($d \geq 4$) de éstos resulta inapropiada su implantación, sobre todo en aquellos procesos en los que los cambios de sus condiciones de operación consumen “gran” cantidad de tiempo o dinero.

2.4.2 Método de Lin y Tu.

(Lin y Tu, 1995), ofrecen un método útil para mejorar el índice C_{pk} , con el que se resuelven las primeras tres desventajas del método anterior y eliminan la necesidad de emplear métodos lagrangianos en la optimización, donde esta eliminación la logran con la transformación del modelo dual (2.5 o 2.6) a un modelo de una sola variable. Para este efecto proponen minimizar el Error Cuadrático Medio (*ECM*) a través de los modelos

$$\text{Minimizar } ECM = (\hat{W}_\mu - T)^2 + \hat{W}_\sigma^2 \quad (2.15)$$

$$\text{Minimizar } ECM = \hat{W}_\mu^2 + \hat{W}_\sigma^2 \quad (2.16)$$

El modelo (2.15) se usa para optimizar procesos en los que lo nominal es lo mejor, mientras que el modelo (2.16) es utilizado en los casos en los que lo menor es lo mejor. Donde \hat{W}_σ y \hat{W}_μ representan, respectivamente, las superficies de respuesta del mejor orden ajustadas para la varianza y la media de la variable de respuesta, T es el valor objetivo para la media de la variable de respuesta.

Este método presenta las siguientes desventajas:

1. Su implantación se complica para un número grande de factores ($d \geq 4$). Es decir, no resuelve la cuarta desventaja del método anterior;
2. Su implantación se torna más complicada con respecto al método anterior, por el hecho de que este método requiere verificar la adecuación y buscar el mejor orden de los modelos que son usados en las ecuaciones (2.15) o (2.16), lo cual constituye una tarea especializada que requiere de personal altamente calificado en el manejo de la estadística;
3. No se puede implantar en los procesos en los que se considera que el valor más grande de la variable de respuesta es lo mejor; es decir, en procesos en los que se buscan las condiciones de operación en las que se logre el máximo valor de la variable de respuesta. Lo anterior se debe a la imposibilidad de definir un *ECM* adecuado para este caso, en el que se busca, simultáneamente,

maximizar μ y minimizar σ , condiciones que se contraponen al tratar de minimizar el *ECM*.

2.4.3 Método de Copeland y Nelson.

(Copeland y Nelson, 1996), proponen minimizar mediante la utilización del método simplex los modelos listados a continuación:

$$\begin{aligned} & \text{Minimizar } \hat{\sigma} \\ & \text{sujeta a : } (\hat{\mu} - T)^2 \leq \Delta^2 \end{aligned} \quad (2.17)$$

$$\begin{aligned} & \text{Minimizar } -\hat{\mu} \\ & \text{sujeta a : } \hat{\sigma} \leq \sigma_T \end{aligned} \quad (2.18)$$

$$\begin{aligned} & \text{Minimizar } \hat{\mu} \\ & \text{sujeta a : } \hat{\sigma} = \sigma_T \end{aligned} \quad (2.19)$$

El modelo (2.17) es usado para el caso en que el valor nominal de la variable de respuesta es lo mejor; los modelos (2.18) y (2.19) son utilizados, respectivamente, para situaciones en las que el valor mayor o menor es lo mejor. De la variable de respuesta, μ corresponde a su valor promedio, T a su valor objetivo y σ a su desviación estándar y de esta última, σ_T representa un valor constante.

Este es un método iterativo que no requiere ajustar modelo alguno, lo cual representa una ventaja sobre los dos métodos anteriores, pero conserva la cuarta desventaja del primer método mencionado; es decir, el número mínimo de observaciones de la variable de respuesta requerido en cada iteración es igual al número de factores controlables más 1.

2.4.4 Método de Kwang y Lin.

(Kwang y Lin, 1998), proponen el siguiente modelo:

$$\begin{aligned} & \text{Maximizar } \lambda \\ & \text{sujeta a :} \\ & m(\hat{W}_\mu)^2 \geq \lambda \\ & m(\hat{W}_\sigma)^2 \end{aligned} \quad (2.10)$$

Donde; $m: \mathfrak{R}^d \rightarrow [0,1]$ representa a una función real llamada función de membresía.

\hat{W}_μ y \hat{W}_σ simbolizan modelos de segundo orden ajustados con el método de mínimos

cuadrados, mismos que a su vez representan, respectivamente, las relaciones funcionales entre el valor promedio y la desviación estándar de la variable de respuesta con respecto a sus factores controlables.

La expresión $m(\hat{W}_\mu(X))$ es usada para medir, para el caso en que lo nominal es lo mejor, el grado con el que el valor de la expresión $\hat{W}_\mu(X)$ se aproxima al valor objetivo de μ (T). Asignándole un 1.0 cuando los valores $\hat{W}_\mu(X)$ y T son iguales entre sí y otra cantidad que tiende a cero; conforme la diferencia entre los valores de $\hat{W}_\mu(X)$ y T se incrementa, siendo esta cantidad igual a cero cuando $\hat{W}_\mu(X)$ excede o es igual al valor mínimo o máximo permitido para μ .

La expresión $m(\hat{W}_\sigma(X))$ es usada para medir, para el caso en que lo nominal es lo mejor, el grado con el que $\hat{W}_\sigma(X)$ se aproxima al valor objetivo de la desviación estándar (σ), con la que se asigna una cantidad igual a 1.0 cuando el valor de $\hat{W}_\sigma(X)$ es menor o igual que el valor mínimo permitido para (σ) y una cantidad que tiende a cero conforme el valor de $\hat{W}_\sigma(X)$ crece hacia el valor máximo permitido (σ_{\max}), siendo esta cantidad igual a cero cuando $\hat{W}_\sigma(X)$ es igual o excede a (σ_{\max}).

Este método tiene la ventaja de que logra un error cuadrático medio igual o inferior al de los métodos anteriores, pero presenta las siguientes desventajas: propone que los modelos \hat{W}_μ y \hat{W}_σ sean modelos cuadráticos ajustados mediante el método de mínimos cuadrados; el número de observaciones de la variable de respuesta, para cada condición de operación, depende del número de factores controlables; no establece la naturaleza de la función de membresía, la que depende de la naturaleza de cada proceso en particular.

2.5. Algoritmo de Aproximaciones Estocásticas con Perturbación Simultánea Modificado.

Considere que el AAEPSM es implantado en un proceso, en el que al evaluar el C_{pk} se refiere a la función que establece la relación entre las variables independientes

(factores controlables), así como, los factores incontrolables o de ruido y el límite de especificación superior $(X, Z; LES)$; Donde, el C_{pk} es una función desconocida, independientes entre sí, de un conjunto de factores controlables y factores de ruido.

Sean: Δ_k ($\Delta_k \in \mathfrak{R}^d$) una sucesión de vectores idéntica e independientemente distribuidos $\Delta_k = (\Delta_{k1}, \Delta_{k2}, \dots, \Delta_{kd})$, donde cada $\Delta_{ki} \in \mathfrak{R}$ con $i = 1, 2, \dots, d$ son variables aleatorias independientes con distribución Bernoulli simétricas con respecto al cero, para cada k ; $\{a_k\}$ y $\{c_k\}$ sucesiones de números reales que convergen a cero, de las que (Sadegh y Spall, 1998), recomiendan que deben construirse de acuerdo al siguiente criterio:

$$a_k = \frac{a}{k^\alpha}, c_k = \frac{1}{k^\gamma} \quad (2.21)$$

Con $a > 0$, $0 < \alpha \leq 1$, $\gamma > 0$, $\alpha - 2\gamma > 0$, $3\gamma - \alpha/2 \geq 0$, se garantiza el cumplimiento de los supuestos para estas dos sucesiones.

2.5.1. Procedimiento propuesto para el AAEPSM.

Paso 1. Considerar las condiciones iniciales de operación del proceso como el punto inicial representado por X_k, Z_l ; obtenido con una distribución uniforme. Con el módulo de *Calc: Random Data: Uniform*, del Software *Minitab*[®] 16.

Paso 2. Dadas las condiciones de operación $X_k, Z_l \in \mathfrak{R}$ $k, l = 1, 2, \dots$ obtener una estimación del valor promedio μ y la desviación estándar σ de la variable de respuesta C_{pk} , en la iteración k, l , mediante la variable Y_t , que representa el valor observado en la t -ésima ($t = 1, 2, \dots, n; n \geq 3$) medición de la variable de respuesta que se presenta en las condiciones de operación correspondientes a la iteración $k, l (X_k, Z_l)$, n representa el total de observaciones de la variable de respuesta en cada iteración.

Paso 3. Desarrollar un diseño cruzado con un arreglo interno que evalúa la $f(X)$ más la combinación de un diseño factorial 2^k como factor de ruido, evaluado con la $f(Z_l \pm v_i) + \varepsilon$, donde (v_i) representa el volumen del diseño factorial 2^k , para $i = 0.05, 0.1, 0.2, 0.3$ en sus variables naturales (X_1, X_2) y la suma de un error aleatorio representado por ε el cual es generado en forma aleatoria con una distribución normal con media cero y desviación estándar finita; (ver tabla 2.1)

Tabla 2.1. Matriz de Diseño Robusto para el AAEPSM

		Factores de Ruido					
		Z_1^\pm	-1	1	-1	1	0
		Z_2^\pm	-1	-1	1	1	0
Iteración	Factores Controlables X_k^\pm	$Y_t^\pm = f(X_k^\pm) + [f(Z_l^\pm \pm v_i) + \varepsilon]$					
1	$f(X_k^\pm)$						
.....						
100	$f(X_k^\pm)$						

Paso 4. Establezca las condiciones de operación:

$$(X_k^+, Z_l^+) = [(X_1, Z_1) + \Delta_{k1}c_k; (X_2, Z_2) + \Delta_{k2}c_k; \dots; (X_d, Z_l) + \Delta_{kd}c_k] \quad (2.22)$$

$$(X_k^-, Z_l^-) = [(X_1, Z_1) - \Delta_{k1}c_k; (X_2, Z_2) - \Delta_{k2}c_k; \dots; (X_d, Z_l) - \Delta_{kd}c_k] \quad (2.23)$$

Paso 5. Obtenga n mediciones o réplicas de la variable de respuesta para cada una de las condiciones de operación $[(X_k^+, Z_l^+), (X_k^-, Z_l^-)]$. Después use los resultados anteriores para estimar los valores del $\hat{C}_{pk}^{(+)}$ y el $\hat{C}_{pk}^{(-)}$ respectivos. Con estos valores a su vez estime el vector gradiente del $\frac{\partial \hat{C}_{pk}(X_k)}{\partial X}$ evaluada en X_k lo que se realiza a través de la ecuación (2.24):

$$\frac{\partial \hat{C}_{pk}(X_k)}{\partial X} = \begin{bmatrix} \frac{\hat{C}_{pk}^{(+)} - \hat{C}_{pk}^{(-)}}{2c_k \Delta_{k1}^{(\pm)}} \\ \bullet \\ \bullet \\ \bullet \\ \frac{\hat{C}_{pk}^{(+)} - \hat{C}_{pk}^{(-)}}{2c_k \Delta_{kd}^{(\pm)}} \end{bmatrix} \quad (2.24)$$

Paso 6. Calcular el nuevo valor para X_{k+1}

$$X_{k+1} = X_k - a_k \bullet \frac{\partial \hat{C}_{pk}(X_k)}{\partial X} \quad (2.25)$$

Paso 7. Considerar $X_{k+1} = X_k$ y regresar al paso 2.

2.5.2. Criterio de terminación.

Se plantea el criterio implementado por (García, 2000), donde, el algoritmo modificado garantiza que las condiciones de operación, que éste genera y las dos observaciones requeridas en cada iteración, se vuelve una herramienta adecuada para el mejoramiento de aquellos procesos que no estén restringidos.

Para efectos de la implantación del algoritmo modificado en los sistemas productivos, es necesario establecer un criterio de terminación; dado que la convergencia de la sucesión de vectores que se genera es una aproximación hacia la posible solución óptima cuando se realizan un número infinito de iteraciones, lo cual no es práctico. En esta tesis se establece un criterio para detener el algoritmo cuando éste es implantado en algún caso práctico o en algún proceso de simulación, dicho criterio es descrito a continuación.

Criterio: Si en las condiciones de operación (X_k, Z_l) , el \hat{C}_{pk} del proceso excede a una cantidad mínima aceptable M , entonces detener el método y declarar a (X_k^*) como las condiciones de operación requeridas o el valor máximo se logre en un intervalo de uno a cien iteraciones.

El valor de M se define en forma individual para cada proceso productivo, dado que este valor depende del nivel de porcentaje de producción dentro de las especificaciones requerido (PDE) o de la máxima cantidad de partes defectuosas por millón (PPM) negociadas con el cliente.

2.5.3. Algoritmo propuesto.

A continuación se describe la metodología para obtener la información del algoritmo modificado. Las entradas que requiere el algoritmo son: $X_0, Z; \{a_k\}; \{c_k\}; \Delta_k; s = 0; M$. Las salidas que se generan corresponden a condiciones de operación en las que el valor esperado del \hat{C}_{pk} es mayor o igual que el valor M o en las que se detiene el algoritmo con la declaración de imposibilidad de lograr mejoras en el valor del C_{pk} en las siguientes iteraciones. Se dice que el algoritmo converge cuando éste alcanza al conjunto de solución $S = \{X^* \in \mathfrak{R} : C_{pk}(X^*) \geq M\}$. Es decir, cuando mediante este algoritmo se identifican las condiciones de operación en las que el valor del índice de capacidad real del proceso es mayor o igual que M .

a1, elaborar una lista de los factores controlables significativos y sus factores de ruido, incluyendo sus rangos; denotados por el vector $X = (X_1, \dots, X_d) \in \mathfrak{R}^d$, $Z = (Z_1, \dots, Z_l) \in \mathfrak{R}^l$ donde \mathfrak{R}^{d+l} representa al espacio cartesiano de dimensión $d+l$. En esta tesis se está considerando que los factores controlables significativos utilizados en el algoritmo son generados aleatoriamente en treinta y tres parejas de puntos centrales para (X_1, X_2) , además estas mismas treinta y tres parejas de puntos centrales para el diseño factorial 2^k (Z_1, Z_2) , considerado como factor de ruido, de manera que estas variables siguen distribuciones independientes uniformes; las cuales serán usadas como representativas de las condiciones iniciales de operación (X, Z) de un proceso. Estos pares de valores iniciales son generados con el módulo *Calc: Random Data: Uniform del Software MINITAB® 16*.

a2, elaborar una lista de las variables dependientes (o de respuesta), sus unidades, denotadas por el vector $(n \times 1)$, observaciones $Y_t = (Y_1, \dots, Y_{d+l}) \in \mathfrak{R}^{d+l}$. Dentro del

proceso de simulación que se presenta en esta sección, la variable de respuesta a optimizar en esta tesis es el índice de capacidad real del proceso dado por: $C_{pk} = f(X, Z; LES)$, utilizado como una medida de razón señal a ruido. Con la que es posible, dado el valor del parámetro LES obtener el valor del C_{pk} al evaluarla en las condiciones de operación (X, Z) . Para lograr lo anterior, de acuerdo a la ecuación (2.26) son necesarios los valores de $\hat{\mu}$ y $\hat{\sigma}$, representados en las ecuaciones (2.27) y (2.28) respectivamente. Donde $\mu: \mathfrak{R}^{d+l} \rightarrow \mathfrak{R}$ y $\sigma: \mathfrak{R}^{d+l} \rightarrow \mathfrak{R}$ son funciones reales. (\mathfrak{R} representa al conjunto de números reales).

$$C_{pk} = \frac{LES - \bar{X}}{3S} \quad (2.26)$$

$$\hat{\mu} = \frac{\sum_{t=1}^n Y_t}{n} \quad (2.27)$$

$$\hat{\sigma} = \sqrt{\sum_{t=1}^n \frac{(Y_t - \hat{\mu})^2}{n-1}} \quad (2.28)$$

a3, implementar el algoritmo según (Spall J. C., 1998). Calcular el algoritmo de acuerdo a las modificaciones siguientes:

a3.1. Inicialización y coeficiente de selección. Seleccionar el índice contador $k = 1$. Tome un valor del vector gradiente inicial, los coeficientes de no negatividad a, c, A, α, γ . (Delyon, 1996) y (Spall, 2003) establecen que el valor que asume típicamente $a_k = 1/k^\alpha$, $c_k = c/(k+1)^\gamma$ son cuando el vector gradiente es igual a la media aritmética de m estimaciones. Valores prácticamente efectivos además de ser teóricamente válidos para α y γ son 0.602 y 0.101 respectivamente (los valores óptimos asintóticos de 1.0 y 1/6 pueden ser usados también); los valores de a, c, A ; Una guía útil al seleccionar A es hacerlo como si fuera mucho menor que el número máximo de iteraciones permitidas o esperadas, es por ello que se selecciona $A = 100, a = 0.16, c = 1$.

a3.2. Generar el Vector de Perturbación Simultánea. Es generado por el método de Montecarlo, un vector de perturbación aleatorio p-dimensional Δ_k , donde cada uno

de los p componentes de Δ_k son generados independientemente de una distribución de probabilidad con una media cero. Una simple (teóricamente válida) opción para cada uno de los componentes del Δ_k es usar una distribución Bernoulli ± 1 con probabilidad de $1/2$ para cada resultado ± 1 . Nótese que variables uniformes y normales aleatorias no son permitidas para los elementos del Δ_k por las condiciones regulares del algoritmo.

a3.3. Evaluar la función a optimizar. Dentro de la Metodología Taguchi (MT) la parte experimental se realiza empleando un diseño para los factores controlables, denominado arreglo interno, que es cruzado con otro diseño para los factores de ruido, llamado arreglo externo. Cada fila en el arreglo interno junto con cada columna del arreglo externo conforman las condiciones de cada uno de los experimentos que se realizan y los resultados de dichos experimentos se presentan en una matriz llamada arreglo o diseño cruzado (Cruz, et al., 2012). Por lo tanto, para este trabajo se desarrolla una estructura de un diseño cruzado, con un arreglo interior que evalúa la función en las condiciones iniciales X y el arreglo exterior por medio de un diseño factorial 2^k que evalúa la función de los factores de ruido Z más un error aleatorio ε , como se muestra en la tabla 2.1.

a3.3.a. Seleccionar los valores iniciales para los factores controlables (X_k) en cada una de las parejas de puntos iniciales obtenidas en **a1**.

a3.3.b. Realizar los experimentos evaluando la función $f(X_k)$ como arreglo interno del diseño cruzado.

a3.3.c. Realizar los experimentos evaluando los factores de ruido con la función $f(Z_l \pm v_i) + \varepsilon$, en base a un diseño factorial 2^k , con un punto central, como arreglo externo del diseño cruzado.

a3.3.d. Obtener las observaciones del experimento, representados como Y_t para cada iteración en base a los planteamientos **a3.3.b.** y **a3.3.c.** donde la $f(X_k)$ representa el resultado del arreglo interno más la $f(Z_l \pm v_i) + \varepsilon$ representa el resultado del arreglo

externo del diseño cruzado. Que en conjunto se obtiene la siguiente ecuación compuesta de $Y_t = f(X_k) + [f(Z_t \pm v_t) + \varepsilon]$.

a3.3.e. Obtener los valores tanto de la media como de la desviación muestral con las ecuaciones (2.27) y (2.28). Para estas ecuaciones se estima el valor promedio muestral ($\hat{\mu}$) y la desviación estándar muestral ($\hat{\sigma}$) de la variable de respuesta. La variable Y_t representa el valor observado en la t -ésima ($t = 1, 2, \dots, n; n \geq 3$) medición de la variable de respuesta que se presenta en las condiciones de operación correspondientes a (X, Z) , n representa el total de observaciones de la variable de respuesta.

a3.3.f. Calcular el índice de capacidad real del proceso C_{pk} . Sustituir los valores de $(\hat{\mu}; \hat{\sigma}; LES)$, en la ecuación (2.26) para calcular el índice de capacidad real del proceso C_{pk}^k , Donde: la fórmula (2.26), evalúa el caso de mayor es lo mejor de la variable de respuesta. $(\hat{\mu}; \hat{\sigma}; LES)$ denotan, respectivamente, el promedio, la desviación estándar muestral de la variable de respuesta y el límite superior de especificación del proceso.

a3.3.g. Obtener dos medidas de la función a optimizar $(\hat{C}_{pk}^{(+)}, \hat{C}_{pk}^{(-)})$ basadas en la perturbación simultánea a partir del valor actual C_{pk}^k , con $\{c_k\}$ y $\{\Delta_k\}$ de **a3.1** y **a3.2** respectivamente; utilizando la ecuación del paso 4 del procedimiento del algoritmo modificado para obtener $(C_{pk}^{(\pm)})$. Generar los valores de $(X_k^{(\pm)})$ con la ecuación (2.29).

$$X_k^{(\pm)} = X_k + c_k (\Delta_k^{(\pm)}) \quad (2.29)$$

a3.3.h. Después de calcular $(X_k^{(\pm)})$ se procede a evaluar de nuevo la función $f(X_k^{(\pm)})$ del arreglo interno del diseño cruzado.

a3.3.i. Realizar los experimentos evaluando los factores de ruido con los nuevos valores de $(Z_l^{(\pm)})$. Se sigue con el mismo planteamiento que en **a3.3.c**, solo que la función quedaría $f(Z_l^{(\pm)} \pm v_i) + \varepsilon^{(\pm)}$.

a3.3.j. Obtener las variables de respuesta para cada iteración en base a los planteamientos **a3.3.h** y **a3.3.i** de la siguiente manera:

$$Y_t^{(\pm)} = f(X_k^{(\pm)}) + [f(Z_l^{(\pm)} \pm v_i) + \varepsilon^{(\pm)}].$$

a3.3.k. Obtener los valores tanto de la media como de la desviación estándar muestral con las ecuaciones (2.27) y (2.28), para estas ecuaciones se estima el valor promedio muestral $(\hat{\mu}^{(\pm)})$ y la desviación estándar muestral $(\hat{\sigma}^{(\pm)})$ de la variable de respuesta. La variable $Y_t^{(\pm)}$ representa el valor observado en la t -ésima ($t = 1, 2, \dots, n; n \geq 3$) medición de la variable de respuesta que se presenta en las condiciones de operación correspondientes a $(X_k^{(\pm)}, Z_l^{(\pm)})$, n representa el total de observaciones de la variable de respuesta.

a3.3.l. Calcular la variable de respuesta a optimizar C_{pk} . Sustituir los valores de $(\hat{\mu}^{(\pm)}; \hat{\sigma}^{(\pm)}; LES)$, en la ecuación (2.26) para calcular el índice de capacidad real del proceso $(C_{pk}^{(\pm)})$. Donde: $(\hat{\mu}^{(\pm)}; \hat{\sigma}^{(\pm)}; LES)$ denotan, respectivamente, el promedio, la desviación estándar muestral de la variable de respuesta y el límite superior de especificación del proceso.

a4. Aproximar el Gradiente. Generar la aproximación de perturbación simultánea del gradiente desconocido como se muestra en la ecuación 2.30:

$$\frac{\partial \hat{C}_{pk}(X)}{\partial X} = \begin{bmatrix} \frac{\hat{C}_{pk}^{(+)} - \hat{C}_{pk}^{(-)}}{2c_k \Delta_{k1}^{(\pm)}} \\ \bullet \\ \bullet \\ \bullet \\ \frac{\hat{C}_{pk}^{(+)} - \hat{C}_{pk}^{(-)}}{2c_k \Delta_{kd}^{(\pm)}} \end{bmatrix} \quad (2.30)$$

Para obtener los resultados de la ecuación (2.30) se sustituyen los valores del $(C_{pk}^{(+)})$ y $(C_{pk}^{(-)})$, la ecuación de sucesión $c_k = 0.932$ y la perturbación simultanea $\{\Delta_k^\pm\}$ buscando la mejor combinación para maximizar el C_{pk} .

a5. Actualizar el valor de (X_k) estimado. Para actualizar el valor de (X_k) a un nuevo valor (X_{k+1}) se hace utilizando las fórmulas estándar del AAEPS como se muestra en la ecuación (2.31):

$$X_{k+1} = X_k - a_k \cdot \frac{\partial \hat{C}_{pk}(X_k)}{\partial X} \quad (2.31)$$

Aplicando los resultados obtenidos en los pasos anteriores, se sustituyen los valores en la ecuación (2.31) obteniendo los nuevos valores para (X_{k+1}) . Los resultados que se obtengan serán utilizados para iniciar la siguiente iteración del estudio en (C_{pk}^{k+1}) .

a6. Calcular el proceso iterativo o terminación. Regresar a la parte de **a3.2** con $k + 1$ reemplazando k . Terminar el algoritmo si se logra obtener un $\hat{C}_{pk} \geq 1.33$ o hay un pequeño cambio en el \hat{C}_{pk} en varias iteraciones sucesivas o el número máximo de 100 iteraciones ha sido rechazado.

El criterio de terminación establecido en este algoritmo garantiza que el procedimiento en algún momento se detendrá. Esto es en virtud de que en la iteración k , el valor esperado del \hat{C}_{pk} tiene únicamente la siguiente opción: ser mayor o igual que M , es decir, que en la implantación del algoritmo se termina con la declaración de que se determinaron las condiciones óptimas de operación.

Si el valor del \hat{C}_{pk} en la iteración $k + 1$ es menor a M y al valor observado del \hat{C}_{pk} en la iteración k , entonces se declara que el algoritmo no logra determinar las

condiciones óptimas de operación, si en una cantidad específica de iteraciones no ocurre que $\hat{C}_{pk}^{k+1} \geq \hat{C}_{pk}^k$. En caso contrario, se continúa iterando el algoritmo propuesto.

Capítulo III. Materiales y Métodos

En este capítulo se presentan tanto los materiales como los métodos que son usados para contrastar las hipótesis propuestas en esta tesis. Los métodos utilizados se llevaron a cabo usando simulación para las cuatro funciones de prueba y los cuatro valores de la distancia que existe entre los puntos centrales y los vértices del diseño factorial 2^k .

