



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

DESARROLLO E IMPLEMENTACIÓN DE
ALGORITMOS DE MODELOS DE ALTA
DIMENSIONALIDAD (“HIGH DIMENSIONAL
MODEL REPRESENTATIONS”) UTILIZANDO
UN PROCESO DE DISEÑO DE SOFTWARE

T E S I S

QUE PARA OBTENER EL GRADO DE:

DOCTORA EN CIENCIAS
(COMPUTACIÓN)

P R E S E N T A:

MARIA DEL CARMEN GÓMEZ FUENTES

DIRECTOR DE TESIS: DR. VLADIMIR TCHIJOV

MÉXICO, D.F.

Marzo 2007.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradezco a:

Mi tutor el Dr. Vladimir Tchijov.

Los miembros del jurado:

Dr. Ismael Herrera Revilla.

Dr. Arón Jazcilevich Diamant.

Dr. Fabián García Noceti.

Dr. Vladislav Kharchenko.

El Posgrado en Ciencia e Ingeniería de la Computación.

CONACYT por la beca No. 165367.

COMECYT por su apoyo de beca-tesis.

Mil gracias!

Dedico esta tesis:

A mi esposo Jorge.

A mis hijos Elisa y José Arturo.

A Dios.

CONTENIDO.

1. Introducción.....	11
1.1. Planteamiento del problema propuesto.....	13
1.2. Objetivos.....	14
1.3. Metas.....	14
1.4. Organización de la tesis.....	15
2. Representaciones de Modelos de Alta Dimensionalidad: High Dimensional Model Representations (HDMR).	
2.1. Determinación de las funciones que componen las HDMR.....	17
2.1.1. Cut-HDMR.....	18
2.1.2. RS-HDMR.....	22
2.2. HDMR y sus aplicaciones.....	26
2.2.1. Construcción de modelos basados en la observación directa desde el laboratorio o de los datos del campo.....	26
2.2.2. Valoración de la incertidumbre global e identificación de las variables clave y su interrelación.....	27
2.2.3. Construcción de un modelo operacional completamente equivalente.....	28
2.2.4. Valoración del riesgo cuantitativamente eficiente.....	28
2.3. Propiedades de las expansiones HDMR.....	29
2.3.1. Convergencia rápida de las expansiones HDMR.....	29
2.3.2. Concordancia de las expansiones HDMR con las leyes de conservación.....	31
2.3.3. Descomposición de la varianza en las expansiones HDMR.....	31
2.4 Expansiones HDMR avanzadas.	
2.4.1. Multi-Cut-HDMR vs. Cut-HDMR con multiples subdominios.....	32
2.4.2. HDMR para un funcional.....	34
Conclusiones del capítulo 2.....	35

3. Modelo Operacional Completamente Equivalente (FEOM) y su desarrollo mediante un proceso de diseño de software.

3.1. Definición.....	37
3.2. Aplicaciones.....	38
3.3. Proceso de diseño del software para implantar un FEOM que resuelve un sistema de ecuaciones diferenciales usando las expansiones Cut-HDMR.....	40
3.3.1. Análisis de requerimientos.....	40
3.3.2. Diseño de alto nivel.....	40
Fase de entrenamiento.....	41
Fase de operación.....	41
3.3.3. Diseño detallado.....	41
Fase de entrenamiento.....	41
Fase de operación.....	50
3.3.4. Codificación.....	52
3.3.5. Pruebas.....	52
Conclusiones del capítulo 3.....	54

4. Aplicación parcial del FEOM en el modelo en 3 dimensiones de difusión/advección.

4.1. El modelo en 3 dimensiones de la difusión/advección en 3 dimensiones.....	55
4.2. Sistemas rígidos.....	56
4.3. Modelo propuesto por Seinfeld como mecanismo de reacciones químicas.	56
4.4. Substitución de CVODE por el FEOM como método de solución del sistema de ecuaciones diferenciales del modelo de reacciones químicas.....	58
4.5. Elección de los parámetros del FEOM.....	58
4.5.1. Elección del grado de cooperatividad.....	59
4.5.2. Elección del número de muestras.....	60
4.5.3. Rango de las variables.....	61
4.5.4. Condiciones iniciales.....	61
4.5.5. Punto de referencia.....	62
4.5.6. Paso de tiempo Δt	62
4.6. Elección del método de interpolación.....	64
4.7. Procedimiento detallado para integrar el sistema de EDO's por medio de un FEOM.....	64
4.8. Resultados obtenidos con el modelo de Seinfeld de 11 variables.....	68
4.9. Otros estudios de caso.	
4.9.1. Resultados obtenidos con el modelo básico de cuatro sustancias.....	69
4.9.2. Resultados obtenidos con el modelo de 27 sustancias.....	70
Conclusiones del capítulo 4.....	71

5. Aplicación del FEOM en el modelo de difusión/advección de contaminantes.	
5.1. Solución de sistema de ecuaciones diferenciales parciales no lineales planteado para las ecuaciones de difusión-advección de contaminantes.....	73
5.2. Malla donde se trabaja el modelo de difusión/advección de contaminantes.....	75
5.3. Intervalo de trabajo y elección del punto de referencia del FEOM.....	76
5.4. Elección de diferentes FEOMs en la malla.....	77
5.4.1. Elección de un solo FEOM para toda la malla.....	77
5.4.2. Elección de cuatro FEOMs para toda la malla.....	79
5.4.3. Elección de nueve FEOMs para toda la malla.....	82
5.5. Resultado final del modelo de difusión/advección al aplicar un FEOM para resolver el sistema de reacciones químicas de Seinfeld de 11 sustancias.....	84
Conclusiones del capítulo 5.....	88
Conclusiones finales.....	89
APÉNDICE A: Algoritmos de búsqueda de proyección.....	91
APÉNDICE B: Funciones de base radial.....	92
APÉNDICE C: Interpolación multidimensional.....	93
APÉNDICE D: El método de Monte Carlo.....	96
APÉNDICE E: El algoritmo de interpolación de Neville para interpolación de Lagrange.....	99
APÉNDICE F: El algoritmo de interpolación spline cúbico.....	101
APÉNDICE G: Valoración del grado global de incertidumbre.....	103
APÉNDICE H: Aseguramiento de la calidad del software (inspecciones).....	106
Referencias.....	108

Capítulo 1.

Introducción.

Las Representaciones de Modelos de Alta Dimensionalidad (HDMR: “High Dimensional Model Representations”) pertenecen a un área emergente en ciencias computacionales y matemáticas aplicadas.

Las expansiones HDMR [Allis & Rabitz, 1999; Li et al., 2001] son herramientas para mejorar el modelado matemático de sistemas físicos en donde lo importante es determinar las relaciones existentes entre las entradas y las salidas del sistema.

Existen sistemas en diferentes áreas de la ciencia que tienen una gran cantidad de variables de entrada, cada una de estas variables influye de diferente manera en el comportamiento de la salida.

Uno de los objetivos mas frecuentes de los científicos es identificar el mapeo de entrada-salida por medio de un muestreo finito de las variables de entrada. Otro de los objetivos es predecir los valores de las variables de salida dados un punto inicial y parámetros de entrada [Rabitz et al., 2000].

Hay métodos tradicionales para atacar este tipo de problemas, sin embargo, estos emplean una gran carga computacional, la cual crece exponencialmente conforme aumenta el número de variables del sistema, de tal manera que para sistemas con $n \gg 10$ el uso de estos métodos es prohibitivo.

HDMR es un modelo que aproxima sistemas multidimensionales y proporciona manipulaciones algebraicas rápidas y estables con una buena exactitud.

Existen varios antecedentes en la teoría de la aproximación de funciones de muchas variables. El teorema de Kolmogorov [Lorentz et al., 1996] es un antecedente de las HDMR y al igual que las HDMR se basa en el hecho de representar aproximaciones a la función por medio de la suma de funciones de menos variables. Este teorema dice que cualquier función continua de n variables definida en un hipercubo unitario k^n puede expresarse como la suma de $2n+1$ funciones, en donde cada una de estas funciones depende de una sola variable. Es importante mencionar que no existe ningún algoritmo numérico viable que pueda calcular las funciones de una sola variable de esta representación.

Sin embargo, hay una serie de representaciones conocidas como *redes de aprendizaje* las cuales aproximan una función de n variables con cualquier precisión deseada [Rabitz et al., 1998]. Algunos ejemplos son los *algoritmos de Búsqueda de Proyección* (ver Apéndice A), los *Perceptrones Multicapa* [Parker, 1985], y las *Funciones de Base Radial* (ver Apéndice B). Ninguna de estas representaciones es una aproximación genérica que se aplique para todas las funciones de muchas variables; es necesario un conocimiento previo de la estructura física del sistema para escoger alguna de las alternativas. Por el contrario, las representaciones HDMR se basan en el supuesto de que solo las funciones de pocas variables (1, 2 y 3) juegan un papel significativo en los sistemas físicos. Las representaciones HDMR pueden ser clasificadas como *redes de aprendizaje no-paramétricas y no-regresivas* [Alis & Rabitz, 1999].

Las técnicas HDMR están basadas en la teoría de operadores de proyección y la teoría de optimización [Li et al., 2000].

HDMR es una expansión de funciones correlacionadas jerárquicamente que capturan las relaciones de entrada-salida de un sistema multidimensional [Allis & Rabitz, 1999; Li et al., 2001] y está dada por la siguiente expresión:

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^n f_i(x_i) + \sum_{1 \leq i < j \leq n} f_{ij}(x_i, x_j) + \sum_{1 \leq i < j < k \leq n} f_{ijk}(x_i, x_j, x_k) + \dots + f_{12\dots n}(x_1, x_2, \dots, x_n) \quad (1)$$

donde f_0 representa el valor medio de $f(\mathbf{x})$ sobre todo el dominio Ω de \mathbf{x} .

El dominio Ω de \mathbf{x} se define de la forma en la que estén acotadas las variables de entrada

x_1, x_2, \dots, x_n .

La función de primer grado de cooperatividad ($l = 1$) $f_i(x_i)$ representa el efecto de las variables de entrada x_i actuando independientemente una de la otra. Sin embargo, en general las variables x_i que actúan independientemente sobre la salida $f(\mathbf{x})$ lo hacen de forma no lineal.

Las funciones de segundo grado de cooperatividad ($l = 2$) $f_{ij}(x_i, x_j)$ describen los efectos cooperativos de las variables de entrada x_i y x_j sobre la función de salida $f(\mathbf{x})$.

Los términos de grado de cooperatividad mas alto reflejan los efectos cooperativos de un número creciente de variables actuando todas juntas para influenciar a la función de salida $f(\mathbf{x})$. El último término representa a todas las variables de entrada actuando en conjunto para influir sobre la salida $f(\mathbf{x})$.

La conjetura fundamental de las HDMR es que en los problemas típicos de la práctica no parece existir un grado alto de cooperatividad entre las variables de entrada, de tal forma que se espera que los términos significativos de las expansiones HDMR (1) satisfagan la relación:

$$l \ll n \quad \text{para } n \gg 1$$

La experiencia ha mostrado que la expresión HDMR de 2º grado de cooperatividad:

$$f(\mathbf{x}) \approx f_0 + \sum_{i=1}^n f_i(x_i) + \sum_{i < j \leq n} f_{ij}(x_i, x_j)$$

con frecuencia proporciona una descripción satisfactoria de $f(\mathbf{x})$ para muchos sistemas multidimensionales [Alis & Rabitz, 1999].

Esta es una hipótesis comprobada con muchos fenómenos a los que se les ha aplicado esta técnica [Shorter et al., 1999; Shorter & Percila, 2000; Wang et al., 1999; Shoendorf et al., 2003] (entre otros), es decir, se han comparado los resultados de la teoría con los de experimentos.

1.1. Planteamiento del problema propuesto.

El modelo que utilizamos en esta tesis para el análisis de la concentración de ozono en forma idealizada, en ausencia de deposición de impurezas, es el modelo en 3 dimensiones de difusión/advección de contaminantes y se describe con el siguiente sistema de ecuaciones diferenciales parciales [Nagornov et al., 2003]:

$$\frac{\partial c_i}{\partial t} + \frac{\partial(uc_i)}{\partial x} + \frac{\partial(vc_i)}{\partial y} + \frac{\partial(wc_i)}{\partial z} = \frac{\partial}{\partial x} \left(K_x \frac{\partial c_i}{\partial x} \right) + \frac{\partial}{\partial y} \left(K_y \frac{\partial c_i}{\partial y} \right) + \frac{\partial}{\partial z} \left(K_z \frac{\partial c_i}{\partial z} \right) + R_i$$

(2)

($i = 1, 2, \dots, n$)

donde c_i representan las concentraciones de contaminantes y R_i son las tasas de cambio en las concentraciones de contaminantes en la mezcla reactiva debido a las transformaciones químicas. Las variables u, v, w son los componentes del vector velocidad de viento, y K_x, K_y, K_z son los coeficientes de difusión turbulenta [Aguilar, 1998]. Los problemas de frontera y valor inicial para el sistema (2) se resuelven numéricamente para un conjunto de n especies químicas, en un dominio Ω tridimensional ($0 < x < L_x, 0 < y < L_y, 0 < z < L_z$).

Los mecanismos de reacciones químicas R_i que están dentro de este modelo de contaminación atmosférica se describen por medio de sistemas rígidos de ecuaciones diferenciales ordinarias (EDO):

$$\frac{dc_i}{dt} = R_i(c_1, c_2, \dots, c_n), \quad i = 1, \dots, n$$

(3)

En los últimos 20 años se han construido diversos modelos de calidad del aire para la ciudad de México. La cantidad de sustancias que intervienen en estos modelos es variable (4, 11, 27, 70,...). Se ha observado que la parte que consume más tiempo cuando se ejecutan los modelos de calidad del aire es la correspondiente a la solución de los sistemas rígidos de ecuaciones diferenciales formados por las reacciones químicas, es decir, la solución de la parte de las reacciones químicas es *computacionalmente intensiva*. Por ejemplo, En la reciente publicación [Jazcilevich, 2005] se dice que en los modelos donde intervienen 70 o más sustancias, el tiempo empleado en la solución del sistema de ecuaciones diferenciales que corresponde a las reacciones químicas ocupa alrededor de un 70% de la carga computacional.

Hasta ahora, se utilizan métodos numéricos convencionales, llamados métodos de GEAR [Press *et al.*, 1999] en la solución de estos sistemas rígidos de EDOs (3).

En nuestro caso utilizamos el paquete CVODE [Cohen & Hindmarsh, 1996]. Este paquete ha sido diseñado específicamente para la solución de sistemas rígidos de EDOs por medio de los métodos de GEAR. El objetivo principal de esta investigación es sustituir el integrador numérico convencional CVODE por una aplicación de las expansiones HDMR llamada FEOM (Fully Equivalent Operational Model).

El uso de un FEOM [Shorter *et al.*, 2000; Wang *et al.*, 1999], en lugar de una parte del sistema (o del sistema completo) es una de las aplicaciones principales de las HDMR. A su vez, una de las aplicaciones importantes del FEOM es el modelado de sistemas descritos por medio de un conjunto de ecuaciones diferenciales. Los resultados de un FEOM se obtienen más rápidamente que los de un integrador numérico convencional, con una buena exactitud.

En esta tesis se pretende demostrar la aplicabilidad del FEOM a la solución de los sistemas de ecuaciones diferenciales planteados por los mecanismos de reacciones fotoquímicas dentro de la ecuación idealizada de difusión/advección en tres dimensiones, operando en tiempos cortos, es decir, trabajando con minutos y segundos.

1.2. Objetivos.

El objetivo principal de esta tesis es sustituir el integrador numérico convencional CVODE por una aplicación de las expansiones HDMR [Alis & Rabitz 1999; Li et al., 2001] llamada FEOM (Fully Equivalent Operational Model) [Shoendorf et al., 2003; Shorter et al., 1999] para lograr una importante reducción en el tiempo de ejecución del modelo de contaminación ambiental.

La aplicación de un modelo matemático para la predicción de los niveles de ozono es un problema complejo. La variación de la velocidad de las reacciones químicas con el ángulo del sol [Nagornov et al., 2003] y la gran cantidad de sustancias que intervienen en la contaminación atmosférica, hacen difícil la predicción de los niveles de ozono.

El objetivo de esta tesis no pretende ser un desarrollo de modelos matemáticos de contaminación ambiental, sino desarrollar y demostrar la utilización del FEOM en lugar de un integrador numérico convencional en dichos modelos para agilizar significativamente su ejecución.

1.3. Metas.

La meta de esta investigación es determinar por medio computacional, que tan rápido es el FEOM para resolver varios mecanismos de reacciones químicas, en comparación con CVODE.

El FEOM mencionado está constituido por un sistema de cómputo, por lo que otra de las metas es utilizar un proceso de desarrollo de software [Gómez, 2004] para hacer al sistema accesible y facilitar su aplicación a otros proyectos.

1.4. Organización de la tesis.

El segundo capítulo incluye las principales aplicaciones de las expansiones HDMR, también expone cómo se determinan las funciones que componen las expansiones HDMR, las propiedades de estas expansiones y las HDMR avanzadas.

En el tercer capítulo se define lo que es un FEOM y se detalla el proceso de diseño de software por medio del cual se elaboró el sistema de cómputo para entrenar y operar los FEOMs con los que se realizaron los experimentos de esta investigación.

En el tercer capítulo se expone la aplicación de un FEOM a diferentes mecanismos de reacciones químicas del modelo de contaminación ambiental, sin incluirlo. Finalmente, en el quinto capítulo se muestran los resultados de sustituir al integrador numérico CVODE por su correspondiente FEOM en el modelo en tres dimensiones de difusión/advección de contaminantes.

CAPÍTULO 2.

Representaciones de Modelos de Alta Dimensionalidad: High Dimensional Model Representations (HDMR).

2.1. Determinación de las funciones que componen las HDMR.

La valiosa propiedad del bajo grado de cooperatividad de las variables de entrada para un sistema multidimensional puede utilizarse solo si se pueden encontrar las funciones componente $f_0, f_i(x_i), f_{ij}(x_i, x_j), \dots$ apropiadas, estas funciones deben hacerse para una $f(\mathbf{x})$ dada sobre todo el dominio Ω de \mathbf{x} tal que los términos de alto orden de la expansión sean despreciables.

Un conjunto particular de funciones componente $f_{i_1 i_2 \dots i_l}(x_{i_1}, x_{i_2}, \dots, x_{i_l})$ ($l = 0, 1, \dots, n-1$ con f_0 que corresponde a $l = 0$) se obtiene mediante un procedimiento que minimiza el siguiente funcional [Rabitz et al., 2000]:

$$\min_{f_{i_1 i_2 \dots i_l}} \int_{\Omega} w_{i_1 i_2 \dots i_l}(\hat{\mathbf{x}}, \mathbf{u}) \left[f(\mathbf{u}) - f_0 - \sum_{i=1}^n f_i(u_i) - \sum_{1 \leq i < j \leq n} f_{ij}(u_i, u_j) - \dots - \sum_{i_1 i_2 \dots i_l} f_{i_1 i_2 \dots i_l}(u_{i_1}, u_{i_2}, \dots, u_{i_l}) \right]^2 d\mathbf{u}$$

bajo una condición específica adecuada que garantice que todas las funciones componente se determinen paso a paso.

Aquí:

$$\hat{\mathbf{x}} = (x_{i_1}, x_{i_2}, \dots, x_{i_l})$$

$$d\mathbf{u} = du_1 du_2 \dots du_n$$

La función $w_{i_1 i_2 \dots i_l}(\hat{\mathbf{x}}, \mathbf{u})$ puede ser considerada como una función de peso.

Funciones de peso diferentes producirán distintas expansiones HDMR.

Existen dos tipos de expansiones HDMR usadas comúnmente:

- 1.- Cut-HDMR.- Depende del valor de $f(\mathbf{x})$ en un punto de referencia específico $\hat{\mathbf{x}}$. Se utiliza cuando se puede hacer un muestreo ordenado para la salida $f(\mathbf{x})$ en puntos escogidos \mathbf{x} .
- 2.- RS-HDMR.- Depende del valor promedio de $f(\mathbf{x})$ sobre todo el dominio Ω . Se utiliza cuando el muestreo se hace de manera aleatoria. Las funciones componente se determinan por medio de la obtención de promedios en un conjunto de puntos muestreados de manera aleatoria sobre el dominio Ω . Otro criterio para la elección de la expansión RS-HDMR es cuando exista una gran cantidad de variables de entrada. Aunque se pueda realizar un muestreo ordenado, es mucho más práctico usar RS-HDMR [Geremia & Rabitz, 2001], ya que, el número requerido

de muestras para la construcción de los polinomios ortonormales que se utilizan, es independiente del número de variables [Li et al., 2002 (28)].

La elección del tipo de expansión HDMR a utilizar depende tanto de lo que se desea saber de la función de salida como del tipo y la cantidad de datos de entrada disponibles.

Todas las funciones componente tanto en Cut-HDMR como en RS-HDMR se pueden construir directamente de los valores de salida $f(\mathbf{x})$ en puntos de \mathbf{x} ordenados o generados de manera aleatoria, lo que hace que la construcción de $f_0, f_i(x_i), f_{ij}(x_i, x_j), \dots$ sea simple y directa.

2.1.1. Cut-HDMR.

Cuando es posible utilizar el *muestreo ordenado* para la salida $f(\mathbf{x})$ en puntos escogidos de \mathbf{x} , (como por ejemplo datos de laboratorio con valores controlados de \mathbf{x}), entonces se puede construir una expansión Cut-HDMR.