3.1. Materiales.

Los materiales empleados en la validación de las hipótesis mediante simulación son una computadora portátil Intel Core i5 de 2.4 GHz, la cual se utilizó tanto para la simulación como para la optimización del algoritmo modificado, además de la redacción del trabajo de tesis. El software utilizado fue el Minitab[®]16 con los módulos de *Calc: Random Data*, este para generar los valores iniciales de X_0 con la distribución uniforme, *Basic Statistics: 2 Sample t*, para verificar la primer hipótesis de esta tesis con respecto a la diferencia de medias (independientes) y *el Análisis de Varianza (ANOVA)*, para verificar la tercera hipótesis de esta tesis; además se utilizó el software MATLAB[®], para programar y validar el algoritmo modificado propuesto en esta tesis. También se utilizó el software STATISTICA[®] para graficar los resultados obtenidos por MATLAB[®], asimismo la hoja de cálculo Excel, esta hoja para concentrar, obtener los valores promedios del C_{pk} además del número de iteraciones, para la verificación de la segunda hipótesis de esta tesis. Con este *software* y *hardware* se verifica a través de simulación que la efectividad del AAEPS es independiente de las condiciones iniciales de operación X_0 en las “primeras” iteraciones y que además mejora el valor del índice de capacidad real del proceso C_{pk} con factores de ruido.

3.2. Métodos.

En esta sección se presentan los métodos utilizados para la verificación de las hipótesis de la sección 1.5 que afirman que el algoritmo modificado cuenta con las

características siguientes: 1) Su efectividad es independiente de las condiciones iniciales de operación X_0 en las “primeras” iteraciones; 2) logra mejorar el C_{pk} con factores de ruido; 3) Su eficacia es afectada por los factores de ruido.

3.2.1 Descripción del Proceso de Simulación

Las hipótesis que afirman que el AAEPSM logra mejorar el índice de capacidad del proceso \hat{C}_{pk} , se evalúan en cada iteración del proceso de simulación. Para cada iteración se estiman treinta y tres valores del \hat{C}_{pk} , estas estimaciones se usan para contrastar a cada una de estas hipótesis. En los siguientes párrafos se describe el proceso de simulación con el cual se genera los valores de los estadísticos empleados en la verificación de estas hipótesis.

El valor del índice de capacidad real del proceso C_{pk} depende de los valores que asuman las funciones reales del arreglo interno y externo del diseño cruzado del AAEPSM, así como de, sus valores medios μ y su desviación estándar σ mismas que a su vez están en función de los niveles o valores de los factores controlables X y la distancia que hay entre los puntos centrales y los vértices del diseño factorial $2^k Z$ utilizado en el arreglo externo del diseño cruzado; esto es, el C_{pk} es una función compuesta de (X, Z) .

$$C_{pk} = f(X, Z; LES) \quad (3.1)$$

Las estimaciones de los valores que asumen los parámetros μ y σ en las condiciones de operación (X, Z) (iteración k) y que se usan en la verificación de que el AAEPSM logra mejorar el valor del C_{pk} , se obtienen, respectivamente, a partir de las ecuaciones (3.2) y (3.3).

$$\hat{\mu} = \frac{\sum_{i=1}^5 Y_i}{5} \quad (3.2)$$

y

$$\hat{\sigma} = \sqrt{\frac{\sum_{t=1}^5 (Y_t - \hat{\mu})^2}{4}} \quad (3.3)$$

Donde Y_t ($t=1,2,\dots,5$) corresponde al valor de la t-ésima medición de

$$Y_t = f(X_k) + [f(Z_t \pm v_i) + \varepsilon] \quad (3.4)$$

Los valores de la variable Y_t corresponden a valores de variables aleatorias utilizando una distribución normal con media $\mu = (X, Z)$ y desviación estándar $\sigma = (X, Z)$, puesto que el error aleatorio representado como ε del *software MINITAB*[®] 16 se generó con números aleatorios usando una distribución normal con media igual a cero y desviación estándar finita.

Con los valores estimados de los parámetros μ y σ , que se obtienen a partir de las ecuaciones (3.2) y (3.3), se estima el valor del C_{pk} , el cual está denotado por \hat{C}_{pk} y definido, respectivamente, por la ecuación (3.5), cuando el valor de la variable de respuesta mayor es lo mejor.

$$\hat{C}_{pk} = \frac{LES - \hat{\mu}}{3\hat{\sigma}} \quad (3.5)$$

Como el lector recordara LES representa, el límite de especificación superior para el valor esperado del C_{pk} . En esta simulación, cuando lo mayor es lo mejor los valores del LES se establece en función del valor de la función $Y_t = f(X_k) + [f(Z_t \pm v_i) + \varepsilon]$ evaluada en los valores de la condiciones iniciales obtenidas con la distribución uniforme.

3.3. Funciones de prueba.

Las funciones que son utilizadas para construir el modelo de simulación (función de prueba) son las funciones de dos variables usadas para la comparación numérica de técnicas experimentales (Parkinson y Hutchinson, 1971), mismas que son listadas a continuación: 1) Función Valle Parabólico de Rosenbrock, FVPDR 2) Función Cúbica de Rosenbrock, FCDR 3) Función Suma de Funciones Cuadradas, FSFC y 4)

Función de Beale, FB. Las tres primeras funciones son diseñadas para dificultar la determinación de sus valores mínimos con la implantación de técnicas experimentales; mientras que en la cuarta ocurre lo contrario, es decir, los valores deseados se obtienen con menor facilidad, por lo que ésta es usada como una medida relativa de comparación de efectividad de los algoritmos (Parkinson y Hutchinson, 1971). Además, el uso de estas funciones permite estimar la utilidad de las técnicas experimentales en la optimización de los procesos productivos (Sánchez, 1991) y (García, 2000).

A continuación se presenta una descripción algebraica y gráfica de las funciones de prueba que son usadas en esta simulación. La representación algebraica de la Función Valle Parabólico de Rosenbrock está dada por la ecuación (3.6).

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \quad (3.6)$$

con $f(2, 2) = 401$ y con su valor mínimo en $(1, 1)$ y $f(1, 1) = 0$. La representación gráfica de esta función se muestra en la figura 3.1.

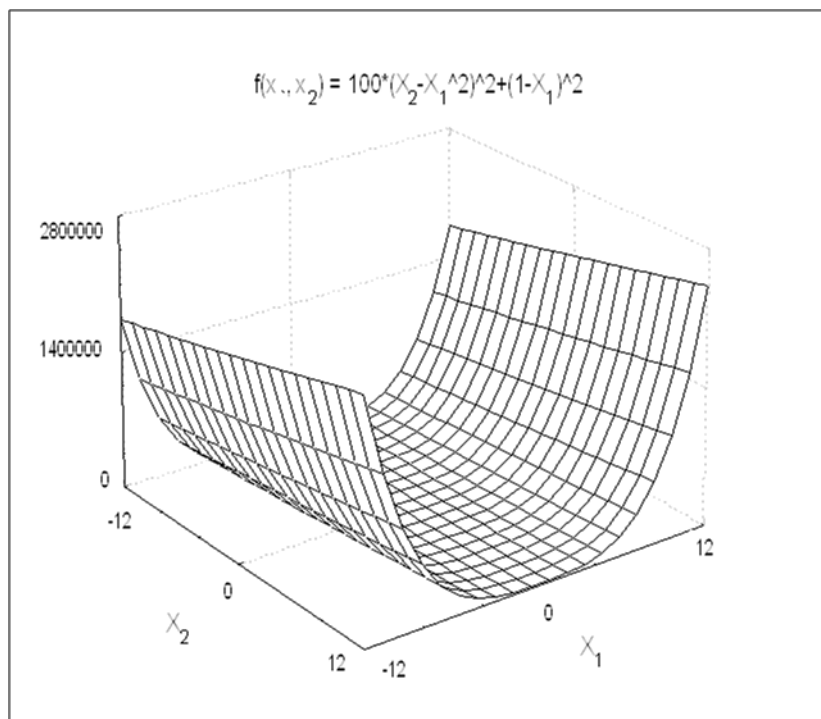


Figura 3.1 Función Valle Parabólico de Rosenbrock.

La representación algebraica de la Función Cúbica de Rosenbrock está dada por la ecuación (3.7).

$$f(x_1, x_2) = 100(x_2 - x_1^3)^2 + (1 - x_1)^2 \quad (3.7)$$

con $f(2, 2) = 3601$, con su valor mínimo en $(1, 1)$ y $f(1, 1) = 0$. La representación gráfica de esta función se muestra en la figura 3.2.

La representación algebraica de la Función Suma de Funciones Cuadradas está dada por la ecuación (3.8).

$$f(x_1, x_2) = x_1^2 + 2x_2^2 \quad (3.8)$$

con $f(2, 2) = 12$, con su valor mínimo absoluto en $(0, 0)$ y $f(0, 0) = 0$. La representación gráfica de esta función se muestra en la figura 3.3.

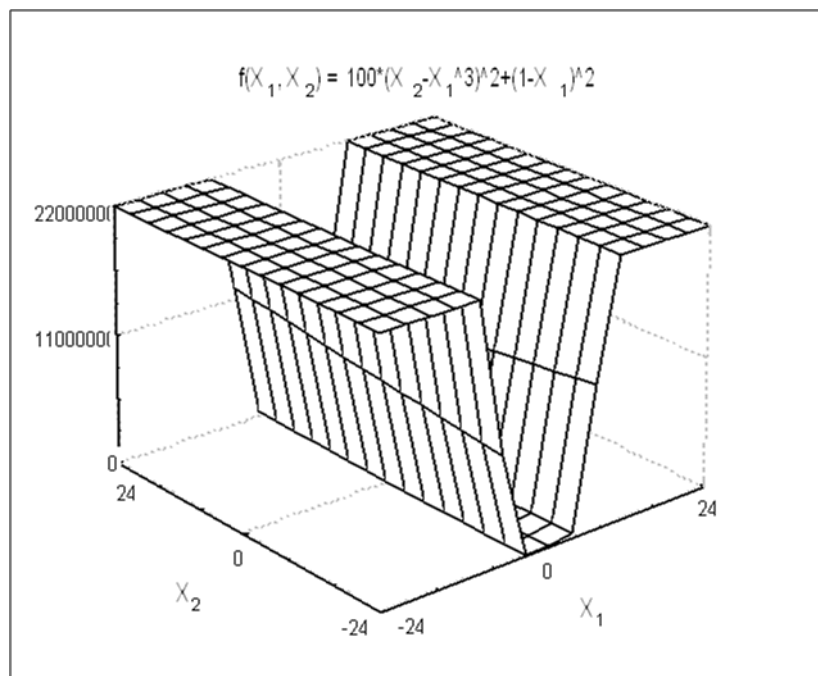


Figura 3.2 Función Cúbica de Rosenbrock.

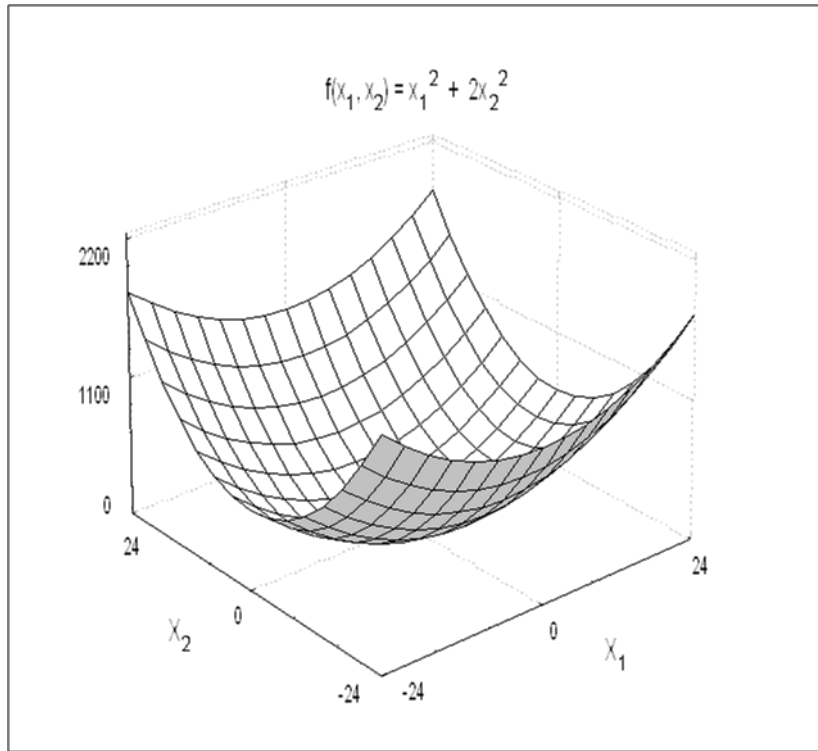


Figura 3.3 Función Suma de Funciones Cuadradas.

La representación algebraica de la Función de Beale está dada por la ecuación (3.9).

$$f(x_1, x_2) = [1.5 - x_1(1 - x_2)]^2 + [2.25 - x_1(1 - x_2^2)]^2 + [2.625 - x_1(1 - x_2^3)]^2 \quad (3.9)$$

con $f(2, 2) = 356.70$, con su valor mínimo en $(3, 0.5)$ $f(3, 0.5) = 0$. La representación gráfica de esta función se muestra en la figura 3.4.

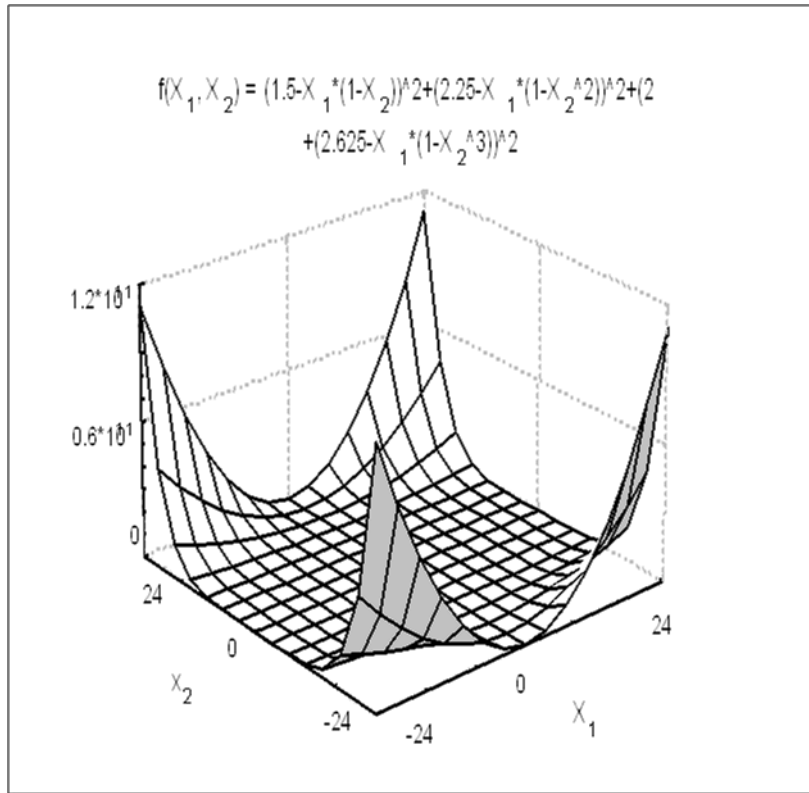


Figura 3.4 Función de Beale.

3.4. Verificación de hipótesis.

3.4.1. Verificación que la efectividad del AAEPSM es independiente de las condiciones iniciales de operación (X_0) en las “primeras” iteraciones.

Para verificar la hipótesis que establece que la efectividad del algoritmo modificado e iterativo en las “primeras” iteraciones es independiente del punto inicial, seleccionar aleatoriamente dos puntos iniciales (X_0) y (X'_0), condicionados a que ($X_0 \neq X'_0$); además, cuatro valores diferentes del volumen del diseño factorial 2^k , planteados en la sección 2.5.1, Con estos dos puntos iniciales y con la metodología indicada en la sección 2.5.3 se procede a simular, para cada punto inicial, el algoritmo modificado en treinta y tres ocasiones diferentes para obtener el valor del C_{pk} , cada una con un total de 10 iteraciones.

Con las treinta y tres estimaciones del valor del C_{pk} , enseguida para cada una de las 10 iteraciones correspondientes a cada uno de los dos puntos iniciales, se procede a contrastar las hipótesis que establece que la efectividad del algoritmo en las

“primeras” diez iteraciones es independiente del punto inicial, esto mediante el análisis de la prueba de hipótesis para diferencias de medias (independientes) del módulo *basic statistics, 2 Sample t* del *software MINITAB®16*. Si con un nivel de significancia del 5% se rechaza H_0 en cualquier iteración k ($k = 1, 2, \dots, 10$), en al menos en un volumen del diseño factorial 2^k evaluado, entonces se concluye que la efectividad del algoritmo depende del punto inicial seleccionado. Por el contrario, si en ninguna de estas iteraciones se rechaza H_0 , entonces se concluye que no existe evidencia suficiente para asegurar la existencia de esta dependencia entre la efectividad del algoritmo y el valor inicial (X_0).

3.4.2. Verificación que el AAEPMSM logra mejorar el C_{pk} con factores de ruido.

La verificación de esta hipótesis se lleva a cabo a través de simulación, empleando las cuatro funciones de prueba (ver tabla 3.1), las cuales se construyen usando las funciones representadas en las ecuaciones (3.6), (3.7), (3.8) y (3.9); siendo cada una de las cuatro, la función que representa al arreglo interno del diseño cruzado representadas como la $f(X_k)$; así mismo, cada una de estas funciones, también representan a la función $f(Z_l \pm v_l) + \varepsilon$, como el arreglo externo del diseño factorial 2^k del mismo diseño cruzado dentro del algoritmo.

La implantación del algoritmo modificado para cada una de las cuatro parejas que resultan de las funciones $f(X_k)$ más $f(Z_l \pm v_l) + \varepsilon$ de la tabla 3.1, estimar el valor del C_{pk} del diseño cruzado para cada iteración, como la media aritmética de treinta y tres valores del \hat{C}_{pk} , los cuales se generan con igual número de puntos iniciales diferentes $X_0 = (X_1, X_2)$. Cada uno de las componentes de (X_0) requeridos para la implantación del algoritmo modificado, se generan en forma independiente y aleatoria con una distribución uniforme a través del generador de números aleatorios del *software MINITAB®16*.

El k -ésimo elemento de las sucesiones $\{a_k\}$; $\{c_k\}$; $\{\Delta_k\}$ son definidos como $a_k = 1/k^\alpha$ y $c_k = c/(k+1)^\gamma$, el vector de perturbación simultánea $\{\Delta_k\}$ fue

optimizado para encontrar la mejor combinación que maximice el C_{pk} dentro del rango de ± 0.1 a ± 1 con incrementos de ± 0.1 ; las cuales satisfacen las condiciones de operación establecidas en (Sadegh y Spall, 1999).

Tabla 3.1 Funciones de Prueba Implementadas en el Proceso Simulación

Función de Prueba	Arreglo Interno $f(X_k)$	Arreglo Externo $[f(Z_l \pm v_l) + \varepsilon]$
FVPDR-FVPDR	Función Valle Parabólico de Rosenbrock	Función Valle Parabólico Rosenbrock
FCDR-FCDR	Función Cúbica de Rosenbrock	Función Cúbica de Rosenbrock
FB-FB	Función de Beale	Función de Beale
FSFC-FSFC	Función Suma de Funciones cuadradas	Función Suma de Funciones cuadradas

Para el análisis del valor esperado y del mejoramiento del C_{pk} que se logra con la implantación del algoritmo modificado, para cada una de las combinaciones de las funciones $f(X_k)$, $f(Z_l \pm v_l) + \varepsilon$, referidas como funciones de prueba en la tabla 4.1, la media aritmética de los treinta y tres valores del \bar{C}_{pk} que se obtienen en cada una de las iteraciones, son dispuestos en tablas, además de estar clasificados por cada uno de los volúmenes del diseño factorial 2^k . La técnica de contrastación que se usa para verificar que el algoritmo modificado mejora el C_{pk} con factores de ruido, es con un análisis numérico. Es decir, la mejora que se logra de la iteración uno con respecto a la iteración 100. Esta prueba se realiza con la revisión y evaluación de los resultados obtenidos gráficamente del C_{pk} . Se utilizó el módulo Graphs: 2D Graphs del *software STATISTICA*[®].

3.4.3. Verificación que la eficacia del AAEPSM es afectada por el volumen del diseño factorial 2^k , considerado como factor de ruido.

La verificación de esta hipótesis se realizó a través de simulación, empleando las cuatro funciones de prueba (ver tabla 3.1). La implantación del algoritmo modificado para cada una de las cuatro combinaciones o parejas de las funciones de prueba se generó con treinta y tres puntos iniciales (X_0), al igual número de valores del C_{pk} , de los cuales se obtiene $f(X_k)$ por función de prueba para obtener el resultado de las condiciones iniciales del proceso que maximizan el C_{pk} . En seguida se implementa el algoritmo para obtener el resultado de las condiciones óptimas de operación del

proceso para evaluar la $f(X_k^*)$ que maximiza el C_{pk} . Finalmente, las variables de respuesta que se analizan comparando el resultado de la simulación con el algoritmo modificado contra el óptimo verdadero de la función de prueba son: La Eficiencia en la Función Ganancia (EFG), la cual se obtiene con la ecuación (3.10), la Distancia al Óptimo (DO) la cual se obtiene con la ecuación (3.11), y el Número de Iteraciones que se refiere a la cantidad de iteraciones realizadas por el algoritmo para obtener las condiciones óptimas de operación de los factores controlables, que maximizan el C_{pk} .

$$EFG = 100 \cdot \left(\frac{f(X_k) - f(X_k^*)}{f(X_k)} \right) \quad (3.10)$$

$$DO = \|\hat{X}_k - X_k^*\| \quad (3.11)$$

Donde, $f(X_k)$ representa el resultado de calcular la función de prueba con los valores iniciales del proceso (X_0). Y la $f(X_k^*)$ representa el resultado de calcular la función de prueba con las condiciones óptimas de operación del proceso (X_k^*). Para el cálculo de la DO (\hat{X}_k) representa las condiciones óptimas de operación del proceso obtenidas por el algoritmo modificado. (X_k^*) representa las condiciones del óptimo verdadero de la función de prueba. La técnica de contrastación que se usa para verificar que la eficacia del algoritmo modificado es afectada por los factores de ruido, es con un Análisis de Varianza, a través de un Diseño Aleatorizado por Bloques Completos (DABC). El planteamiento de este diseño es que una vez obtenidos los valores de los experimentos realizados para la obtención del \hat{C}_{pk} para las treinta y tres réplicas o puntos iniciales (representados como bloques) se realizan considerando los cuatro volúmenes del diseño factorial 2^k como un factor de efectos fijos. Este factor determina el valor de la variable de respuesta del proceso y el efecto en el número volúmenes resultantes que sitúan el valor máximo del \hat{C}_{pk} . En cada replica (bloque) se prueban los cuatro volúmenes del diseño factorial 2^k uno a la vez aleatoriamente, quedando las observaciones de cada réplica en la forma como se muestra en la tabla 4.2.

Tabla 3.2. Formato utilizado en el análisis de varianza para la eficacia del AAEPSPM

Volumen del diseño 2^k (factor de ruido)	Distribución Normal
0.05	$\hat{y}_{0.05j}$
0.1	$\hat{y}_{0.1j}$
0.2	$\hat{y}_{0.2j}$
0.3	$\hat{y}_{0.3j}$

Donde:

\hat{y}_{ij} : Representa el valor de la variable de respuesta cuando el volumen del diseño factorial 2^k es $i = 0.05, 0.1, 0.2$ y 0.3 , corresponde a la réplica j , donde $j = 1, 2, \dots, 33$.

Se desea que el error experimental sea lo más pequeño posible, sustrayendo del error experimental la variabilidad producida por los bloques. El diseño a considerarse para el análisis de varianza, es el diseño aleatorizado por bloques completos (Saha y Khuri, 2009). El modelo estadístico para este diseño se muestra en la ecuación (4.12).

$$\hat{y}_{ij} = \mu + \tau_i + \beta_j + \varepsilon_{ij} \quad (4.12)$$

Donde:

\hat{y}_{ij} : representa la observación correspondiente a la réplica j con el volumen i .

μ : representa la media general de la variable de respuesta.

τ_i : representa el efecto del i -ésimo volumen del diseño factorial 2^k . Para $i = 0.05, 0.1, 0.2$ y 0.3 .

β_j : representa el efecto de la j -ésima réplica (bloque). Para $j = 1, 2, \dots, 33$.

ε_{ij} : representa el error aleatorio.

Los efectos de los tratamientos y bloques se consideran como desviaciones de la media general, por lo cual, se desea probar la igualdad de las medias de los tratamientos. La hipótesis nula es $H_o : \tau_{0.05} = \tau_{0.1} = \tau_{0.2} = \tau_{0.3} = 0$ o $H_o : \tau_i \neq 0$ al menos para una i .

Capítulo IV. Análisis y Discusión de Resultados

En este capítulo se presenta tanto el análisis como la discusión de los resultados que se obtienen en la comprobación y validación de las hipótesis planteadas en esta tesis mediante simulación. Hipótesis que describen al AAEPSM como un algoritmo iterativo y robusto, que determina las condiciones óptimas de operación en las variables controlables. Es decir, logran mejorar el valor del C_{pk} . De las hipótesis, su comprobación y validación se realiza tanto con las cuatro funciones de prueba como con los cuatro valores del volumen del diseño factorial 2^k dentro del diseño cruzado desarrollado en el AAEPSM, como se mencionó en la sección 2.5 y 3.3.

Este planteamiento se desarrolla de acuerdo a la siguiente forma: primeramente se expone la verificación de que el AAEPSM es efectivo, además que logra mejorar el valor del C_{pk} , y también determinar si su eficacia es afectada por el factor de ruido.

4.1. Verificar que la efectividad del AAEPSM es independiente de las condiciones iniciales de operación (X_0) en las “primeras” diez iteraciones.

Para comprobar la hipótesis que afirma que la efectividad del AAEPSM o que el valor máximo del C_{pk} con este algoritmo se generan en la primeras iteraciones, es independiente de las condiciones iniciales de operación (X_0), Los datos que se obtienen para las primeras iteraciones del proceso de simulación de la implantación del algoritmo modificado con factores de ruido, sus resultados son presentados en la tabla 4.1 para la FSFC, en la que se muestra la media aritmética de treinta y tres estimaciones del C_{pk} que se producen al implantar en igual cantidad de veces este método para un mismo punto inicial. También son expuestos los resultados que se obtienen de la verificación de la hipótesis nula que afirma que la efectividad del algoritmo modificado es independiente de las condiciones iniciales de operación (X_0).

En la tabla 4.1 se puede observar que los valores de las medias aritméticas (promedios) del índice de capacidad del proceso C_{pk} de la función de prueba FSFC, con un valor del volumen del diseño factorial 2^k igual a 0.1 del arreglo externo del

diseño cruzado, son generados con la simulación del algoritmo modificado dados los puntos iniciales (X_0) y (X'_0), que corresponden a cada una de las diez “primeras” iteraciones.

Tabla 4.1. Valores Promedios del C_{pk} para Dos Puntos Iniciales Distintos, para la FSFC con un $\nu = 0.1$ del diseño factorial 2^k

Iteración	1	2	3	4	5	6	7	8	9	10
$\bar{C}_{pk} X_0$ (1)	-1.7	-9.9	-51.0	-80.2	-71.9	-76.4	-58.5	-49.9	-28.7	-24.1
$\bar{C}_{pk} X'_0$ (2)	-2.4	-17.7	-61.4	-67.7	-78.9	-61.0	-38.7	-35.2	-31.6	-12.8
(1) \neq (2)	Si	si	si	si	si	si	si	si	si	si

Si: diferencia significativa con $\alpha = 0.05$

No: diferencia no significativa con $\alpha = 0.05$

La hipótesis en la que se establece que la efectividad del algoritmo modificado e iterativo o que el valor máximo del C_{pk} con este algoritmo es independiente de las condiciones iniciales de operación (X_0) en las “primeras” iteraciones, es rechazada, ya que se muestra un ejemplo en el que, con un nivel de significancia del 5%, se contradice esta afirmación. Lo anterior, implica que la efectividad del AAEPSM o que el valor máximo del C_{pk} dependen del punto inicial, ya que es un parámetro que afecta el desempeño del algoritmo con factores de ruido.

La comprobación de las hipótesis restantes, contempladas en esta sección, se llevaron a cabo considerando para cada iteración la media aritmética de treinta y tres valores del C_{pk} . Estos valores se generan con igual número de puntos iniciales distintos en forma aleatoria con la distribución uniforme. Dado que esto es lo más apegado a la realidad, ya que en los casos prácticos no es posible establecer un criterio de selección de las condiciones iniciales de operación y usualmente no se llevan a cabo repeticiones o réplicas de la aplicación de los métodos experimentales en diferentes condiciones iniciales de operación.

4.2. Verificar que el AAEPSM logra mejorar el C_{pk} con factores de ruido.

En esta sección se presenta un análisis del desempeño del algoritmo, con el propósito de mostrar el comportamiento de los valores del \hat{C}_{pk} que son usados para comprobar

la hipótesis 1.5.2 de la sección 1.5. Análisis que se realizó a través de simular su implantación en procesos en los que el valor mayor de la característica de calidad es lo mejor, en el que se consideran como variable de respuesta al índice de capacidad del proceso \bar{C}_{pk} que corresponden a la media aritmética de treinta y tres estimaciones del valor del C_{pk} para cada iteración y los resultados de contrastar la hipótesis alternativa (con un nivel de significancia del 5%) de que el AAEPMS no logra mejorar el valor del C_{pk} en más de 1 sin factores de ruido.

4.2.1. Análisis de Resultados para cada Función de Prueba.

Los valores de la media aritmética \bar{C}_{pk} de los treinta y tres \hat{C}_{pk} estimados en cada iteración por volumen del diseño factorial 2^k implementados en cada una de las cuatro funciones de prueba descritas en la sección 3.3, son presentados en la siguiente sección de esta tesis, para los casos de contar con un criterio de paro y no contar con un criterio de paro del algoritmo modificado. Por otro lado, el comportamiento gráfico del \bar{C}_{pk} se muestra en las figuras 4.1, 4.2, 4.3 y 4.4, para el caso de contar con un criterio de paro del algoritmo. En las figuras 4.5, 4.6, 4.7 y 4.8, para el caso de no contar con un criterio de paro del mismo. Estos valores y resultados se obtienen mediante la simulación del algoritmo modificado; simulación en la que se considera a cada una de las funciones de prueba, como las funciones que evalúan los puntos iniciales dentro del arreglo interno, también evalúan cada uno de los experimentos del diseño factorial 2^k en el arreglo externo del diseño cruzado presentado en la tabla 2.1. Esto como parte de la relación funcional que rige el comportamiento tanto del valor promedio como de la desviación estándar para la variable de respuesta.

4.2.1.1. Análisis de Resultados para la Función Valle Parabólico de Rosenbrock (FVPDR).

Los resultados presentados en la tabla 4.2 para la FVPDR del algoritmo modificado con un criterio de paro, se pueden observar todos los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.2 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -1.49 y que este valor decrementa en forma constante hasta la iteración cinco con un

valor de -2.74; así en la iteración diez este valor llega a -2.11, mientras que en la iteración cien es igual a 0.64. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.06 unidades.

Tabla 4.2. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FVPDR con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-6.94	-3.45	-1.49	-0.04
2	-28.74	-8.59	-2.46	-1.80
3	-33.98	-10.04	-2.63	-2.30
4	-36.92	-10.71	-2.80	-2.48
5	-38.80	-11.02	-2.74	-2.58
6	-40.07	-11.29	-2.74	-2.62
7	-40.93	-11.34	-2.62	-2.78
8	-41.52	-11.45	-2.45	-2.72
9	-41.90	-11.42	-2.29	-2.74
10	-42.12	-11.41	-2.11	-2.73
20	-40.04	-10.14	-0.75	-3.03
30	-34.81	-8.11	0.35	-3.20
40	-28.39	-5.35	0.50	-3.30
50	-21.36	-3.92	0.56	-3.41
60	-13.88	-2.69	0.65	-3.52
70	-4.75	-1.53	0.64	-3.53
80	-0.75	-0.23	0.61	-3.63
90	-0.64	0.50	0.62	-3.58
100	-0.56	0.48	0.64	-3.64

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

En la tabla 4.3, se muestran todos los valores del \bar{C}_{pk} con la FVPDR del algoritmo modificado sin un criterio de paro, para cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.2 unidades, se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -1.08 y que este valor empieza incrementarse a partir de la iteración cuatro con un valor del \bar{C}_{pk} igual a -0.91; así en la iteración diez este valor llega a -0.41, mientras que en la iteración cien es igual a -0.56. Además, también ocurre que a partir de la iteración treinta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de -0.04 unidades.

Tabla 4.3. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FVPDR sin Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-4.06	-2.04	-1.08	-0.79
2	-21.70	-4.84	-1.43	-0.70
3	-25.40	-5.21	-1.29	-0.53
4	-26.70	-5.00	-0.91	-0.33
5	-26.92	-4.47	-0.59	-0.33
6	-26.54	-3.67	-0.49	-0.37
7	-25.76	-2.72	-0.44	-0.37
8	-24.71	-2.10	-0.43	-0.35
9	-23.45	-0.89	-0.46	-0.36
10	-22.03	-0.86	-0.41	-0.35
20	-2.89	-0.35	-0.45	-0.43
30	-0.58	-0.55	-0.52	-0.46
40	-0.61	-0.80	-0.56	-0.47
50	-0.69	-0.92	-0.57	-0.47
60	-1.05	-0.98	-0.58	-0.46
70	-1.19	-0.99	-0.57	-0.46
80	-1.33	-1.00	-0.57	-0.46
90	-1.52	-1.00	-0.57	-0.45
100	-1.64	-1.00	-0.56	-0.45

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

En la tabla 4.4, se muestran los valores del \bar{C}_{pk} evaluados con el algoritmo modificado sin factores de ruido y con criterio de paro. Se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a 0.31, siendo este su valor máximo y que este valor se decrementa constantemente hasta la iteración cuarenta con un valor del \bar{C}_{pk} igual a -3.96; fluctuando entre los valores de -3.69 y -3.29 de la iteración cincuenta a la iteración ochenta respectivamente, mientras que de la iteración noventa a la cien el valor del \bar{C}_{pk} se estabiliza en -3.46.

4.2.1.2. Análisis de Resultados para la Función Cúbica de Rosenbrock (FCDR).

Los resultados presentados en la tabla 4.5 para la FCDR evaluados con el algoritmo modificado con un criterio de paro, se pueden observar todos los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.2 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es

igual a -1.26 y que este valor decremanta en forma constante hasta la iteración tres con un valor de -1.92; así en la iteración diez este valor llega a -0.15, mientras que en la iteración cien es igual a 0.21. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.03 unidades.

Tabla 4.4. Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FVPDR sin Factores de Ruido Con y Sin Criterio de Paro

Iteración	Sin Criterio de Paro	Con Criterio de Paro
1	-0.04	0.31
2	0.29	-0.81
3	0.54	-1.60
4	0.57	-2.27
5	0.62	-2.38
6	0.59	-2.76
7	0.63	-3.04
8	0.73	-3.05
9	0.76	-3.51
10	0.70	-3.57
20	0.73	-3.99
30	0.75	-4.26
40	0.69	-3.96
50	0.73	-3.69
60	0.73	-3.36
70	0.67	-3.18
80	0.66	-3.26
90	0.61	-3.46
100	0.66	-3.46

En la tabla 4.6, se muestran todos los valores del \bar{C}_{pk} con la FCDR evaluados con el algoritmo modificado sin un criterio de paro, para cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -0.67 y que este valor decremanta inmediatamente en la iteración dos con un valor máximo del \bar{C}_{pk} igual a -0.79; así en la iteración diez este valor llega a -0.56, mientras que en la iteración cien es igual a -0.60. Además, también ocurre que a partir de la iteración cuatro estos valores se

estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de -0.03 unidades.

En la tabla 4.7, se muestran los valores del \bar{C}_{pk} evaluados con el algoritmo modificado sin factores de ruido y con criterio de paro, para la FCDR. Donde se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -0.56 y que este valor decrementa constantemente hasta la iteración diez con un valor del \bar{C}_{pk} igual a -1.26, mientras que en la iteración cien es igual a 2.84. Además, también ocurre que de la iteración once a la iteración cien estos valores presentan un constante incremento de un valor de -1.25 a 2.84 respectivamente.

Tabla 4.5. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FCDR con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-4.12	-2.40	-1.26	-0.85
2	-20.97	-5.66	-1.82	-1.03
3	-22.82	-6.02	-1.92	-0.90
4	-23.81	-6.14	-1.83	-0.78
5	-24.39	-6.13	-1.65	-0.63
6	-24.72	-6.05	-1.50	-0.50
7	-24.90	-5.95	-1.41	-0.37
8	-24.97	-5.80	-1.36	-0.27
9	-24.95	-5.64	-1.23	-0.17
10	-24.87	-5.47	-1.02	-0.15
20	-22.38	-2.92	-0.28	0.05
30	-18.66	-1.21	0.04	0.10
40	-14.45	0.09	0.13	0.10
50	-9.99	0.38	0.21	0.08
60	-5.80	0.53	0.22	0.08
70	-5.00	0.29	0.24	0.09
80	-5.49	0.16	0.23	0.07
90	-5.99	0.06	0.23	0.08
100	-6.34	-0.07	0.21	0.08

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

Tabla 4.6. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FCDR sin Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-4.14	-1.65	-0.91	-0.67
2	-13.72	-3.72	-1.21	-0.79
3	-14.11	-3.37	-1.22	-0.68
4	-14.04	-2.84	-1.05	-0.57
5	-13.56	-2.30	-0.92	-0.58
6	-12.74	-2.21	-0.95	-0.58
7	-11.81	-2.07	-0.95	-0.58
8	-10.80	-2.10	-0.95	-0.58
9	-9.70	-2.03	-0.94	-0.56
10	-8.43	-2.10	-0.95	-0.56
20	-10.03	-2.24	-0.92	-0.54
30	-8.11	-1.77	-0.92	-0.56
40	-6.56	-1.69	-0.95	-0.58
50	-5.87	-1.72	-0.98	-0.59
60	-5.64	-1.77	-1.00	-0.60
70	-5.55	-1.83	-1.01	-0.60
80	-5.57	-1.88	-1.02	-0.60
90	-5.60	-1.89	-1.02	-0.60
100	-5.64	-1.93	-1.02	-0.60

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

Tabla 4.7. Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FCDR sin Factores de Ruido Con y Sin Criterio de Paro

Iteración	Sin Criterio de Paro	Con Criterio de Paro
1	-0.12	-0.56
2	0.22	-0.88
3	0.21	-1.03
4	0.41	-1.11
5	0.53	-1.17
6	0.57	-1.20
7	0.83	-1.23
8	0.82	-1.25
9	0.85	-1.26
10	0.82	-1.26
20	0.87	-1.11
30	0.88	-0.77
40	0.92	0.40
50	0.92	1.72
60	0.98	2.15
70	0.88	2.60
80	0.92	2.72
90	0.91	2.82
100	0.89	2.84

4.2.1.3. Análisis de Resultados para la Función de Beale (FB).

Los resultados presentados en la tabla 4.8 para la FB con el algoritmo modificado con un criterio de paro, se pueden observar todos los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -0.52 y que este valor se incrementa en forma constante de la iteración tres hasta la iteración cien; así en la iteración diez este valor llega a 0.08, mientras que en la iteración cien es igual a 1.2. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.15 unidades.

En la tabla 4.9, se muestran todos los valores del \bar{C}_{pk} para la FB con el algoritmo modificado sin un criterio de paro, para cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -0.16 y que este valor se incrementa inmediatamente en la iteración cuatro con un valor máximo del \bar{C}_{pk} igual a 0.48; así en la iteración diez este valor llega a 0.33, mientras que en la iteración cien es igual a 0.25. Además, también ocurre que a partir de la iteración treinta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.01 unidades.

En la tabla 4.10, también se muestran los valores del \bar{C}_{pk} para la FB con el algoritmo modificado sin factores de ruido y con criterio de paro, se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -1.54, además de que este valor se incrementa inmediatamente en la iteración dos con un valor del \bar{C}_{pk} igual a -0.54; así en la iteración diez este valor decremanta hasta llegar a -2.12, mientras que en la iteración cien es igual a 0.53. Además, también ocurre que de la iteración cinco a la iteración veinte decremantan estos valores aproximadamente -0.25 unidades, y a partir de la iteración treinta a la cien hay un constante incremento de un valor de -1.72 a 0.53 respectivamente.

Tabla 4.8. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FB con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-2.74	-1.38	-0.75	-0.52
2	-2.91	-1.63	-0.80	-0.53
3	-4.01	-1.70	-0.68	-0.41
4	-4.14	-1.69	-0.71	-0.25
5	-4.10	-1.62	-0.69	-0.33
6	-4.06	-1.46	-0.62	-0.29
7	-3.99	-1.33	-0.50	-0.17
8	-3.92	-1.30	-0.41	-0.14
9	-3.82	-1.26	-0.43	-0.12
10	-3.67	-1.18	-0.33	-0.08
20	-2.22	0.16	0.24	0.33
30	-1.19	1.40	0.83	0.68
40	0.29	2.00	1.19	0.92
50	0.99	2.12	1.34	1.05
60	1.24	2.05	1.35	1.06
70	1.56	2.07	1.35	1.11
80	1.74	2.07	1.42	1.14
90	1.76	2.11	1.46	1.20
100	1.72	2.00	1.54	1.20

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

Tabla 4.9. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FB sin Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-0.97	-0.36	-0.16	-0.16
2	-1.94	-0.47	0.57	0.44
3	-0.99	0.53	0.68	0.34
4	0.01	0.98	0.79	0.48
5	0.83	0.88	0.92	0.44
6	1.37	0.91	0.86	0.45
7	1.09	0.96	0.80	0.37
8	0.96	1.00	0.89	0.35
9	1.01	1.00	1.03	0.33
10	0.95	1.02	1.10	0.33
20	0.86	1.15	0.77	0.29
30	0.97	0.99	0.62	0.26
40	0.83	0.86	0.55	0.24
50	0.87	0.75	0.56	0.24
60	0.84	0.74	0.56	0.24
70	0.80	0.62	0.54	0.23
80	0.82	0.66	0.55	0.25
90	0.73	0.65	0.55	0.25
100	0.68	0.58	0.57	0.25

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

Tabla 4.10. Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FB sin Factores de Ruido Con y Sin Criterio de Paro

Iteración	Sin Criterio de Paro	Con Criterio de Paro
1	-0.05	-1.54
2	0.58	-0.54
3	1.22	-1.70
4	1.52	-1.69
5	2.49	-1.86
6	2.69	-2.04
7	2.85	-2.00
8	2.90	-2.13
9	2.84	-2.16
10	2.68	-2.12
20	2.63	-2.11
30	2.52	-1.72
40	2.58	-1.22
50	2.49	-0.77
60	2.55	-0.32
70	2.41	-0.08
80	2.45	0.25
90	2.39	0.49
100	2.49	0.53

4.2.1.4. Análisis de Resultados para la Función Suma de Funciones Cuadráticas (FSFC).

Los resultados presentados en la tabla 4.11 para la FSFC con el algoritmo modificado con un criterio de paro, se pueden observar los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a 0.79 y que este valor decrementa en forma constante hasta la iteración cien; Además, también ocurre que a partir de la iteración sesenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de -0.46 unidades.

En la tabla 4.12, se muestran los valores del \bar{C}_{pk} para la FSFC con el algoritmo modificado sin un criterio de paro, para cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen

del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a 0.92 y que este valor se decrementa inmediatamente hasta la iteración tres con un valor del \bar{C}_{pk} igual a 0.61; así en la iteración diez este valor llega a 1.22, mientras que en la iteración cien es igual a 1.26. Además, también ocurre que a partir de la iteración seis estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.19 unidades.

En la tabla 4.13, también se muestran los valores del \bar{C}_{pk} para la FSFC evaluados con el algoritmo modificado sin factores de ruido y con criterio de paro, se puede observar que en la iteración 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a 2.1, además de que este valor se decrementa constantemente hasta la iteración cien con un valor del \bar{C}_{pk} igual a -13.65.

Tabla 4.11. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FSFC con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	0.34	0.45	0.49	0.79
2	-14.40	-9.79	-6.56	-4.38
3	-127.94	-48.58	-10.73	-6.56
4	-376.32	-70.28	-12.03	-7.19
5	-515.42	-80.22	-11.80	-6.89
6	-636.88	-84.78	-12.14	-7.15
7	-692.52	-91.66	-12.80	-7.36
8	-690.71	-90.55	-12.83	-7.50
9	-686.29	-87.86	-12.64	-7.44
10	-690.71	-90.75	-12.52	-7.38
20	-677.39	-85.37	-12.63	-7.76
30	-691.04	-85.70	-13.55	-8.19
40	-692.34	-87.52	-14.12	-8.73
50	-698.08	-90.69	-14.22	-8.93
60	-703.71	-92.57	-14.72	-9.10
70	-714.50	-93.95	-15.00	-9.29
80	-716.31	-94.07	-15.26	-9.46
90	-714.26	-93.82	-15.57	-9.60
100	-719.56	-95.70	-15.57	-9.56

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

Tabla 4.12. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FSFC sin Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	0.36	0.44	0.56	0.92
2	-1.46	-0.69	-0.03	0.62
3	-2.89	-1.58	-0.24	0.61
4	-5.05	-2.46	-0.17	0.78
5	-7.74	-2.93	-0.03	0.91
6	-9.27	-3.12	0.10	1.07
7	-9.43	-3.17	0.17	1.12
8	-12.07	-3.50	0.32	1.16
9	-12.56	-3.76	0.42	1.23
10	-13.52	-3.90	0.42	1.22
20	-19.04	-4.22	0.75	1.22
30	-20.45	-4.37	0.91	1.20
40	-23.10	-3.81	0.97	1.23
50	-24.86	-3.58	1.02	1.27
60	-26.71	-3.35	1.06	1.29
70	-26.60	-3.28	1.01	1.27
80	-26.80	-3.07	1.05	1.27
90	-27.52	-2.84	1.07	1.29
100	-28.49	-2.99	1.03	1.26

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

Tabla 4.13. Valores del \bar{C}_{pk} por Iteración para la Función de Prueba FSFC sin Factores de Ruido Con y Sin Criterio de Paro.

Iteración	Sin Criterio de Paro	Con Criterio de Paro
1	2.10	2.10
2	1.88	0.13
3	1.79	-1.84
4	1.79	-3.12
5	1.78	-3.98
6	1.76	-4.71
7	1.75	-5.13
8	1.74	-5.57
9	1.72	-6.14
10	1.64	-6.36
20	1.72	-7.70
30	1.80	-9.13
40	1.85	-10.74
50	1.94	-11.63
60	2.07	-11.70
70	2.03	-12.05
80	2.05	-12.38
90	2.08	-12.77
100	2.11	-13.65

4.2.1.5. Análisis gráfico para la Función Valle Parabólico de Rosenbrock (FVPDR).

El comportamiento gráfico de la FVPDR con factor de ruido y sin factor de ruido; con criterio de paro; sus valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones que se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.1, por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar de forma constante un decremento a partir de la primera iteración hasta la iteración seis con un valor del \bar{C}_{pk} de -1.49 hasta -2.74 respectivamente. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.08 unidades. A diferencia de la FVPDR sin factor de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a 0.31, además de que este valor decrementa constantemente hasta la iteración número cien con un valor de -3.46.

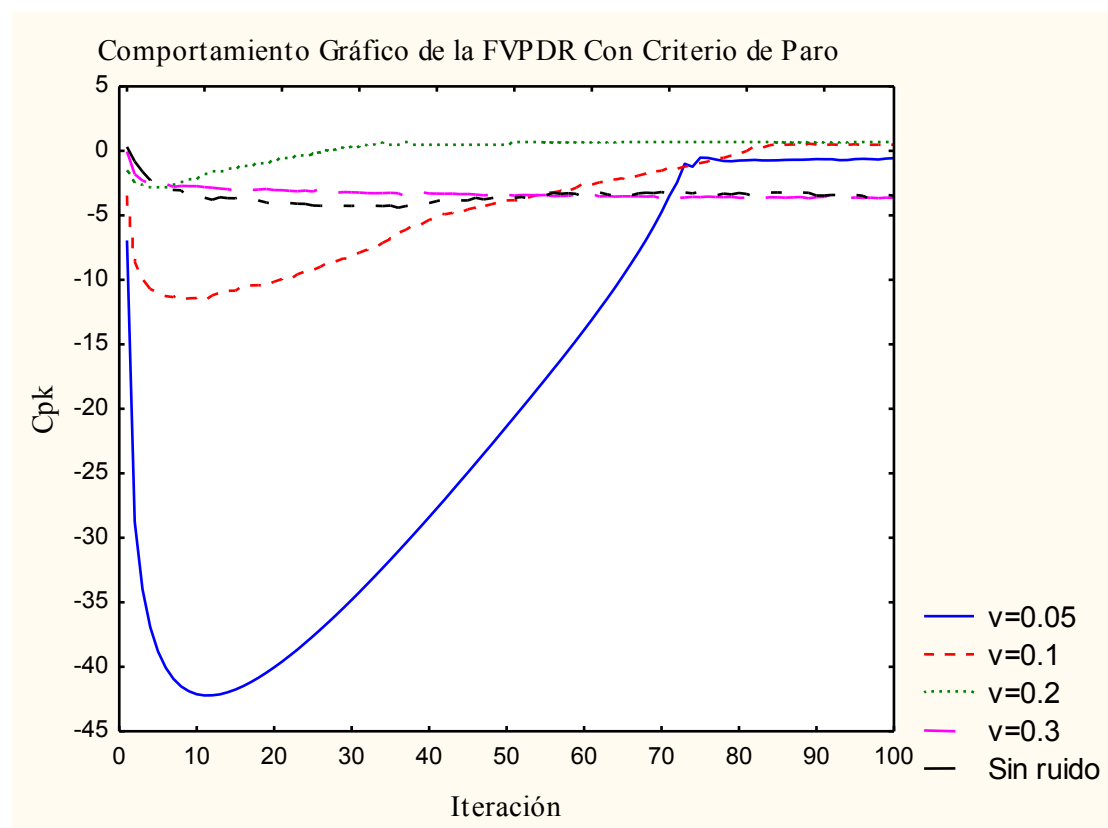


Figura 4.1. Gráfica de los \bar{C}_{pk} Generados para la FVPDR con Factores de Ruido y Sin Factor de Ruido y Con Criterio de Paro.

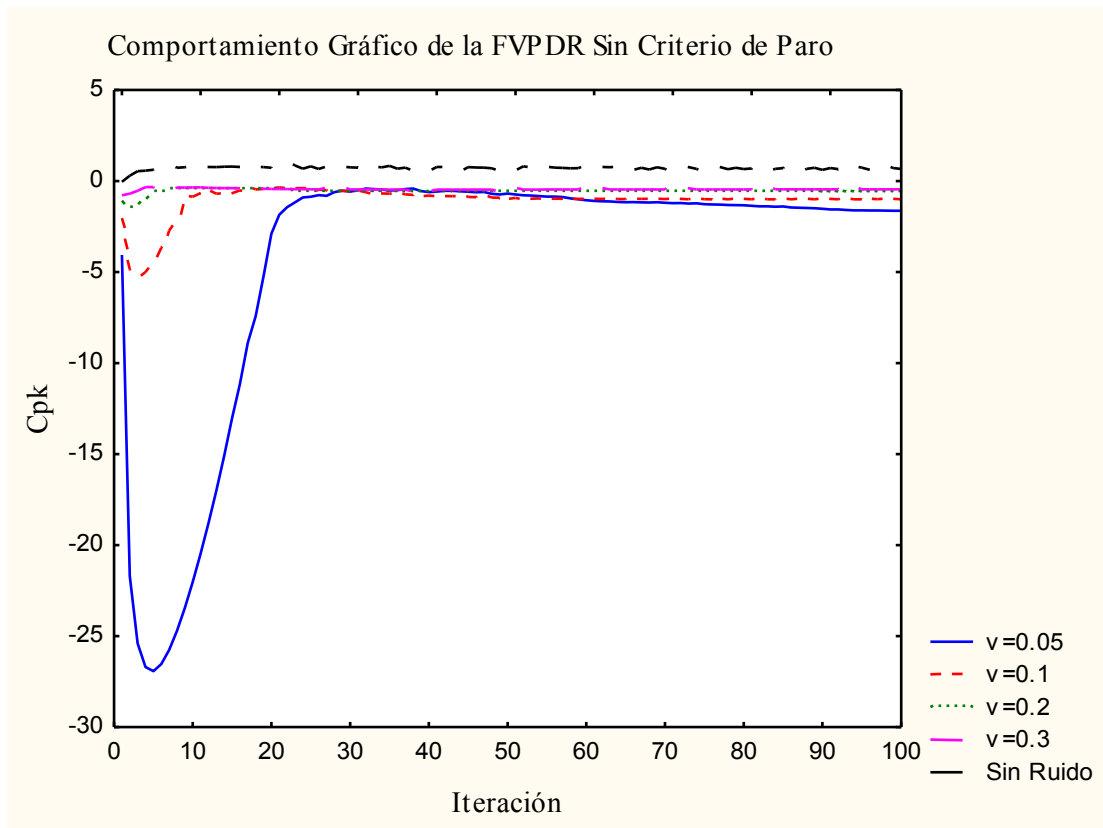


Figura 4.2. Gráfica de los \bar{C}_{pk} Generados para la FVPDR con Factores de Ruido y Sin Factor de Ruido y Sin Criterio de Paro.