Para el Cut-HDMR es necesario establecer primero un punto de referencia $\bar{\mathbf{x}} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$ seleccionado dentro del dominio Ω . Las funciones componente óptimas de la ec. (1) en el Cut-HDMR tienen la siguiente estructura [Allis & Rabitz, 1999]:

$$f_0 = f(\bar{\mathbf{x}}) \quad (4)$$

$$f_i(x_i) = f(x_i, \bar{\mathbf{x}}^i) - f_0 \quad (5)$$

$$f_{ij}(x_i, x_j) = f(x_i, x_j, \bar{\mathbf{x}}^{ij}) - f_i(x_i) - f_j(x_j) - f_0 \quad (6)$$

.....

donde:

$$(x_i, \bar{\mathbf{x}}^i) = (\bar{x}_1, \dots, \bar{x}_{i-1}, x_i, \bar{x}_{i+1}, \dots, \bar{x}_n)$$

$$(x_i, x_j, \bar{\mathbf{x}}^{ij}) = (\bar{x}_1, \dots, \bar{x}_{i-1}, x_i, \bar{x}_{i+1}, \dots, \bar{x}_{j-1}, x_j, \bar{x}_{j+1}, \dots, \bar{x}_n)$$

El último término $f_{12\dots n}(x_1, x_2, \dots, x_n)$ está determinado por la diferencia entre $f(\mathbf{x})$ y todas las demás funciones componente de la ecuación (1).

Las formulas mencionadas anteriormente se obtienen substituyendo $(x_{i_1}, x_{i_2}, \dots, x_{i_l}, \bar{x}^{\bar{i}_1 \bar{i}_2 \dots \bar{i}_l})$ con diferentes conjuntos de $\{i_1, i_2, \dots, i_l\} \subset \{1, 2, \dots, n\}$ para \mathbf{x} en ambos lados de la ecuación (1) y usando la siguiente condición:

Una función componente de Cut-HDMR desaparece cuando cualquiera de sus variables propias toma el valor del elemento correspondiente en $\bar{\mathbf{x}}$, es decir:

$$f_{i_1 i_2 \dots i_l}(x_{i_1}, x_{i_2}, \dots, x_{i_l}) \Big|_{x_s = \bar{x}_s} = 0 \quad s \in \{i_1, i_2, \dots, i_l\}$$

La cual define una relación ortogonal entre dos funciones componente diferentes del Cut-HDMR como:

$$f_{i_1 i_2 \dots i_l}(x_{i_1}, x_{i_2}, \dots, x_{i_l}) f_{j_1 j_2 \dots j_k}(x_{j_1}, x_{j_2}, \dots, x_{j_k}) \Big|_{x_s = \bar{x}_s} = 0 \quad s \in \{i_1, i_2, \dots, i_l\} \cup \{j_1, j_2, \dots, j_k\}$$

Si cualquiera de las $x_{i_1}, x_{i_2}, \dots, x_{i_s}$ o de las $x_{j_1}, x_{j_2}, \dots, x_{j_t}$ es igual a la x correspondiente del vector de referencia $\bar{\mathbf{x}}$, el producto es cero.

La manera típica de presentar las funciones componente del Cut-HDMR $f_i(x_i), f_{ij}(x_i, x_j), \dots$ es en forma tabular a partir de la función original $f(\mathbf{x})$ evaluándola en repetidas ocasiones dejando fijas algunas componentes y variando otras.

Se hace notar que las funciones componente del Cut-HDMR $f_i(x_i), f_{ij}(x_i, x_j), \dots$ se definen a lo largo de líneas de corte o a lo largo y ancho de ciertos planos ó en el interior de algún volumen, etc., los cuales incluyen al punto de referencia $\bar{\mathbf{x}}$ en el dominio de la función Ω , de ahí el nombre Cut-HDMR.

Como todas las funciones componente son obtenidas con procedimientos de minimización, éstas son las elecciones óptimas para una salida $f(\mathbf{x})$ dada, de modo que solo se necesitan términos de bajo orden de la expansión Cut-HDRM para tener una buena aproximación de $f(\mathbf{x})$.

Para encontrar el valor aproximado de $f(\mathbf{x})$ en cualquier punto $\mathbf{x} \in \Omega$ se lleva a cabo una interpolación dentro de las tablas de datos generadas por las funciones componente.

Las tablas de datos de las funciones componente se elaboran de la siguiente manera:

Para las funciones componente de grado de cooperatividad $l=1$ $f_i(x_i)$ se forma una tabla para cada $f_i(x_i) = f(x_i, \bar{\mathbf{x}}) - f_0$.

x_i puede tomar s valores diferentes. Si el número de variables de entrada es n , la tabla para cada $f_i(x_i)$ se obtiene fijando el valor de las $n-1$ variables en el punto de referencia y variando solo la x_i en los s valores diferentes:

$\bar{x}_1 \bar{x}_2 \dots \bar{x}_i \dots \bar{x}_n$	$f(x_i, \bar{\mathbf{x}})$	$f_i(x_i) = f(x_i, \bar{\mathbf{x}}) - f_0$
x_{i_1}	$f(x_{i_1}, \bar{\mathbf{x}})$	$f(x_{i_1}, \bar{\mathbf{x}}) - f_0$
x_{i_2}	$f(x_{i_2}, \bar{\mathbf{x}})$	$f(x_{i_2}, \bar{\mathbf{x}}) - f_0$
...	...	
x_{i_s}	$f(x_{i_s}, \bar{\mathbf{x}})$	$f(x_{i_s}, \bar{\mathbf{x}}) - f_0$

Tabla 1: Función tabular $f_i(x_i)$.

Se pueden formar n tablas diferentes para las funciones componente con grado de cooperatividad $l=1$.

El número de muestras requerido para construir las tablas de $f_i(x_i)$ es del orden $O(sn)$.

Para las funciones componente con grado de cooperatividad $l=2$ se obtienen tablas para $f_{ij}(x_i, x_j)$ de tamaño $s_1 \times s_2$, en donde s_1 es el número de valores que puede tomar la variable x_i y s_2 es el número de valores que puede tomar la variable x_j .

$\overline{x_1 x_2 \dots x_i \dots x_j \dots x_n}$	$f(x_i, x_j, \overline{\mathbf{x}}^{-ij})$	$f_{ij}(x_i, x_j) = f(x_i, x_j, \overline{\mathbf{x}}^{-ij}) - f_i(x_i) - f_j(x_j) - f_0$
$x_{i_1} \quad x_{j_1}$	$f(x_{i_1}, x_{j_1}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_1}, x_{j_1}, \overline{\mathbf{x}}^{-ij}) - f_{i_1}(x_{i_1}) - f_{j_1}(x_{j_1}) - f_0$
$x_{i_2} \quad x_{j_1}$	$f(x_{i_2}, x_{j_1}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_2}, x_{j_1}, \overline{\mathbf{x}}^{-ij}) - f_{i_2}(x_{i_2}) - f_{j_1}(x_{j_1}) - f_0$
...
$x_{i_{s_1}} \quad x_{j_1}$	$f(x_{i_{s_1}}, x_{j_1}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_{s_1}}, x_{j_1}, \overline{\mathbf{x}}^{-ij}) - f_{i_{s_1}}(x_{i_{s_1}}) - f_{j_1}(x_{j_1}) - f_0$
$x_{i_1} \quad x_{j_2}$	$f(x_{i_1}, x_{j_2}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_1}, x_{j_2}, \overline{\mathbf{x}}^{-ij}) - f_{i_1}(x_{i_1}) - f_{j_2}(x_{j_2}) - f_0$
$x_{i_2} \quad x_{j_2}$	$f(x_{i_2}, x_{j_2}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_2}, x_{j_2}, \overline{\mathbf{x}}^{-ij}) - f_{i_2}(x_{i_2}) - f_{j_2}(x_{j_2}) - f_0$
...
$x_{i_{s_1}} \quad x_{j_2}$	$f(x_{i_{s_1}}, x_{j_2}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_{s_1}}, x_{j_2}, \overline{\mathbf{x}}^{-ij}) - f_{i_{s_1}}(x_{i_{s_1}}) - f_{j_2}(x_{j_2}) - f_0$
.....
$x_{i_1} \quad x_{j_{s_2}}$	$f(x_{i_1}, x_{j_{s_2}}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_1}, x_{j_{s_2}}, \overline{\mathbf{x}}^{-ij}) - f_{i_1}(x_{i_1}) - f_{j_{s_2}}(x_{j_{s_2}}) - f_0$
$x_{i_2} \quad x_{j_{s_2}}$	$f(x_{i_2}, x_{j_{s_2}}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_2}, x_{j_{s_2}}, \overline{\mathbf{x}}^{-ij}) - f_{i_2}(x_{i_2}) - f_{j_{s_2}}(x_{j_{s_2}}) - f_0$
...
$x_{i_{s_1}} \quad x_{j_{s_2}}$	$f(x_{i_{s_1}}, x_{j_{s_2}}, \overline{\mathbf{x}}^{-ij})$	$f(x_{i_{s_1}}, x_{j_{s_2}}, \overline{\mathbf{x}}^{-ij}) - f_{i_{s_1}}(x_{i_{s_1}}) - f_{j_{s_2}}(x_{j_{s_2}}) - f_0$

Tabla 2: Función tabular $f_{ij}(x_i, x_j)$.

Los valores de $f_{ij}(x_i, x_j)$ se obtienen de los resultados de evaluar la función en $f(x_i, x_j, \overline{\mathbf{x}}^{-ij})$ y de las funciones tabulares f_i, f_j obtenidas anteriormente.

Se pueden formar $\binom{n}{2} = \frac{n!}{2!(n-2)!}$ tablas diferentes para las funciones componente con orden de cooperatividad $l = 2$.

Si cada variable de entrada toma la misma cantidad de valores diferentes s , es decir, si $s_1 = s_2 = s$, el número requerido de evaluaciones de $f(\mathbf{x})$ para construir las tablas de todas las funciones componente $f_i(x_i), f_{ij}(x_i, x_j), \dots$ es:

$$1 + ns + \frac{n(n-1)}{2} s^2 + \frac{n(n-1)(n-2)}{3!} s^3 + \dots + s^n$$

donde n es el número de variables de entrada.

Puede observarse que el número de evaluaciones de la función $f(\mathbf{x})$ es de complejidad polinomial con respecto al número de muestras n , pero es de complejidad exponencial con respecto al número de variables n , es decir, el *costo computacional* crece de manera exponencial con el número de variables de la función. El *costo computacional* se refiere

al número de experimentos necesarios para construir la representación de la función de muchas variables.

Sin embargo, basándose en la *conjetura fundamental de las expansiones HDMR* (ver introducción), es suficiente aproximar a la función $f(\mathbf{x})$ usando un orden de cooperatividad $l = 2$, de esta manera solo se tendrían que hacer: $1 + ns + \frac{n(n-1)}{2}s^2$ evaluaciones de $f(\mathbf{x})$. Esto reduce el problema de elaborar las funciones tabulares, de complejidad exponencial a complejidad polinomial. En este caso el *costo computacional* crece cuadráticamente. El número de muestras requerido para construir las tablas de $f_{ij}(x_i, x_j)$ es del orden $O(s^2n^2)$ donde n es el número de variables de la función y s el número de muestras de cada variable [Geremia & Rabitz, 2001].

Basándose en la conjetura fundamental de las HDMR, solo se necesitan tablas para variables actuando de manera independiente o cuando mucho actuando de dos en dos sobre la salida, de esta manera el ahorro en el muestreo para valores grandes de n es significativo comparado con el muestreo tradicional de s^n .

Si una función de muchas variables se representa con Cut-HDMR de grado de cooperatividad $l=1$ ó $l=2$ el problema del muestreo tradicional de una función con n variables que tiene complejidad s^n se reduce a un problema de complejidad polinomial.

Finalmente, el problema de evaluar la función $f(\mathbf{x})$ en cualquier punto arbitrario \mathbf{x} , se reduce a elaborar las tablas de las funciones componente de orden de cooperatividad $l=1$ y $l=2$ y a una interpolación en varias dimensiones (ver Apéndice C: Interpolación multidimensional).

El laborioso esfuerzo inicial de construir las tablas para todas las funciones componente $f_i(x_i), f_{ij}(x_i, x_j), \dots$ de la expansión HDMR se ve compensado posteriormente cuando es necesario hacer una gran cantidad de evaluaciones del modelo, ya que los resultados que se obtienen son rápidos y exactos.

2.1.2. RS-HDMR.

Para el RS-HDMR, las funciones componente son determinadas por medio de un proceso de obtener promedios en un conjunto de puntos *muestreados de manera aleatoria* sobre el dominio Ω .

Para el RS-HDMR, es necesario normalizar primero las variables x_i de modo que sea $0 \leq x_i \leq 1$ para $i=1,2,\dots, n$. La función de salida $f(\mathbf{x})$ está definida entonces en el hipercubo unitario:

$$K^n = \{(x_1, x_2, \dots, x_n) | 0 \leq x_i \leq 1, i = 1, 2, \dots, n\}$$

Las funciones componente de RS-HDMR poseen las siguientes formas:

$$f_0 = \int_{k^n} f(\mathbf{x}) \, d\mathbf{x} \quad (7)$$

$$f_i(x_i) = \int_{k^{n-1}} f(\mathbf{x}) \, d\mathbf{x}^i - f_0 \quad (8)$$

$$f_{ij}(x_i, x_j) = \int_{k^{n-2}} f(\mathbf{x}) \, d\mathbf{x}^{ij} - f_i(x_i) - f_j(x_j) - f_0 \quad (9)$$

.....

Donde dx^i es el producto $dx_1 dx_2 \dots dx_n$ sin dx_i , y dx^{ij} es el producto $dx_1 dx_2 \dots dx_n$ sin dx_i ni dx_j .

De manera similar, el último término $f_{12\dots n}(x_1, x_2, \dots, x_n)$ se determina con la diferencia entre $f(\mathbf{x})$ y todas las demás funciones componente de la ecuación (1).

Como el hipercubo está normalizado, los intervalos de integración tienen valor 1 para las n dimensiones, al integrar sobre estos intervalos se obtienen n áreas. Como el promedio se obtiene dividiendo el área entre el intervalo de integración que tiene el valor de 1, entonces, en este caso, el promedio de la función $\bar{f}(\mathbf{x})$ es igual a las n áreas.

Considerando que el dominio Ω es un hipercubo unitario, f_0 es en realidad el valor promedio de $f(\mathbf{x})$ sobre todo el dominio, mientras que en el Cut-HDMR f_0 es el valor de $f(\mathbf{x})$ en un solo punto de referencia $\bar{\mathbf{x}}$.

Las ecuaciones (7), (8) y (9) pueden obtenerse rápidamente con la integración de ambos lados de la ecuación (1) con respecto a diferentes conjuntos de variables de entrada: $\{x_{i_1}, x_{i_2}, \dots, x_{i_l}\} (l = n, n-1, \dots, 1)$ y tomando en cuenta la siguiente condición:

La integral de una función componente de RS-HDMR con respecto a cualquiera de sus propias variables es cero, es decir:

$$\int_0^1 f_{i_1 i_2 \dots i_l}(x_{i_1}, x_{i_2}, \dots, x_{i_l}) dx_s = 0 \quad s \in \{i_1, i_2, \dots, i_l\}$$

lo que define la relación ortogonal entre dos funciones componente RS-HDMR diferentes como:

$$\int_{k^n} f_{i_1 i_2 \dots i_l}(x_{i_1}, x_{i_2}, \dots, x_{i_l}) f_{j_1 j_2 \dots j_l}(x_{j_1}, x_{j_2}, \dots, x_{j_l}) dx = 0$$

donde los conjuntos de variables utilizadas deben ser diferentes, es decir: $\{i_1, i_2, \dots, i_l\} \neq \{j_1, j_2, \dots, j_l\}$

La evaluación de las integrales multidimensionales en la expansión RS-HDMR puede llevarse a cabo por medio del método de integración por muestreo aleatorio de Monte Carlo [Hughes, 1998; Press et al., 1999] (ver Apéndice D). El nombre RS-HDMR viene de *muestreo aleatorio* (Random Sampling).

Para construir las tablas de datos para las funciones componente del RS-HDMR es necesario evaluar las integrales:

$$\int_{k^n} f(\mathbf{x}) dx, \int_{k^{n-1}} f(\mathbf{x}) dx^i, \int_{k^{n-2}} f(\mathbf{x}) dx^{ij}, \dots$$

Estas pueden evaluarse por medio del método de Monte Carlo, para lo que es necesario obtener las muestras. Por ejemplo, se generan de manera aleatoria y uniforme N muestras de un vector de dimensión n llamado $\mathbf{x}^{(s)} = (x_1^{(s)}, x_2^{(s)}, \dots, x_n^{(s)})$ ($s = 1, 2, \dots, N$) en el espacio de n dimensiones \mathbf{K}^n .

f_0 es el valor promedio de $f(\mathbf{x})$ para todas las muestras $\mathbf{x}^{(s)}$, entonces:

$$f_0 = \int_{k^n} f(\mathbf{x}) dx \approx \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}^{(s)})$$

Como normalmente las integrales del método de Monte Carlo convergen bastante rápido, un valor relativamente pequeño de N puede dar un resultado muy bueno. También se tiene que con mucha frecuencia la aproximación de la integral por el método de Monte Carlo no depende de manera significativa de la dimensión n del vector \mathbf{x} al integrar $f(\mathbf{x})$. Lo anterior es de mucha utilidad para trabajar con sistemas multidimensionales [Li et al., 2002 (29)].

La determinación directa de todas las funciones componente de la expansión RS-HDMR requiere integrar cada $f(\mathbf{x}) dx^i, f(\mathbf{x}) dx^{ij}, \dots$ con N muestras cada integral, esto significa que se requiere de una gran cantidad de muestras aleatorias.

Por ejemplo, para funciones con orden de cooperatividad $l=1$ es necesario hacer N muestras de $f(x_i, \mathbf{x}^i)$ en los puntos $(x_i, \mathbf{x}^i)^{(s)} = (x_1^{(s)}, x_2^{(s)}, \dots, x_{i-1}^{(s)}, x_i, x_{i+1}^{(s)}, \dots, x_n^{(s)})$ con $s = 1, 2, \dots, N$, así:

$$f_i(x_i) = \int_{k^{n-1}} f(\mathbf{x}) d\mathbf{x}^i - f_0 \approx \frac{1}{N} \sum_{s=1}^N f((x_i, \mathbf{x}^i)^{(s)}) - \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}^{(s)})$$

(Se aplicó el método de Monte Carlo para hacer la aproximación de la integral).

Si x_i toma s_I valores diferentes $x_{i_1}, x_{i_2}, \dots, x_{i_{s_I}}$, se toman N muestras para cada valor x_i fijo tomando al azar el resto de los valores de las demás variables y se obtiene una tabla como la *Tabla 3* de tamaño N para cada uno de los s_I valores de x_i .

$f(x_i, \mathbf{x}^i)$
$f(x_1^{(1)}, x_2^{(1)}, \dots, x_i, x_{i+1}^{(1)}, \dots, x_n^{(1)})$
$f(x_1^{(2)}, x_2^{(2)}, \dots, x_i, x_{i+1}^{(2)}, \dots, x_n^{(2)})$
...
$f(x_1^{(N)}, x_2^{(N)}, \dots, x_i, x_{i+1}^{(N)}, \dots, x_n^{(N)})$

Tabla 3 : N muestras diferentes de la función $f(\mathbf{x})$ para valores de x_i fijos.

Con esta tabla se obtiene el valor promedio de $f(x_i, \mathbf{x}^i)$ mediante:

$$f(x_i) \approx \frac{1}{N} \sum_{s=1}^N f((x_i, \mathbf{x}^i)^{(s)}) - \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}^{(s)})$$

Y este sería solo el valor del primer renglón y segunda columna de la *Tabla 4*:

$f(x_i, \mathbf{x}^i)$	$f_i(x_i) \approx \frac{1}{N} \sum_{s=1}^N f((x_i, \mathbf{x}^i)^{(s)}) - \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}^{(s)})$
$f(x_{i_1}, \mathbf{x}^i)$	$f(x_{i_1})$
$f(x_{i_2}, \mathbf{x}^i)$	$f(x_{i_2})$
...	
$f(x_{i_{s_1}}, \mathbf{x}^i)$	$f(x_{i_{s_1}})$

Tabla 4: Valores promedio de $f_i(x_i)$.

Donde s_l es el número de valores que toma la variable x_i .

De tal manera que para completar la *Tabla 4*, es necesario elaborar s_l tablas similares a la *Tabla 3* tomando N muestras en cada tabla.

En total se requieren $s \times N$ muestras aleatorias para construir la función $f_i(x_i)$ de la *Tabla 4*.

Nótese que se requieren n tablas similares a la *Tabla 4* si el tamaño del vector de entrada $\mathbf{x} = (x_1, x_2, \dots, x_n)$ es n .