Por otro lado, el comportamiento gráfico de la FVPDR con factor de ruido y sin factor de ruido; sin criterio de paro; sus valores de la media aritmética de los treinta y tres $\hat{C}_{pk} (\bar{C}_{pk})$ que se obtienen para cada una de las cien iteraciones que se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.2, por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar que el valor en la iteración uno o en el punto inicial el valor del \bar{C}_{pk} es igual a -1.08, además de que decremanta hasta la iteración dos con un valor de -1.43, a partir de la iteración tres incrementa de forma constante hasta la iteración cuarenta con un valor del \bar{C}_{pk} de -0.56, dado que pasa de un valor aproximado de -1.29 a otro de -0.56 respectivamente. Además, también ocurre que a partir de la iteración treinta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de -0.06 unidades. A diferencia de la FVPDR sin factor de ruido y sin criterio de paro, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -0.04, además de que este valor

incrementa constantemente hasta la iteración número nueve con un valor de \bar{C}_{pk} 0.76. Además, también ocurre que a partir de la iteración diez hasta la iteración 100 estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.05 unidades.

4.2.1.6. Análisis gráfico para la Función Cúbica de Rosenbrock (FCDR).

El comportamiento gráfico de la FCDR con factor de ruido y sin factor de ruido; con criterio de paro; los valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones que se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.3, por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar que el valor de la iteración uno es de -1.26, además de que decrementa hasta la iteración tres con un valor de -1.92, entonces a partir de la iteración cuatro incrementa de forma constante hasta la iteración setenta el valor del \bar{C}_{pk} , dado que pasa de un valor aproximado de -1.83 a otro de 0.24 respectivamente. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.03. A diferencia de la FCDR sin factor de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -0.56, además de que este valor decrementa constantemente hasta la iteración número diez con un valor de -1.26; entonces a partir de la iteración once a la cien hay un constante incremento de un valor de -1.26 a 2.84 respectivamente.

Por otro lado, el comportamiento gráfico de la FCDR con factor de ruido y sin factor de ruido; sin criterio de paro; los valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones que se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.4, por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar que el valor de la iteración uno es de -0.91, además de que decrementa hasta la iteración tres con un valor de -1.22, entonces a partir de la iteración cuatro incrementa de forma constante hasta la iteración treinta el valor del \bar{C}_{pk} , dado que pasa de un valor aproximado de -1.05 a otro de -0.92

respectivamente. Además, también ocurre que a partir de la iteración cuatro estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.13 unidades. A diferencia de la FCDR sin factor de ruido y sin criterio de paro, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -0.12, además de que este valor incrementa constantemente hasta la iteración número sesenta con un valor de 0.98; entonces a partir de la iteración once a la iteración cien hay un constante incremento de un valor de 0.82 a 0.89 respectivamente.

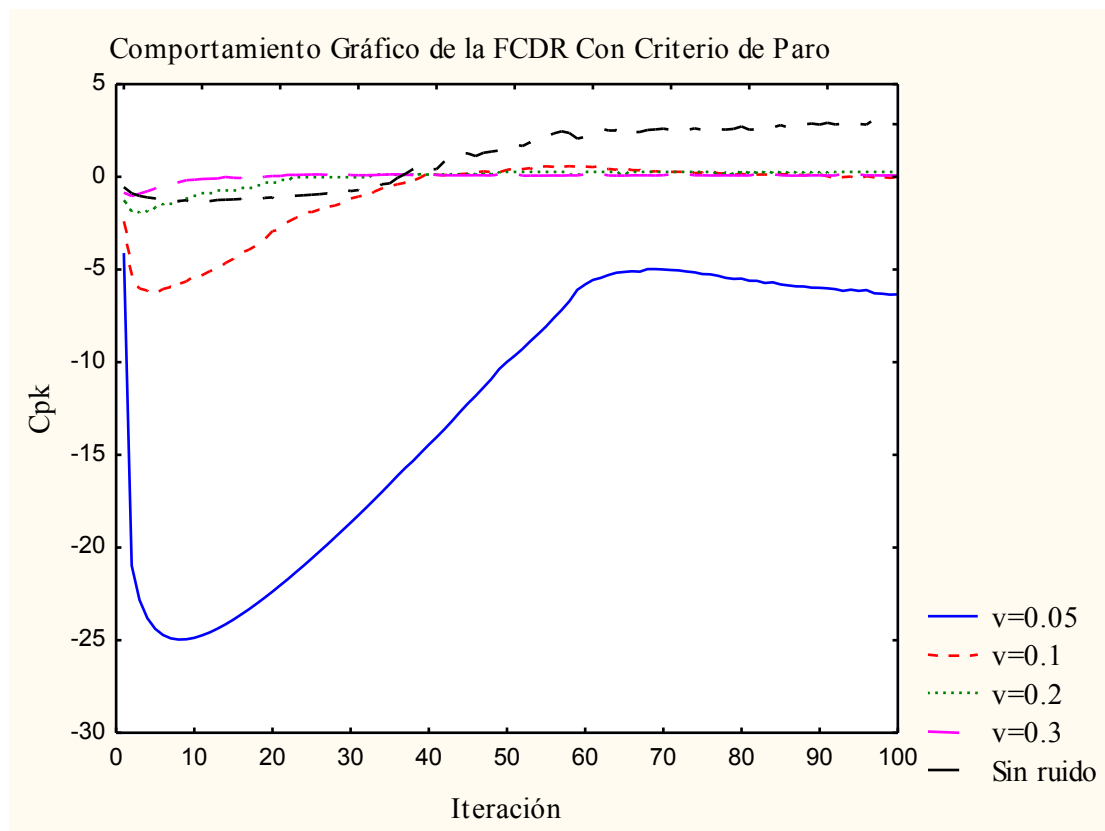


Figura 4.3. Gráfica de los \bar{C}_{pk} Generados para la FCDR con Factor de Ruido y Sin Factor de Ruido con Criterio de Paro.

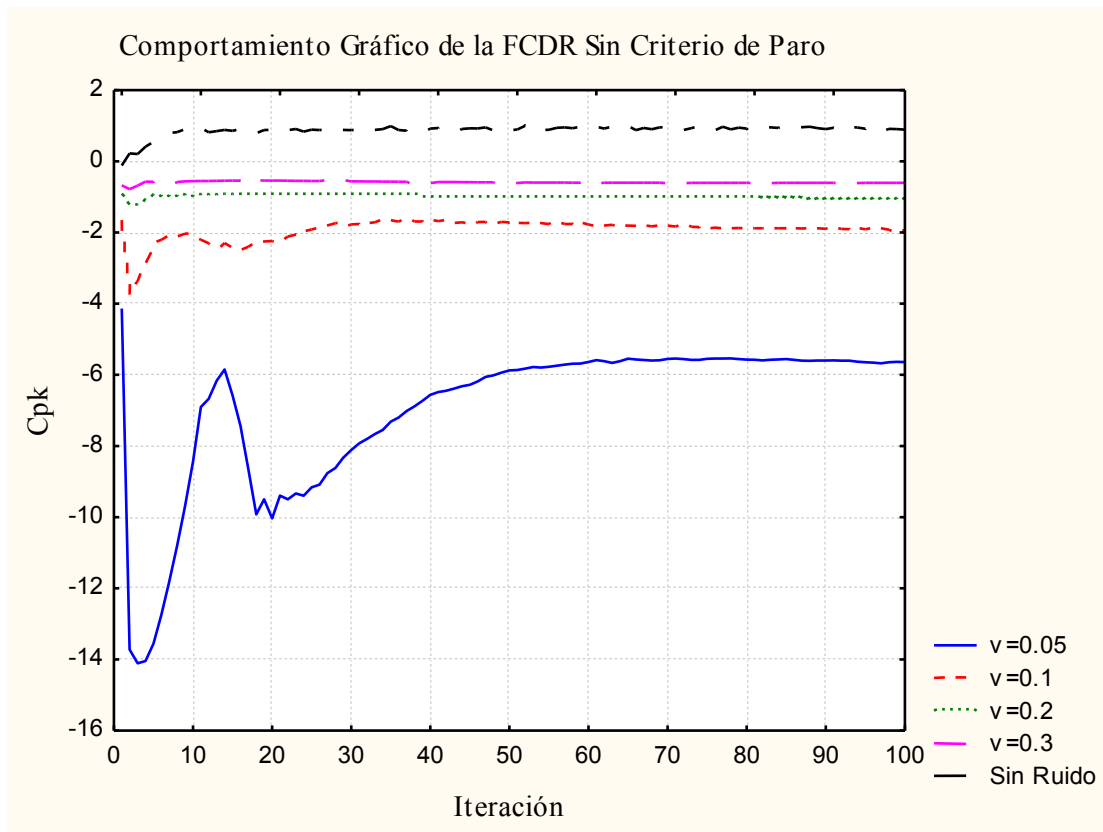


Figura 4.4. Gráfica de los \bar{C}_{pk} Generados para la FCDR con Factor de Ruido y Sin Factor de Ruido y Sin Criterio de Paro.

4.2.1.7. Análisis Gráfico para la Función de Beale (FB).

El comportamiento gráfico de la FB con factor de ruido y sin factor de ruido, con criterio de paro; sus valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.5. Por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.3 unidades, se puede observar que el valor de la iteración uno es de -0.52. Además, se observa de forma constante un incremento a partir de la iteración cinco hasta la iteración cien con un valor máximo del (\bar{C}_{pk}) de 1.20. Además, ocurre que a partir de la iteración cincuenta los valores del (\bar{C}_{pk}) se estabilizan en el sentido de que la máxima diferencia entre estos valores no excede de 0.15 unidades. A diferencia de la FB sin factores de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -1.54, además de que este valor se incrementa inmediatamente en la iteración número dos hasta -0.54; Además, ocurre que de la iteración tres a la

iteración nueve decrementan estos valores del (\bar{C}_{pk}) de -1.7 a -2.16 unidades respectivamente, entonces a partir de la iteración diez a la cien hay un constante incremento de un valor de -2.12 a 0.53 respectivamente.

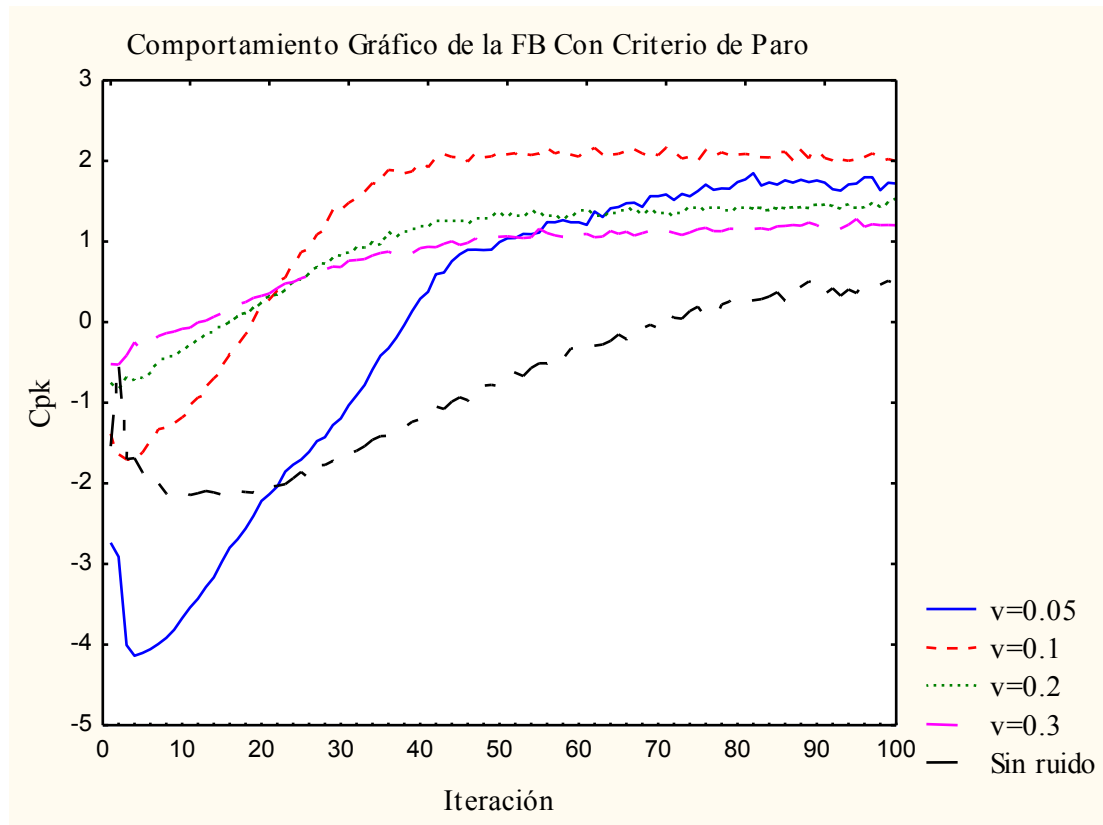


Figura 4.5. Gráfica de los \bar{C}_{pk} Generados para la FB con Factor de Ruido y Sin Factor de Ruido con Criterio de Paro.

Por otro lado, el comportamiento gráfico de la FB con factor de ruido y sin factor de ruido, sin criterio de paro; sus valores de la media aritmética de los treinta y tres $\hat{C}_{pk}(\bar{C}_{pk})$ que se obtienen para cada una de las cien iteraciones se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.6. Por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar que el valor de la iteración uno es de -0.16. Además, se observa de forma constante un incremento a partir de la iteración dos hasta la iteración diez con un valor máximo del (\bar{C}_{pk}) de 1.10. Además, ocurre que a partir de la iteración cuarenta los valores del (\bar{C}_{pk}) se estabilizan en el sentido de que la máxima diferencia entre estos valores no excede de 0.03 unidades. A diferencia de la

FB sin factores de ruido y sin criterio de paro, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -0.05, además de que este valor se incrementa constantemente hasta la iteración número ocho con un valor del \bar{C}_{pk} de 2.9; Además, ocurre que de la iteración nueve a la iteración treinta decremantan estos valores del (\bar{C}_{pk}) de 2.84 a 2.52 unidades respectivamente, entonces a partir de la iteración setenta estos valores se estabilizan en el sentido de que la máxima diferencia entre estos no excede de 0.1 unidades.

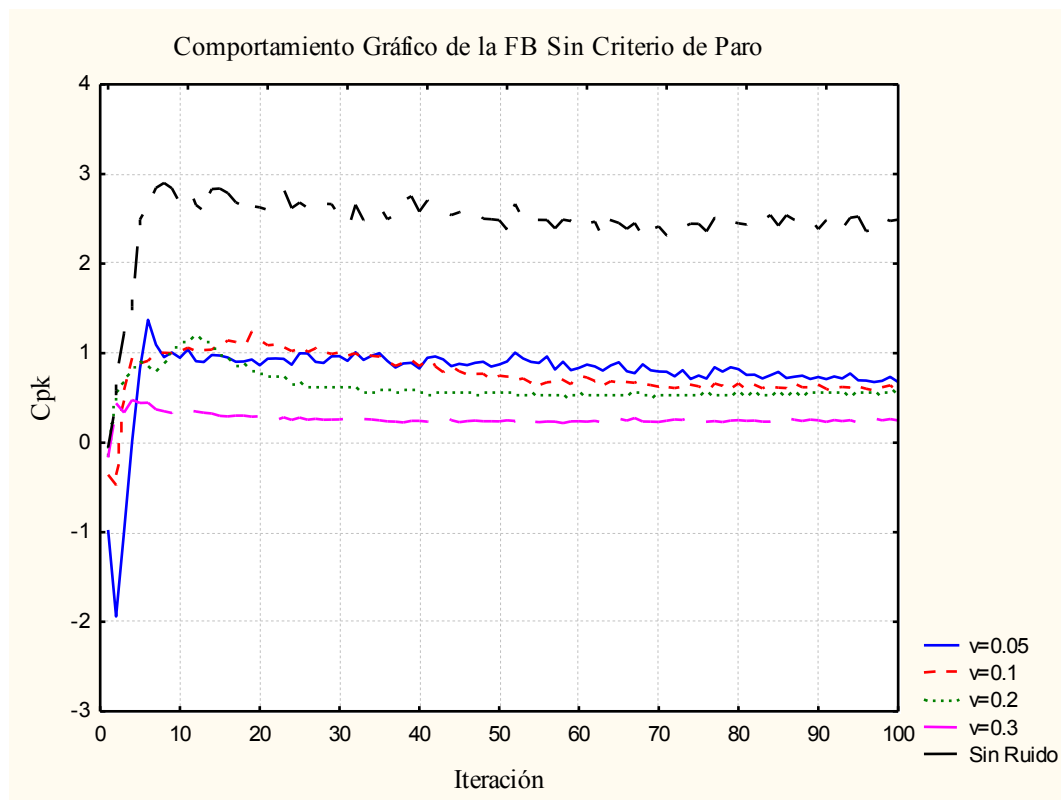


Figura 4.6. Gráfica de los \bar{C}_{pk} Generados para la FB con Factores de Ruido y Sin Factor de Ruido, Sin Criterio de Paro.

4.2.1.8. Análisis Gráfico para la Función Suma de Funciones Cuadráticas (FSFC).

El comportamiento gráfico de la FSFC con factor de ruido y sin factor de ruido, con criterio de paro; sus valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones se generan mediante la simulación de la implantación del algoritmo modificado, como se muestran en la figura 4.7. Por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.3 unidades, se puede observar que el valor de la iteración uno es de 0.79. Además, se observa de

forma constante un decremento a partir de la iteración dos hasta la iteración cien con un valor del (\bar{C}_{pk}) de -4.38 hasta -9.56 respectivamente. A diferencia de la FSFC sin factores de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a 2.1, además de que este valor decrementa constantemente hasta la iteración cien con un valor del (\bar{C}_{pk}) de -12.9.

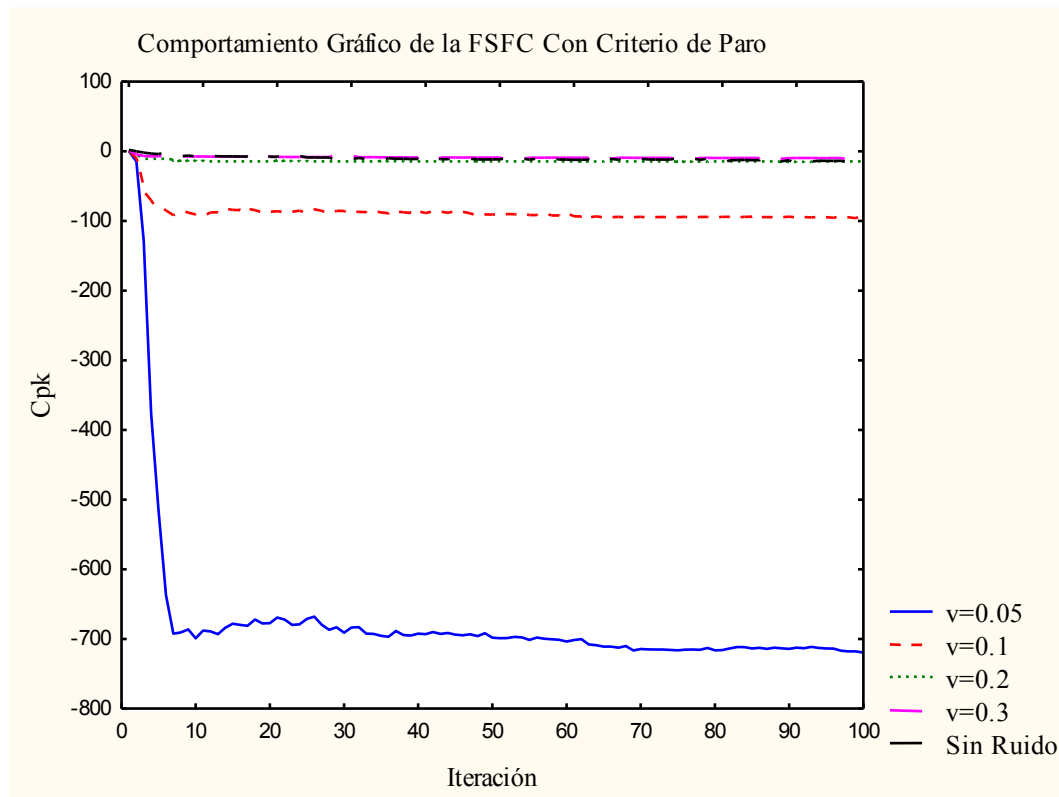


Figura 4.7. Gráfica de los \bar{C}_{pk} Generados para la FSFC con Factor de Ruido y Sin Factor de Ruido con Criterio de Paro.

Por otro lado, el comportamiento gráfico de la FSFC con factor de ruido y sin factor de ruido, sin criterio de paro; sus valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones se generan mediante la simulación de la implantación del algoritmo modificado, como se muestran en la figura 4.8. Por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.3 unidades, se puede observar que el valor de la iteración uno es de 0.92. Además, se observa un decremento a partir de la iteración dos y tres con un valor del (\bar{C}_{pk}) de 0.62 y 0.61 respectivamente. Además, se observa de forma constante un incremento a partir de la iteración cinco hasta la iteración nueve, con un valor máximo del (\bar{C}_{pk})

de 1.23. También ocurre que a partir de la iteración diez estos valores se estabilizan en el sentido de que la máxima diferencia entre estos valores no excede de 0.07 unidades. A diferencia de la FSFC sin factores de ruido y sin criterio de paro, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a 2.1, además de que este valor decrementa constantemente hasta la iteración diez con un valor del (\bar{C}_{pk}) de 1.64 unidades. Además, ocurre que a partir de la iteración sesenta hasta la iteración cien estos valores se estabilizan en el sentido de que la máxima diferencia entre estos valores no excede de 0.08 unidades.

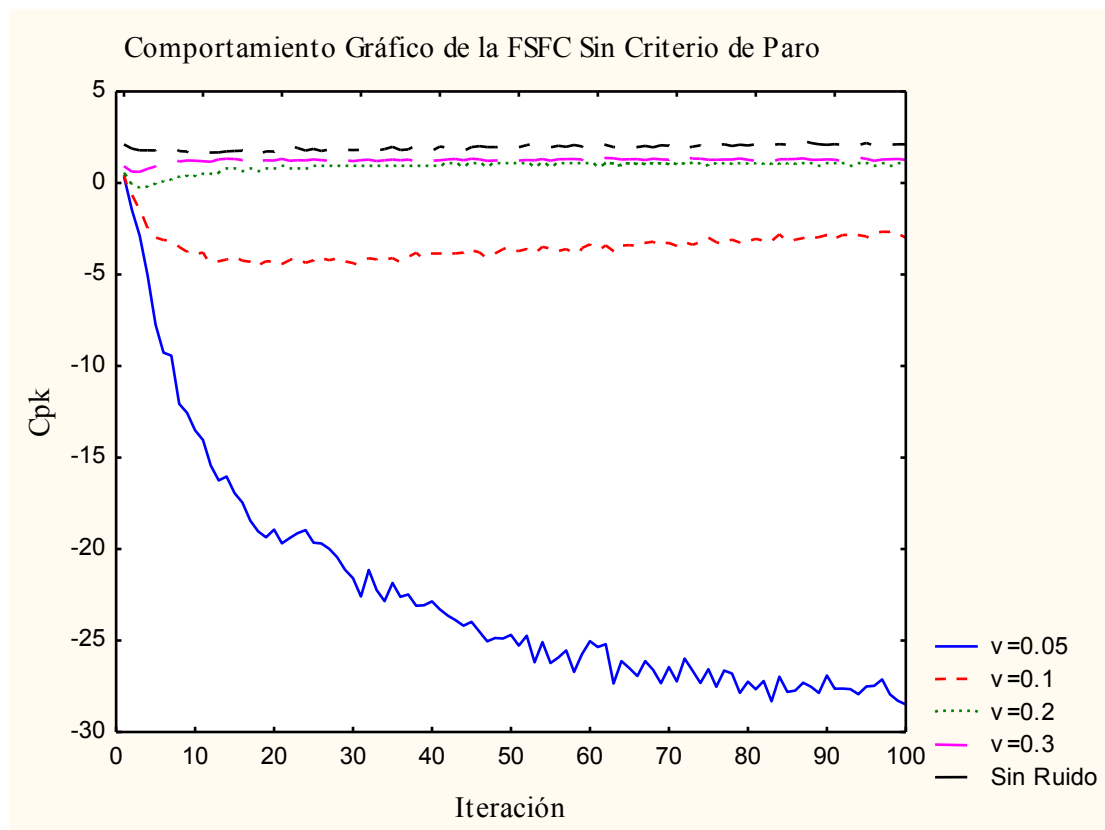


Figura 4.8. Gráfica de los \bar{C}_{pk} Generados para la FSFC con Factor de Ruido y Sin Factor de Ruido y Sin Criterio de Paro

4.3. Verificar que la eficacia del AAEPSM es afectada por el volumen del diseño factorial 2^k , considerado como factor de ruido.