Para construir la tabla de una función con orden de cooperatividad $l = 2$, el proceso de obtención de cada tabla para $f_{ij}(x_i, x_j)$ se hace aún más complejo, ya que es necesario construir $s_1 \times s_2$ tablas similares a la *Tabla 3* para obtener cada uno de los valores de la segunda columna de la *Tabla 5*:

$f(x_i, x_j, \mathbf{x}^{ij})$	$f_{ij}(x_i, x_j)$
--------------------------------	--------------------

$f(x_{i_1}, x_{j_1}, \mathbf{x}^{ij})$	$f(x_{i_1}, x_{j_1})$
$f(x_{i_2}, x_{j_1}, \mathbf{x}^{ij})$	$f(x_{i_2}, x_{j_1})$
...
$f(x_{i_{s_1}}, x_{j_1}, \mathbf{x}^{ij})$	$f(x_{i_{s_1}}, x_{j_1})$
$f(x_{i_1}, x_{j_2}, \mathbf{x}^{ij})$	$f(x_{i_1}, x_{j_2})$
$f(x_{i_2}, x_{j_2}, \mathbf{x}^{ij})$	$f(x_{i_2}, x_{j_2})$
...
$f(x_{i_{s_1}}, x_{j_2}, \mathbf{x}^{ij})$	$f(x_{i_{s_1}}, x_{j_2})$
.....
$f(x_{i_1}, x_{j_{s_2}}, \mathbf{x}^{ij})$	$f(x_{i_1}, x_{j_{s_2}})$
$f(x_{i_2}, x_{j_{s_2}}, \mathbf{x}^{ij})$	$f(x_{i_2}, x_{j_{s_2}})$
...
$f(x_{i_{s_1}}, x_{j_{s_2}}, \mathbf{x}^{ij})$	$f(x_{i_{s_1}}, x_{j_{s_2}})$

Tabla 5: Valores promedio de $f_{ij}(x_i, x_j)$.

Donde:

- s_1 es el número de valores que toma la variable x_i
- s_2 es el número de valores que toma la variable x_j

La obtención de las $f(x_i, x_j)$ se hace por medio de la siguiente ecuación:

$$f_{ij}(x_i, x_j) = \int_{k^{n-2}} f(\mathbf{x}) d\mathbf{x}^{ij} - f_i(x_i) - f_j(x_j) - f_0 \approx$$

$$\frac{1}{N} \sum_{s=1}^N f((x_i, x_j, \mathbf{x}^{ij})^{(s)}) + \frac{1}{N} \sum_{s=1}^N f((x_i, \mathbf{x}^i)^{(s)}) - \frac{1}{N} \sum_{s=1}^N f((x_j, \mathbf{x}^j)^{(s)}) + \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}^{(s)})$$

Suponiendo que tanto x_i como x_j toman m valores diferentes, se requieren de $m^2 N$ muestras diferentes para construir la tabla 5. Nótese que para un vector de entrada de tamaño n se requieren $\frac{n(n-1)}{2}$ tablas similares a la *Tabla 5*.

El número requerido de muestras aleatorias crece exponencialmente con el orden de cooperatividad de las funciones componente RS-HDMR. Por esta razón con frecuencia es prohibitivamente costoso construir las tablas de las funciones componente RS-HDMR con grado de cooperatividad $l=2$ y $l=3$.

Para reducir el esfuerzo del muestreo en el RS-HDMR se utilizan algunas funciones convenientes (polinomios ortonormales) para aproximar las funciones componente de la expansión [Li et al., 2002 (29)].

2.2. HDMR y sus aplicaciones.

Hay una gran variedad de áreas en donde son útiles las técnicas HDMR, de hecho, ya se han publicado una serie de artículos con aplicaciones concretas; en cinética química [Shorter et al., 1999; Wang *et al.*, 1999], en transporte de radiación [Shorter & Percila, 2000], en descubrimiento de materiales [Rabitz & Shim, 1999; Shim & Rabitz, 1998] y en análisis estadístico [Li et al., 2002 (29)]. A continuación se mencionan las cuatro áreas principales en donde se aplican las expansiones HDMR (1).

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^n f_i(x_i) + \sum_{1 \leq i < j \leq n} f_{ij}(x_i, x_j) + \sum_{1 \leq i < j < k \leq n} f_{ijk}(x_i, x_j, x_k) + \dots + f_{12\dots n}(x_1, x_2, \dots, x_n) \quad (1)$$

2.2.1. Construcción de modelos basados en la observación directa desde el laboratorio o de los datos del campo.

Cuando no es fácil construir un modelo matemático de un sistema, el comportamiento de las salidas en función de las entradas puede basarse en observaciones en el laboratorio o en el campo. En este caso, el sistema se considera simplemente como una caja negra. Las entradas consisten en las variables medidas en el laboratorio u observadas en el campo y las salidas son la respuesta observada del sistema.

Con frecuencia la entrada consiste en un vector de muchas variables $\mathbf{x} = (x_1, x_2, \dots, x_n)$ donde $n \approx 10^2 - 10^3$ o mas, mientras que la salida es una sola cantidad $f(\mathbf{x})$.

Uno de los objetivos de los científicos es identificar el mapeo entrada-salida por medio de un muestreo finito de las variables de entrada.

Hay métodos tradicionales para atacar este tipo de problemas, sin embargo, estos emplean una gran carga computacional, la cual crece exponencialmente conforme aumenta el número de variables del sistema, de tal manera que para sistemas con $n \gg 10$ el uso de estos métodos es prohibitivo.

Las técnicas HDMR son herramientas muy poderosas para modelar este tipo de sistemas. Cuando es posible llevar a cabo un muestreo ordenado, se utilizan las expansiones Cut-HDMR y cuando se considera que los datos disponibles están producidos de forma aleatoria se utilizan las expansiones RS-HDMR.

Una vez que se ha construido la expansión HDMR dada por la Ec.(1) se puede obtener una reproducción bastante precisa de los datos obtenidos del campo o laboratorio y una predicción razonable de la salida $f(\mathbf{x})$ para una entrada \mathbf{x} dada.

La expansión HDMR sirve como un modelo matemático con ventajas específicas sobre aquellos modelos generados con métodos estadísticos convencionales, estas ventajas son [Rabitz et al., 2000]:

- El modelo no depende de ninguna restricción física, química o de suposiciones teóricas. Su exactitud solo depende de la cantidad y de la calidad de los datos. Es especialmente útil cuando no se tiene un conocimiento completo y detallado del sistema.
- El modelo puede utilizarse para un análisis estadístico de su propia calidad por medio de la colocación de banderas cuantitativas en las predicciones resultantes, con estas banderas se puede determinar si es necesario hacer mediciones adicionales para mejorar la calidad del modelo.
- El modelo puede mejorarse continuamente sin tener que descartar los términos ya obtenidos; para mejorarlo solo es necesario ir añadiendo más términos nuevos a la expansión cuando se haya identificado la información sobre las nuevas variables de entrada.

2.2.2. Valoración de la incertidumbre global e identificación de las variables clave y su interrelación.

Las funciones componente individuales de las expansiones RS-HDMR se pueden interpretar directamente como una correlación estadística, esto permite que la varianza de la salida σ_f^2 pueda descomponerse en las contribuciones $\sigma_i^2, \sigma_{ij}^2, \dots$ de las variables de entrada (La demostración matemática de esta propiedad está en la 2.3.3. Descomposición de la varianza en las expansiones HDMR).

La información que proporciona esta descomposición permite alcanzar una comprensión física de los orígenes de la incertidumbre a la salida.

Una de las características de los modelos obtenidos con expansiones HDMR es que se obtienen funciones con pocas variables las cuales se pueden muestrear exhaustivamente, lo que permite una mejora en la calidad de los datos.

Por otra parte, analizando detalladamente las subcomponentes $\sigma_i^2, \sigma_{ij}^2, \dots$ de la varianza total σ_f^2 es posible identificar cuales son las variables de entrada importantes del modelo y como interactúan una con otra. La información resultante de este análisis proporcionará sugerencias para la mejora del modelo. Este tema se trata más ampliamente en el Apéndice G.

2.2.3. Construcción de un modelo operacional completamente equivalente.

Una de las aplicaciones más interesantes de las HDMR es la obtención de modelos operacionales completamente equivalentes (en Ingles FEOM: Fully Equivalent Operational Model).

Teniendo en cuenta que los modelos complejos pueden estar compuestos de la unión de varios submodelos, puede partirse a los sistemas complejos de tal forma que las partes que requieran de una gran carga computacional puedan modelarse de manera independiente con expansiones HDMR.

Cuando se reemplaza el o los componentes lentos del sistema por expansiones HDMR, la ejecución del modelo se vuelve mucho más eficiente.

Las funciones componente $f_i(x_i), f_{ij}(x_i, x_j), \dots$ para un submodelo se obtienen de manera aislada, es decir, sin tomar en cuenta el resto de los subsistemas.

El Modelo Operacional Completamente Equivalente (FEOM) de un sistema se obtiene con la expansión de funciones componente HDMR. El FEOM se utiliza para predecir el comportamiento de la salida de un subsistema en cualquier punto \mathbf{x} dentro del dominio Ω . En algunos casos el FEOM puede reemplazar a todo un sistema completo.

El uso del FEOM tiene un gran campo de aplicación, sobre todo en el reemplazo de componentes que implican un gran esfuerzo computacional.

Solo en el caso de que una entrada \mathbf{x} esté fuera de la región que abarca el FEOM, o que se determine que la predicción del FEOM tiene errores considerables, se utiliza el componente original del modelo para obtener resultados exactos. Además, con estos nuevos resultados se mejora el FEOM para su uso posterior.

El capítulo 3 está dedicado a la descripción e implantación del FEOM para aplicarlo posteriormente en la solución de diversos sistemas de ecuaciones diferenciales que corresponden a las reacciones químicas del modelo de difusión/advección de contaminantes para la ciudad de México.

2.2.4. Valoración del riesgo cuantitativamente eficiente.

Para la valoración cuantitativa del riesgo, generalmente es necesario reconsiderar el conjunto original de variables y dividirlos en 2 componentes $(x_1, x_2, \dots, x_s; y_1, y_2, \dots, y_r; s + r = n)$ donde el conjunto $\{x_i\}$ son las variables bajo el control humano (variables de escenario) y el conjunto $\{y_i\}$ corresponde a las demás variables del modelo, las cuales están sujetas a cierto grado de incertidumbre [Rabitz et al., 2000].

Para medir el riesgo, se identifica si la salida $f(\mathbf{x}, \mathbf{y})$ sobrepasa (o no llega) a un valor crítico f_c .

El riesgo se define como la probabilidad $p(f > f_c)$ de que ocurra este evento tomando en cuenta la incertidumbre de entre el conjunto de variables de entrada $\{y_i\}$.

Así que el riesgo está definido como:

$$R = \int H[f(\mathbf{x}, \mathbf{y}) - f_c] d\mathbf{x}d\mathbf{y} \quad (10)$$

y la varianza del riesgo es:

$$\sigma_r^2 = \int H[f(\mathbf{x}, \mathbf{y}) - f_c] H[f(\mathbf{u}, \mathbf{y}) - f_c] d\mathbf{u}d\mathbf{y} - R^2 \quad (11)$$

donde $H[z] = \begin{cases} 1 & z \geq 1 \\ 0 & z < 0 \end{cases}$

La distribución de probabilidad para $\{y_i\}$ se pliega a la transformación de las variables a la forma de hipercubo unitario.

Con las expansiones HDMR se tiene la gran ventaja de poder evaluar el riesgo de una manera cuantitativa. Estas tareas se facilitan debido a la capacidad de evaluar fácilmente $f(\mathbf{x}, \mathbf{y})$ en la ec. (10) y a la amplia cobertura del espacio $\{x_i, y_i\}$.

En suma, es posible determinar la porción de variables de escenario $\{x_1, x_2, \dots, x_s\}$ que contribuye independientemente o de forma correlacionada al riesgo.

El análisis no solo proporciona el riesgo R (10), sino también el aseguramiento de la calidad del riesgo a través de su varianza σ_R^2 (11), ya que se tienen las variables $\{y_i\}$ del modelo y su respectiva incertidumbre.

Una aplicación práctica del análisis de incertidumbre en la cinética de las reacciones fotoquímicas, por medio de las técnicas HDMR, puede consultarse en [Wang et al., 2001].

2.3. Propiedades de las expansiones HDMR.

Las siguientes propiedades de las expansiones HDMR son importantes para su aplicación en diferentes problemas científicos.

2.3.1. Convergencia rápida de las expansiones HDMR.

Las expansiones HDMR convergen rápidamente. Esta propiedad puede demostrarse de la siguiente manera [Rabitz et al., 2000]:

Supongamos que una salida $f(\mathbf{x})$ definida en un hipercubo unitario de \mathbf{x} puede expandirse como una serie de Taylor convergente en un punto de referencia $\bar{\mathbf{x}}$:

$$f(\mathbf{x}) = f(\bar{\mathbf{x}}) + \sum_{i=1}^n \frac{\partial f(\bar{\mathbf{x}})}{\partial x_i} (x_i - \bar{x}_i) + \sum_{i,j=1}^n \frac{1}{2!} \frac{\partial^2 f(\bar{\mathbf{x}})}{\partial x_i \partial x_j} (x_i - \bar{x}_i)(x_j - \bar{x}_j) + \dots$$

Las funciones componente $f_0, f_i(x_i), f_{ij}(x_i, x_j) \dots$ del Cut-HDMR según las ecuaciones (4), (5) y (6) son:

$$f_0 = f(\bar{\mathbf{x}})$$

$$f_i(x_i) = f(x_i, \bar{\mathbf{x}}^i) - f_0$$

$$f_{ij}(x_i, x_j) = f(x_i, x_j, \bar{\mathbf{x}}^{ij}) - f_i(x_i) - f_j(x_j) - f_0$$

∴

Substituyendo $(x_i, \bar{\mathbf{x}}^i)$ por \mathbf{x} y restando $f(\bar{\mathbf{x}})$ de ambos lados de la expansión en la ecuación (1) se obtiene $f_i(x_i)$.

Como todos los términos que contienen x_j ($j \neq i$) desaparecen, la función componente de primer orden $f_i(x_i)$ es la suma de todos los términos de la serie de Taylor que contienen únicamente a la variable x_i .

De manera similar, la función componente de segundo orden $f_{ij}(x_i, x_j)$ es la suma de todos los términos de la serie de Taylor que contienen únicamente las variables x_i y x_j .

Así, el número infinito de términos de la serie de Taylor está dividido en un número finito de grupos diferentes. Cada grupo corresponde a una función componente del Cut-HDMR.

En otras palabras, cada función componente del Cut-HDMR está compuesta por un subconjunto infinito de términos de la serie de Taylor. Por esto, cualquier expansión Cut-HDMR truncada, proporciona una mejor aproximación a $f(\mathbf{x})$ que cualquier serie de Taylor truncada, ya que esta última solo contiene un número finito de la serie de Taylor, mientras que la primera contiene a todas las funciones componente del orden en donde se trunque la expansión.

Aún mas, considerando que $0 \leq x_i \leq 1$ ($i = 1, 2, \dots, n$) y que $(x_i - \bar{\mathbf{x}}_i) < 1$, las funciones componente de alto orden están compuestas por el producto: $\prod_{s=1}^l (x_{i_s} - \bar{\mathbf{x}}_{i_s})^{k_s}$ con l grande, por lo que el valor de la multiplicatoria es muy pequeño.

Hay que mencionar que lo anterior no es una prueba estrictamente matemática, pero si alcanza a dar una idea del porqué la expansión Cut-HDMR converge rápido.

Además, los subconjuntos de términos de la serie de Taylor que corresponden a diferentes funciones componente, no se traslapan la una con la otra, lo cual es la base de la relación ortogonal entre dos funciones componente del Cut-HDMR.

Las expansiones RS-HDMR poseen la misma propiedad que Cut-HDMR, ya que puede establecerse una relación uno a uno entre RS-HDMR y Cut-HDMR [Rabitz et al., 2000].

2.3.2. Concordancia de las expansiones HDMR con las leyes de conservación.

Si un conjunto de salidas $\{f^{(1)}(x), f^{(2)}(x), \dots, f^{(s)}(x)\}$ obedecen a un conjunto de leyes de conservación de superposición lineal, entonces sus aproximaciones HDMR de cualquier orden también obedecen esas leyes de conservación, por ejemplo:

Si:

$$\sum_{i=1}^s w_{ki} f^{(i)}(x) = c_k \quad k = 1, 2, \dots, m \quad (12)$$

donde w_{ki} y c_k son dos conjuntos de constantes, entonces:

$$\sum_{i=1}^s w_{ki} [\mu_l f^{(i)}(x)] = c_k \quad k = 1, 2, \dots, m \quad \text{y} \quad l = 0, 1, 2, \dots, n$$

Aquí:

$$\mu_l = \rho_0 + \sum_{i=1}^n \rho_i + \dots + \sum_{i_1 i_2 \dots i_l} \rho_{i_1 i_2 \dots i_l}$$

y $\mu_l f^{(i)}(x)$ es la aproximación de $f^{(i)}(x)$ de orden l . Esta propiedad puede probarse aplicando el operador μ_l a ambas partes de la ecuación (12) y usando la identidad:

$$\mu_l c = c \quad \text{donde } c \text{ es una constante}$$

La concordancia de las expansiones HDMR con las leyes de conservación es muy útil para la aplicación de las HDMR en física, química y otras disciplinas científicas en donde las leyes de conservación (por ejemplo, de la masa, de la energía, del momentum, etc.) son importantes.

2.3.3. Descomposición de la varianza en las expansiones HDMR.

Utilizando la propiedad de ortogonalidad de las funciones componente de las expansiones RS-HDMR, puede probarse que la varianza total σ_f^2 de $f(\mathbf{x})$ causada por todas las variables de entrada, puede ser descompuesta en la suma de las diferentes contribuciones a la varianza total de cada una de las variables de entrada. Esto, debido a que la contribución a la varianza total σ_i^2 de x_i actúa de manera independiente a la contribución σ_{ij}^2 de las variables correlacionadas x_i, x_j , etc.

Así:

$$\begin{aligned} \sigma_f^2 &= \int_{k^n} [f(\mathbf{x}) - \bar{f}]^2 dx = \int_{k^n} [f(\mathbf{x}) - f_0]^2 dx = \\ &= \int_{k^n} \left[\sum_{i=1}^n f_i(x_i) + \sum_{1 \leq i < j \leq n} f_{ij}(x_i, x_j) + \dots \right]^2 dx \end{aligned}$$

donde \bar{f} es el valor medio de $f(\mathbf{x})$ sobre todo el dominio Ω .

Por la ortogonalidad de las funciones componente, los productos entre ellas son cero, y solo quedan los términos:

$$\begin{aligned}
&= \sum_{i=1}^n \int_0^1 f_i^2(x_i) dx_i + \sum_{1 \leq i < j \leq n} \int_0^1 \int_0^1 f_{ij}^2(x_i, x_j) dx_i dx_j + \dots \\
&= \sum_{i=1}^n \sigma_i^2 + \sum_{1 \leq i < j \leq n} \sigma_{ij}^2 + \dots
\end{aligned}$$

Esta propiedad es muy útil para el análisis global de incertidumbre (Apéndice G), ya que la descomposición es válida sobre todo el dominio. Es fácil encontrar cómo influye cada una de las incertidumbres de las variables de entrada en la incertidumbre total de la salida analizando las magnitudes de $\sigma_i^2, \sigma_{ij}^2$, etc. También es posible determinar con estos datos cuáles son las *variables clave* de entrada y qué clase de cooperatividad existe entre las variables de entrada.

2.4. Expansiones HDMR avanzadas.

2.4.1. MultiCut-HDMR vs. Cut-HDMR con múltiples subdominios.

En algunos casos la expansión Cut-HDMR truncada no puede aproximar bien a una función $f(\mathbf{x})$ cuando el dominio Ω de \mathbf{x} es muy grande, porque en este caso muchos puntos del dominio quedan lejos del punto de referencia $\bar{\mathbf{x}}$. Esta es una restricción del Cut-HDMR. MultiCut-HDMR es una herramienta que hace frente a este problema utilizando múltiples puntos de referencia para mejorar la exactitud de la aproximación [Li et al., 2004].

Los principios básicos de la expansión HDMR se pueden extender a casos más generales. Una expansión MultiCut-HDMR consta de varias expansiones Cut-HDMR de orden l , en distintos puntos de referencia $\mathbf{a}(1), \mathbf{a}(2), \dots, \mathbf{a}(m)$. La aproximación a $f(\mathbf{x})$ se construye a partir de las m expansiones Cut-HDMR:

$$f(\mathbf{x}) = \sum_{k=1}^m w_k(\mathbf{x}) \left[f_0^{(k)} + \sum_{i=1}^n f_i^{(k)}(x_i) + \dots + \sum_{1 \leq i_1 < i_2 < \dots < i_l \leq n} f_{i_1 i_2 \dots i_l}^{(k)}(x_{i_1}, x_{i_2}, \dots, x_{i_l}) \right],$$

donde los coeficientes $w_k(\mathbf{x})$ poseen las siguientes propiedades:

$$w_k(\mathbf{x}) = \begin{cases} 1 & \text{si } \mathbf{x} \text{ esta en cualquier subvolumen de corte del } k\text{-ésimo punto de referencia} \\ 0 & \text{si } \mathbf{x} \text{ esta en cualquier subvolumen de corte de otro punto de referencia} \end{cases},$$

$$\sum_{k=1}^m w_k(\mathbf{x}) = 1.$$

Las propiedades de los coeficientes $w_k(\mathbf{x})$ implican que si el punto \mathbf{x} está localizado sobre una línea de corte, un plano o cualquier subvolumen de la expansión Cut-HDMR para un punto de referencia k ($1 \leq k \leq m$), la contribución de todas las demás expansiones Cut-HDMR desaparece y la expansión MultiCut-HDMR se reduce a una sola.

La aproximación usando una expansión Cut-HDMR de orden l para un punto \mathbf{x} , no tiene error cuando \mathbf{x} está localizado en alguno de los subvolúmenes mencionados anteriormente.

Cuando se utilizan m expansiones Cut-HDMR para construir una expansión MultiCut-HDMR, la región libre de error para una entrada \mathbf{x} es m veces mas grande que para una sola expansión Cut-HDMR, es por esto que se mejora la exactitud de la aproximación.

Una de las formas de definir $w_k(\mathbf{x})$ es: la distancia $\rho_k^{i_1, \dots, i_l}$ entre el punto \mathbf{x} donde se va a evaluar la función y el subvolumen de dimensión l que pasa por el punto de referencia $\mathbf{a}(k)$ ($=1, 2, \dots, m$) de la k -ésima expansión. Como se muestra en la fórmula (13).

$$\rho_k^{i_1, \dots, i_l}(\mathbf{x}) = \left[\sum_{i=1}^l [x_i - \mathbf{a}_i(k)]^2 \right]^{1/2} \quad \{i_1, i_2, \dots, i_l\} \subset \{1, 2, \dots, n\} \quad (13)$$

Una vez calculadas las distancias, se define a $\overline{w}_k(\mathbf{x})$ como:

$$\overline{w}_k(\mathbf{x}) = \prod_{\substack{s=1 \\ s \neq k}}^m \prod_{1 \leq i_1 < \dots < i_l \leq n} \rho_s^{i_1, \dots, i_l}(\mathbf{x})$$

$$w_k(\mathbf{x}) = \frac{\overline{w}_k(\mathbf{x})}{\sum_{s=1}^m \overline{w}_s(\mathbf{x})}$$

Puede demostrarse que la $\overline{w}_k(\mathbf{x})$ definida satisface las propiedades requeridas siempre y cuando los diferentes puntos de referencia $\mathbf{a}(k)$ no compartan ninguna coordenada [Li et al., 2004].

Cuando los puntos de referencia $\mathbf{a}(k)$ tienen el mismo valor para alguno de sus elementos, entonces debe usarse una definición modificada para $\overline{w}_k(\mathbf{x})$.

Otra manera de calcular un MultiCut-HDMR es usando interpolación lineal de Lagrange [Li et al., 2004]. Hay que considerar que los cálculos para un MultiCut-HDMR llevan tiempo y hacen más lenta la evaluación de la aproximación a la función.

Cuando el dominio de una función es demasiado grande para que una expansión Cut-HDMR pueda aproximar con exactitud a $f(\mathbf{x})$ puede aplicarse el MultiCut-HDMR o bien puede optarse por el Cut-HDMR con múltiples dominios. En Shoendorf et al. (2003) utilizan Cut-HDMR con múltiples dominios para ampliar el área de trabajo de un FEOM. Con esta técnica se divide el dominio en varios subdominios, se elige un punto de referencia en cada uno de estos subdominios y se construye un FEOM para cada punto de referencia. La aproximación a la función se obtiene calculando la expansión Cut-HDMR del subdominio en donde se encuentra el punto a evaluar. Con esta técnica se tiene la ventaja de la rapidez, ya que se calcula una sola expansión HDMR pero, se requiere memoria para todos los FEOMs que cubren el dominio de la función.

Li et al. (2004) reportaron una aplicación donde se compararan los resultados obtenidos de MultiCut-HDMR con los de Cut-HDMR con múltiples dominios. La exactitud resultó

ser ligeramente mejor con el MultiCut-HDMR, sin embargo en la publicación no se menciona nada respecto a la rapidez de los cálculos. Es de suponer que aunque este último método es ligeramente más exacto, también es más lento por la cantidad de cálculos adicionales que implica. No obstante, necesita de menos FEOMs para abarcar con exactitud una amplia área de trabajo, por lo tanto requiere de menos memoria.

2.4.2. HDMR para un funcional.

Si las entradas de un sistema son un conjunto de funciones, por ejemplo, el vector de entrada está en función del tiempo: $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_n(t))$, entonces la salida del sistema se convierte en un funcional.

Una forma de hacer la aproximación a este funcional es usar la siguiente discretización:

$$x_i(t) = \sum_{k=1}^{N_i} c_{ik} \phi_k(t),$$

donde $\{\phi_k(t)\}$ es una familia de funciones ortogonales. De esta manera, cualquier funcional se convierte en una función cuyos parámetros son c_{ik} y así son aplicables las fórmulas estándar de las expansiones HDMR.

Este tipo de aproximaciones, ya se han llevado a cabo con éxito en el problema de quantums dispersos [Geremia et al., 2001; Geremia & Rabitz, 2001] y en el problema del calentamiento por irradiación atmosférica [Wang et al., 1999].

Conclusiones del capítulo I.

Los modelos matemáticos utilizados para representar fenómenos físicos y químicos del mundo real, o de procesos de laboratorio, tienen con frecuencia una gran cantidad de variables de entrada.

Uno de los objetivos generales es deducir la estructura detallada del vector de entrada de dimensión n para identificar cuales son las variables que tienen mas impacto sobre el modelo o sobre la salida observada.

Otro objetivo es poder predecir los valores de salida del sistema dados un punto (x_1, x_2, \dots, x_n) cualquiera y el modelo matemático del sistema.

Un análisis de todo el espacio al que pertenecen las variables, es un problema de tipo NP-completo, ya que la carga computacional crece exponencialmente con el número de variables de entrada n de forma: $\sim s^n$ (donde “ s ” es el número de muestras del sistema). A esto se le llama la *maldición de la dimensionalidad* [Rabitz et al., 1998].

El calificativo de “gran cantidad” de variables de entrada depende del sistema. En algunas situaciones, por razones prácticas, puede considerarse que $n \sim 3 - 5$ es una “gran cantidad”, mientras que en otras circunstancias este calificativo se le da a sistemas con $n \gg 10$.

La expansión HDMR modela las relaciones existentes entre la entrada y la salida de un sistema, con una serie de funciones correlacionadas que representan con gran exactitud al sistema original.

Las HDMR's se basan en la siguiente conjetura fundamental: “*en muchos sistemas físicos las correlaciones de orden alto entre las variables de entrada son despreciables y la magnitud de esta correlación decrece muy rápidamente después de $l=2$* ” (en donde l es el grado de cooperatividad entre las variables de entrada) [Saltelli & Sobol, 1995]. Por esto, muchos sistemas físicos pueden ser descritos estadísticamente por medio de un análisis de la covarianza [Schoendorf et al., 2003].

Si con un grado de cooperatividad $l=1$ no se puede representar correctamente al sistema original, entonces se agregan los efectos cooperativos con dos variables de entrada, y así sucesivamente hasta lograr el grado de exactitud deseado. Sin embargo, la experiencia ha demostrado que solo en muy raros casos existe un grado de cooperatividad $l=3$.

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^n f_i(x_i) + \sum_{1 \leq i < j \leq n} f_{ij}(x_i, x_j) + \sum_{1 \leq i < j < k \leq n} f_{ijk}(x_i, x_j, x_k) + \dots + f_{12\dots n}(x_1, x_2, \dots, x_n)$$

Como se ha observado en la práctica que existe un grado bajo de cooperatividad ($l=2$ o $l=3$) en los fenómenos de la ciencia y de la ingeniería, incluyendo los problemas con dimensiones atómicas, entonces el problema original se reduce de complejidad exponencial a complejidad polinomial.

El propósito de aplicar HDMR a un modelo complejo es reducir significativamente la carga computacional.

Cuando es posible realizar experimentos controlados sobre un sistema, es decir, hay libertad para elegir los valores de las variables de entrada, entonces se utiliza la expansión Cut-HDMR para representar a $f(\mathbf{x})$ de una forma computacionalmente eficiente.

Sin embargo, en algunas ocasiones no es posible controlar los experimentos, es decir, la naturaleza de las variables de entrada es aleatoria o éstas están dispersas arbitrariamente. En estos casos se usan las expansiones RS-HDMR [Alis & Rabitz, 2000].

Cuando existe una gran cantidad de variables de entrada, aunque se pueda realizar un muestreo ordenado, es más práctico usar RS-HDMR ya que el número requerido de muestras para la construcción de los polinomios ortonormales es independiente del número de variables [Li et al., 2000].

Cuando la obtención de una salida del sistema a analizar requiere de mucho esfuerzo, se produce la llamada *escasez de los datos* la cual representa una severa dificultad para la obtención de un buen modelo [Li et al., 2000].

HDMR tiene una gran variedad de aplicaciones en los problemas que tienen datos escasos, ya que puede representar de manera eficiente a sistemas de muchas variables que producen pocos datos.

Por otra parte, el método de integración de Monte Carlo es el único medio práctico para integrar funciones de muchas variables y por lo tanto es el que se utiliza para obtener las funciones componente de las expansiones RS-HDMR. Los errores producidos por las aproximaciones de Monte Carlo son independientes del número de variables de entrada n y son del orden $1/\sqrt{N}$ (N es el número de muestras). [Alis & Rabitz, 2000].

Además, el número requerido de muestras para representar a un sistema con una tolerancia dada es independiente del número de variables [Shreider, 1967].

Las expansiones HDMR tienen tres propiedades importantes:

- 1.- Convergen rápidamente.
- 2.- Satisfacen las leyes de conservación.
- 3.- La varianza total σ_f^2 de $f(\mathbf{x})$ causada por todas las variables de entrada, puede

ser descompuesta en la suma de las diferentes contribuciones a la varianza total de cada una de las variables de entrada, es decir:

$$\sigma_f^2 = \sum_{i=1}^n \sigma_i^2 + \sum_{1 \leq i < j \leq n} \sigma_{ij}^2 + \dots \quad .$$

En el siguiente capítulo se define que es un FEOM (Fully Equivalent Operational Model), como se construye y como se aplica en la solución de sistemas de ecuaciones diferenciales. Además se incluye el proceso de diseño de software que se utilizó para elaborar un FEOM.

CAPÍTULO 3.

Modelo Operacional Completamente Equivalente (FEOM: Fully Equivalent Operational Model) y su desarrollo mediante un proceso de diseño de software.

3.1. Definición.

Un FEOM (Fully Equivalent Operational Model) es un modelo que reemplaza a la parte lenta de un sistema y que permite obtener salidas equivalentes a las del modelo original pero más rápidamente.

El uso de un FEOM en lugar de una parte del sistema (o del sistema completo) es una de las aplicaciones principales de las HDMR. El FEOM se usa para calcular las salidas aproximadas de un modelo a partir de ciertas entradas, con la ventaja de que el tiempo para obtener la salida es una fracción del que se emplearía para obtener la salida con el modelo original, además, con una muy buena exactitud.

El FEOM de un sistema se obtiene con la expansión de las funciones componente HDMR, y se utiliza para predecir el comportamiento de la salida de un sistema en cualquier punto x dentro de cierto dominio Ω . En algunos casos el FEOM puede reemplazar a todo un sistema completo. Cuando se reemplaza a los componentes lentos del sistema por expansiones HDMR, la ejecución del modelo se vuelve mucho más eficiente [Shorter et al., 1999].

Una de las aplicaciones importantes de los FEOM's es el modelado de sistemas descritos por medio de un conjunto de ecuaciones diferenciales. La ventaja de usar un FEOM es que la obtención de la salida, dada cierta entrada, es mucho más rápida que la obtención de la salida utilizando el modelo original.

Una condición importante para que se pueda implementar un FEOM es que el paralelepípedo de dimensión n de todas las condiciones iniciales posibles $[c_{1_{\min}}, \dots, c_{1_{\max}}] \times [c_{2_{\min}}, \dots, c_{2_{\max}}] \times \dots \times [c_{n_{\min}}, \dots, c_{n_{\max}}]$ esté mapeado a sí mismo en cada paso del procedimiento de integración, de otra manera, el FEOM no puede ser utilizado. (En nuestro caso n es el número de sustancias consideradas en el mecanismo de reacciones químicas).

En la figura 1 se presenta la forma esquemática de generar un FEOM:

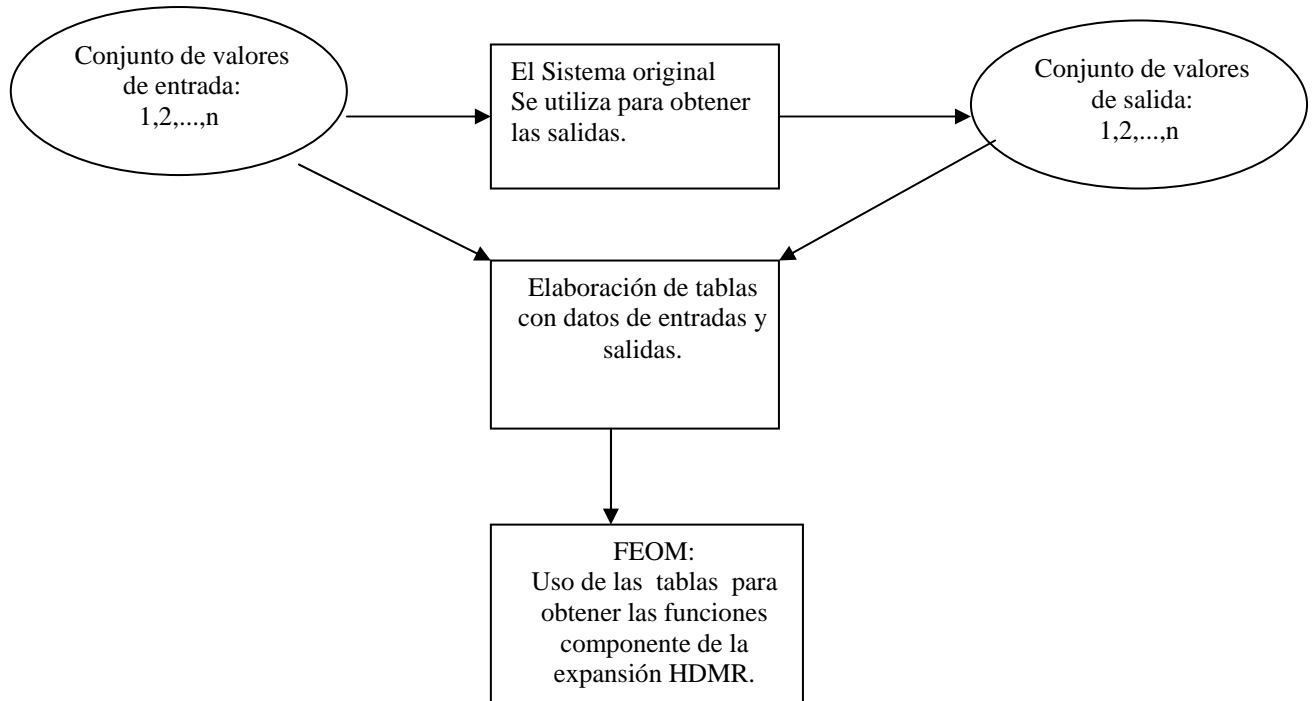


Figura 1 generación de un FEOM.

El modelo original se ejecuta un determinado número de veces para obtener la salida correspondiente a un conjunto de entradas seleccionadas de tal forma que cubran suficientemente el área de interés. Es decir, cuando se elabore la expansión HDMR es necesario que se cubra el espacio pertinente de entradas del modelo.

Con los datos del modelo original se obtienen tablas que contienen la relación de entradas y salidas del sistema para un cierto número de muestras de cada variable. Interpolando en estas tablas se obtienen las funciones componente $f_i(x_i)$, $f_{ij}(x_i, x_j)$, ... de la expansión HDMR que aproxima la salida del sistema. Dicha expansión es el FEOM de la parte del sistema que se modeló.

3.2. Aplicaciones.

La substitución de un integrador numérico para resolver un sistema de ecuaciones diferenciales, por su correspondiente FEOM, permite obtener resultados rápidos para valores grandes de la variable independiente.

El FEOM puede utilizarse solamente en sistemas *autónomos* de ecuaciones diferenciales.

Los sistemas *autónomos* de ecuaciones diferenciales son aquellos en los que la variable independiente no aparece en el lado derecho de las ecuaciones diferenciales. La palabra *autónomo* significa “autogobernante”. Un sistema autónomo es autogobernante porque evoluciona de acuerdo con ecuaciones diferenciales que están determinadas enteramente por el valor de las variables dependientes [Blanchard et al., 1999].

En sistemas autónomos sencillos, es decir, no *rígidos* (ver 4.2), en donde se usan métodos con espaciado constante, la ganancia al aplicar HDMR (FEOM) es muy

buena. En el caso de Runge-Kutta orden-4 el FEOM resultó ser alrededor de 600 veces más rápido que el integrador numérico utilizado. Los resultados obtenidos con el FEOM tuvieron un error porcentual del orden de 10^{-4} . En(14) se muestra el sistema de ecuaciones diferenciales que se utilizó para hacer los experimentos.

$$\frac{du}{dt} = \frac{dx_1}{dt} = -\frac{2}{25}u(t) + \frac{1}{50}v(t)$$

$$\frac{dv}{dt} = \frac{dx_2}{dt} = \frac{2}{25}u(t) - \frac{2}{25}v(t)$$

(14)

Para explicar como funciona un FEOM en la solución de ecuaciones diferenciales, supóngase que se tiene la siguiente ecuación diferencial:

$$\frac{dy}{dt} = f(y)$$

en donde la familia de *curvas solución* en función del tiempo, es la siguiente:

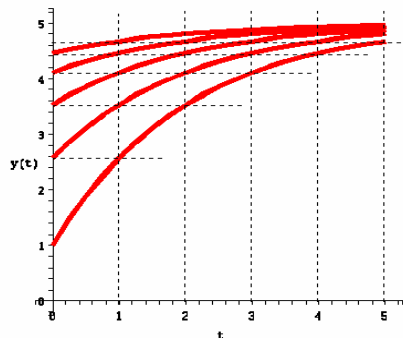


Figura 2: Familia de curvas solución.

Dado un conjunto de condiciones iniciales $\{y_0, y_0, \dots, y_0\}$ en el tiempo $t_0 = 0$, se obtiene, por medio de un integrador numérico, el estado final correspondiente a cada condición inicial $\{y_1, y_1, \dots, y_1\}$ en el tiempo $t_1 = t_0 + \Delta t$ y se construye una tabla en donde se relaciona cada condición inicial y_0 con su estado final correspondiente y_1 después del incremento de tiempo Δt . El rango de las condiciones iniciales $\{y_0, y_0, \dots, y_0\}$ debe cubrir apropiadamente el conjunto de situaciones iniciales que cubran el espacio pertinente de entrada del modelo.

El FEOM se obtiene con tablas construidas como se menciona en 2.1.1, de tal forma que contengan el estado del sistema cuando se pasa del tiempo inicial t_0 al tiempo t_1 partiendo de cualquier condición inicial dentro del espacio de interés. Con este FEOM es posible obtener resultados para períodos largos de tiempo $t_i = t_0 + i \Delta t$ ($i = 1, 2, \dots$). De hecho, si se usa el FEOM para la condición inicial $y = y_0$ en $t_0 = 0$, se obtiene el estado final en $t = t_1$: $y_1 = y(y_0, t_1)$ (ver figura 2), luego se utiliza y_1 como condición inicial en el

mismo FEOM para obtener el estado final en el tiempo $t = t_2$: $y_2 = y(y_1, t_1) = y(y_0, t_2)$. Generalizando, si el estado final $y(y_0, t_i)$ se utiliza en el FEOM como condición inicial, el siguiente estado final que se obtiene $y(y_i, t_1)$ es la aproximación a $y(y_0, t_{i+1})$. Este procedimiento brinda una reducción significativa en el esfuerzo computacional en comparación con los integradores numéricos convencionales.

3.3. Proceso de diseño del software para implantar un FEOM que resuelve un sistema de ecuaciones diferenciales usando las expansiones Cut-HDMR.

Se siguió el método de diseño de software en espiral para la elaboración de este sistema. Los detalles de este método pueden consultarse en [Gómez, 2004].

3.3.1 Análisis de requerimientos.

En el análisis de requerimientos se especifican las características que debe tener el sistema y las funciones con las que debe cumplir.

- El sistema sirve para aproximar la solución de un sistema de ecuaciones diferenciales, por medio de expansiones Cut-HDMR.
- El sistema de ecuaciones diferenciales debe especificarse desde un principio.
- Se debe contar con un integrador numérico que permita construir el FEOM.
- Debe elegirse un conjunto de diferentes condiciones iniciales que cubran suficientemente el área de interés $\{y_{0_1}, y_{0_2}, \dots, y_{0_k}\}$.
- Las posibles salidas deben quedar dentro del área que cubren las entradas seleccionadas.
- Incluir la métrica de los resultados que permita evaluar el desempeño del sistema.
- El sistema deberá ser lo más general posible.

3.3.2. Diseño de alto nivel.

El sistema consiste de dos fases; la fase de entrenamiento y la de operación. Adicionalmente se hizo un pequeño sistema para generar el dominio que sirve como entrada tanto a la fase de entrenamiento como a la de operación.

Fase de entrenamiento.

Se diseñó un sistema que, basándose en un archivo con los datos del dominio, elabora las tablas de las funciones $f_i(x_i), f_{ij}(x_i, x_j), f_{ijk}(x_i, x_j, x_k)$.

Puntos que se tomaron en cuenta para hacer el diseño:

- Determinación el tamaño del vector de entrada (x_1, x_2, \dots, x_n) .
- Determinación de cómo se almacenan las salidas del sistema (variables de estado).
- Interfase entre el sistema y el usuario.
- Solicitud de un archivo con el dominio Ω como entrada.