La eficacia del algoritmo modificado está definida por dos medidas de rendimiento. La primera es el porcentaje de la mejora que se logra en el rendimiento del proceso, denominada Eficiencia en la Función Ganancia (EFG). La segunda, la distancia del óptimo generado por el algoritmo al óptimo verdadero de la función de prueba, la cual

es llamada Distancia al Óptimo (DO). Adicionalmente, se incluirá el Número de Iteraciones (NI), la cual se refiere a la cantidad de nuevas regiones generadas por el algoritmo para la mejora del C_{pk} .

Los resultados presentados en la tabla 4.14 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FVPDR con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 1.56, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 2.32. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.67$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.570$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.14. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Valle Parabólico de Rosenbrock con Criterio de Paro.

Análisis de varianza para los datos de:			<i>Eficiencia en la Función Ganancia con la Función Valle Parabólico de Rosenbrock</i>			
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	4.69	3	1.56	0.67	Aceptar Ho	0.570
Error	296.82	128	2.32			
TOTAL	301.51	131				

Por otro lado, los resultados presentados en la tabla 4.15 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FVPDR con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados

entre los experimentos llamado “Volumen” igual a 19.183, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 0.787. Esto indica que las medias de los tratamientos no son iguales. Dado que, al obtener el valor de $F_0 = 24.38$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.000$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.15. Análisis de Varianza para la Distancia al Óptimo de la Función Valle Parabólico de Rosenbrock con Criterio de Paro.

Análisis de varianza para los datos de: <i>Distancia al Óptimo con la Función Valle Parabólico de Rosenbrock</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	57.549	3	19.183	24.38	Rechazar Ho	0.000
Error	100.735	128	0.787			
TOTAL	158.284	131				

Los resultados presentados en la tabla 4.16 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FVPDR con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 19476, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 680. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 28.63$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.000$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

Tabla 4.16. Análisis de Varianza para el Número de Iteraciones de la Función Valle Parabólico de Rosenbrock con Criterio de Paro.

Análisis de varianza para los datos de: <i>Número de Iteraciones con la Función Valle Parabólico de Rosenbrock</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	58428	3	19476	28.63	Rechazar Ho	0.000
Error	87079	128	680			
TOTAL	145506	131				

Tabla 4.17. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Cúbica de Rosenbrock con Criterio de Paro.

Análisis de varianza para los datos de: <i>Eficiencia en la Función Ganancia con la Función Cúbica de Rosenbrock</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	1.076	3	0.359	0.66	Aceptar Ho	0.577
Error	69.323	128	0.542			
TOTAL	70.399	131				

Los resultados presentados en la tabla 4.17 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FCDR con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.359, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 0.542. Esto indica que las medias de los tratamientos son iguales. Sin embargo, que al obtener el valor de $F_0 = 0.66$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.577$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.18. Análisis de Varianza para la Distancia al Óptimo de la Función Cúbica de Rosenbrock con Criterio de Paro.

Análisis de varianza para los datos de:			<i>Distancia al Óptimo con la Función Cúbica de Rosenbrock</i>			
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	43.749	3	14.583	25.74	Rechazar Ho	0.000
Error	72.519	128	0.567			
TOTAL	116.269	131				

Por otro lado, los resultados presentados en la tabla 4.18 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FCDR con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 14.583, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 0.567. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 25.74$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.000$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.19. Análisis de Varianza para el Número de Iteraciones de la Función Cúbica de Rosenbrock con Criterio de Paro.

Análisis de varianza para los datos de: <i>Número de Iteraciones con la Función Cúbica de Rosenbrock</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	23589	3	7863	12.54	Rechazar Ho	0.000
Error	80263	128	627			
TOTAL	103852	131				

Los resultados presentados en la tabla 4.19 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FCDR con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 7863, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 627. Esto indica que las medias de los tratamientos no son iguales. Es decir, al obtener el valor de $F_0 = 12.54$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.000$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

Tabla 4.20. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función de Beale con Criterio de Paro.

Análisis de varianza para los datos de: <i>Eficiencia en la Función Ganancia con la Función de Beale</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	2.37	3	0.79	0.68	Aceptar Ho	0.566
Error	149.13	128	1.17			
TOTAL	151.5	131				

Los resultados presentados en la tabla 4.20 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FB con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.79, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 1.17. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.68$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.566$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.21. Análisis de Varianza para la Distancia al Óptimo de la Función de Beale con Criterio de Paro.

Análisis de varianza para los datos de: <i>Distancia al Óptimo con la Función de Beale</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	263.97	3	87.99	8.86	Rechazar Ho	0.000
Error	1271.23	128	9.93			
TOTAL	1535.2	131				

Los resultados presentados en la tabla 4.21 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FB con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 87.99, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 9.93. Esto indica que las medias de los tratamientos no son iguales. Sin embargo, al obtener el valor de $F_0 = 8.86$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras

palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.000$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Los resultados presentados en la tabla 4.22 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FB con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 1489, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 719. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 2.07$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.107$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en el NI del algoritmo modificado e iterativo.

Tabla 4.22. Análisis de Varianza para el Número de Iteraciones de la Función de Beale con Criterio de Paro.

Análisis de varianza para los datos de:			<i>Número de Iteraciones con la Función de Beale</i>			
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	4468	3	1489	2.07	Aceptar Ho	0.107
Error	92029	128	719			
TOTAL	96497	131				

Los resultados presentados en la tabla 4.23 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FSFC con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 592, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 971. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.61$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.610$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.23. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Suma de Funciones Cuadráticas con Criterio de Paro.

Análisis de Varianza para los datos de: <i>Eficiencia en la Función Ganancia con la Función Suma de Funciones Cuadráticas</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	1777	3	592	0.61	Aceptar Ho	0.610
Error	124350	128	971			
TOTAL	126127	131				

Tabla 4.24. Análisis de Varianza para la Distancia al Óptimo de la Función Suma de Funciones Cuadráticas con Criterio de Paro.

Análisis de varianza para los datos de: <i>Distancia al Óptimo con la Función Suma de Funciones Cuadráticas</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	2.516	3	0.839	0.99	Aceptar Ho	0.402
Error	108.941	128	0.851			
TOTAL	111.458	131				

Los resultados presentados en la tabla 4.24 muestra el Análisis de Varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FSFC con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.839, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 0.851. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.99$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.402$. Por lo tanto, hay razón para dudar de que el “Volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.25. Análisis de Varianza para el Número de Iteraciones de la Función Suma de Funciones Cuadráticas con Criterio de Paro.

Análisis de varianza para los datos de:			<i>Número de Iteraciones con la Función de Baile</i>			
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	1253	3	418	0.33	Aceptar Ho	0.806
Error	163640	128	1278			
TOTAL	164894	131				

Los resultados presentados en la tabla 4.25 muestra el Análisis de Varianza (ANOVA) para el Numero de Iteraciones (NI) para la función FSFC con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 418, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 1278. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.33$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño

factorial 2^k no influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.806$. Por lo tanto, hay razón para dudar de que el “Volumen” del diseño factorial 2^k influye significativamente en la NI del algoritmo modificado e iterativo.

Tabla 4.26. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Valle Parabólico de Rosenbrock sin Criterio de Paro.

<i>Eficiencia en la Función Ganancia con la Función Valle Parabólico de Rosenbrock sin paro</i>						
Análisis de varianza para los datos de:						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	0.246	3	0.082	0.17	Aceptar Ho	0.914
Error	60.488	128	0.473			
TOTAL	60.735	131				

Los resultados presentados en la tabla 4.26 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FVPDR sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.082, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 0.473. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.17$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.914$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.27. Análisis de Varianza para la Distancia al Óptimo de la Función Valle Parabólico de Rosenbrock sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Distancia al Óptimo con la Función Valle Parabólico de Rosenbrock sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	233.4	3	77.8	1.05	Aceptar Ho	0.373
Error	9486.3	128	74.1			
TOTAL	9719.7	131				

Por otro lado, los resultados presentados en la tabla 4.27 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FVPDR sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 77.8, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 74.1. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 1.05$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.373$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.28. Análisis de Varianza para el Número de Iteraciones de la Función Valle Parabólico de Rosenbrock sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Número de Iteraciones con la Función Valle Parabólico de Rosenbrock sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	14575	3	4858	17.74	Rechazar Ho	0.000
Error	35057	128	274			
TOTAL	49632	131				

Los resultados presentados en la tabla 4.28 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FVPDR sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 4858, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 274. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 17.74$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.000$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

Tabla 4.29. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Cúbica de Rosenbrock sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Eficiencia en la Función Ganancia con la Función Cúbica de Rosenbrock sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	7.50	3	2.5	1.05	Aceptar Ho	0.375
Error	306.1	128	2.39			
TOTAL	313.6	131				

Los resultados presentados en la tabla 4.29 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FCDR sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 2.5, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 2.39. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 1.05$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen”

del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.375$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.30. Análisis de Varianza para la Distancia al Óptimo de la Función Cúbica de Rosenbrock sin Criterio de Paro.

Análisis de varianza para los datos de:			<i>Distancia al Óptimo con la Función Cúbica de Rosenbrock sin paro</i>			
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	1.985	3	0.662	1.97	Aceptar Ho	0.122
Error	43.061	128	0.336			
TOTAL	45.045	131				

Por otro lado, los resultados presentados en la tabla 4.30 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FCDD sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.662, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 0.336. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 1.97$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.122$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.31. Análisis de Varianza para el Número de Iteraciones de la Función Cúbica de Rosenbrock sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Número de Iteraciones con la Función Cúbica de Rosenbrock sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	4408	3	1469	2.08	Aceptar Ho	0.107
Error	90572	128	708			
TOTAL	94980	131				

Los resultados presentados en la tabla 4.31 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FCDR sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 1469, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 708. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 2.08$ y compararlo con el valor de $F_{\alpha, v_1, v_2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.107$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

Tabla 4.32. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función de Beale sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Eficiencia en la Función Ganancia con la Función de Baile sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	1.12	3	0.37	0.13	Aceptar Ho	0.945
Error	381.84	128	2.98			
TOTAL	382.97	131				

Los resultados presentados en la tabla 4.32 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FB sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.37, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 2.98. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.13$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.945$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.33. Análisis de Varianza para la Distancia al Óptimo de la Función de Beale sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Distancia al Óptimo con la Función de Baile sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	16.6	3	5.53	3.57	Rechazar Ho	0.016
Error	198.41	128	1.55			
TOTAL	215.01	131				

Los resultados presentados en la tabla 4.33 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FB sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 5.53, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 1.55. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 3.57$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras

palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.016$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.34. Análisis de Varianza para el Número de Iteraciones de la Función de Beale sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Número de Iteraciones con la Función de Baile sin paro</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	4009	3	1336	1.79	Aceptar Ho	0.152
Error	95332	128	745			
TOTAL	99340	131				

Los resultados presentados en la tabla 4.34 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FB sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 1336, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 745. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 1.79$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.152$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

Tabla 4.35. Análisis de Varianza para la Eficiencia en la Función Ganancia de la Función Suma de Funciones Cuadráticas sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Eficiencia en la Función Ganancia con la Función Suma de Funciones Cuadradas sin paro.</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	1291	3	430	1.08	Aceptar Ho	0.359
Error	50885	128	398			
TOTAL	52176	131				

Los resultados presentados en la tabla 4.35 muestra el análisis de varianza (ANOVA) para la Eficiencia en la Función Ganancia (EFG) para la función FSFC sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 430, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 398. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 1.08$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la EFG del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.359$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influya significativamente en la EFG del algoritmo modificado e iterativo.

Tabla 4.36. Análisis de Varianza para la Distancia al Óptimo de la Función Suma de Funciones Cuadráticas sin Criterio de Paro.

Análisis de varianza para los datos de: <i>Distancia al Óptimo con la Función Suma de Funciones Cuadradas sin paro.</i>						
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	0.321	3	0.107	0.18	Aceptar Ho	0.908
Error	74.905	128	0.585			
TOTAL	75.226	131				

Los resultados presentados en la tabla 4.36 muestra el análisis de varianza (ANOVA) para la Distancia al Óptimo (DO) para la función FSFC sin criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 0.107, el cual es menor al de las medias de los tratamientos llamado “Error” igual a 0.585. Esto indica que las medias de los tratamientos son iguales. Es decir, que al obtener el valor de $F_0 = 0.18$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos no difieren. En otras palabras, que con un 95% de confianza el “Volumen” del diseño factorial 2^k no influye significativamente en la DO del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que no existen diferencias significativas entre los tratamientos, con un $p - value = 0.908$. Por lo tanto, hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en la DO del algoritmo modificado e iterativo.

Tabla 4.37. Análisis de Varianza para el Número de Iteraciones de la Función Suma de Funciones Cuadráticas sin Criterio de Paro.

Análisis de varianza para los datos de:				<i>Número de Iteraciones con la Función Suma de Funciones Cuadradas sin paro.</i>		
Fuente de Variación	Suma de cuadrados	Grados de Libertad	Media de Cuadrados	Fo	Decisión	*P<
Volumen	9122	3	3041	3.26	Rechazar Ho	0.024
Error	119487	128	933			
TOTAL	128609	131				

Los resultados presentados en la tabla 4.37 muestra el análisis de varianza (ANOVA) para el Número de Iteraciones (NI) para la función FSFC con criterio de paro, en la cual se pueden observar los valores de la media de cuadrados entre los experimentos llamado “Volumen” igual a 3041, el cual es mayor al de las medias de los tratamientos llamado “Error” igual a 933. Esto indica que las medias de los tratamientos no son iguales. Es decir, que al obtener el valor de $F_0 = 3.23$ y compararlo con el valor de $F_{\alpha, v1, v2} = 2.60$, se concluye que las medias de los tratamientos difieren. En otras palabras, que con un 95% de confianza el “Volumen”

del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

El ANOVA realizado a partir del experimento mostró que existen diferencias significativas entre los tratamientos, con un $p - value = 0.024$. Por lo tanto, no hay razón para dudar de que el “volumen” del diseño factorial 2^k influye significativamente en el NI del algoritmo modificado e iterativo.

4.4. Validación de los resultados obtenidos para el mejoramiento del C_{pk} con el AAEPMSM.

En esta sección se presentan los resultados obtenidos con el AAEPMSM que valida la hipótesis planteada en esta tesis. La validación se desarrolló multiplicando a la función del factor de ruido presentado en la sección 3.3. Apartado M3.3.i el valor de 0.1, como parte del fundamento de la precisión y exactitud que se presentan dentro de los sistemas de medición, dado que, son dos manifestaciones de la variabilidad en cualquier proceso de medición. Por lo tanto, la función quedaría como sigue:

$$\left[f(Z_i^{(\pm)} \pm v_i) + \varepsilon^{(\pm)} \right] * 0.1.$$

La precisión es la variación que presentan los resultados al medir varias veces una misma magnitud o midiendo con el mismo equipo (sus componentes principales son la repetitividad y la reproducibilidad). En otras palabras, la precisión es la habilidad de un instrumento de medición para repetir o reproducir su propia medición, con independencia de si dicha medición es correcta o incorrecta. La exactitud o sesgo se refiere al desfase desplazamiento que tienen las mediciones en relación al estándar o verdadero valor que se supone conocido: la exactitud se estima mediante la diferencia entre la media observada y el verdadero valor (N) del mensurando.

Para poder estudiar la exactitud se requiere contar con un estándar o patrón de modo que se pueda suponer la magnitud verdadera a medir. El objetivo de un estudio de repetibilidad y reproducibilidad llamado R&R es cuantificar la variabilidad que aportan a los datos el instrumento de medición (repetibilidad) y la que aportan los operadores (reproducibilidad). Así, la repetibilidad se refiere a la variabilidad de las mediciones sucesivas del mismo objeto con un instrumento y el mismo operador,

mientras que la reproducibilidad es la variabilidad de las mediciones que es atribuible a los diferentes operadores que miden una misma pieza.

Por lo tanto, el cálculo del índice de precisión/tolerancia (P/T) expresa en porcentaje la comparación de la expansión del error de medición (EM) con la variabilidad tolerada (ES-EI) para la característica de calidad que se está midiendo. De aquí que es deseable que EM sea más pequeño que la tolerancia para poder asegurar que la calidad del proceso de medición es aceptable. Además, este índice hace evidente que un instrumento de medición será preciso en función de la característica de calidad que se pretende medir. De manera específica, los criterios de aceptación del índice de P/T se interpretan como sigue: un resultado por debajo de 10% o 0.1 indica un excelente proceso, de 10 a 20% o 0.1 a 0.2 indica un bueno o aceptable, de 20 a 30% o 0.2 a 0.3 indica uno marginalmente aceptable, también arriba de 30% indica un inaceptable y debe ser corregido. (Gutiérrez y De la Vara, 2004)

4.4.1. Validación y Análisis de Resultados del AAEPSM con la Función Valle Parabólico de Rosenbrock (FVPDR).

Los resultados para la validación del algoritmo modificado e iterativo son presentados en la tabla 4.38 para la FVPDR con un criterio de paro. Se puede observar todos los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -5.77, además de que este valor decrementa en forma constante hasta la iteración veinte con un valor de -32.89; así en la iteración treinta este valor llega a -27.77 y se mantiene en constante incremento hasta la iteración cien con un valor del \bar{C}_{pk} igual a 0.5.

4.4.2. Análisis Gráfico para la Validación del AAEPSM con la Función Valle Parabólico de Rosenbrock (FVPDR).

El comportamiento gráfico para la validación del algoritmo modificado con la FVPDR con factor de ruido y sin factor de ruido; con criterio de paro, sus valores de la media aritmética de los treinta y tres $\hat{C}_{pk}(\bar{C}_{pk})$ que se obtienen para cada una de las cien iteraciones que se generan mediante la simulación de la implantación del

algoritmo modificado, se muestran en la figura 4.9, por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar de forma constante un decremento a partir de la primera iteración hasta la iteración seis con un valor del \bar{C}_{pk} de -1.49 hasta -2.74 respectivamente. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.08 unidades. A diferencia de la FVPDR sin factor de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a 0.31, además de que este valor decrementa constantemente hasta la iteración número cien con un valor de -3.46.

Tabla 4.38. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FVPDR con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-19.08	-9.74	-7.25	-5.77
2	-909.48	-241.81	-122.44	-24.84
3	-1066.11	-279.40	-138.90	-29.14
4	-1166.12	-302.48	-148.56	-31.52
5	-1238.09	-318.46	-155.22	-33.01
6	-1293.33	-330.22	-159.92	-33.99
7	-1337.46	-339.20	-163.35	-34.63
8	-1373.68	-346.20	-165.99	-35.05
9	-1404.00	-351.74	-168.01	-35.28
10	-1429.75	-356.14	-169.44	-35.39
20	-1559.41	-368.70	-170.10	-32.89
30	-1591.69	-357.69	-160.55	-27.77
40	-1586.31	-337.77	-147.43	-21.08
50	-1561.13	-313.26	-132.34	-15.53
60	-1523.91	-285.94	-116.08	-12.03
70	-1478.64	-256.63	-99.21	-8.73
80	-1427.69	-225.64	-80.53	-5.89
90	-1372.54	-193.52	-70.05	-2.47
100	-1314.18	-165.24	-62.25	0.50

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

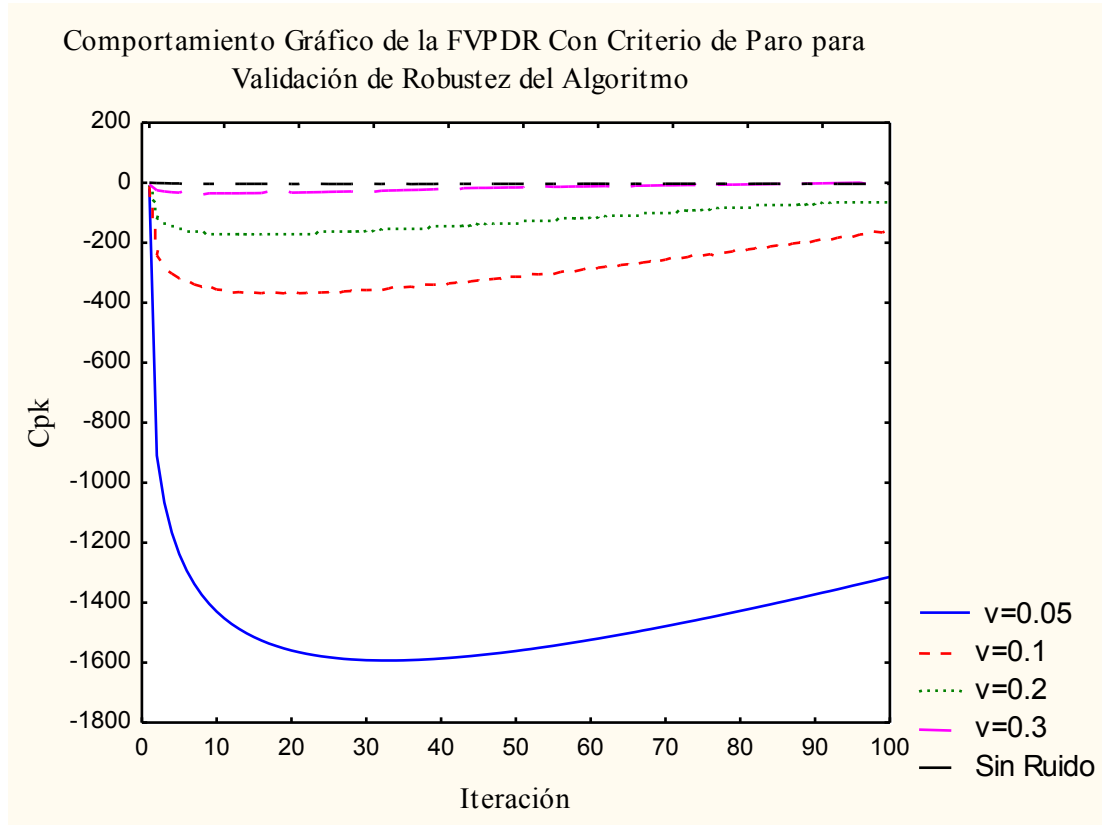


Figura 4.9. Gráfica de los \bar{C}_{pk} Generados para la Validación del Algoritmo Modificado con la FVPDR Con Criterio de Paro.

4.4.3. Validación y Análisis de Resultados del AAEPSM con la Función Cúbica de Rosenbrock (FCDR).

Los resultados para la validación del algoritmo modificado e iterativo son presentados en la Tabla 4.39 para la FCDR con un criterio de paro. Se pueden observar todos los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -3.71, además de que este valor decrementa en forma constante hasta la iteración cinco con un valor de -17.12; así en la iteración diez este valor llega a -16.22. Además, también ocurre que a partir de la iteración diez hasta la iteración cincuenta estos valores se incrementan constantemente con un valor del \bar{C}_{pk} de -16.22 hasta -0.62 respectivamente; Por otro lado, vuelve a comportarse con un valor decremental de la iteración sesenta a la iteración cien con un valor del \bar{C}_{pk} de -2.24 hasta -12.70 respectivamente.

Tabla 4.39. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FCDR con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-12.47	-8.63	-5.39	-3.71
2	-580.40	-146.34	-31.82	-14.66
3	-646.90	-162.17	-35.50	-16.32
4	-688.74	-171.82	-37.19	-16.91
5	-718.30	-178.42	-38.18	-17.12
6	-740.52	-183.19	-38.74	-17.10
7	-757.87	-186.76	-39.01	-17.00
8	-771.75	-189.46	-39.08	-16.82
9	-783.05	-191.53	-38.99	-16.53
10	-792.35	-193.11	-38.79	-16.22
20	-829.56	-195.24	-33.59	-11.79
30	-824.62	-187.18	-26.06	-5.32
40	-803.60	-175.32	-17.76	-2.11
50	-774.27	-161.55	-13.07	-0.62
60	-739.96	-146.63	-12.50	-2.24
70	-702.40	-131.07	-15.17	-4.90
80	-662.56	-112.64	-19.20	-7.61
90	-620.78	-122.07	-22.78	-10.09
100	-582.39	-132.99	-28.10	-12.70

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

4.4.4. Análisis Gráfico para la Validación del AAEPMS con la Función Cúbica de Rosenbrock (FCDR).

El comportamiento gráfico para la validación del algoritmo modificado con la FCDR con factor de ruido y sin factor de ruido; con criterio de paro; los valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones que se generan mediante la simulación de la implantación del algoritmo modificado, se muestran en la figura 4.10, por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.05 unidades, se puede observar que el valor de la iteración uno es de -12.47, además de que decrementa constantemente hasta la iteración veinte con un valor de -829.56, y a partir de la iteración treinta incrementa de forma constante hasta la iteración cien el valor del \bar{C}_{pk} , dado que pasa de un valor aproximado de -824.62 a otro de -582.39 respectivamente. A diferencia de la FCDR sin factor de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -0.56, además de que este valor decrementa constantemente

hasta la iteración número diez con un valor de -1.26; y a partir de la iteración once a la cien hay un constante incremento de un valor de -1.26 a 2.84 respectivamente.