- El conjunto de condiciones iniciales $\{y_{0_1}, y_{0_2}, \dots, y_{0_k}\}$ (dominio) debe tener como subconjuntos a los conjuntos de valores posibles de salida de cada una de las variables de estado.
- El usuario debe establecer el punto de referencia \mathbf{x}_{ref} .
- Dado el conjunto de condiciones iniciales $\{y_{0_1}, y_{0_2}, \dots, y_{0_k}\}$ (dominio), el sistema construye un archivo para cada conjunto de funciones $f_i(x_i), f_{ij}(x_i, x_j), f_{ijk}(x_i, x_j, x_k)$, es decir, uno para las funciones con grado de cooperatividad $l=1$, otro para las de grado de cooperatividad $l=2$, y otro para las de $l=3$.
- El sistema solicita al usuario el paso de tiempo Δt con el que trabajará el FEOM.
- También solicita el tiempo final al que trabajará el integrador numérico.

Fase de operación.

Se diseñó otro sistema, independiente del de entrenamiento el cual, basándose en los archivos con las tablas de funciones Cut-HDMR, elabora un FEOM para hallar la solución de un sistema de ecuaciones diferenciales ordinarias.

Puntos que se tomaron en cuenta para hacer el diseño:

- Interfase para pedir al usuario un punto de entrada $\mathbf{x} = (x_1, x_2, \dots, x_n)$ y el grado de cooperatividad deseado l .
- La salida del sistema contiene el resultado del FEOM, el resultado obtenido con el integrador numérico (solver) y el error relativo porcentual.
- Diseño de las subrutinas de interpolación de dos y de tres dimensiones.
- Elaboración de un algoritmo de búsqueda para encontrar el intervalo en el que se va a interpolar.
- Determinar que método se utilizará para la interpolación en una dimensión, (polinomial o spline).
- Para aumentar la velocidad de acceso, bajar los datos de los archivos a tablas en la memoria RAM.
- Medir el tiempo de operación del FEOM.
- Solicitar el número de veces que se aplicará el FEOM para llegar a la aproximación de la solución al tiempo t .

3.3.3. Diseño detallado.

Fase de entrenamiento. (El sistema se llama “*trainer.cpp*”).

Determinación del dominio Ω .

Alojamiento en memoria del dominio Ω .

- Se usa un arreglo para guardar la cantidad de valores diferentes que toma cada variable de entrada:

```
int muestreo[n];
```

Estos datos se solicitan al usuario.

- El dominio se guarda en un arreglo con apuntadores a los valores que toma cada x_i :

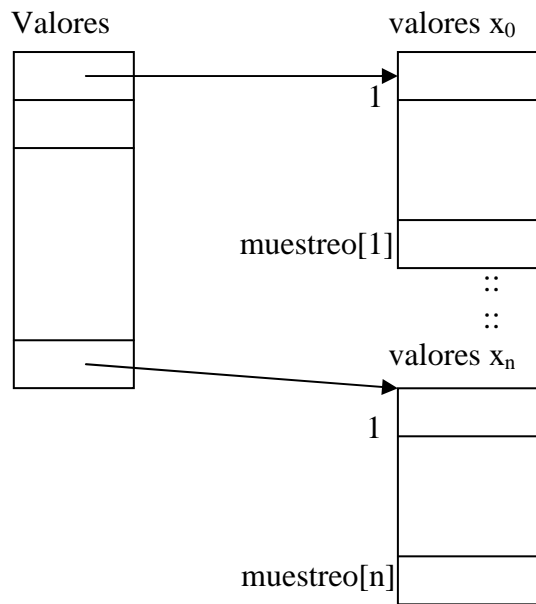


Figura 3: forma en que se guardan las muestras de cada variable.

El dominio Ω se guarda en un archivo de entrada que contiene el arreglo `muestreo[]` y la tabla con los valores que toma cada una de las variables.

Formato del archivo de entrada (*nombre_dominio.hdm*).

```

          Número de variables: NumVars
  ...   muestreo[NumVars]

   ....   valores[1][muestreo[1]]
   .....   valores[2][muestreo[2]]
  ::                               ::
   .....   valores[NumVars][muestreo[NumVars]]

```

Figura 4: formato del archivo de entrada.

Se hizo un sistema adicional llamado Domain para optimizar la creación de los archivos de entrada.

El sistema “Domain” tiene como entrada el número de variables, el número de muestras para cada variable, el valor inferior y el valor superior que toman las muestras de cada variable.

La salida de este sistema es un archivo *.hdm con el formato que se indicó para los archivos de entrada (figura 4).

A continuación se presenta la interfase del sistema “Domain” con el usuario:

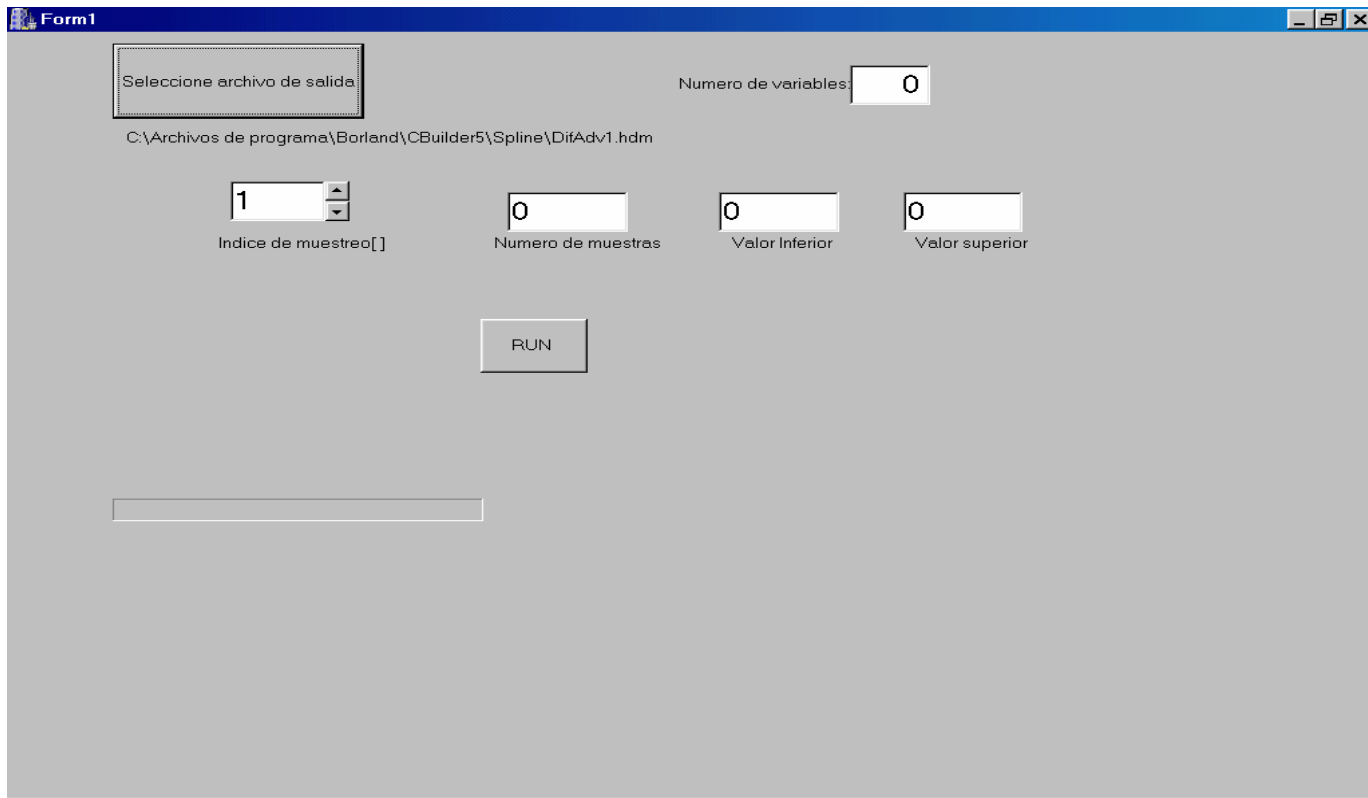


Figura 5: Interface del sistema “Domain” con el usuario.

Elaboración de las tablas Cut-HDMR:

Elaboración de las tablas $f_i(x_i)$:

Existe una expansión HDMR para cada variable de salida: t-HDMR, donde $t= 1,2,\dots,n$. Se forma un arreglo de apuntadores para cada una de las n expansiones HDMR. Cada apuntador de la expansión t-HDMR apunta a la $f_i(x_i)$ de la expansión t:

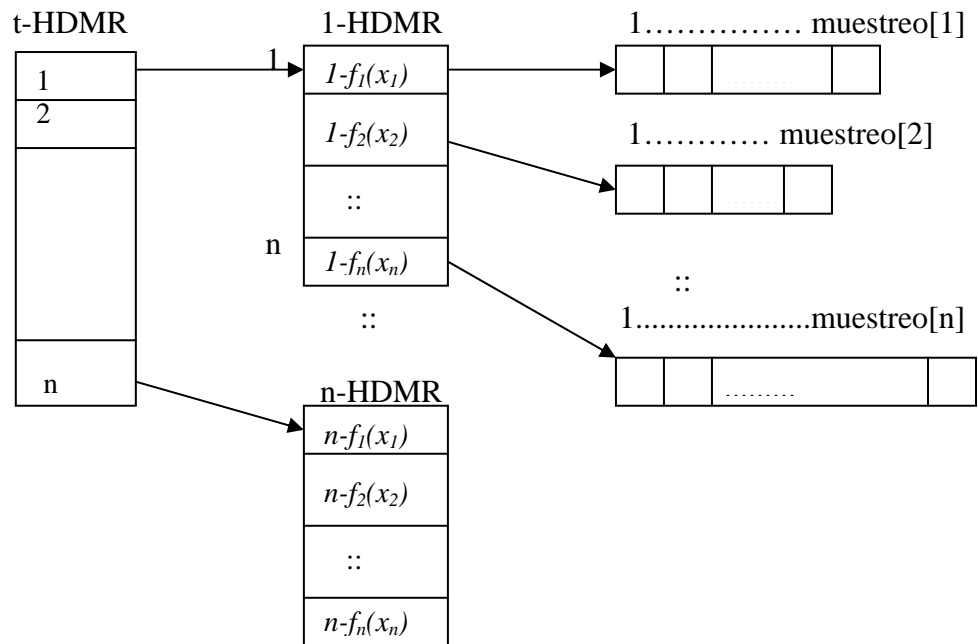


Figura 6: Formato de las tablas de grado de cooperatividad 1.

El nombre del archivo en el que se guardan las tablas con grado de cooperatividad $l=1$ es "nombre_f_i.tab".

Formato del archivo de salida (nombre_f_i.tab).

```

□ □ ..... □      f0[ n]
□ deltaT
Expansión 1-HDMR
1 2      muestreo[1]
□ □ ..... □      f_i[1][1][ ]
1 2      muestreo[2]
□ □ ..... □      f_i[1][2][ ]
::
1 2      muestreo[n]
□ □ ..... □      f_i[1][n][ ]
-----
Expansión n-HDMR
1 2      muestreo[1]
□ □ ..... □      f_i[n][1][ ]
1 2      muestreo[2]
□ □ ..... □      f_i[n][2][ ]
::
1 2      muestreo[n]
□ □ ..... □      f_i[n][n][ ]

```

Figura 7: formato del archivo de salida con las tablas de cooperatividad 1
Elaboración de las tablas $f_{ij}(x_i, x_j)$:

Si el grado de cooperatividad es $l=2$, entonces existen $\binom{n}{2} = \frac{n!}{(n-2)!2!} = \frac{n(n-1)}{2}$ tablas

$f_{ij}(x_i, x_j)$ las cuales se guardan de la siguiente manera:
 Existe una expansión HDMR para cada variable de salida: t-HDMR, donde $t=1,2..n$.

Se forma un arreglo de apuntadores para cada una de las n expansiones HDMR.
 Cada apuntador de la expansión t-HDMR apunta a las $f_{ij}(x_i, x_j)$ de la expansión t:

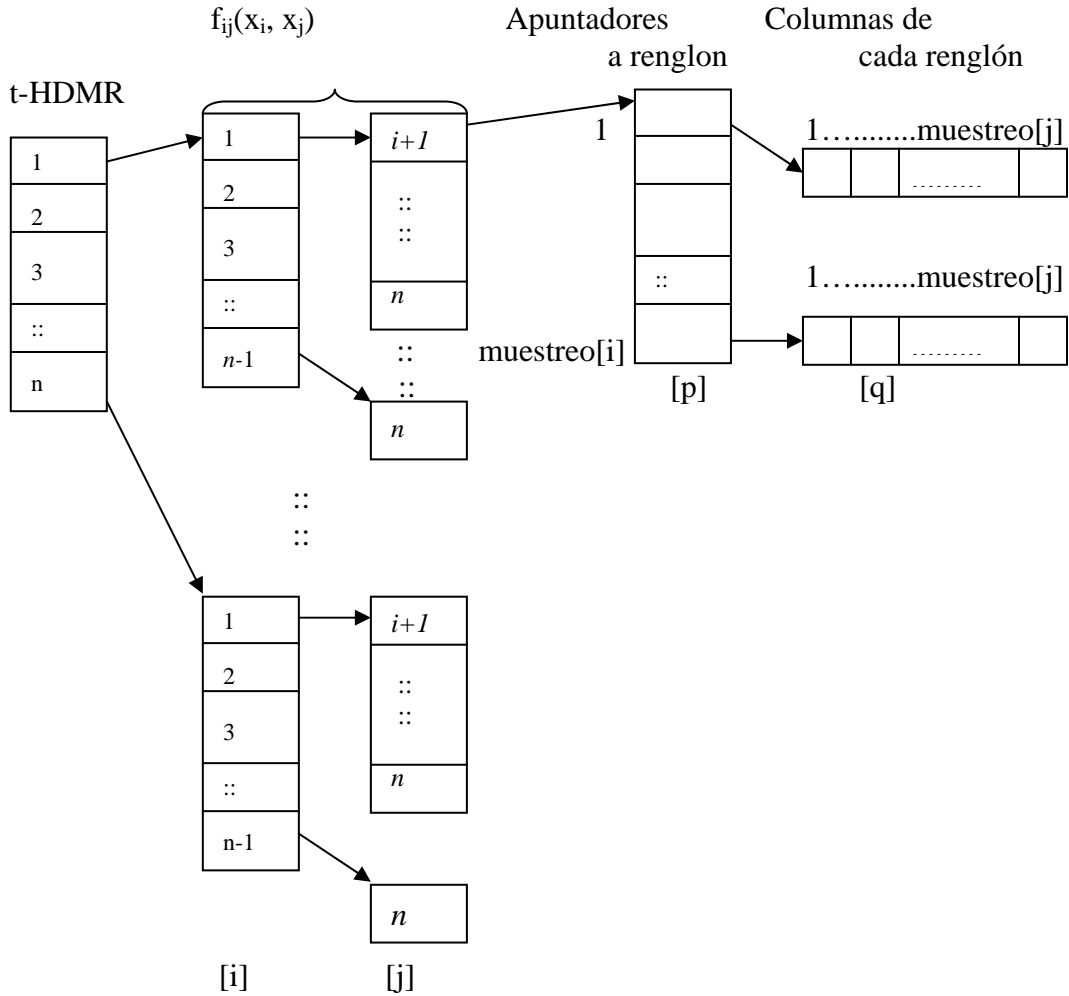


Figura 8: formato de las tablas de grado de cooperatividad 2.

De tal forma que para acceder a la tabla se usan los subíndices:

$$F[t][i][j]p[q]$$

- t para la expansión HDMR.
- i, j para la función $f_{ij}(x_i, x_j)$.
- p, q para el renglón y la columna respectivamente.

El nombre del archivo en el que se guardan las tablas con grado de cooperatividad $l=2$ es "nombre_f_i_j.tab".

Formato del archivo de salida (*nombre_f_i_j.tab*).

Expansión 1-HDMR

$$i = 1 \quad j = 2$$

```

1 2  muestreo[2]
□ □ ..... □  f_i_j[1][2][ ]      1
□ □ ..... □  f_i_j[1][2][ ]      ::
::
□ □ ..... □  f_i_j[1][2][ ]      muestreo[1]
-----

i = 1 j = n
1 2  muestreo[n]
□ □ ..... □  f_i_j[1][2][ ]      1
□ □ ..... □  f_i_j[1][2][ ]      ::
::
□ □ ..... □  f_i_j[1][2][ ]      muestreo[1]
-----
-----

Expansión n-HDMR
i = 1 j = 2
1 2  muestreo[j]
□ □ ..... □  f_i_j[i][j][ ]      1
□ □ ..... □  f_i_j[i][j][ ]      ::
::
□ □ ..... □  f_i_j[i][j][ ]      muestreo[i]
-----
-----

i = n-1 j = n
1 2  muestreo[n]
□ □ ..... □  f_i_j[i][j][ ]      1
□ □ ..... □  f_i_j[i][j][ ]      ::
::
□ □ ..... □  f_i_j[i][j][ ]      muestreo[i]

```

Figura 9: formato del archivo de salida con las tablas de cooperatividad 2.

Elaboración de las tablas $f_{ijk}(x_i, x_j, x_k)$.

Si el grado de cooperatividad es $l=3$, la estructura de los datos es la siguiente:

Se forma un arreglo de apuntadores para cada una de las n expansiones HDMR.

Cada apuntador de la expansión t-HDMR apunta a las $f_{ijk}(x_i, x_j, x_k)$ de la expansión t:

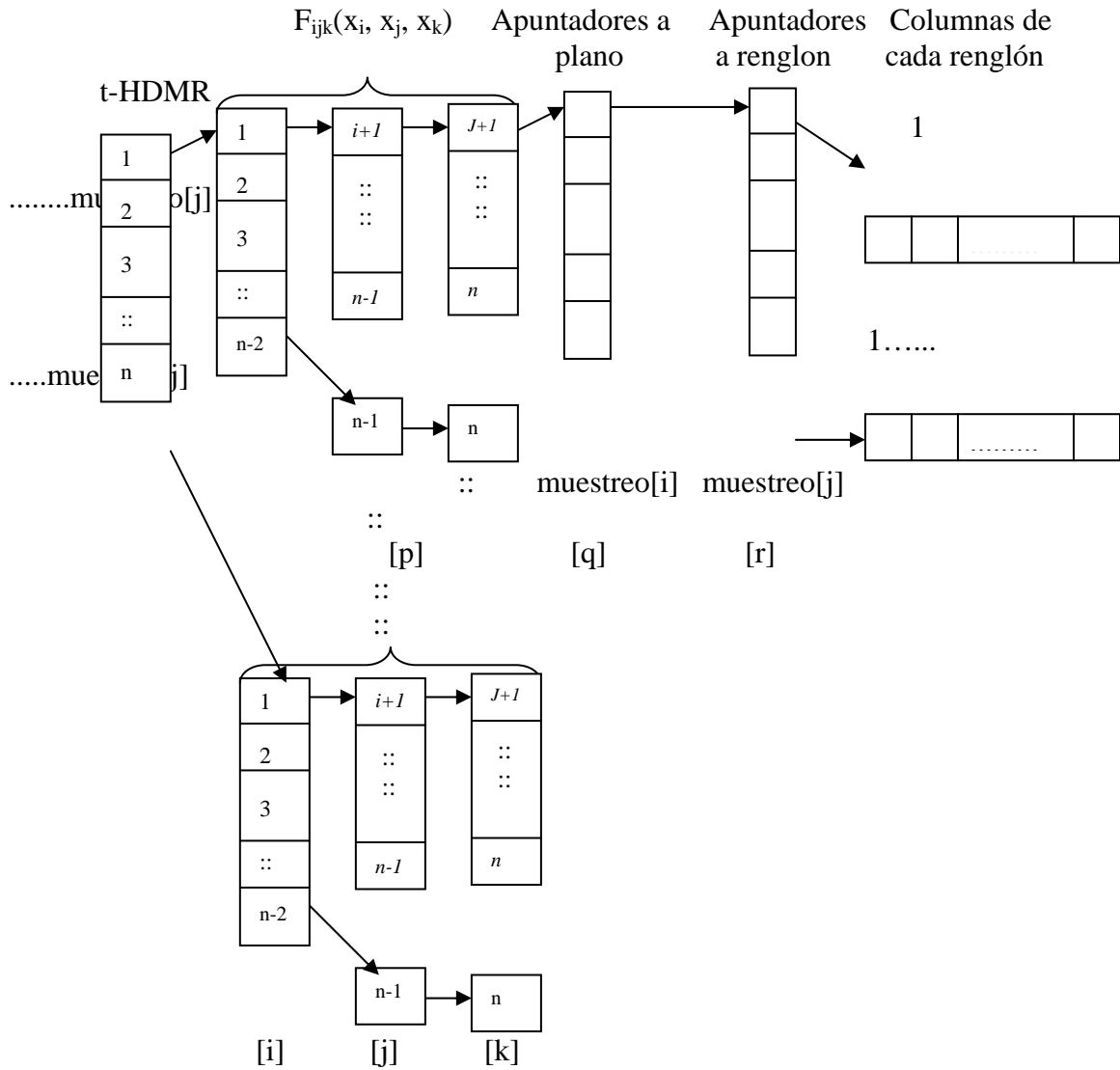


Figura 10: formato de las tablas de grado de cooperatividad 3.

De tal forma que para acceder a la tabla se usan los subíndices:

$$F[t][i][j][k][p][q][r]$$

t para el arreglo de variables de estado.

i, j, k para la función $f_{ijk}(x_i, x_j, x_k)$.

p, q, r para el plano, el renglón y la columna respectivamente.

El nombre del archivo en el que se guardan las tablas con grado de cooperatividad $l=3$ es "nombre_f_i_j_k.tab".

Formato del archivo de salida (nombre_f_i_j_k.tab).

Expansión 1-HDMR

i = 1 j = 2 k = 3

```

    1  2  muestreo[3]
    □ □ ..... □  f_i_j_j[1][2][3][ ]          1
    □ □ ..... □  f_i_j_k[1][2][3][ ]          ::
    ::
    □ □ ..... □  f_i_j_k[1][2][3][ ]          muestreo[2]
(1) -----

```

i = 1 j = 2 k = n

```

    1  2  muestreo[n]
    □ □ ..... □  f_i_j_k[1][2][n][ ]          1
    □ □ ..... □  f_i_j_k[1][2][n][ ]          ::
    ::
    □ □ ..... □  f_i_j_k[1][2][n][ ]          muestreo[2]
(2) -----

```

i = 1 j = n-1 k = n

```

    1  2  muestreo[n]
    □ □ ..... □  f_i_j_k[1][2][n][ ]          1
    □ □ ..... □  f_i_j_k[1][2][n][ ]          ::
    ::
    □ □ ..... □  f_i_j_k[1][2][n][ ]          muestreo[2]
(muestreo[1]) -----

```

Expansión n-HDMR

i = 1 j = 2 k = 3

```

    1  2  muestreo[3]
    □ □ ..... □  f_i_j_j[1][2][3][ ]          1
    □ □ ..... □  f_i_j_k[1][2][3][ ]          ::
    ::
    □ □ ..... □  f_i_j_k[1][2][3][ ]          muestreo[2]
(1) -----

```

i = 1 j = 2 k = n

```

    1  2  muestreo[n]
    □ □ ..... □  f_i_j_k[1][2][n][ ]          1
    □ □ ..... □  f_i_j_k[1][2][n][ ]          ::
    ::

```

```

(2)      □ □ ..... □      f_i_j_k[1][2][n][ ]      muestreo[1]
-----
-----
i = 1    j = n-1    k = n
1      2      muestreo[n]
□ □ ..... □      f_i_j_k[1][2][n][ ]      1
□ □ ..... □      f_i_j_k[1][2][n][ ]      ::
::
□ □ ..... □      f_i_j_k[1][2][n][ ]      muestreo[n-1]
(muestreo[1])
-----
-----
i = n-2  j = n-1  k = n
1      2      muestreo[n]
□ □ ..... □      f_i_j_k[1][2][n][ ]      1
□ □ ..... □      f_i_j_k[1][2][n][ ]      ::
::
□ □ ..... □      f_i_j_k[1][2][n][ ]      muestreo[n-1]
(muestreo[n-2])
===== f_{n-2,n-1,n}(x_{n-2}, x_{n-1}, x_n )

```

Figura 11: formato del archivo de salida con las tablas de cooperatividad 3.

A continuación se presenta la interfase del sistema entrenador con el usuario. Las derivadas solo se usan cuando se trabaja con la interpolación de spline cúbico.

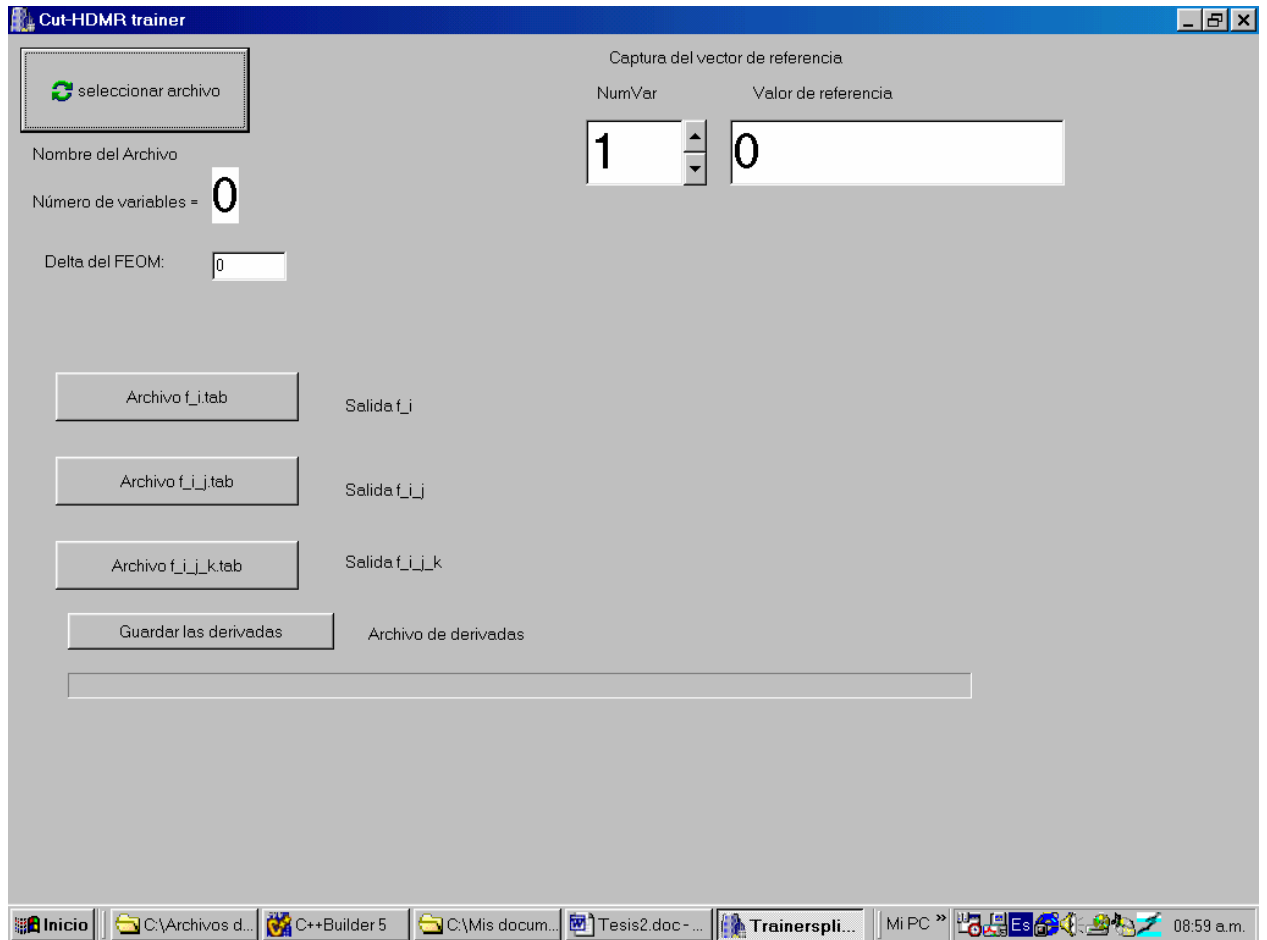


Figura 12: interface del sistema “Trainer” con el usuario.

Fase de operación. (El sistema se llama “*FEOM.cpp*”).

Determinación del dominio.- Se utiliza el mismo archivo de entrada que el de la fase de entrenamiento (*.hdm).

Tablas $f_i(x_i), f_{ij}(x_i, x_j), f_{ijk}(x_i, x_j, x_k)$.

Antes de comenzar la operación, es necesario leer las tablas de los archivos de salida de la fase de entrenamiento:

- *f_i.tab
- *f_i_j.tab
- *f_i_j_k.tab

De esta manera se bajan los datos de los archivos a tablas en la memoria RAM para aumentarla velocidad de acceso. Las tablas $f_i(x_i), f_{ij}(x_i, x_j), f_{ijk}(x_i, x_j, x_k)$ a usar están en función del grado de cooperatividad seleccionado.

A continuación se presenta la interfase del sistema FEOM con el usuario.

The screenshot shows a Windows application window titled "Form1". The interface includes several input fields and buttons:

- Cooperatividad:** A dropdown menu set to "1".
- Numero de Variables:** A text field containing "11".
- Seleccionar dominio:** A button with the path "C:\Archivos de programa\Borland\CBuilder5\Spline\DifAdv0-0.5.hdm".
- Cambiar Tablas f_i:** A button with the path "C:\Archivos de programa\Borland\CBuilder5\Spline\f_i_Dif_Adv5.tab".
- Cambiar tablas f_i_j:** A button with the path "c:\Archivos de programa\Borland\CBuilder5\Spline\Seinfeld_f_i_j.tab".
- Cambiar tablas f_i_j_k:** A button with the path "c:\Archivos de programa\Borland\CBuilder5\Spline\Seinfeld_f_i_j_k.tab".
- Estado inicial:** A dropdown menu set to "1" and a text field containing "0.00954".
- Numero de deltasT (FEOM):** A text field containing "30".
- Numero de pasos (60seg):** A text field containing "30".
- Buttons:** "RUN Integrador", "RUN FEOM", and "Archivo de derivadas".
- Output fields:** "Estado final: 0.171363614472225", "Tiempo empleado integrador: 0: 0.71", "Estado final: 0.171304501444446", "Tiempo empleado (FEOM): 0: 0.17", and "Error relativo: 6.87360788120817E-5".
- Index:** A dropdown menu set to "5".

The Windows taskbar at the bottom shows the Start button, several open applications (C:\Archivos d..., C++Builder 5, C:\Mis docum..., Tesis2.doc-..., Splinefeom...), and the system clock showing "09:16 a.m.".

Figura 13: interfase del sistema "FEOM" con el usuario.

Nota: El punto a evaluar que se introduzca en el sistema debe estar dentro del área que cubre el archivo de entrada *.htm.

Interpolación. Para interpolar en 2 y 3 dimensiones se utiliza el método de interpolación multidimensional descrito en el Apéndice C.

Para interpolar en una dimensión se utilizaron tres métodos para hacer experimentos:

- 1.- Un polinomio de tercer grado de Lagrange (ver Apéndice E).
- 2.- Interpolación spline cúbico (ver Apéndice F).
- 3.- Interpolación lineal; este tipo de interpolación dio los resultados más rápidos y por esto fue la elegida finalmente.

Algoritmo de búsqueda del intervalo de interpolación.

Se utilizó el método de bisección para encontrar el offset necesario para obtener los arreglos Xa y Ya con cuatro elementos que incluyan de forma adecuada la x a interpolar, el algoritmo que se utilizó es el siguiente:

```
// conseguir offset de los arreglos X y Y
```

```

while(a<b)
{
  m=(a+b)/2;
  if(X[m]>p)
    b=m;
  else
    a=m+1;
};
a=a-2;
if(a<0) a=0;
if(a>size-4) a=size-4;

// Llamar a un interpolador que construye un polinomio de grado "n".
polint(&X[a],&Y[a],n,p,&y_p);

```

3.3.4. Codificación.

El sistema está implantado en lenguaje C++. Se utilizó C++ Builder *versión 5*, como herramienta de desarrollo.

El código de las dos fases del sistema se encuentra en el compact disk incluido en esta tesis.

3.3.5. Pruebas.

Se realizaron dos tipos de pruebas, las de operación y las del comportamiento del FEOM.

Pruebas de operación:

Antes de utilizar los resultados finales de los sistemas *Trainer.cpp* y *FEOM.cpp*, se hicieron pruebas preeliminares para verificar que los resultados sean los correctos:

1.- Pruebas para comprobar que los datos de las $f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{ijk}(x_i, x_j, x_k)$ que se cargan y se leen en/de los archivos estén correctos.

2.- Pruebas para verificar que se están calculando bien las expansiones HDMR de diferente grado de cooperatividad.

Pruebas del comportamiento del FEOM:

Una vez corroborado que los sistemas *Trainer.cpp* y *FEOM.cpp*, funcionan correctamente, éstos se utilizaron para ajustar los parámetros del FEOM para el sistema de [Seinfeld, 1986] de 11 variables y una vez ajustados se utilizaron también para un sistema básico de contaminación ambiental con 4 sustancias y para otro con 27 sustancias. Se hicieron las siguientes pruebas.

1.- *Pruebas para determinar el Δt adecuado.* Se encontró, después de hacer pruebas con $\Delta t=40$ seg., $\Delta t=60$ seg., $\Delta t=120$ seg. y $\Delta t=600$ seg. que $\Delta t=60$ seg. es el óptimo para balancear rapidez con exactitud.

2.- *Pruebas variando el número de muestras del dominio.* Se determinó que los resultados son prácticamente los mismos para 20, 30, 50, 100 y 200 muestras de cada variable.

3.- *Pruebas para encontrar el dominio.* En este punto fue necesario observar el comportamiento de cada variable a través del tiempo y así encontrar el rango en el que se mueve cada una.

4.- *Pruebas para encontrar un buen punto de referencia.*

Para determinar en que tiempo se debe tomar el punto se llegó a la conclusión de que tomar los valores de un punto en $t = 0$ no es muy adecuado, ya que la mayoría de las concentraciones tanto en el sistema de 11 como en el de 27 variables, tienen un valor inicial de cero, mientras que tomar los valores cuando ya transcurrió un tiempo intermedio de $t = 30$ minutos, pone al punto de referencia más cerca de los valores a los que tienden cada una de las concentraciones.

Después de varios experimentos se observó que para medidas a 10 km de altura ($z=10$ km) el punto de referencia óptimo se encuentra en $x=30$ km, $y=35$ km, (la descripción de la malla donde trabaja el modelo de difusión/advección se encuentra en el capítulo 5).

5.- *Pruebas para determinar el grado de cooperatividad "l".* Aunque el grado de cooperatividad $l=2$ disminuyó ligeramente el error de los resultados, hizo que el FEOM fuera notablemente más lento que CVODE. Esto hizo inviable un grado de cooperatividad $l=2$ y se concluyó que es suficiente $l=1$.

Conclusiones del capítulo 3.

La implantación completa de la técnica Cut-HDMR para construir un FEOM consta de dos fases, la *fase de entrenamiento o aprendizaje* y la *fase de operación del modelo*.

A continuación se describe en que consiste cada una de las fases:

Fase 1.
Fase de entrenamiento o aprendizaje.

Elaboración de las tablas con las funciones componente:

- 1.- Por medio del sistema original (el que será se reemplazado por el FEOM) se obtienen las salidas correspondientes a un conjunto de diferentes situaciones iniciales (entradas). para un incremento de tiempo Δt .
- 2.- Se utilizan las ecuaciones con la estructura del Cut-HDMR (4), (5) y (6) para determinar las funciones tabulares: $f_i(x_i), f_{ij}(x_i, x_j), \dots$ se debe elegir apropiadamente el numero de muestras para cada variable.

Fase 2.
Fase de operación del modelo.

- 1.- Para un punto \mathbf{x} dado de entrada, evalúan las funciones componente, por interpolación.
- 2.- La salida aproximada del sistema para el punto de entrada dado, se determina evaluando la expansión HDMR (1) usando las funciones componente $f_i(x_i), f_{ij}(x_i, x_j), \dots$

Hay que notar que:

- 1.- La *fase de entrenamiento* se realiza una sola vez para obtener las tablas con las funciones componente; ya que se obtuvieron las tablas, se ejecuta solamente la *fase de operación* cada vez que se requiera obtener el estado final a partir de un punto \mathbf{x} después de cierto tiempo t .

- 2.- El uso del FEOM para simular sistemas de ecuaciones diferenciales solo es válido en *sistemas autónomos*, es decir, en sistemas en donde la variable independiente no aparece explícitamente en ninguna ecuación.

CAPÍTULO 4.

Aplicación del FEOM al modelo en 3 dimensiones de difusión/advección con reacciones químicas.

4.1. El modelo en 3 dimensiones de difusión/advección con reacciones químicas.

El modelo que utilizamos en esta tesis para el análisis del ozono, en ausencia de deposición de impurezas, se describe con el siguiente sistema de ecuaciones diferenciales parciales [Nagornov *et al.*, 2003]:

$$\frac{\partial c_i}{\partial t} + \frac{\partial(uc_i)}{\partial x} + \frac{\partial(vc_i)}{\partial y} + \frac{\partial(wc_i)}{\partial z} = \frac{\partial}{\partial x} \left(K_x \frac{\partial c_i}{\partial x} \right) + \frac{\partial}{\partial y} \left(K_y \frac{\partial c_i}{\partial y} \right) + \frac{\partial}{\partial z} \left(K_z \frac{\partial c_i}{\partial z} \right) + R_i$$

(14)

($i = 1, 2, \dots, n$)

donde c_i representan las concentraciones de contaminantes y R_i son las tasas de cambio en las concentraciones de contaminantes en la mezcla reactiva debido a las transformaciones químicas. Los problemas de valor inicial del sistema (14) se resuelven numéricamente para un conjunto de n especies químicas, en un dominio Ω tridimensional ($0 < x < L_x, 0 < y < L_y, 0 < z < L_z$).

Los mecanismos de reacciones químicas R_i que están dentro de este modelo de contaminación atmosférica se describen por medio de sistemas rígidos de Ecuaciones Diferenciales Ordinarias (EDO):

$$\frac{dc_i}{dt} = R_i(c_1, c_2, \dots, c_n), \quad i = 1, \dots, n$$

(15)

La solución de estos *sistemas rígidos* (ver sección 4.2) de EDO (15) es computacionalmente intensiva.

Los *sistemas rígidos* de ecuaciones diferenciales que surgen de los mecanismos de reacciones químicas pueden ser de 4,11,30,70,100 o mas variables, dependiendo del modelo utilizado. Los métodos de GEAR [Press *et al.*, 1996], que se usan tradicionalmente para resolver el Problema de Valor Inicial (PVI) son sofisticados debido a la rigidez de este tipo de sistemas e implican una gran carga computacional.

En esta tesis se hicieron experimentos con tres mecanismos diferentes de reacciones químicas; el modelo básico de contaminación ambiental con 4 sustancias (17), el modelo de Seinfeld de 11 sustancias (16) y el modelo de Seinfeld de 27 sustancias [León, 1998]. Sin embargo, para poder comparar la velocidad del modelo de difusión/advección de contaminantes usando CVODE y usando el FEOM, se tomó como referencia el modelo de Seinfeld de 11 sustancias, ya que en una tesis previa

[Aguilar, 1998] ya se tenía implantado el modelo de difusión/advección de contaminantes usando CVODE en la solución del mecanismo de reacciones químicas de 11 sustancias.

4.2. Sistemas rígidos.

Los sistemas rígidos son sistemas modelados por medio de ecuaciones diferenciales que contienen términos cuyos coeficientes son muy diferentes entre sí. La diferencia grande en magnitud entre los términos de una ecuación diferencial implica que existen cambios bruscos y suaves en el comportamiento del sistema.

Los métodos numéricos tradicionales (*explícitos*) para resolver ecuaciones diferenciales no son apropiados para resolver sistemas rígidos ya que un paso constante “grande” no detecta los cambios bruscos, no es posible reproducir el sistema en forma correcta y se genera el *problema de estabilidad* [Borelli et al., 2002]. Para poder detectar estos cambios, el espaciamiento tendría que ser extremadamente pequeño, lo que vuelve al método inoperante por su lentitud.

Los métodos *implícitos* (métodos de Gear [Press et al., 1999]), se idearon para resolver los sistemas rígidos, con éstos es posible obtener una buena aproximación a la solución con un espaciamiento tal que permita resolver el problema en un tiempo razonable.

La desventaja de los métodos implícitos es que para despejar a y_{n+1} en la mayoría de los casos es necesario resolver una ecuación no lineal.

Método explícito:

$$\frac{dy}{dt} = f(t_n, y_n)$$

Método implícito:

$$\frac{dy}{dt} = f(t_{n+1}, y_{n+1})$$

4.3. Modelo propuesto por Seinfeld (1986) como mecanismo de reacciones químicas.

A continuación se presenta el mecanismo de reacciones químicas de Seinfeld de 11 sustancias, que es el que tomamos en esta tesis como modelo para evaluar la velocidad del FEOM en el modelo en 3 dimensiones de difusión/advección de contaminantes.

$$\begin{aligned}
\frac{dc_1}{dt} &= k_1c_2 - k_3c_1c_3 - k_7c_1c_{10} - k_8c_1c_7 - k_9c_1c_9 \\
\frac{dc_2}{dt} &= -k_1c_2 + k_3c_3c_1 + k_7c_1c_{10} + k_8c_1c_7 + k_9c_1c_9 - k_{10}c_2c_6 - k_{11}c_2c_9 + k_{12}c_{11} \\
\frac{dc_3}{dt} &= k_2c_4 * C_{-02} * C_{-M} - k_3c_3c_1 \\
\frac{dc_4}{dt} &= k_1c_2 - k_2c_4 * C_{-02} * C_{-M} \\
\frac{dc_5}{dt} &= -k_4c_5c_6 \\
\frac{dc_6}{dt} &= -k_4c_5c_6 - k_5c_6c_8 + k_7c_1c_{10} - k_{10}c_2c_6 \\
\frac{dc_7}{dt} &= k_4c_5c_6 + k_6c_8 - k_8c_1c_7 + k_9c_1c_9 \\
\frac{dc_8}{dt} &= -k_5c_6c_8 - k_6c_8 + k_8c_1c_7 \\
\frac{dc_9}{dt} &= k_5c_6c_8 - k_9c_1c_9 - k_{11}c_2c_9 + k_{12}c_{11} \\
\frac{dc_{10}}{dt} &= k_6c_8 - k_7c_1c_{10} + k_8c_1c_7 \\
\frac{dc_{11}}{dt} &= k_{11}c_2c_9 - k_{12}c_{11}
\end{aligned} \tag{16}$$

Cada concentración representa a un compuesto químico:

$c_1 = \text{NO}$	(Monóxido de Nitrógeno).
$c_2 = \text{NO}_2$	(Dióxido de Nitrógeno).
$c_3 = \text{O}_3$	(Ozono).
$c_4 = \text{O}$	(Oxígeno).
$c_5 = \text{RH}$	(Hidrocarburos).
$c_6 = \text{OH}$	(Radical Hidroxilo).
$c_7 = \text{RO}_2$	(Radical alquiperóxido).
$c_8 = \text{RCHO}$	(Compuestos carbonílicos).
$c_9 = \text{RC[O]O}_2$	(Radical paroxiacilo).
$c_{10} = \text{HO}_2$	(Radical hidroperóxido).
$c_{11} = \text{RC[O]O}_2\text{NO}_2$	(Nitratos de peróxiacilos).