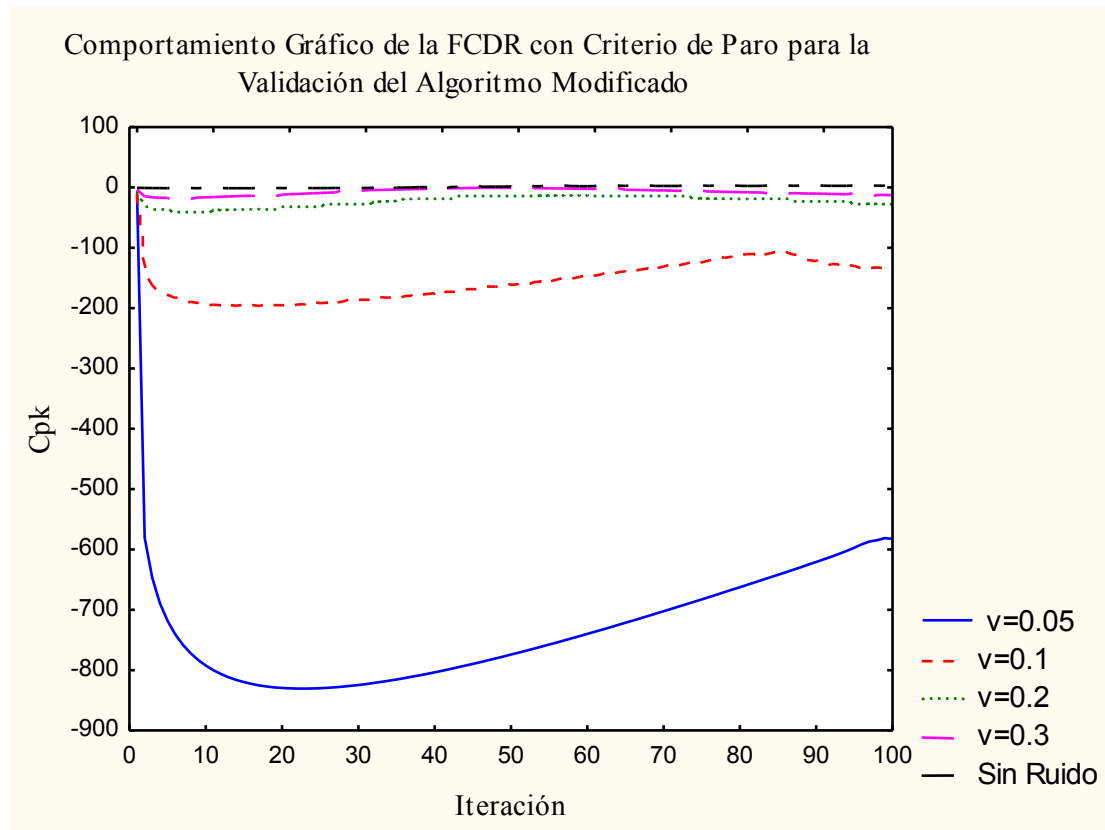


Figura 4.10. Gráfica de los \bar{C}_{pk} Generados para la Validación del Algoritmo Modificado con la FCDR Con Criterio de Paro.

4.4.5. Validación y Análisis de Resultados del AAEPSM con la Función de Beale (FB).

Los resultados presentados en la tabla 4.40 para la FB con el algoritmo modificado con un criterio de paro, se pueden observar todos los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a -0.69 y que este valor se incrementa en forma constante de la iteración ocho hasta la iteración veinte con un valor del \bar{C}_{pk} de -0.53 hasta 2.35. Además, también ocurre que a partir de la iteración cincuenta estos valores se estabilizan, en el sentido de que la máxima diferencia entre estos valores no excede de 0.4 unidades.

Tabla 4.40. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FB con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	-4.07	-2.09	-1.01	-0.69
2	-0.69	-0.49	-1.30	-0.88
3	-16.12	-4.90	-1.54	-0.65
4	-16.26	-5.01	-1.46	-0.88
5	-17.01	-4.81	-1.43	-0.79
6	-16.97	-4.90	-1.37	-0.35
7	-16.87	-4.68	-1.23	-0.39
8	-16.96	-4.41	-1.21	-0.53
9	-16.89	-4.26	-0.63	-0.16
10	-17.02	-3.97	-0.36	0.11
20	-14.98	-2.94	0.95	2.35
30	-14.03	-2.59	0.98	2.28
40	-13.29	-2.75	0.29	1.30
50	-13.54	-3.08	0.30	1.57
60	-12.75	-3.19	0.61	1.80
70	-12.20	-3.52	0.80	1.46
80	-11.39	-3.36	1.32	1.56
90	-11.10	-4.65	1.22	1.58
100	-13.69	-4.77	1.23	1.72

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

4.4.6. Análisis Gráfico para la Validación del Algoritmo Modificado con la Función de Beale (FB).

El comportamiento gráfico para la validación del algoritmo modificado con la FB con factor de ruido y sin factor de ruido, con criterio de paro; sus valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones se generan mediante la simulación de la implantación del algoritmo modificado, como se muestran en la figura 4.11. Por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar que el valor de la iteración uno es de -1.01. Además, se observa de forma constante un incremento a partir de la iteración cuatro hasta la iteración cien con un valor máximo del (\bar{C}_{pk}) de 1.32. Además, ocurre que a partir de la iteración ochenta los valores del (\bar{C}_{pk}) se estabilizan en el sentido de que la máxima diferencia entre estos valores no excede de 0.1 unidades. A diferencia de la FB sin factores de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a -1.54, además de que este

valor se incrementa inmediatamente en la iteración número dos hasta -0.54; Además, ocurre que de la iteración tres a la iteración nueve decremantan estos valores del (\bar{C}_{pk}) de -1.7 a -2.16 unidades respectivamente, y a partir de la iteración diez a la cien hay un constante incremento de un valor de -2.12 a 0.53 respectivamente.

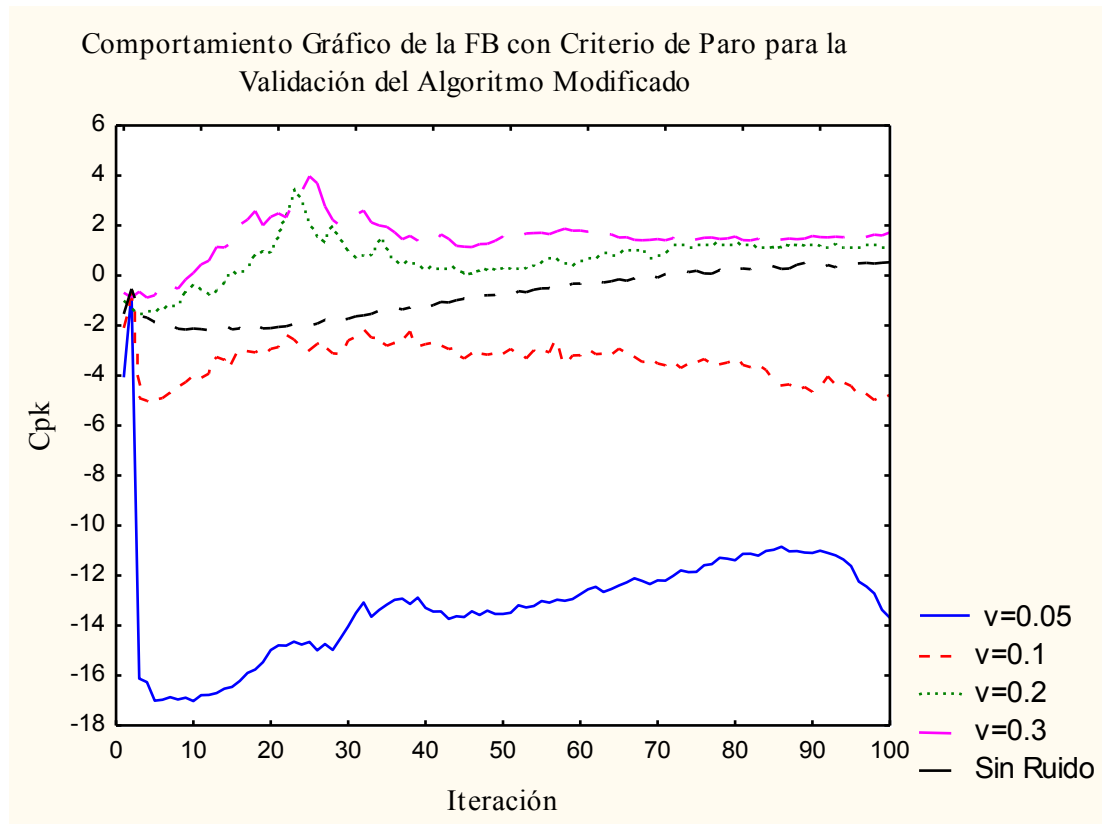


Figura 4.11. Gráfica de los \bar{C}_{pk} Generados para la Validación del Algoritmo Modificado con la FCDR Con Criterio de Paro.

4.4.7. Validación y Análisis de Resultados del AAEPSM con la Función Suma de Funciones Cuadráticas (FSFC).

Los resultados presentados en la tabla 4.41 para la FSFC con el algoritmo modificado con un criterio de paro, se pueden observar los valores de los \bar{C}_{pk} por cada uno de los volúmenes del diseño factorial 2^k (considerados como los factores de ruido). Por ejemplo, para el volumen del diseño factorial 2^k de 0.3 unidades, se puede observar que en la iteración número 1 o en el punto inicial el valor del \bar{C}_{pk} es igual a 22.72 y que este valor decremanta en forma constante hasta la iteración cien.

4.4.8. Análisis Gráfico para la Validación del AAEPSM con la Función Suma de Funciones Cuadráticas (FSFC).

El comportamiento gráfico para la validación del algoritmo modificado con la FSFC con factor de ruido y sin factor de ruido, con criterio de paro; sus valores de la media aritmética de los treinta y tres \hat{C}_{pk} (\bar{C}_{pk}) que se obtienen para cada una de las cien iteraciones se generan mediante la simulación de la implantación del algoritmo modificado, como se muestran en la figura 4.12. Por ejemplo: para el factor de ruido del diseño factorial 2^k con volumen de 0.2 unidades, se puede observar que el valor de la iteración uno es de 20.77. Además, se observa de forma constante un decremento a partir de la iteración dos hasta la iteración cien con una valor del (\bar{C}_{pk}) de -488.64 hasta -2298.2 respectivamente. A diferencia de la FSFC sin factores de ruido, en la que se puede observar que en la iteración número uno el valor del \bar{C}_{pk} es igual a 2.1, además de que este valor decrementa constantemente hasta la iteración cien con un valor del (\bar{C}_{pk}) de -12.9.

Tabla 4.41. Valores del \bar{C}_{pk} por Iteración y por Factor de Ruido para la Función de Prueba FSFC con Criterio de Paro.

Iteración	*Factor de Ruido			
	0.05	0.1	0.2	0.3
1	24.07	22.58	20.77	22.72
2	-2038.12	-1051.30	-488.64	-327.27
3	-54909.81	-7840.04	-1202.05	-481.20
4	-75297.94	-10246.98	-1423.60	-560.20
5	-87294.50	-11345.69	-1530.93	-651.82
6	-93457.37	-12084.82	-1633.59	-622.11
7	-102117.36	-13159.71	-1764.45	-644.80
8	-105984.45	-13506.98	-1796.49	-702.46
9	-107008.86	-13684.04	-1802.49	-639.26
10	-108555.40	-13878.73	-1832.26	-659.01
20	-116051.46	-14828.03	-1996.03	-717.54
30	-121039.50	-15344.14	-2097.97	-773.71
40	-122465.53	-15533.14	-2152.10	-841.33
50	-123646.88	-15700.04	-2191.65	-870.79
60	-124732.39	-15875.32	-2219.11	-896.38
70	-125220.56	-15988.05	-2242.25	-925.43
80	-126004.88	-16114.88	-2268.17	-923.89
90	-126229.03	-16192.82	-2286.02	-941.45
100	-126527.72	-16232.01	-2298.20	-959.13

*Factor de ruido es representado por cada volumen del diseño factorial 2^k

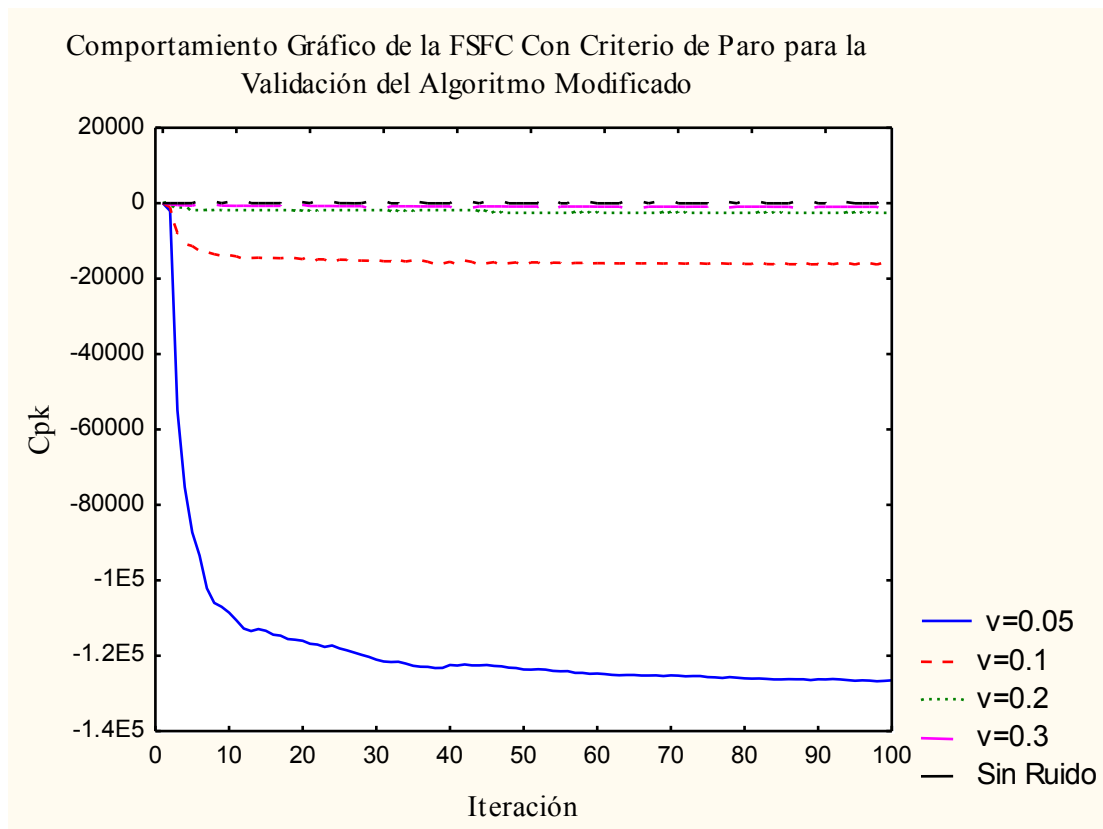


Figura 4.12. Gráfica de los \bar{C}_{pk} Generados para la Validación del Algoritmo Modificado con la FSFC Con Criterio de Paro.

4.5. Discusión de Resultados.

En esta sección se presenta una discusión de los resultados que se obtienen en la verificación de las hipótesis planteadas en esta tesis, el proceso de simulación del AAEPSM; finalmente se presenta una discusión general comparando los resultados de la simulación del AAEPSM con criterio de paro y sin criterio de paro, así como, el análisis del AAEPS sin factores de ruido con criterio de paro y sin criterio de paro.

4.5.1. Comprobación de las Hipótesis.

De acuerdo con los resultados mostrados al verificar las hipótesis planteadas en esta tesis por medio de simulación, se obtiene que el algoritmo modificado e iterativo: 1) es efectivo o el valor máximo del C_{pk} es independiente del punto inicial; 2) logra mejorar el C_{pk} con factores de ruido; 3) La eficacia del algoritmo es afectada por el volumen del diseño factorial 2^k , considerado como factor de ruido.

4.5.2. Discusión de los Resultados de Simulación.

En el proceso de simulación del AAEPSM en las cuatro funciones de prueba, se observa que para la primera hipótesis que dice que el algoritmo modificado e iterativo es efectivo o el valor máximo del C_{pk} es independiente del punto inicial. Esta hipótesis es rechazada, debido a que se muestra un ejemplo en el que, con un nivel de significancia del 5%, se contradice esta afirmación. Lo anterior, implica que la efectividad del algoritmo modificado e iterativo o que el valor máximo del C_{pk} dependen del punto inicial, ya que es un parámetro que afecta el desempeño del algoritmo con factores de ruido.

Por otro lado, para la segunda hipótesis que dice, el AAEPSM logra mejorar el C_{pk} con factores de ruido. En esta hipótesis se observa que en todas las gráficas presentadas por cada función de prueba el algoritmo logra mejorar el C_{pk} . Es decir, las mejoras que se obtienen considerando en cuantas ocasiones ocurre que el valor $E(C_{pk}^{k+1}) \geq E(C_{pk}^k)$ en un intervalo de 1 a 100 iteraciones. Los resultados obtenidos muestran un valor promedio de 59 %, considerando todos los resultados de las cuatro funciones de prueba con criterio de paro, y en 49 % sin criterio de paro.

Comparando los resultados anteriores con respecto a los valores promedios obtenidos por el algoritmo sin factores de ruido, con criterio de paro y sin criterio de paro son muy similares. Por ejemplo, en el caso de los resultados del algoritmo sin factores de ruido con criterio de paro, el valor promedio es de 53 %, y el de sin criterio de paro es de 51 %. Por lo tanto, podemos afirmar que el algoritmo modificado e iterativo logra mejorar el C_{pk} . Además, los resultados de validación obtenidos mediante el algoritmo modificado e iterativo muestran que el valor $E(C_{pk}^{k+1}) \geq E(C_{pk}^k)$ en promedio es de 54%, considerando todos los resultados de las cuatro funciones de prueba con criterio de paro. Por otro lado, el valor del C_{pk} mejora con respecto al aumento en el tamaño del volumen del diseño factorial 2^k , dado que se obtienen los valores máximos en los volúmenes $v = 0.3$ en cada función de prueba analizada.

Por último, para la tercera hipótesis que dice, la eficacia del algoritmo modificado e iterativo es afectada por el volumen del diseño factorial 2^k . Los resultados obtenidos con el algoritmo para las cuatro funciones de prueba, consistió en evaluar la eficacia del algoritmo con factores de ruido, los cuales son considerados con cuatro volúmenes diferentes de un diseño factorial 2^k . Las variables utilizadas para la comprobación de esta hipótesis son: la Eficiencia en la Función Ganancia, la Distancia al óptimo y el Número de Iteraciones obtenidas al resultar el máximo C_{pk} .

La tabla 4.42 y 4.43 muestran el resumen de los resultados de la significancia por cada función de prueba. La tabla 4.42 muestra los resultados del algoritmo por cada función de prueba con criterio de paro; la tabla 4.43 muestra los resultados del algoritmo por cada función de prueba sin un criterio de paro.

Tabla 4.42. Resultados de Significancia por Cada Función de Prueba Con Criterio de Paro para el AAEPSM.

Función de prueba	Medidas de Rendimiento		
	EFG	DO	NI
Valle Parabólico de Rosenbrock	No Significativo	Significativo	Significativo
Cúbica de Rosenbrock	No Significativo	Significativo	Significativo
Beale	No Significativo	Significativo	No Significativo
Suma de Funciones Cuadráticas	No Significativo	No Significativo	No Significativo

Los resultados presentados en la tabla 4.42 indican que existe suficiente evidencia estadística, con un nivel de significancia del cinco por ciento $\alpha = 0.05$, para afirmar que el volumen del diseño factorial 2^k afecta la eficacia del AAEPSM en la variable Eficiencia en la Función Ganancia para las cuatro funciones de prueba. Así mismo, observe que la Distancia al Óptimo, generada con el algoritmo para las funciones de prueba: Valle Parabólico de Rosenbrock, Cúbica de Rosenbrock y Beale, no están afectadas por el volumen del diseño factorial 2^k , a diferencia de la Función de Suma de Funciones Cuadráticas, que es afectada por dicho volumen, a un nivel de significancia del cinco por ciento $\alpha = 0.05$. Por último, observe que el Número de Iteraciones generado con el algoritmo para las funciones de prueba: Valle Parabólico de Rosenbrock y Cúbica de Rosenbrock no están afectadas por el volumen del diseño

factorial 2^k , a diferencia de la Función de Beale y Función de Suma de Funciones Cuadráticas que es afectada por dicho volumen. A un nivel de significancia del cinco por ciento $\alpha = 0.05$.

Tabla 4.43. Resultados de Significancia por Cada Función de Prueba Sin Criterio de Paro para el AAEPSM.

Función de prueba	Variable de Respuesta		
	EFG	DO	NI
Valle Parabólico de Rosenbrock	No Significativo	No Significativo	Significativo
Cúbica de Rosenbrock	No Significativo	No Significativo	No Significativo
Beale	No Significativo	Significativo	No Significativo
Suma de Funciones Cuadráticas	No Significativo	No Significativo	Significativo

Los resultados presentados en la tabla 4.43 indican que existe suficiente evidencia estadística, con un nivel de significancia del cinco por ciento $\alpha = 0.05$, para afirmar que el volumen del diseño factorial 2^k afecta la eficacia del AAEPSM en la variable Eficiencia en la Función Ganancia para las cuatro funciones de prueba. Así mismo, observe que la Distancia al Óptimo, generadas con el AAEPSM para las funciones de prueba: Valle Parabólico de Rosenbrock, Cúbica de Rosenbrock y Suma de Funciones Cuadráticas, están afectadas por el volumen del diseño factorial 2^k , a diferencia de la Función de Beale, que no es afectada por dicho volumen, a un nivel de significancia del cinco por ciento $\alpha = 0.05$. Referente al Número de Iteraciones realizados por el AAEPSM, para las funciones de prueba: Valle Parabólico de Rosenbrock y Suma de Funciones Cuadráticas, no son afectadas por el volumen del diseño factorial 2^k , a diferencia de las funciones de prueba: Cúbica de Rosenbrock y Beale, las cuales son afectadas, con un nivel de significancia del cinco por ciento $\alpha = 0.05$.

Los valores promedios de las variables EFG, DO y NI generados con el AAEPSM con un criterio de paro y sin un criterio de paro, para cada función de prueba se presentan, en las tablas 4.44 y 4.45 respectivamente. Los resultados referentes a la tabla 4.40 son: para la variable EFG el valor más alto lo obtuvo la función de prueba Cúbica de Rosenbrock, con un 99.7 por ciento. Mientras que la función de Suma de Funciones Cuadráticas obtuvo el valor más bajo con un 45.2 por ciento. Por otro lado,

para la variable DO nótese que para la máxima distancia de separación de los óptimos generados por el AAEPMS y el óptimo verdadero por función de prueba es de 3.7, que pertenece a la función de prueba de Beale, la mínima distancia es de 1.8, que pertenece a la función de prueba Cúbica de Rosenbrock. Por último, los resultados obtenidos para la variable NI presentan que la función de prueba Suma de Funciones Cuadráticas es la que logra obtener el máximo valor del C_{pk} en la menor cantidad de iteraciones con 23, la que logra en la mayor cantidad de iteraciones es la de Beale con 67 iteraciones. A diferencia de las funciones Valle Parabólico de Rosenbrock y Cúbica de Rosenbrock que son de 53 y 38 respectivamente.

Tabla 4.44. Valores Promedios de las Variables de Respuesta para la hipótesis 1.4.3 para cada Función de Prueba Con Criterio de Paro para el AAEPMS.

Función de prueba	Variables de Respuesta		
	EFG	DO	NI
Valle Parabólico de Rosenbrock	99.6	1.8	53
Cúbica de Rosenbrock	99.7	1.9	38
Beale	99.4	3.7	67
Suma de Funciones Cuadráticas	45.2	2.8	23

Los resultados referentes a la tabla 4.45 para la variable EFG el valor más alto lo obtuvo la función de prueba Valle Parabólico de Rosenbrock, con un 99.7 por ciento. Mientras que la función de Suma de Funciones Cuadráticas obtuvo el valor más bajo con un 81 por ciento. Por otro lado, para la variable DO nótese que para la máxima distancia de separación de los óptimos generados por el AAEPMS y el óptimo verdadero por función de prueba es de 3.3, que pertenece a la función de prueba Valle Parabólico de Rosenbrock, la mínima distancia es de 1.5, que pertenece a la función de prueba Cúbica de Rosenbrock. Por último, los resultados obtenidos para la variable NI son que la función de prueba Cúbica de Rosenbrock es la que logra obtener el máximo C_{pk} en la menor cantidad de iteraciones con 7, la que logra en la mayor cantidad de iteraciones es la función de prueba Suma de Funciones Cuadráticas con 48 iteraciones. A diferencia de las funciones Valle Parabólico de Rosenbrock y Beale que son de 11 y 15 respectivamente.

Tabla 4.45. Valores Promedios de las Variables de Respuesta para la hipótesis 1.4.3 para cada Función de Prueba Sin Criterio de Paro para el AAEPS.

Función de prueba	Variables de Respuesta		
	EFG	DO	NI
Valle Parabólico de Rosenbrock	99.7	3.3	11
Cúbica de Rosenbrock	99.6	1.5	7
Beale	99.3	2.0	15
Suma de Funciones Cuadráticas	81	1.8	48

En general, los resultados que se obtienen, sobre el comportamiento observado del valor del C_{pk} , a través de la implantación del AAEPSM con factores de ruido en los casos de contar con un criterio de paro y sin contar con un criterio de paro muestran que el algoritmo modificado e iterativo es viable para problemas del Diseño de Parámetro Robusto.

Capítulo V. Conclusiones y Recomendaciones.

En este capítulo se presentan las conclusiones generales que se obtienen con el desarrollo de esta tesis, conclusiones que se generan de la comprobación tanto de las hipótesis como de los objetivos establecidos en el capítulo I, además de la simulación experimental del AAEPSM con factores de ruido. También se exponen las recomendaciones que constituyen las propuestas de proyectos de investigación futuros.

5.1. Conclusiones.

El AAEPSM es una herramienta útil para el DPR y el mejoramiento de medidas de rendimiento de procesos de simulación, dado que, se desconoce la relación funcional entre su variable de respuesta, sus factores controlables y sus factores de ruido, por el hecho de que el método de perturbación simultánea posee las características de ser iterativo, recursivo y converger hacia un punto que es candidato a ser el óptimo del modelo a optimizar.

A continuación se analizan con más detalle las hipótesis planteadas al inicio de esta tesis, con el fin de verificar si se aceptan o rechazan con base a los resultados obtenidos en la simulación experimental. Referente a la primer hipótesis planteada que menciona que si la efectividad del AAEPSM es independiente de las condiciones iniciales de operación X_0 en las “primeras” iteraciones. Esta hipótesis se puede considerar falsa ya que los resultados mostraron que la efectividad del AAEPSM o el valor máximo del C_{pk} dependen del punto inicial, ya que es un parámetro que afecta el desempeño del algoritmo.