Los coeficientes de la cinética química son los siguientes[Seinfeld, 1986; León F., 1998; Nagornov et al., 2003]:

$$\begin{aligned}
k_1 &= 0.533/60.0 = 0.00888333 && \text{(depende del ángulo del sol)} \\
k_2 &= 2.183\text{e-}5/60.0 = 2.45 \times 10^{-4} \\
k_3 &= 26.59/60.0 = 0.4481666 \\
k_4 &= 3.775\text{e}3/60.0 = 1.26371503 \\
k_5 &= 2.341\text{e}4/60.0 = 2.13023782 \\
k_6 &= 1.91\text{e-}4/60.0 = 5.83 \times 10^{-4} && \text{(depende del ángulo del sol)} \\
k_7 &= 1.214\text{e}4/60.0 = 1.1047025 \\
k_8 &= 1.127\text{e}4/60.0 = 1.0255352 \\
k_9 &= 1.127\text{e}4/60.0 = 1.0255352
\end{aligned}$$

$$k_{10}=1.613e4/60.0 = 1.4677802$$

$$k_{11}=6.893e3/60.0 = 2.3074934$$

$$k_{12}=2.143e-2/60.0 = 0.0048337$$

La concentración de un *cuerpo*:

$$c_M = [M]=79.0 \times 10^4 \text{ ppm} \quad (M= N_2)$$

$$c_{O_2} = [O_2] = 21.0 \times 10^4 \text{ ppm}$$

4.4. Substitución de CVODE por el FEOM como método de solución del sistema de ecuaciones diferenciales del modelo de reacciones químicas.

Los sistemas de EDO que corresponden a reacciones químicas, en particular los sistemas (16), (17) y el de Seinfeld de 27 sustancias [León, 1998], se resuelven numéricamente usando paquetes de programación especialmente desarrollados para sistemas rígidos. En esta tesis hemos utilizado el reconocido paquete CVODE [Aguilar, 1998; León F., 1998; Nagornov et al., 2003].

Lo que se hizo en esta investigación fue reemplazar el paquete CVODE por su correspondiente FEOM para reducir el tiempo de ejecución del modelo en 3 dimensiones de difusión/advección de contaminantes. Una vez elaborado el sistema de cómputo que entrena y opera al FEOM, fue necesario elegir los parámetros a utilizar y un método de interpolación. Después de comparar resultados con interpolación lineal, Lagrange de orden 2. Lagrange de orden 3 y spline cúbico se eligió interpolación lineal por ser un método más rápido, sin pérdida de exactitud, además se determinó experimentalmente que un grado de cooperatividad $l = 1$ en las expansiones HDMR es suficiente, los detalles se presentan en 4.5.

4.5. Elección de los parámetros del FEOM.

Para que el FEOM funcione adecuadamente es necesario definir los siguientes parámetros.

- 1.- Grado de cooperatividad de las expansiones HDMR con las que se trabajará.
- 2.- El número de muestras de cada variable de entrada: (m_1, m_2, \dots, m_n)
- 3.- El rango en el que se mueve cada una de las variables del sistema c_i .
- 4.- Condiciones iniciales.
- 5.- El punto de referencia: $(c_{1_{ref}}, c_{2_{ref}}, \dots, c_{n_{ref}})$
- 6.- El incremento de tiempo Δt para entrenar al FEOM.

4.5.1. Elección del grado de cooperatividad.

El grado de cooperatividad $l = 2$ se utiliza cuando con la cooperatividad $l = 1$ no se alcanza la exactitud requerida. En los experimentos realizados la cooperatividad $l = 2$ dio como resultado un desempeño del FEOM mucho más lento que el de CVODE sin mejorar notablemente los resultados obtenidos con $l = 1$.

Cuando el número de variables de entrada sea lo suficientemente grande, un FEOM con $l=2$ puede llegar a ser más rápido que el sistema original que se piensa reemplazar. Utilizar cooperatividad $l = 2$ es viable si la contribución al resultado de dos variables actuando juntas es importante.

A continuación, en la Tabla 6 se muestran los errores relativos utilizando un grado de cooperatividad $l = 1$ y en la Tabla 7 los errores relativos utilizando cooperatividad $l = 2$. Puede observarse que el error disminuye muy poco con cooperatividad $l = 2$, ya que de por sí, con cooperatividad $l = 1$ se obtienen muy buenos resultados.

Definimos el error obtenido, como la diferencia entre el resultado del FEOM con el de CVODE.

	30 min	60 min	90 min	120 min	150 min
c_1	0.074%	0.770%	1.463%	3.090%	4.633%
c_2	0.017%	0.168%	0.320%	0.678%	1.015%
c_3	0.005%	0.009%	0.054%	0.176%	0.412%
c_4	0.018%	0.171%	0.326%	0.691%	1.035%
c_5	0.001%	0.001%	0.004%	0.014%	0.031%
c_6	0.211%	0.931%	1.899%	3.219%	3.653%
c_7	0.036%	0.149%	0.389%	0.698%	0.911%
c_8	0.006%	0.018%	0.040%	0.077%	0.133%
c_9	0.004%	0.019%	0.048%	0.091%	0.156%
c_{10}	0.030%	0.122%	0.342%	0.609%	0.752%
c_{11}	0.004%	0.012%	0.067%	0.175%	0.352%

Tabla 6: Errores relativos con cooperatividad $l = 1$ (20 muestras, $\Delta t = 30$ seg., interpolación lineal).

	30 min	60 min	90 min	120 min	150 min
c_1	0.996%	1.108%	0.754%	0.942%	0.784%
c_2	0.223%	0.249%	0.172%	0.217%	0.187%
c_3	0.015%	0.005%	0.013%	0.009%	0.013%
c_4	0.227%	0.254%	0.176%	0.221%	0.190%
c_5	0.002%	0.004%	0.005%	0.006%	0.006%
c_6	0.332%	0.582%	0.724%	1.186%	6.237%
c_7	0.015%	0.078%	0.081%	0.093%	0.802%
c_8	0.004%	0.009%	0.015%	0.022%	0.027%
c_9	0.002%	0.004%	0.002%	0.001%	0.026%
c_{10}	0.007%	0.145%	0.033%	0.430%	1.433%
c_{11}	0.013%	0.023%	0.026%	0.029%	0.044%

Tabla 7: Errores relativos con cooperatividad $l=2$ (20 muestras, $\Delta t = 30$ seg., interpolación lineal).

4.5.2. Número de muestras.

Una de las ventajas de las técnicas HDMR es que éstas son útiles incluso cuando hay escasez en los datos, debido a que las interpolaciones se hacen sobre curvas suaves y los resultados son prácticamente independientes del número de muestras. Para verificar esta característica se hicieron experimentos con 20, 30, 50, 100 y 200 muestras y como los resultados son muy parecidos se llegó a la conclusión de que en nuestro caso particular de 11 variables, con 20 muestras se obtienen buenos resultados (ver Tablas 8 y 9). Los resultados con 20 muestras están en la Tabla 6.

	30 min	60 min	90 min	120 min	150 min
c_1	0.462%	0.779%	1.767%	3.420%	5.041%
c_2	0.102%	0.170%	0.388%	0.752%	1.106%
c_3	0.025%	0.012%	0.041%	0.161%	0.394%
c_4	0.103%	0.174%	0.395%	0.766%	1.127%
c_5	0.002%	0.001%	0.003%	0.013%	0.031%
c_6	0.179%	0.790%	1.789%	2.062%	5.154%
c_7	0.019%	0.158%	0.384%	0.512%	1.224%
c_8	0.006%	0.017%	0.039%	0.077%	0.133%
c_9	0.005%	0.020%	0.049%	0.097%	0.148%
c_{10}	0.009%	0.139%	0.346%	0.417%	1.137%
c_{11}	0.004%	0.012%	0.067%	0.176%	0.352%

Tabla 8: Errores relativos usando 200 muestras.

	30 min	60 min	90 min	120 min	150 min
c_1	0.522%	0.768%	1.599%	3.222%	4.697%
c_2	0.115%	0.168%	0.351%	0.708%	1.030%
c_3	0.026%	0.011%	0.049%	0.170%	0.409%
c_4	0.117%	0.171%	0.357%	0.721%	1.049%
c_5	0.002%	0.001%	0.003%	0.014%	0.031%
c_6	0.227%	0.837%	1.923%	0.497%	5.108%
c_7	0.012%	0.151%	0.389%	0.324%	1.207%
c_8	0.006%	0.018%	0.040%	0.077%	0.133%
c_9	0.004%	0.020%	0.048%	0.105%	0.148%
c_{10}	0.008%	0.125%	0.340%	0.118%	1.125%
c_{11}	0.004%	0.013%	0.068%	0.176%	0.353%

Tabla 9: Errores relativos usando 30 muestras.

El número de muestras no debe ser muy alto, depende de la cantidad de variables de entrada. Según [Shorter et al., 1999], desde el punto de vista realista, se espera que el número de muestras de cada variable esté entre 10 y 20.

Cuando el comportamiento de las curvas del sistema (funciones componente) es suave, con un bajo número de muestras se pueden obtener buenos resultados. Mientras mas bruscos sean los cambios, se requerirá de un mayor número de muestras para mejorar la interpolación.

4.5.3. Rango de las variables.

La elección del conjunto de valores de entrada requiere un minucioso análisis, ya que es necesario examinar el rango en el que se mueve cada una de las variables para tiempos cortos y largos.

El rango de cada concentración es un factor clave en el buen funcionamiento del FEOM, ya que los sistemas rígidos presentan problemas de inestabilidad. De hecho, en un sistema rígido de ecuaciones diferenciales con tan solo dos variables, si el rango no está razonablemente elegido pueden presentarse puntos singulares. Por ejemplo, dada una cierta EDO de primer orden de dos variables, es posible que $\frac{dy}{dx}$ esté indefinida para ciertas condiciones iniciales. A

esto se le llama *punto singular* y

entonces, si los rangos de x y de y incluyen al punto singular, es posible que la integración numérica lleve al sistema hacia este punto.

A continuación se muestran los rangos con los que trabajan las 11 concentraciones del modelo de Seinfeld (encontrados de manera experimental).

```
Concentracion[1]=(0,0.18);
Concentracion[2]=(0,0.12);
Concentracion[3]=(0,0.4);
Concentracion[4]=(0,2e-8);
Concentracion[5]=(0,0.86);
Concentracion[6]=(0,1e-6);
Concentracion[7]=(0,1.5e-5);
Concentracion[8]=(0,0.84);
Concentracion[9]=(0,8e-6);
Concentracion[10]=(0,1.5e-5);
Concentracion[11]=(0,0.03);
```

El comportamiento de los sistemas rígidos de ecuaciones diferenciales, puede ser muy irregular cuando se opera en regiones que están fuera del rango de las variables, este comportamiento irregular ocasiona que no se pueda llevar a cabo una buena interpolación y que por lo tanto el FEOM no tenga un buen funcionamiento. En suma, es muy importante que los valores de entrada estén dentro del rango en el que el sistema rígido de ecuaciones diferenciales tiene un comportamiento adecuado.

4.5.4. Condiciones Iniciales.

Utilizamos como condiciones iniciales las concentraciones de sustancias químicas con máxima presencia de luz en una atmósfera contaminada según los datos proporcionados por los químicos especialistas [Aguilar, 1998; León 1998].

```
Concentracion[1]= 0.00954;
Concentracion[2]= 0.042;
Concentracion[3]= 0.1;
Concentracion[4]= 0;
Concentracion[5]= 0.17192;
```

Concentracion[6]= 0;
Concentracion[7]= 0;
Concentracion[8]= 0.17192;
Concentracion[9]= 0;
Concentracion[10]= 0;
Concentracion[11]= 0;

4.5.5. Punto de referencia.

Se observó experimentalmente que el punto de referencia no puede ser construido con valores arbitrarios dentro del rango de cada variable, sino que *debe ser un punto real*. La estrategia para elegir este punto debe basarse en encontrar un punto medio de $f(\mathbf{x})$ para evitar que los valores extremos queden lejos del punto de referencia.

Un FEOM trabaja apropiadamente solamente en un área cercana al punto de referencia. En nuestro estudio de casos observamos que no es conveniente utilizar la condición inicial c_{0_2} como punto de referencia ya que la mayoría de las concentraciones se encuentran en cero. Sin embargo, si se deja correr el sistema hasta la mitad del tiempo del intervalo de trabajo, obtendremos unos valores intermedios entre las concentraciones iniciales y las finales, que es precisamente la característica que se requiere para un buen punto de referencia.

La elección entre FEOM y un integrador numérico convencional (CVODE en nuestro estudio) es un compromiso entre rapidez de cálculos y amplio espacio de operación. El FEOM es más rápido que CVODE, sin embargo su desempeño correcto está limitado a un área del espacio multidimensional de las concentraciones relativamente cerca del punto de referencia. Para hacer mas amplio el espacio de operación de un FEOM se puede utilizar la técnica Multi-Cut HDMR [Li et al., 2004] la cual está descrita en 2.4.1 de esta tesis. Sin embargo, cabe hacer notar que el uso de Multi-Cut HDMR implica más esfuerzo en el entrenamiento del FEOM, porque implica la construcción de varios FEOM's.

4.5.6. Paso de tiempo Δt .

No existe un criterio específico para determinar el paso de tiempo Δt con el que debe ser entrenado un FEOM, el valor Δt debe encontrarse experimentalmente de manera que sea lo suficientemente grande como para que la velocidad sea mayor que la del integrador numérico, sin embargo no debe ser tan grande que se pierda exactitud en el resultado. En este trabajo se encontró experimentalmente que un $\Delta t = 60$ seg. proporciona una buena velocidad del FEOM y exactitud. En los resultados en las Tablas 10, 11 y 12 se puede observar como aumenta considerablemente el error cuando el paso de tiempo Δt es grande.

Los resultados con $\Delta t = 60$ seg. están en la Tabla 6.

	30 min	60 min	90 min	120 min	150 min
c_1	0.216%	0.578%	1.234%	2.580%	4.248%
c_2	0.051%	0.122%	0.265%	0.539%	0.890%
c_3	0.002%	0.001%	0.062%	0.186%	0.396%
c_4	0.052%	0.125%	0.270%	0.550%	0.907%
c_5	0.004%	0.006%	0.004%	0.003%	0.017%
c_6	0.241%	0.835%	1.940%	6.768%	0.211%
c_7	0.032%	0.148%	0.388%	5.740%	2.494%
c_8	0.000%	0.004%	0.020%	0.046%	0.091%
c_9	0.004%	0.019%	0.046%	0.102%	0.172%
c_{10}	0.030%	0.134%	0.362%	6.806%	3.297%
c_{11}	0.003%	0.013%	0.066%	0.146%	0.302%

Tabla 10. Errores relativos usando $\Delta t = 40$ s.

	30 min	60 min	90 min	120 min	150 min
c_1	0.224%	1.029%	1.917%	3.727%	5.583%
c_2	0.044%	0.216%	0.405%	0.797%	1.197%
c_3	0.010%	0.004%	0.063%	0.189%	0.425%
c_4	0.045%	0.220%	0.413%	0.813%	1.220%
c_5	0.008%	0.013%	0.015%	0.010%	0.001%
c_6	0.202%	0.724%	1.796%	3.248%	4.986%
c_7	0.020%	0.160%	0.367%	0.724%	1.187%
c_8	0.012%	0.031%	0.060%	0.104%	0.168%
c_9	0.005%	0.021%	0.049%	0.091%	0.150%
c_{10}	0.006%	0.137%	0.319%	0.652%	1.048%
c_{11}	0.001%	0.018%	0.075%	0.189%	0.371%

Tabla 11. Errores relativos usando $\Delta t = 120$ s..

	30 min	60 min	90 min	120 min	150 min
c_1	3.647%	6.882%	9.752%	12.17%	14.16%
c_2	1.366%	2.799%	4.294%	5.868%	7.519%
c_3	0.868%	1.959%	3.218%	4.617%	6.124%
c_4	1.392%	2.852%	4.375%	5.979%	7.661%
c_5	0.109%	0.234%	0.372%	0.521%	0.677%
c_6	2.719%	4.882%	6.502%	7.639%	8.313%
c_7	0.658%	1.407%	2.267%	3.254%	4.368%
c_8	0.347%	0.659%	0.947%	1.216%	1.472%
c_9	0.079%	0.167%	0.264%	0.373%	0.495%
c_{10}	0.613%	1.323%	2.148%	3.104%	4.186%
c_{11}	2.612%	5.123%	7.565%	9.958%	12.32%

Tabla 12. Errores relativos usando $\Delta t = 600$ s.

4.6. Elección del método de interpolación.

Otra de las decisiones importantes para el buen funcionamiento de un FEOM es la elección del método de interpolación. En este trabajo se hicieron experimentos para comparar 2 métodos de interpolación :

1.- La interpolación de Lagrange usando el algoritmo de Neville (ver Apéndice E). Se hicieron experimentos para interpolar con polinomios de grados 2 y 3, sin embargo los mejores resultados se obtuvieron usando interpolación lineal, ya que además de ser más rápida proporciona resultados bastante aceptables. Los resultados con interpolación lineal y con los demás parámetros encontrados como los óptimos están en la Tabla 6.

2.- La interpolación “spline cubico” (ver Apéndice F). Spline cúbico sirve para interpolar en una dimensión, es bastante exacto sin embargo más lento que la interpolación lineal. Las matrices de segundas derivadas que utiliza este método deben generarse en la fase de entrenamiento. Si se requiere un grado de cooperatividad $l = 2$ este método ya no es conveniente, ya que habría que encontrar las matrices de las segundas derivadas para interpolar en dos o mas dimensiones y esto es demasiado lento.

	30 min	60 min	90 min	120 min	150 min
c_1	0.213%	0.842%	1.657%	3.215%	4.743%
c_2	0.046%	0.184%	0.363%	0.706%	1.040%
c_3	0.017%	0.015%	0.043%	0.167%	0.402%
c_4	0.047%	0.188%	0.370%	0.719%	1.059%
c_5	0.002%	0.001%	0.003%	0.013%	0.031%
c_6	0.153%	0.956%	1.780%	3.090%	5.935%
c_7	0.043%	0.155%	0.375%	0.662%	1.553%
c_8	0.006%	0.017%	0.039%	0.077%	0.133%
c_9	0.005%	0.019%	0.049%	0.091%	0.143%
c_{10}	0.038%	0.126%	0.329%	0.575%	1.349%
c_{11}	0.004%	0.012%	0.066%	0.174%	0.351%

Tabla 13. Interpolación hecha con spline cúbico.
(20 muestras, $\Delta t = 30$ seg, coop. $l = 1$).

En la Tabla 13 se observan los resultados de interpolar con spline cúbico, los demás parámetros son los mismos que los de la Tabla 6. Puede verse que los errores relativos mostrados en la Tabla 13 son muy similares a los de la Tabla 6.

4.7. Procedimiento detallado para integrar un sistema de EDO's por medio de un FEOM.

A continuación se describen cada una de las etapas que es necesario seguir para la construcción de un FEOM usando expansiones HDMR de grado de cooperatividad $l=1$. Se utiliza $l=1$ porque se encontró experimentalmente que, para nuestro estudio de casos, la cooperatividad $l=2$ prácticamente no mejora los resultados pero sí hace al FEOM considerablemente más lento que CVODE.

Etapa 1.

Determinar los siguientes parámetros.

- 1.- Grado de cooperatividad de las expansiones HDMR con las que se trabajará.
- 2.- El número de muestras de cada variable de entrada: (m_1, m_2, \dots, m_n)
- 3.- El rango en el que se mueve cada una de las variables del sistema c_i .
- 4.- Condiciones iniciales.
- 5.- El punto de referencia: $(c_{1_{ref}}, c_{2_{ref}}, \dots, c_{n_{ref}})$
- 6.- El incremento de tiempo Δt para entrenar al FEOM.

Etapa 2.

Fase de entrenamiento (construcción de las tablas).

2a).- Obtener el vector \mathbf{f}_0 integrando con el integrador numérico, en nuestro caso CVODE, con el punto de referencia para un Δt .

$$\mathbf{f}_0 = (f_0^{(1)}, f_0^{(2)}, \dots, f_0^{(n)})$$

2b).- Obtener la tabla de *contribuciones de cada variable* a la concentración C_1 .