La segunda hipótesis planteada expresa que el AAEPSM logra mejorar el C_{pk} . Esta hipótesis se puede considerar verdadera ya que en los resultados obtenidos se verificó que el algoritmo con un criterio de paro y sin un criterio de paro mejoran el valor del \bar{C}_{pk} en promedio en 59 % y 49 %, respectivamente, considerando todos los resultados de las cuatro funciones de prueba. Además, los resultados de validación obtenidos mediante el algoritmo propuesto muestran que el valor del C_{pk} en promedio es de 54 %, considerando todos los resultados de las cuatro funciones de prueba con criterio de

paro. Por otro lado, también se obtuvo que el valor del C_{pk} mejora con respecto al aumento en el tamaño del volumen del diseño factorial 2^k , considerado como el arreglo externo de la matriz del diseño cruzado del algoritmo. Dado que se obtienen los valores máximos en los tamaños de volumen $v = 0.3$ en cada función de prueba analizada.

La tercera hipótesis planteada expresa que la eficacia del algoritmo propuesto es afectada por el volumen del diseño factorial 2^k . Las variables utilizadas para la comprobación de esta hipótesis son: la Eficiencia en la Función Ganancia, la Distancia al óptimo y el Número de Iteraciones obtenidas al resultar el valor máximo del \bar{C}_{pk} .

Para el caso del algoritmo con un criterio de paro, existe suficiente evidencia estadística, con un nivel de significancia del cinco por ciento $\alpha = 0.05$, para afirmar que el volumen del diseño factorial 2^k afecta la eficacia del AAEPSM en la variable Eficiencia en la Función Ganancia, Así mismo, que la Distancia al Óptimo, generada con el algoritmo para las funciones de prueba: Valle Parabólico de Rosenbrock, Cúbica de Rosenbrock y Beale, no están afectadas por el volumen del diseño factorial 2^k , a diferencia de la Función de Suma de Funciones Cuadráticas, que es afectada por dicho volumen, a un nivel de significancia del cinco por ciento $\alpha = 0.05$. Por último, se obtuvo que el Número de Iteraciones generado con el algoritmo para las funciones de prueba: Valle Parabólico de Rosenbrock y Cúbica de Rosenbrock no están afectadas por el volumen del diseño factorial 2^k , a diferencia de la Función de Beale y Función de Suma de Funciones Cuadráticas que son afectadas por dicho volumen. A un nivel de significancia del cinco por ciento $\alpha = 0.05$.

Por otro lado, para el caso del algoritmo sin un criterio de paro, existe suficiente evidencia estadística, con un nivel de significancia del cinco por ciento $\alpha = 0.05$, para afirmar que el volumen del diseño factorial 2^k afecta la eficacia del AAEPSM en la variable Eficiencia en la Función Ganancia para las cuatro funciones de prueba. Así mismo, observe que la Distancia al Óptimo, generadas con el AAEPSM para las funciones de prueba: Valle Parabólico de Rosenbrock, Cúbica de Rosenbrock y Suma de Funciones Cuadráticas, están afectadas por el volumen del diseño factorial 2^k , a

diferencia de la Función de Beale, que no es afectada por dicho volumen, a un nivel de significancia del cinco por ciento $\alpha = 0.05$. Referente al Número de Iteraciones realizados por el AAEPSM, para las funciones de prueba: Valle Parabólico de Rosenbrock y Suma de Funciones Cuadráticas, no son afectadas por el volumen del diseño factorial 2^k , a diferencia de las funciones de prueba: Cúbica de Rosenbrock y Beale, las cuales son afectadas, con un nivel de significancia del cinco por ciento $\alpha = 0.05$.

Con relación al objetivo general de esta tesis se puede concluir que se cumplió, al igual que con los objetivos específicos, dado que se validó el desarrollo del AAEPSM para comprobar su efectividad y su independencia de las condiciones iniciales de operación X_0 en las “primeras” iteraciones; comprobar que logra mejorar el valor del C_{pk} ; y comprobar que su eficacia es afectada por el volumen del diseño factorial 2^k . El AAEPSM proporciona una solución al problema de contar con una herramienta para el DPR que mejora el valor del índice de capacidad del proceso C_{pk} , en el desarrollo experimental por medio de simulación.

5.2. Recomendaciones.

Se propone implementar el AAEPSM en casos prácticos, con todos sus valores continuos, discretos o mixtos (continuos y discretos), con el objetivo de incrementar el desempeño de este algoritmo, con respecto al mejoramiento del índice de capacidad real de los procesos o alguna otra medida de rendimiento.

Se propone determinar el tipo de relación que se establece entre el punto inicial de operación y los valores de los índices de capacidad real de los procesos en las “primeras” iteraciones con el AAEPSM, con el objetivo de establecer una selección óptima del punto inicial.

Se propone verificar la combinación óptima del vector de perturbación simultánea Δ_k tal que maximice el valor del C_{pk} , con todos sus valores continuos, discretos o mixtos (continuos y discretos).

Se propone establecer nuevos criterios de paro para el AAEPSM con todos sus valores continuos, discretos o mixtos (continuos y discretos), con el objetivo de incrementar el desempeño de este algoritmo, con respecto al mejoramiento de medidas de rendimiento.

Capítulo VI. Bibliografía.

- Andradóttir, S., (1995) i), A stochastic Approximation Algorithm with Varying Bounds, Operations Research, volumen 43, número 6, páginas 1037-1048.
- Andradóttir, S., (1995) ii), A method for Discrete Stochastic Approximation, Management Science, volumen 41, número 12, páginas 1946-1961.
- Andradóttir, S., (1996), A Scaled Stochastic Approximation Algorithm, Management Science, volumen 42, número 4, páginas 475-498.
- Blum, J.R., (1954), Multidimensional Stochastic Approximation Methods, Annals of Mathematical Statistic, volumen 25, páginas 737-744.
- Box, G.E.P., (1957), Evolutionary Operation: A Method for Increasing Industrial Productivity. Applied Statistics, número 6, No. 2, páginas 110-115.
- Box, G.E.P., (1999), Statistics as a catalyst to learning by scientific method, part II-a discussion, Journal of Quality Technology, número 31, Vol. 1, páginas 16-29.
- Box, G.E.P., y Behnken, D.W., (1960), Some new three level designs for the study of quantitative variables. Technometrics, número 2.
- Box, G.E.P., y Draper, N.R., (1969), Evolutionary Operation: A Statistical Method for Process Management. John Wiley & Sons, Inc., Toronto.
- Box, G.E.P., y Draper, N.R., (1975), Robust Designs. Journal of Biometrika, número 62, páginas 347-352.
- Box, G.E.P., y Draper, N.R., (1987), Empirical Model Building and Response Surfaces. New York: John Wiley & Sons.
- Box, G.E.P., y Hunter, J.S., (1957), Multi-factor Experimental Desings for Exploring Response Surfaces. Annals of Mathematical Statistics, volumen 28, 195-241.
- Box, G.E.P., y Wilson, K.B., (1951), On the Experimental Attainment of Optimum Conditions. Journal of the Royal Statistical Society Series B13, número 1, páginas 1-38.
- Chen, H.F., (1985), Recursive Estimation and Control for Stochastic Systems, John Wiley & Sons, primera edición, Estados Unidos.

- Chin, D.C., (1997), Comparative Study of Stochastic Algorithms for System Optimization Based on Gradient Approximation, IEEE Transaction on Systems, Man, and Cybernetics-partb: Cybernetics, volumen 27, número 2, páginas 244-249.
- Chung, K.L., (1954), On a Stochastic approximation Method, Annals of Mathematical Statistica, número 25, páginas 463-483.
- Copeland, K.A.F. y Nelson, P.R., (1996), Dual Response Optimization via Direct Function Minimization, Journal of Quality Technology, volumen 28, número 3, páginas 331-336.
- Cornell, J.A., (1990), How to apply Response Surface Methodology. American Society for Quality Control.
- Cruz T. E. A., Medina V. P. D. y Silva D. C. A., (2012), Una revisión crítica de la Razón Señal Ruido usada por Taguchi, Scientia et Technica, Año XVII, No. 50, páginas 52-56.
- Delyon, B., (1996), General Results on the Convergence of Stochastic Algorithms, IEEE Transaction on Automatic Control, volumen 41, número 3, páginas 1245-1255.
- Donohue, J.M., Houck, E.C., Myers, R.H., (1993), Simulation designs and correlation induction for reducing-order bias in first-order response surfaces. Operations Research, Vol. 41, Num. 5, páginas 880-902.
- Donohue, J.M., Houck, E.C., Myers, R.H., (1995), Simulation designs for the estimation of response surface gradients in the presence of model misspecification. Management Science Vol. 41, Num. 2, páginas 244-262.
- Draper, N.R., y Smith H., (1981), Applied Regression Analysis. Wiley series In Probability and Mathematical Statistics, New York: John Wiley & Sons.
- Fabian, V., (1968), On Asymptotic Normality in Stochastic Approximation, Annals of Mathematical Statistic, número 39, páginas 1327-1332.
- Franklin, L.A., y Wasserman, G.S., (1992), Bootstrap Lower Confidence Limits for Capability Indices, Journal of Quality Technology, volumen 24, num. 4, páginas 196-210.

- Fu, M.C., y Hill, S.D., (1997), Optimization of Discrete Event Systems Via Simultaneous Stochastic Approximation, IIE Transactions, número 29, páginas 233-243.
- García M.R., (1993), Evaluación de la Metodología de Superficie de Respuesta. Tesis de Maestría en Ingeniería Industrial, Instituto Tecnológico de Cd. Juárez.
- García M.R., (2000), Algoritmo de Aproximaciones Estocásticas para el Mejoramiento del Índice de Capacidad Real de Procesos CPK. Tesis de Doctorado en Ingeniería Industrial, Instituto Tecnológico de Cd. Juárez.
- Gutiérrez, P.H., (1997), Calidad Total y Productividad, McGraw Hill, primera Edición, México, D. F.
- Hood, S.J., Welch, P.D., (1993), Response surface methodology and its application in simulation. In: Evans, G.W., Mollaghasemi, M., Russell, E.C., Biles, W.E. (Eds.), Proceedings of the 1993 Winter Simulation Conference, páginas 115–122.
- Irizarry, M., Wilson, J.R., Trevino, J., (2001), A flexible simulation tool for manufacturing-cell design, II: Response surface analysis and case study. IIE Transactions Vol. páginas 33, 837–846.
- Kenworthy, I.C., (1967), Some Examples of Simplex Evolutionary Operation in the paper Industry. Applied Statistics, volumen 16, número 3, páginas 211-24.
- Khuri, A.I., (1996), Multiresponse Surface Methodology, In: Ghosh, S., Rao, C.R. (Eds), Handbook of statistics, Vol. 13 ElSevier, Amsterdam.
- Khuri, A.I., y Cornell, J.A, (1996), Response Surface: Design and Analyses. New York: Marcel Dekker, Inc.
- Kiefer, J., y Wolfowitz, J., (1952), Stochastic Estimation of the Maximun of a Regression Function, Annals of Mathematical Statistic, volumen 23, páginas 462-466.
- Kleijnen, J.P.C., (1998), Experimental Design for Sensitivity Analysis, Optimization and Validation of Simulation Models. In Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice. New York: John Wiley & Sons.

- Kleijnen, J.P.C., Hertog D. den y Angun E, (2004), Response Surface Methodology's Steepest Ascent and Step Size Revisited, European Journal of Operation Research, Vol. 159, páginas 121-131.
- Kleijnen, J.P.C., Hertog D. den y Angun E, (2006), Response surface methodology's steepest Accent and step size revisited: correction, European Journal of Operation Research, Vol. 170, páginas 664-666.
- Kuehl, R.O., (2001), Diseño de Experimentos: Principios Estadísticos para el Diseño y Análisis de Investigaciones. Segunda Edición en Español. Editorial Thompson Learning.
- Kushler, R.H., y Hurley, P., (1992), Confidence Bounds for Capability Indices, Journal of Quality Technology, volumen 24, número 4, páginas 188-195.
- Kulkarni, S.R., y Horn, C.S., (1996), An Alternative Proof for Convergence of Stochastic Approximation Algorithms, IEEE Transactions on Automatic Control, volumen 41, número 3, páginas 419-424.
- Kushner, H.J., y Clark, D.J., (1978), Stochastic Approximation for Constrained and Unconstrained Systems, Springer-Verrlag, New York, N.Y.
- Kwang, K.J., y Lin, D.K.J., (1998), Dual Response Surface Optimization: A Fuzzy Modeling Approach, Journal of Quality Technology, volumen 30, número 1, páginas 1-10.
- Lagunes R. Y. y Sanchez L. J., (2011), Comparacion de Superficies de Respuesta con Cpk, Senal Ruido y Respuesta Dual para el Diseño Robusto. 4º Congreso Internacional de Investigación CIPITECH, paginas 695-703.
- Law, A.M., Kelton, W.D., (2000), Simulation Modeling and Analysis, third ed. McGraw-Hill, Boston. páginas 646-655.
- L'Ecuyer, P., N. Girox, y P. W. Glynn, (1994), Stochastic Optimization by Simulation: Numerical Experiments with M/M/1 Queue in Steady-State, Management Science, volumen 40, páginas 1245-1261.
- Lin, D.K.J., y Tu, W., (1995), Dual Response Optimization, Journal of Quality Technology, volumen 27, número 1, páginas 34-39.
- Lowe, C. W. (1974), Evolutionary Operation in Action. Applied Statistics, volumen 23, número 2, páginas 218-26.
- Maeda, Y., (1996), Time difference Simultaneous Perturbation Method, Electronic Letters, volumen 32, número 11, páginas 1016-1017.

- Minitab, (2012), CSS:STATISTICA 16, serie SW8088444813M51, Estados Unidos.
- Miranda A. K. y Del Castillo E., (2011), Robust Parameter Desing Optimization of Simulation Experiments Using Stochastic Perturbation Methods, Journal Operational Research Society, Vol. 62, Num. 1, páginas 198-205.
- Montgomery, D.C., (1991), Diseño y Análisis de Experimentos, Grupo Editorial Iberoamérica, segunda edición, México.
- Montgomery, D.C., (2004), Diseño y Análisis de Experimentos, Limusa Wiley Editorial primera edición, México.
- Myers, R.H. y Carter, W.H., (1973), Response Surface Techniques for Dual Response Systems, Technometrics, volumen 15, páginas 301-317.
- Myers, R.H. y Montgomery, D.C., (1995), Response Surface Methodology. Wiley series In Probability and Mathematical Statistics, New York: John Wiley & Sons.
- Myers, R.H., (1999), Response Surface Methodology-current status and future directions (including discussion). Journal of Quality Technology, Vol. 31, Num. 1, páginas 30-74.
- Myers, R.H. and Montgomery, D.C. (2009), Response Surface Methodology: Process and Product optimization Using Designed Experiments, Second ed., Wiley series, New York: John Wiley & Sons.
- Neddermeijer, H.G., van Ootmarsum, G.J., PieMSRa, N., Dekker, R., (2000), A framework for response surface methodology for simulation optimization models. In: Joines, J.A., Barton, R.R., Kang, K., Fishwick, P.A. (Eds.), Proceedings of the 2000 Winter Simulation Conference, páginas 129–136.
- Pearn, W.L., Kotz, S., Johnson, N. L., (1992), Distributional and Inferential Properties of Process Capability Indices, Journal of Quality Control, volumen 24, número 4, páginas 216-231
- Polyak, B.T., (1991), New Method of Stochastic Approximation Type Procedures, Automatica I telemekhanika, 51 (1990) páginas 98-107 en Ruso, trasladado al Inglés en Automatica Remote Control, volumen 51 páginas 937-945.

- Polyak, B.T., y Juditsky, A.B., (1992), Acceleration of Stochastic Approximation by Avering, SIAM Journal on Control and Optimization, volumen 30, número 4, páginas 838-855.
- Robbins, H., y S. Monro, (1951), A Stochastic Approximation Method, Annals of Mathematical Statistica, volumen 22, páginas 400-407.
- Rodriguez, R.N., (1992), Recent Developments in Process Capability Analysis, Journal of Quality Technology, volumen 24, número 4, páginas 176-187.
- Sacks, J., (1958), Asymptotic Distribution os Stochastic Approximation, Annals of Mathematical Statistica, número 29, páginas 373-405.
- Sadegh, P., (1997), Constrained Optimization via Stochastic Approximation with a Simultaneous Perturbation gradiente Approximation, Automatica, volumen 33, páginas 889-892.
- Sadegh, P., y Spall, J. C., (1998), Optimal Random for Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation, IEEE Transactions on Automatic Control, volumen 43, número 10, páginas 1480-1486.
- Safizadeh, M.H., (2002), Minimizing the bias and variance of the gradient estimate in MSR simulation studies. European Journal of Operational Research Vol. 136, Num. 1, páginas 121–135.
- Spall, J.C., (1988), A Stochastic Approximation Algorithm for Large- Dimensional Systems in the Kiefer-Wolfowitz Setting, IEEE Proceeding of the 27th Conference on Decision and Control, páginas 1544-1548.
- Spall, J.C., (1992), Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation, IEEE trans. Automatica, volumen 37, número 3, páginas 332-341.
- Spall, J.C., (1998), Implementation of the Simultaneous Perturbation Algorithm for Stochastic Optimization, IEEE Transactions on Aerospace and Electronics Systems, Vol. 34, páginas 817-823.
- Spall, J.C., (2003), Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, Wiley, Hoboken, NJ.
- Sánchez, L.J., (1991), A Method of Experimental Optimization, Ph. D. dissertation, University of New Mexico at Las Cruces.

- Vardeman, S.B., y Jobe, J.M., (1999), Statistical Quality Assurance Methods For Engineers, John Wiley & Sons Inc., Estados Unidos de Norteamérica.
- Vining, G.G., y Myers, R.H., (1990), Combining Taguchi and Response Surface Philosophies: A Dual Response Approach, Journal of Quality Technology, volumen 22, páginas38-45.

APENDICE 1

Publicaciones Derivadas del Trabajo de Tesis Doctoral

Publicaciones En Revistas Indizadas.

Jesús Everardo Olgún Tizado, Rafael García Martínez, Claudia Camargo Wilson, Juan Andres López Barreras, 2011. Stochastic Approximation Algorithm for Industrial Process Optimization, Revista Ingeniería e Investigación, Vol. 31, No. 3, paginas 100-111. ISSN 0120-5609.

Ponencias en Congresos Nacionales e Internacionales.

Juana Iris Cruz Alcantar, **Jesús Everardo Olgún Tizado**, Rafael García Martínez, Claudia Camargo Wilson, 2011. Análisis Comparativo del Método de Máxima Pendiente Ascendente y el Algoritmo de Aproximaciones Estocásticas Evaluando el Índice de Capacidad del Proceso, XXI Escuela Nacional de Optimización y Análisis Numérico (ENOAN), Cuernava, Morelos, México.

Jesús Everardo Olgún Tizado, Rafael García Martínez, Claudia Camargo Wilson, Juan Andres López Barreras, Everardo Inzunza González, Javier Ordorica Villalvazo, 2012. Effect of Size of the Step in the Response Surface Methodology Using Nonlinear Test Functions, World Academy of Science, Engineering and Technology, Vol. 66. ISSN 2010-376X.

Rafael García Martínez, **Jesús Everardo Olgún Tizado**, Claudia Camargo Wilson, 2012, Evaluación del Efecto del Tamaño del Paso en la Metodología de Superficie de Respuesta, 16va. Convención Científica de Ingeniería y Arquitectura, La Habana, Cuba.

APENDICE 2

En este apéndice se presentan los programas desarrollados en el módulo de Model Editor del software MATLAB[®] para las funciones de prueba FVPDR, FCDR, FB y FSFC (ver Tabla 4.2) con criterio de paro y sin criterio de paro, los cuales son usados para simular la implantación del Algoritmo de Aproximaciones Estocásticas con Perturbación Simultanea Modificado (AAEPSM) en procesos productivos en los que tanto el valor promedio como la desviación estándar de la variable de respuesta están en función de factores controlables y factores de ruido de dicho proceso. En este proceso de simulación son usadas cuatro funciones de prueba, donde se muestran los programas correspondientes para estimar la función de los valores iniciales de los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPSM.

A.1.1. Programa Usado para la Función de Prueba FVPDR.

Programa para la función de prueba FVPDR con criterio de paro, en la cual usa a la Función Valle Parabólico de Rosenbrock respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPSM. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```
clc
clear all
close all
% Criterio de Paro para el Cpk;
cpk_deseado=1.33;
bandera=0;
iteraciones=0;
% Intervalo para el valor del delta sub "k" positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
%Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
```

```

ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
% Cálculo del diseño externo dentro del AAEPS;
fb1=(100*(x2-x1^2)^2 + (1-x1)^2) + (randn(1,1)*5);
fb2=(100*(x4-x3^2)^2 + (1-x3)^2) + (randn(1,1)*5);
fb3=(100*(x6-x5^2)^2 + (1-x5)^2) + (randn(1,1)*5);
fb4=(100*(x8-x7^2)^2 + (1-x7)^2) + (randn(1,1)*5);
fb5=(100*(x10-x9^2)^2 + (1-x9)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb1;
y2=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb2;
y3=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb3;
y4=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb4;
y5=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos-v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=(100*(x2pos-x1pos^2)^2 + (1-x1pos)^2) + (randn(1,1)*5);
fb2pos=(100*(x4pos-x3pos^2)^2 + (1-x3pos)^2) + (randn(1,1)*5);
fb3pos=(100*(x6pos-x5pos^2)^2 + (1-x5pos)^2) + (randn(1,1)*5);
fb4pos=(100*(x8pos-x7pos^2)^2 + (1-x7pos)^2) + (randn(1,1)*5);
fb5pos=(100*(x10pos-x9pos^2)^2 + (1-x9pos)^2) + (randn(1,1)*5);

```

```

% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb1pos;
y2pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb2pos;
y3pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb3pos;
y4pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb4pos;
y5pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg+v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=(100*(x2neg-x1neg^2)^2 + (1-x1neg)^2) + (randn(1,1)*5);
fb2neg=(100*(x4neg-x3neg^2)^2 + (1-x3neg)^2) + (randn(1,1)*5);
fb3neg=(100*(x6neg-x5neg^2)^2 + (1-x5neg)^2) + (randn(1,1)*5);
fb4neg=(100*(x8neg-x7neg^2)^2 + (1-x7neg)^2) + (randn(1,1)*5);
fb5neg=(100*(x10neg-x9neg^2)^2 + (1-x9neg)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb1neg;
y2neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb2neg;
y3neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb3neg;
y4neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb4neg;
y5neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);

```

```

if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
% Ciclo para el Criterio de Paro del Cpk>=2;
if cpk_max(i,j)>=cpk_deseado;
    bandera=1;
    break;
end % Fin del If;
end % Fin del For j;
if bandera
    break;
end % Fin del If;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion Valle parabolica de Rosenbrock')
[cpk_max_1,renglones]=max(cpk_max);

```

```

[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
    fprintf(fid,'%4.4f\n',cpk_ok);

```

Programa para la función de prueba FVPDR sin criterio de paro, en la cual usa a la Función Valle Parabólico de Rosenbrock respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPSM. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```

clc
clear all
close all
iteraciones=0;
% Intervalo para el valor del delta sub "k" positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
%Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);

```

```

x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
% Cálculo del diseño externo dentro del AAEPS;
fb1=(100*(x2-x1^2)^2 + (1-x1)^2) + (randn(1,1)*5);
fb2=(100*(x4-x3^2)^2 + (1-x3)^2) + (randn(1,1)*5);
fb3=(100*(x6-x5^2)^2 + (1-x5)^2) + (randn(1,1)*5);
fb4=(100*(x8-x7^2)^2 + (1-x7)^2) + (randn(1,1)*5);
fb5=(100*(x10-x9^2)^2 + (1-x9)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb1;
y2=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb2;
y3=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb3;
y4=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb4;
y5=(100*(y0(k)-x0(k)^2)^2 + (1-x0(k))^2)+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos-v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=(100*(x2pos-x1pos^2)^2 + (1-x1pos)^2) + (randn(1,1)*5);
fb2pos=(100*(x4pos-x3pos^2)^2 + (1-x3pos)^2) + (randn(1,1)*5);
fb3pos=(100*(x6pos-x5pos^2)^2 + (1-x5pos)^2) + (randn(1,1)*5);
fb4pos=(100*(x8pos-x7pos^2)^2 + (1-x7pos)^2) + (randn(1,1)*5);
fb5pos=(100*(x10pos-x9pos^2)^2 + (1-x9pos)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb1pos;
y2pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb2pos;
y3pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb3pos;
y4pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb4pos;

```

```

y5pos=(100*(y0pos-x0pos^2)^2 + (1-x0pos)^2)+fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg+v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=(100*(x2neg-x1neg^2)^2 + (1-x1neg)^2) + (randn(1,1)*5);
fb2neg=(100*(x4neg-x3neg^2)^2 + (1-x3neg)^2) + (randn(1,1)*5);
fb3neg=(100*(x6neg-x5neg^2)^2 + (1-x5neg)^2) + (randn(1,1)*5);
fb4neg=(100*(x8neg-x7neg^2)^2 + (1-x7neg)^2) + (randn(1,1)*5);
fb5neg=(100*(x10neg-x9neg^2)^2 + (1-x9neg)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb1neg;
y2neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb2neg;
y3neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb3neg;
y4neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb4neg;
y5neg=(100*(y0neg-x0neg^2)^2 + (1-x0neg)^2)+fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1) ;
if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);

```

```

y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
end % Fin del For j;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion Valle parabolica de Rosenbrock')
[cpk_max_1,renglones]=max(cpk_max);
[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
fprintf(fid,'%4.4f\n',cpk_ok);

```

A.1.2. Programa Usado para la Función de Prueba FCDR.

Programa para la función de prueba FCDR con criterio de paro, en la cual usa a la Función Cúbica de Rosenbrock respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPSM. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```
clc
clear all
close all
% Criterio de Paro para el Cpk;
cpk_deseado=1.33;
bandera=0;
iteraciones=0;
% Intervalo para el valor del delta sub "k" positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
% Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
```

```

% Cálculo del diseño externo dentro del AAEPS;
fb1=(100*(x2-x1^3)^2 + (1-x1)^2) + (randn(1,1)*5);
fb2=(100*(x4-x3^3)^2 + (1-x3)^2) + (randn(1,1)*5);
fb3=(100*(x6-x5^3)^2 + (1-x5)^2) + (randn(1,1)*5);
fb4=(100*(x8-x7^3)^2 + (1-x7)^2) + (randn(1,1)*5);
fb5=(100*(x10-x9^3)^2 + (1-x9)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb1;
y2=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb2;
y3=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb3;
y4=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb4;
y5=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos-v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=(100*(x2pos-x1pos^3)^2 + (1-x1pos)^2) + (randn(1,1)*5);
fb2pos=(100*(x4pos-x3pos^3)^2 + (1-x3pos)^2) + (randn(1,1)*5);
fb3pos=(100*(x6pos-x5pos^3)^2 + (1-x5pos)^2) + (randn(1,1)*5);
fb4pos=(100*(x8pos-x7pos^3)^2 + (1-x7pos)^2) + (randn(1,1)*5);
fb5pos=(100*(x10pos-x9pos^3)^2 + (1-x9pos)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb1pos;
y2pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb2pos;
y3pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb3pos;
y4pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb4pos;
y5pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));

```

```

cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg+v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=(100*(x2neg-x1neg^3)^2 + (1-x1neg)^2) + (randn(1,1)*5);
fb2neg=(100*(x4neg-x3neg^3)^2 + (1-x3neg)^2) + (randn(1,1)*5);
fb3neg=(100*(x6neg-x5neg^3)^2 + (1-x5neg)^2) + (randn(1,1)*5);
fb4neg=(100*(x8neg-x7neg^3)^2 + (1-x7neg)^2) + (randn(1,1)*5);
fb5neg=(100*(x10neg-x9neg^3)^2 + (1-x9neg)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb1neg;
y2neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb2neg;
y3neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb3neg;
y4neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb4neg;
y5neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);
if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;

```

```

yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
% Ciclo para el Criterio de Paro del Cpk>=2;
if cpk_max(i,j)>=cpk_deseado;
    bandera=1;
    break;
end % Fin del If;
end % Fin del For j;
if bandera
    break;
end % Fin del If;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion Cubica de Rosenbrock')
[cpk_max_1,renglones]=max(cpk_max);
[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
fprintf(fid,'%4.4f\n',cpk_ok);

```

Programa para la función de prueba FCDR sin criterio de paro, en la cual usa a la Función Cúbica de Rosenbrock respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPSM. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```

clc
clear all
close all
iteraciones=0;
% Intervalo para el valor del delta sub "k" positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
% Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
% Cálculo del diseño externo dentro del AAEPS;
fb1=(100*(x2-x1^3)^2 + (1-x1)^2) + (randn(1,1)*5);
fb2=(100*(x4-x3^3)^2 + (1-x3)^2) + (randn(1,1)*5);
fb3=(100*(x6-x5^3)^2 + (1-x5)^2) + (randn(1,1)*5);
fb4=(100*(x8-x7^3)^2 + (1-x7)^2) + (randn(1,1)*5);
fb5=(100*(x10-x9^3)^2 + (1-x9)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;

```

```

y1=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb1;
y2=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb2;
y3=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb3;
y4=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb4;
y5=(100*(y0(k)-x0(k)^3)^2 + (1-x0(k))^2)+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos+v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=(100*(x2pos-x1pos^3)^2 + (1-x1pos)^2) + (randn(1,1)*5);
fb2pos=(100*(x4pos-x3pos^3)^2 + (1-x3pos)^2) + (randn(1,1)*5);
fb3pos=(100*(x6pos-x5pos^3)^2 + (1-x5pos)^2) + (randn(1,1)*5);
fb4pos=(100*(x8pos-x7pos^3)^2 + (1-x7pos)^2) + (randn(1,1)*5);
fb5pos=(100*(x10pos-x9pos^3)^2 + (1-x9pos)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb1pos;
y2pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb2pos;
y3pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb3pos;
y4pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb4pos;
y5pos=(100*(y0pos-x0pos^3)^2 + (1-x0pos)^2)+fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));

```

```

% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg-v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=(100*(x2neg-x1neg^3)^2 + (1-x1neg)^2) + (randn(1,1)*5);
fb2neg=(100*(x4neg-x3neg^3)^2 + (1-x3neg)^2) + (randn(1,1)*5);
fb3neg=(100*(x6neg-x5neg^3)^2 + (1-x5neg)^2) + (randn(1,1)*5);
fb4neg=(100*(x8neg-x7neg^3)^2 + (1-x7neg)^2) + (randn(1,1)*5);
fb5neg=(100*(x10neg-x9neg^3)^2 + (1-x9neg)^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb1neg;
y2neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb2neg;
y3neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb3neg;
y4neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb4neg;
y5neg=(100*(y0neg-x0neg^3)^2 + (1-x0neg)^2)+fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);
if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);

```

```

else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
end % Fin del For j;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion Cubica de Rosenbrock')
[cpk_max_1,renglones]=max(cpk_max);
[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
fprintf(fid,'%4.4f\n',cpk_ok);

```

A.1.3. Programa Usado para la Función de Prueba FB.

Programa para la función de prueba FB con criterio de paro, en la cual usa a la Función de Beale respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPMS. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```
clc
clear all
close all
% Criterio de Paro para el Cpk;
cpk_deseado=1.33;
bandera=0;
iteraciones=0;
% Intervalo para el valor del delta sub "k" positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
% Condiciones de No Negatividad del AAEPMS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPMS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
```

```

% Cálculo del diseño externo dentro del AAEPS;
fb1=(1.5-(x1*(1-x2)))^2 + (2.25-(x1*(1-x2^2)))^2 + (2.625-(x1*(1-x2^3)))^2 +
(randn(1,1)*5);
fb2=(1.5-(x3*(1-x4)))^2 + (2.25-(x3*(1-x4^2)))^2 + (2.625-(x3*(1-x4^3)))^2 +
(randn(1,1)*5);
fb3=(1.5-(x5*(1-x6)))^2 + (2.25-(x5*(1-x6^2)))^2 + (2.625-(x5*(1-x6^3)))^2 +
(randn(1,1)*5);
fb4=(1.5-(x7*(1-x8)))^2 + (2.25-(x7*(1-x8^2)))^2 + (2.625-(x7*(1-x8^3)))^2 +
(randn(1,1)*5);
fb5=(1.5-(x9*(1-x10)))^2 + (2.25-(x9*(1-x10^2)))^2 + (2.625-(x9*(1-x10^3)))^2 +
(randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb1;
y2=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb2;
y3=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb3;
y4=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb4;
y5=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos+v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=(1.5-(x1pos*(1-x2pos)))^2 + (2.25-(x1pos*(1-x2pos^2)))^2 + (2.625-
(x1pos*(1-x2pos^3)))^2 + (randn(1,1)*5);
fb2pos=(1.5-(x3pos*(1-x4pos)))^2 + (2.25-(x3pos*(1-x4pos^2)))^2 + (2.625-
(x3pos*(1-x4pos^3)))^2 + (randn(1,1)*5);
fb3pos=(1.5-(x5pos*(1-x6pos)))^2 + (2.25-(x5pos*(1-x6pos^2)))^2 + (2.625-
(x5pos*(1-x6pos^3)))^2 + (randn(1,1)*5);

```

```

fb4pos=(1.5-(x7pos*(1-x8pos)))^2 + (2.25-(x7pos*(1-x8pos^2)))^2 + (2.625-
(x7pos*(1-x8pos^3)))^2 + (randn(1,1)*5);
fb5pos=(1.5-(x9pos*(1-x10pos)))^2 + (2.25-(x9pos*(1-x10pos^2)))^2 + (2.625-
(x9pos*(1-x10pos^3)))^2 + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb1pos;
y2pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb2pos;
y3pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb3pos;
y4pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb4pos;
y5pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg-v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=(1.5-(x1neg*(1-x2neg)))^2 + (2.25-(x1neg*(1-x2neg^2)))^2 + (2.625-
(x1neg*(1-x2neg^3)))^2 + (randn(1,1)*5);
fb2neg=(1.5-(x3neg*(1-x4neg)))^2 + (2.25-(x3neg*(1-x4neg^2)))^2 + (2.625-
(x3neg*(1-x4neg^3)))^2 + (randn(1,1)*5);
fb3neg=(1.5-(x5neg*(1-x6neg)))^2 + (2.25-(x5neg*(1-x6neg^2)))^2 + (2.625-
(x5neg*(1-x6neg^3)))^2 + (randn(1,1)*5);
fb4neg=(1.5-(x7neg*(1-x8neg)))^2 + (2.25-(x7neg*(1-x8neg^2)))^2 + (2.625-
(x7neg*(1-x8neg^3)))^2 + (randn(1,1)*5);
fb5neg=(1.5-(x9neg*(1-x10neg)))^2 + (2.25-(x9neg*(1-x10neg^2)))^2 + (2.625-
(x9neg*(1-x10neg^3)))^2 + (randn(1,1)*5);

```

```

% Cálculo del diseño interno más el diseño externo dentro del AAEPs para el Cpk(-);
y1neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb1neg;
y2neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb2neg;
y3neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb3neg;
y4neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb4neg;
y5neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPs;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);
if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);

```

```

xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
% Ciclo para el Criterio de Paro del Cpk>=2;
if cpk_max(i,j)>=cpk_deseado;
    bandera=1;
    break;
end % Fin del If;
end % Fin del For j;
if bandera
    break;
end % Fin del If;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion de Beale')
[cpk_max_1,renglones]=max(cpk_max);
[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
fprintf(fid,'%4.4f\n',cpk_ok);

```

Programa para la función de prueba FB sin criterio de paro, en la cual usa a la Función de Beale respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPMS. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```

clc
clear all
close all
iteraciones=0;
% Intervalo para el valor del delta sub "k" positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
% Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
% Cálculo del diseño externo dentro del AAEPS;
fb1=(1.5-(x1*(1-x2)))^2 + (2.25-(x1*(1-x2^2)))^2 + (2.625-(x1*(1-x2^3)))^2 +
(randn(1,1)*5);
fb2=(1.5-(x3*(1-x4)))^2 + (2.25-(x3*(1-x4^2)))^2 + (2.625-(x3*(1-x4^3)))^2 +
(randn(1,1)*5);
fb3=(1.5-(x5*(1-x6)))^2 + (2.25-(x5*(1-x6^2)))^2 + (2.625-(x5*(1-x6^3)))^2 +
(randn(1,1)*5);

```

```

fb4=(1.5-(x7*(1-x8)))^2 + (2.25-(x7*(1-x8^2)))^2 + (2.625-(x7*(1-x8^3)))^2 +
(randn(1,1)*5);
fb5=(1.5-(x9*(1-x10)))^2 + (2.25-(x9*(1-x10^2)))^2 + (2.625-(x9*(1-x10^3)))^2 +
(randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb1;
y2=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb2;
y3=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb3;
y4=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb4;
y5=((1.5-x0(k)*(1-y0(k)))^2 + ((2.25-x0(k)*(1-y0(k)^2))^2 + ((2.625-x0(k)*(1-
y0(k)^3))))+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos-v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=(1.5-(x1pos*(1-x2pos)))^2 + (2.25-(x1pos*(1-x2pos^2)))^2 + (2.625-
(x1pos*(1-x2pos^3)))^2 + (randn(1,1)*5);
fb2pos=(1.5-(x3pos*(1-x4pos)))^2 + (2.25-(x3pos*(1-x4pos^2)))^2 + (2.625-
(x3pos*(1-x4pos^3)))^2 + (randn(1,1)*5);
fb3pos=(1.5-(x5pos*(1-x6pos)))^2 + (2.25-(x5pos*(1-x6pos^2)))^2 + (2.625-
(x5pos*(1-x6pos^3)))^2 + (randn(1,1)*5);
fb4pos=(1.5-(x7pos*(1-x8pos)))^2 + (2.25-(x7pos*(1-x8pos^2)))^2 + (2.625-
(x7pos*(1-x8pos^3)))^2 + (randn(1,1)*5);
fb5pos=(1.5-(x9pos*(1-x10pos)))^2 + (2.25-(x9pos*(1-x10pos^2)))^2 + (2.625-
(x9pos*(1-x10pos^3)))^2 + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb1pos;

```

```

y2pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb2pos;
y3pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb3pos;
y4pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb4pos;
y5pos=((1.5-x0pos*(1-y0pos))^2 + ((2.25-x0pos*(1-y0pos^2))^2 + ((2.625-
x0pos*(1-y0pos^3)))) + fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg+v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=(1.5-(x1neg*(1-x2neg)))^2 + (2.25-(x1neg*(1-x2neg^2)))^2 + (2.625-
(x1neg*(1-x2neg^3)))^2 + (randn(1,1)*5);
fb2neg=(1.5-(x3neg*(1-x4neg)))^2 + (2.25-(x3neg*(1-x4neg^2)))^2 + (2.625-
(x3neg*(1-x4neg^3)))^2 + (randn(1,1)*5);
fb3neg=(1.5-(x5neg*(1-x6neg)))^2 + (2.25-(x5neg*(1-x6neg^2)))^2 + (2.625-
(x5neg*(1-x6neg^3)))^2 + (randn(1,1)*5);
fb4neg=(1.5-(x7neg*(1-x8neg)))^2 + (2.25-(x7neg*(1-x8neg^2)))^2 + (2.625-
(x7neg*(1-x8neg^3)))^2 + (randn(1,1)*5);
fb5neg=(1.5-(x9neg*(1-x10neg)))^2 + (2.25-(x9neg*(1-x10neg^2)))^2 + (2.625-
(x9neg*(1-x10neg^3)))^2 + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb1neg;
y2neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb2neg;
y3neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb3neg;
y4neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb4neg;
y5neg=((1.5-x0neg*(1-y0neg))^2 + ((2.25-x0neg*(1-y0neg^2))^2 + ((2.625-
x0neg*(1-y0neg^3)))) + fb5neg;

```

```

% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);
if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
end % Fin del For j;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')

```

```

title ('Funcion de Beale')
[cpk_max_1,renglones]=max(cpk_max);
[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
    fprintf(fid,'%4.4f\n',cpk_ok);

```

A.1.4. Programa Usado para la Función de Prueba FSFC.

Programa para la función de prueba FSFC con criterio de paro, en la cual usa a la Función de Suma de Funciones Cuadráticas respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPMS. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```

clc
clear all
close all
% Criterio de Paro para el Cpk;
cpk_deseado=1.33;
bandera=0;
iteraciones=0;
% Intervalo para el valor del delta sub “k” positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
% Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;

```

```

ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);
x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
% Cálculo del diseño externo dentro del AAEPS;
fb1=x1^2 + (2*x2^2) + (randn(1,1)*5);
fb2=x3^2 + (2*x4^2) + (randn(1,1)*5);
fb3=x5^2 + (2*x6^2) + (randn(1,1)*5);
fb4=x7^2 + (2*x8^2) + (randn(1,1)*5);
fb5=x9^2 + (2*x10^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=x0(k)^2 + (2*y0(k)^2)+fb1;
y2=x0(k)^2 + (2*y0(k)^2)+fb2;
y3=x0(k)^2 + (2*y0(k)^2)+fb3;
y4=x0(k)^2 + (2*y0(k)^2)+fb4;
y5=x0(k)^2 + (2*y0(k)^2)+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos-v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=x1pos^2 + (2*x2pos^2)+ (randn(1,1)*5);
fb2pos=x3pos^2 + (2*x4pos^2) + (randn(1,1)*5);
fb3pos=x5pos^2 + (2*x6pos^2) + (randn(1,1)*5);
fb4pos=x7pos^2 + (2*x8pos^2) + (randn(1,1)*5);

```

```

fb5pos=x9pos^2 + (2*x10pos^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=x0pos^2 + (2*y0pos^2) + fb1pos;
y2pos=x0pos^2 + (2*y0pos^2) + fb2pos;
y3pos=x0pos^2 + (2*y0pos^2) + fb3pos;
y4pos=x0pos^2 + (2*y0pos^2) + fb4pos;
y5pos=x0pos^2 + (2*y0pos^2) + fb5pos;
% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg+v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=x1neg^2 + (2*x2neg^2) + (randn(1,1)*5);
fb2neg=x3neg^2 + (2*x4neg^2) + (randn(1,1)*5);
fb3neg=x5neg^2 + (2*x6neg^2) + (randn(1,1)*5);
fb4neg=x7neg^2 + (2*x8neg^2) + (randn(1,1)*5);
fb5neg=x9neg^2 + (2*x10neg^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=x0neg^2 + (2*y0neg^2) + fb1neg;
y2neg=x0neg^2 + (2*y0neg^2) + fb2neg;
y3neg=x0neg^2 + (2*y0neg^2) + fb3neg;
y4neg=x0neg^2 + (2*y0neg^2) + fb4neg;
y5neg=x0neg^2 + (2*y0neg^2) + fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximacion del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);

```

```

if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
% Ciclo para el Criterio de Paro del Cpk>=2;
if cpk_max(i,j)>=cpk_deseado;
    bandera=1;
    break;
end % Fin del If;
end % Fin del For j;
if bandera
    break;
end % Fin del If;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion Suma de Funciones Cuadradas')
[cpk_max_1,renglones]=max(cpk_max);

```

```

[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
    fprintf(fid,'%4.4f\n',cpk_ok);

```

Programa para la función de prueba FSFC sin criterio de paro, en la cual usa a la Función de Suma de Funciones Cuadráticas respectivamente, como la función que relaciona la estimación de la función de los valores iniciales para los factores controlables más la función para estimar los valores de los niveles del diseño factorial 2^k empleado como factor de ruido dentro del AAEPSM. Esto para relacionar la función que estima el valor promedio y la desviación estándar de la variable de respuesta.

Inicio

```

clc
clear all
close all
iteraciones=0;
% Intervalo para el valor del delta sub “k” positivo y negativo;
increkpos=0.1:0.1:1; % Renglones;
increkneg=-0.1:-0.1:-1; % Columnas;
% Condiciones de No Negatividad del AAEPS;
a=0.16;
c=1;
A=100;
alfa=0.602;
gama=0.101;
n=5;
v1=0.2; % Volumen del diseño factorial 2k;
esp_sup=50; % Especificación Superior para el Cpk;
k=1; % Contador inicial de iteraciones;
x0(k)=2.7821; % Valor inicial de x1;
y0(k)=4.4122; % Valor inicial de x2;
% Ciclo para iniciar el proceso iterativo del AAEPS;
for i=1:length(increkpos)
for j=1:length(increkneg)
for k=1:100 % k son las iteraciones, va de 1 al 100;
ak=1/k;
ck=c/(k+1)^gama;
% Diseño 2k con un punto central;
x1=(x0(k)-v1);
x2=(y0(k)-v1);
x3=(x0(k)-v1);

```

```

x4=(y0(k)+v1);
x5=(x0(k)+v1);
x6=(y0(k)-v1);
x7=(x0(k)+v1);
x8=(y0(k)+v1);
x9=x0(k);
x10=y0(k);
% Cálculo del diseño externo dentro del AAEPS;
fb1=x1^2 + (2*x2^2) + (randn(1,1)*5);
fb2=x3^2 + (2*x4^2) + (randn(1,1)*5);
fb3=x5^2 + (2*x6^2) + (randn(1,1)*5);
fb4=x7^2 + (2*x8^2) + (randn(1,1)*5);
fb5=x9^2 + (2*x10^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS;
y1=x0(k)^2 + (2*y0(k)^2)+fb1;
y2=x0(k)^2 + (2*y0(k)^2)+fb2;
y3=x0(k)^2 + (2*y0(k)^2)+fb3;
y4=x0(k)^2 + (2*y0(k)^2)+fb4;
y5=x0(k)^2 + (2*y0(k)^2)+fb5;
% Análisis de la media y desviación estándar para el Cpk(inicial);
fbtotal=sum(y1+y2+y3+y4+y5);
yprom1=fbtotal/n;
ysigma1=sqrt(sum((yprom1-y1)^2+(yprom1-y2)^2+(yprom1-y3)^2+(yprom1-
y4)^2+(yprom1-y5)^2)/(n-1));
cpk=(esp_sup-yprom1)/(3*ysigma1);
% Iniciar el Análisis del Cpk(+);
x0pos=x0(k)+(ck*increkpos(i));
y0pos=y0(k)+(ck*increkneg(j));
% Diseño 2k con un punto central para xk positivo;
x1pos=(x0pos-v1);
x2pos=(y0pos-v1);
x3pos=(x0pos-v1);
x4pos=(y0pos+v1);
x5pos=(x0pos+v1);
x6pos=(y0pos-v1);
x7pos=(x0pos+v1);
x8pos=(y0pos+v1);
x9pos=(x0pos);
x10pos=(y0pos);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(+);
fb1pos=x1pos^2 + (2*x2pos^2)+ (randn(1,1)*5);
fb2pos=x3pos^2 + (2*x4pos^2) + (randn(1,1)*5);
fb3pos=x5pos^2 + (2*x6pos^2) + (randn(1,1)*5);
fb4pos=x7pos^2 + (2*x8pos^2) + (randn(1,1)*5);
fb5pos=x9pos^2 + (2*x10pos^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el
Cpk(+);
y1pos=x0pos^2 + (2*y0pos^2) + fb1pos;
y2pos=x0pos^2 + (2*y0pos^2) + fb2pos;
y3pos=x0pos^2 + (2*y0pos^2) + fb3pos;

```

```

y4pos=x0pos^2 + (2*y0pos^2) + fb4pos;
y5pos=x0pos^2 + (2*y0pos^2) + fb5pos;

% Análisis de la media y desviación estándar para el Cpk(+);
fbtotalpos=sum(y1pos+y2pos+y3pos+y4pos+y5pos);
yprom1pos=fbtotalpos/n;
ysigma1pos=sqrt(sum((yprom1pos-y1pos)^2+(yprom1pos-y2pos)^2+(yprom1pos-
y3pos)^2+(yprom1pos-y4pos)^2+(yprom1pos-y5pos)^2)/(n-1));
cpkpos=(esp_sup-yprom1pos)/(3*ysigma1pos);
% Iniciar el Análisis del Cpk(-);
x0neg=x0(k)-(ck*increkpos(i));
y0neg=y0(k)-(ck*increkneg(j));
% Diseño 2k con un punto central para xk negativo;
x1neg=(x0neg-v1);
x2neg=(y0neg-v1);
x3neg=(x0neg-v1);
x4neg=(y0neg+v1);
x5neg=(x0neg+v1);
x6neg=(y0neg-v1);
x7neg=(x0neg+v1);
x8neg=(y0neg+v1);
x9neg=(x0neg);
x10neg=(y0neg);
% Cálculo del diseño externo dentro del AAEPS para el Cpk(-);
fb1neg=x1neg^2 + (2*x2neg^2) + (randn(1,1)*5);
fb2neg=x3neg^2 + (2*x4neg^2) + (randn(1,1)*5);
fb3neg=x5neg^2 + (2*x6neg^2) + (randn(1,1)*5);
fb4neg=x7neg^2 + (2*x8neg^2) + (randn(1,1)*5);
fb5neg=x9neg^2 + (2*x10neg^2) + (randn(1,1)*5);
% Cálculo del diseño interno más el diseño externo dentro del AAEPS para el Cpk(-);
y1neg=x0neg^2 + (2*y0neg^2) + fb1neg;
y2neg=x0neg^2 + (2*y0neg^2) + fb2neg;
y3neg=x0neg^2 + (2*y0neg^2) + fb3neg;
y4neg=x0neg^2 + (2*y0neg^2) + fb4neg;
y5neg=x0neg^2 + (2*y0neg^2) + fb5neg;
% Análisis de la media y desviación estándar para el Cpk(-);
fbtotalneg=sum(y1neg+y2neg+y3neg+y4neg+y5neg);
yprom1neg=fbtotalneg/n;
ysigma1neg=sqrt(sum((yprom1neg-y1neg)^2+(yprom1neg-y2neg)^2+(yprom1neg-
y3neg)^2+(yprom1neg-y4neg)^2+(yprom1neg-y5neg)^2)/(n-1));
cpkneg=(esp_sup-yprom1neg)/(3*ysigma1neg);
% Aproximación del Vector Gradiente del AAEPS;
fikpos=(cpkpos-cpkneg)/((2)*(ck)*(increkpos(i)));
fikneg=(cpkpos-cpkneg)/((2)*(ck)*(increkneg(j)));
% Ciclo para (k+1);
if (cpk>cpkpos)&&(cpk>cpkneg);
xopt=x0(k);
yopt=y0(k);
cpk_ok(k)=cpk;

```

```

xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
elseif (cpkpos>cpkneg)&&(cpkpos>cpk);
xopt=x0pos;
yopt=y0pos;
cpk_ok(k)=cpkpos;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
else (cpkneg>cpk)&&(cpkneg>cpkpos);
xopt=x0neg;
yopt=y0neg;
cpk_ok(k)=cpkneg;
xopt_ok(k)=xopt;
yopt_ok(k)=yopt;
x0(k+1)=xopt_ok(k)-(ak*fikpos);
y0(k+1)=yopt_ok(k)-(ak*fikneg);
end
end % fin del For k;
% Obtener el valor máximo del Cpk;
[cpk_max(i,j),pos]=max(cpk_ok);
xopt_ok2(i,j)=xopt_ok(pos);
yopt_ok2(i,j)=yopt_ok(pos);
iteraciones=iteraciones+1;
% Obtener la iteración que obtuvo el máximo Cpk;
iteracion_opt(iteraciones)=pos;
end % Fin del For j;
end % Fin del For i;
% Obtener el gráfico Cpk máximo;
figure
plot(cpk_ok)
xlabel('Numero de iteraciones')
ylabel('Cpk')
title ('Funcion Suma de Funciones Cuadradas')
[cpk_max_1,renglones]=max(cpk_max);
[cpk_max_ok,col]=max(cpk_max_1)
renglon=renglones(col);
xopt_ok2(renglon,col) % Valor óptimo para las condiciones de x1;
yopt_ok2(renglon,col) % Valor óptimo para las condiciones de x2;
increkpos_opt=increkpos(renglon) % Valor óptimo para delta k positivo;
increkneg_opt=increkneg(col) % Valor óptimo para delta k negativo;
iteraciones;
iteracion_opt(iteraciones)
% Guardar los datos como archivos txt del Cpk;
fid = fopen('cpk_ok.txt','w');
fprintf(fid,'%4.4f\n',cpk_ok);

```