Expansión HDMR_1 : Tablas de la contribución de todas las variables a

C_1 .

Tabla 1: Contribución de c_1 a C_1 .

Valores de C_1	Punto a integrar : $f(c_1, \bar{c}^1)$	$f_1 = f(c_1, \bar{c}^1) - f_0^{(1)}$
c_{1_1}	$f(c_{1_1}, \bar{c}^1) = \text{CVODE}(c_{1_1}, c_{2_{ref}}, \dots, c_{n_{ref}})$	$f_{1_1} = f(c_{1_1}, \bar{c}^1) - f_0^{(1)}$
c_{1_2}	$f(c_{1_2}, \bar{c}^1) = \text{CVODE}(c_{1_2}, c_{2_{ref}}, \dots, c_{n_{ref}})$	$f_{1_2} = f(c_{1_2}, \bar{c}^1) - f_0^{(1)}$
	:: ::	
$c_{1_{m_1}}$	$f(c_{1_{m_1}}, \bar{c}^1) = \text{CVODE}(c_{1_{m_1}}, c_{2_{ref}}, \dots, c_{n_{ref}})$	
$f_{1_{m_1}} = f(c_{1_{m_1}}, \bar{c}^1) - f_0^{(1)}$		
	:: ::	

Tabla n: Contribución de c_n a c_1 .

Valores de C_1	Punto a integrar : $f(c_n, \bar{c}^n)$	$f_n = f(c_n, \bar{c}^n) - f_0^{(n)}$
c_{1_1}	$f(c_{1_1}, \bar{c}^n) = \text{CVODE}(c_{1_{ref}}, c_{2_{ref}}, \dots, c_{n_1})$	$f_{n_1} = f(c_{n_1}, \bar{c}^n) - f_0^{(n)}$
c_{1_2}	$f(c_{1_2}, \bar{c}^n) = \text{CVODE}(c_{1_{ref}}, c_{2_{ref}}, \dots, c_{n_2})$	$f_{n_2} = f(c_{n_2}, \bar{c}^n) - f_0^{(n)}$
	::	
	::	
$c_{1_{m_1}}$	$f(c_{1_{m_1}}, \bar{c}^n) = \text{CVODE}(c_{1_{ref}}, c_{2_{ref}}, \dots, c_{n_{m_1}})$	
$f_{n_{m_1}} = f(c_{n_{m_1}}, \bar{c}^n) - f_0^{(n)}$		

2c).- Repetir el paso 2b) $n-1$ veces mas para encontrar la tabla de contribuciones de cada variable a la concentración c_1 , $i = 2, 3, \dots, n$. Cada conjunto de n tablas de contribuciones a c_1 forman el grupo para HDMR_i.

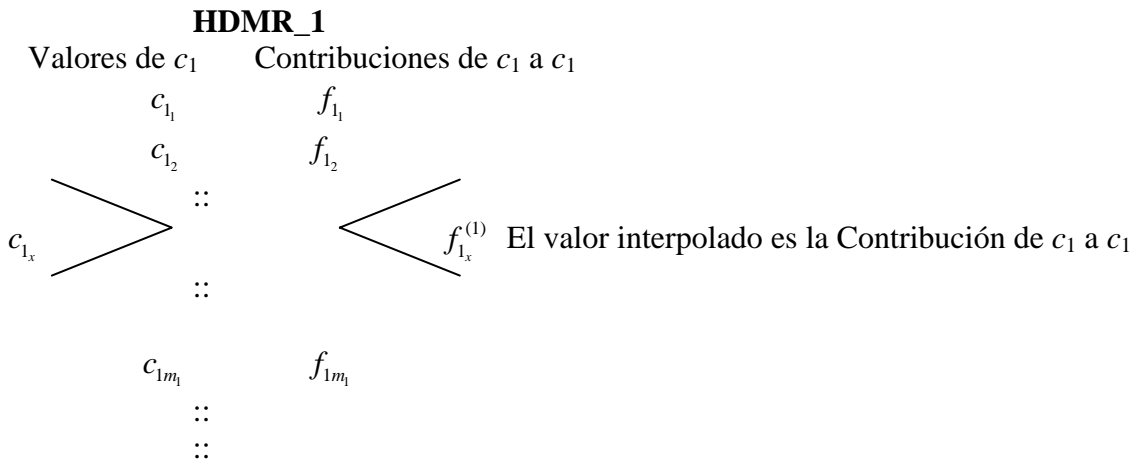
En realidad ya no es necesario integrar con CVODE, se utiliza el vector resultado de la integración en cada uno de los puntos de las tablas del HDMR₁ y se va restando la $f_0^{(i)}$ correspondiente para formar los demás grupos de tablas HDMR₂, HDMR₃, ..., HDMR_n.

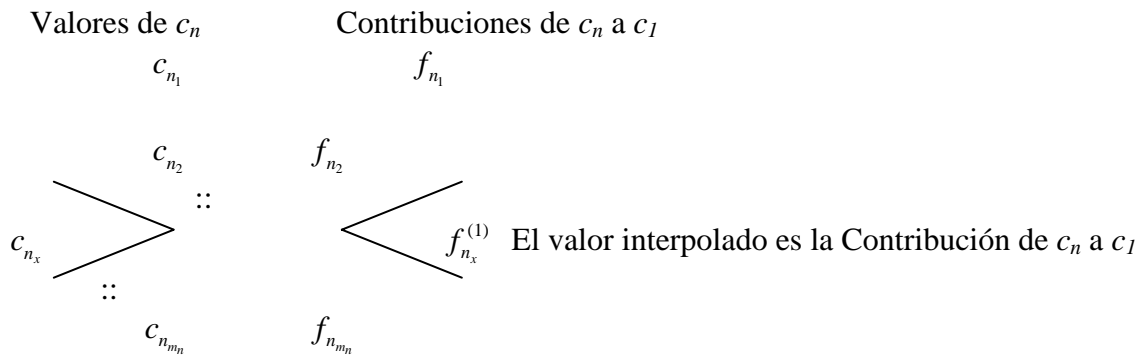
Etapa 3.

Operación del FEOM para un paso de tiempo Δt .

Se tiene una condición inicial $(c_{1_x}, c_{2_x}, \dots, c_{n_x})$ para la cual se desea obtener su estado final después de un tiempo $t = \Delta t$.

3a).- Para obtener el estado final de la concentración c_1 después de $t = \Delta t$ se obtiene la contribución a c_1 de cada una de las concentraciones interpolando c_x en las tablas HDMR₁.





3b).- El estado final de c_l después de $t = \Delta t$ se aproxima con la expansión $HDMR_1$ ($l = 1$):

$$\begin{aligned}
 c_1 \approx HDMR_1 &= f_0^{(1)} + f_{1_x}^{(1)} + f_{2_x}^{(1)} + \dots + f_{n_x}^{(1)} \\
 &= f_0^{(1)} + \sum_{i=1}^n f_{i_x}^{(1)}
 \end{aligned}$$

3c).- Se repiten los pasos 3a) y 3b) para obtener los estados finales de c_2, c_3, \dots, c_n :

$$\begin{aligned}
 c_2 \approx HDMR_2 &= f_0^{(2)} + f_{1_x}^{(2)} + f_{2_x}^{(2)} + \dots + f_{n_x}^{(2)} \\
 &= f_0^{(2)} + \sum_{i=1}^n f_{i_x}^{(2)} \\
 &\vdots \\
 &\vdots \\
 c_n \approx HDMR_n &= f_0^{(n)} + f_{1_x}^{(n)} + f_{2_x}^{(n)} + \dots + f_{n_x}^{(n)} \\
 &= f_0^{(n)} + \sum_{i=1}^n f_{i_x}^{(n)}
 \end{aligned}$$

Así, después de esta etapa se tiene un estado final \mathbf{c} .

Etapa 4.

Iteraciones de la fase de operación.

El estado final \mathbf{c} después de $t = \Delta t$ obtenido en la etapa 3 se usa como condición inicial y se ejecuta una vez mas la etapa 3.

El estado final que se obtiene es para $t = 2\Delta t$, así sucesivamente hasta obtener el estado final para $t = x\Delta t$ donde x es el número deseado de iteraciones del FEOM.

El ciclo puede apreciarse en la figura 14.

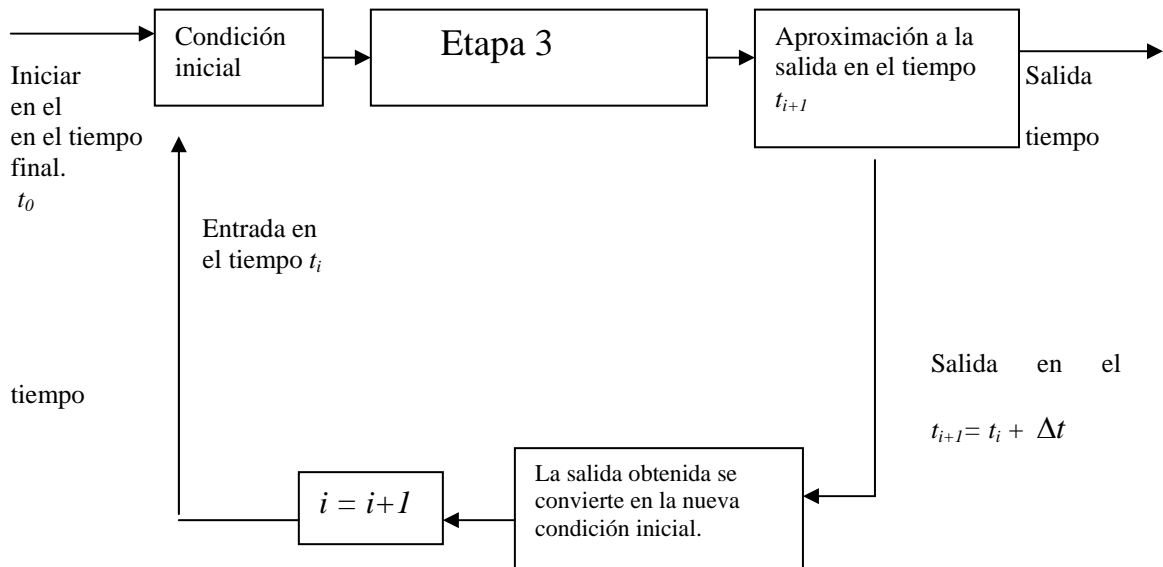


Figura 14: ciclo de operación de un FEOM.

Cabe hacer notar que los tiempos en los que se pueden obtener los estados finales del sistema deben ser múltiplos de Δt .

4.8. Resultados obtenidos con el modelo de Seinfeld de 11 sustancias.

Para poder comparar los resultados obtenidos con CVODE y con el FEOM se utilizó el mismo paso de tiempo $\Delta t = 60$ seg en los experimentos.

Como los coeficientes k_1 y k_6 del modelo de Seinfeld dependen del ángulo del sol [Nagornov et al., 2003] y por esta razón dependen explícitamente del tiempo, dividimos el problema en intervalos de 1 hora en donde podemos asumir que estos coeficientes son constantes, de esta manera obtenemos sistemas autónomos. A continuación se muestran los resultados obtenidos al comparar el desempeño de un FEOM con CVODE cada 30 minutos durante un período de 2.5 horas.

Si las funciones componente forman curvas suaves, las predicciones del FEOM serán muy exactas incluso para períodos muy largos de tiempo, sin embargo, en algunos casos, el comportamiento de estas curvas es brusco y entonces las expansiones HDMR no pueden aproximar correctamente el estado final del sistema en las regiones donde las curvas pierden la suavidad.

Existe un método llamado Factorized High Dimensional Model Representation (FHDMR) [Tunga & Demiralp, 2006] que pretende corregir este problema. En nuestros estudios de caso, el sistema presenta singularidades en las curvas de las funciones componente de algunas concentraciones. Estas singularidades introducen un error que es significativo solo para tiempos mas largos que 2.5 horas.

En la Tabla 6, donde se utilizó interpolación lineal, grado de cooperatividad $l=1$, paso de tiempo $\Delta t = 60$ segundos y 20 muestras para cada concentración, puede apreciarse que los errores son muy bajos para la primera hora y media.

	30 min	60 min	90 min	120 min	150 min
c_1	0.522%	0.768%	1.599%	3.222%	4.697%
c_2	0.115%	0.168%	0.351%	0.708%	1.030%
c_3	0.026%	0.011%	0.049%	0.170%	0.409%
c_4	0.117%	0.171%	0.357%	0.721%	1.049%
c_5	0.002%	0.001%	0.003%	0.014%	0.031%
c_6	0.227%	0.837%	1.923%	0.497%	5.108%
c_7	0.012%	0.151%	0.389%	0.324%	1.207%
c_8	0.006%	0.018%	0.040%	0.077%	0.133%
c_9	0.004%	0.020%	0.048%	0.105%	0.148%
c_{10}	0.008%	0.125%	0.340%	0.118%	1.125%
c_{11}	0.004%	0.013%	0.068%	0.176%	0.353%

Tabla 6: Errores relativos de los resultados obtenidos con un FEOM.

4.9. Otros estudios de caso.

4.9.1. Resultados obtenidos con el modelo básico de cuatro sustancias.

El sistema rígido de ecuaciones diferenciales (17) es el modelo de reacciones químicas básico de contaminación ambiental y esta compuesto por cuatro sustancias .

$$\begin{aligned}
 \frac{dc_1}{dt} &= k_1c_2 - k_3c_1c_3 \\
 \frac{dc_2}{dt} &= -k_1c_2 + k_3c_3c_1 \\
 \frac{dc_3}{dt} &= k_2c_4 * C_{-02} * C_{-M} - k_3c_3c_1 \\
 \frac{dc_4}{dt} &= k_1c_2 - k_2c_4 * C_{-02} * C_{-M}
 \end{aligned} \tag{17}$$

Los coeficientes de la cinética química son los siguientes [Seinfeld, 1986; León F., 1998]:

$$\begin{aligned}
 k_1 &= 0.533/60.0 = 0.00888333 && \text{(depende del ángulo del sol)} \\
 k_2 &= 2.183e-5/60.0 = 2.45 \times 10^{-4} \\
 k_3 &= 26.59/60.0 = 0.4481666 \\
 k_4 &= 3.775e3/60.0 = 1.26371503 \\
 k_5 &= 2.341e4/60.0 = 2.13023782
 \end{aligned}$$

Las condiciones iniciales proporcionadas por los químicos [León, 1998] para realizar los experimentos fueron:

$$\begin{aligned}
 \text{Concentración}[1] &= 0.00954 \\
 \text{Concentración}[2] &= 0.042 \\
 \text{Concentración}[3] &= 0.1 \\
 \text{Concentración}[4] &= 0
 \end{aligned}$$

Al igual que para el modelo de 11 y 27 sustancias, se utilizó interpolación lineal, grado de cooperatividad $l=1$, paso de tiempo $\Delta t = 60$ segundos y 20 muestras para cada concentración. Usando el criterio de 4.5.5 se fijó el punto de referencia en el valor de las concentraciones iniciales después de 30 segundos.

En la Tabla 14 se pueden ver los resultados del FEOM y los de CVODE para un tiempo de 1.5 horas. Puede observarse que el error relativo es pequeño.

Concentración	FEOM	CVODE	Error
c_1	0.006378	0.0063328	0.713%
c_2	0.04515	0.04520	0.005%
c_3	0.09685	0.09679	0.061%
c_4	4.6726e-9	4.6780e-9	0.115%

Tabla 14: resultados obtenidos con un FEOM y con CVODE Después de 1.5 horas con un paso de tiempo $\Delta t = 60$ seg.

Los resultados con el FEOM se obtuvieron 11 veces mas rápido que con CVODE.

4.9.2. Resultados obtenidos con el modelo de 27 sustancias.

También se hicieron experimentos con el modelo de 27 sustancias propuesto por Seinfeld y proporcionado por [León, 1998] para un intervalo de tiempo: $30 \text{ min} \leq t \leq 90 \text{ min}$, los parámetros del FEOM elaborado se eligieron con los mismos criterios utilizados para el mecanismo de 11 sustancias (16), los errores obtenidos fueron menores al 3.2% (la mayoría de los resultados menores al 1%), y la velocidad de este FEOM fue alrededor de 3 veces mayor que CVODE.

Conclusiones del capítulo 4.

Los mecanismos de reacciones químicas dentro de los modelos de contaminación ambiental se describen por medio de sistemas rígidos de ecuaciones diferenciales ordinarias en los que intervienen muchas variables. Estos sistemas se resuelven, hasta la fecha, mediante métodos numéricos convencionales. CVODE es un paquete diseñado especialmente para resolver este tipo de sistemas por medio de los métodos computacionales de GEAR.

Para reemplazar CVODE por su correspondiente FEOM se analizó experimentalmente como afecta cada uno de los parámetros que intervienen en la construcción del FEOM, el tipo de interpolación más conveniente y el grado de cooperatividad necesario, en nuestros estudios de caso obtuvimos como parámetros óptimos; interpolación lineal, grado de cooperatividad $l = 1$, paso de tiempo $\Delta t = 60$ segundos y 20 muestras para cada concentración.

Se construyeron FEOMs que resuelven eficientemente los sistemas rígidos de ecuaciones diferenciales ordinarias de los modelos de 11 y 27 sustancias propuesto por Seinfeld [Seinfeld, 1986; León, 1998] y del modelo básico de contaminación ambiental de 4 sustancias [León, 1998].

De acuerdo con los resultados obtenidos en este capítulo, para nuestro estudio de casos, una interpolación lineal y un grado de cooperatividad $l = 1$ en las expansiones HDMR del FEOM proporcionan resultados entre 2.5 y 11 veces más rápidos en comparación con los resultados obtenidos con CVODE. Además, se obtuvo una buena exactitud con los FEOMs construidos (ver Tablas 6 y 14).

CAPÍTULO 5.

Aplicación del FEOM en el modelo de difusión/advección de contaminantes.

5.1 Solución de sistema de ecuaciones diferenciales parciales no lineales planteado para las ecuaciones de difusión-advección de contaminantes.

A continuación se describe como se resuelve el sistema de ecuaciones diferenciales parciales no lineales planteado para la calidad del aire en la ciudad de México [Aguilar, 1998].

Para encontrar una solución particular al sistema de ecuaciones diferenciales parciales no lineales acopladas (14), es necesario establecer las condiciones iniciales y las condiciones de frontera.

$$(14) \quad \frac{\partial c_i}{\partial t} + \frac{\partial(uc_i)}{\partial x} + \frac{\partial(vc_i)}{\partial y} + \frac{\partial(wc_i)}{\partial z} = \frac{\partial}{\partial x} \left(K_x \frac{\partial c_i}{\partial x} \right) + \frac{\partial}{\partial y} \left(K_y \frac{\partial c_i}{\partial y} \right) + \frac{\partial}{\partial z} \left(K_z \frac{\partial c_i}{\partial z} \right) + R_i$$

$(i = 1, 2, \dots, n)$

Si $c(x, y, z, t)$ es la concentración de la especie i , en un punto del dominio Ω en el tiempo t , entonces las condiciones iniciales están dadas por:

$$c(x, y, z, 0) = \Phi(x, y, z), \quad (x, y, z) \in \Omega$$

y las condiciones a la frontera están dadas como sigue:

En las fronteras $x=0$ y $x=L_x$,

- El flujo de entrada:

$$uc_i - K_x \frac{\partial c_i}{\partial x} = uc_{in}$$

- El flujo de salida:

$$-K_x \frac{\partial c_i}{\partial x} = 0$$

donde c_{in} son las concentraciones en la frontera.

Las condiciones en las fronteras $y=0$ y $y=L_y$,

- El flujo de entrada:

$$vc_i - K_y \frac{\partial c_i}{\partial y} = vc_{in}$$

- El flujo de salida:

$$-K_y \frac{\partial c_i}{\partial y} = 0$$

Las condiciones en las fronteras $z=0$ y $z=L_z$ son,

- Para $z=0$:

$$Q_i = -K_z \frac{\partial c_i}{\partial z}$$

donde Q_i es el flujo del contaminante i (emisiones en superficie).

- Para $z=L_z$ suponemos flujo nulo, entonces:

$$-K_z \frac{\partial c_i}{\partial z} = 0$$

Los datos de las mediciones de concentraciones iniciales se limitan a valores de medición en la superficie. Por lo general, estos se encuentran esparcidos irregularmente en el dominio. Bajo estas condiciones, es necesario obtener un campo de concentraciones iniciales representativo. Esto se logra utilizando técnicas de interpolación [Calvo, 2002].

En la técnica de solución numérica del problema de difusión/advección en tres dimensiones, las ecuaciones diferenciales se reducen a una secuencia de ecuaciones en una dimensión, a esto se le llama técnica de separación de operadores [Yanenko, 1971]:

$$L_1 : \quad \frac{\partial c_i}{\partial t} = K_x \frac{\partial^2 c_i}{\partial x^2} - \frac{\partial(uc_i)}{\partial x} \quad (18)$$

$$L_2 : \quad \frac{\partial c_i}{\partial t} = K_y \frac{\partial^2 c_i}{\partial y^2} - \frac{\partial(vc_i)}{\partial y} \quad (19)$$

$$L_3 : \quad \frac{\partial c_i}{\partial t} = K_z \frac{\partial^2 c_i}{\partial z^2} - \frac{\partial(wc_i)}{\partial z} \quad (20)$$

La ecuación de difusión en la dimensión x (18) se resuelve con las condiciones iniciales y a la frontera $x=0$ y $x=L_x$. Las concentraciones calculadas para cada nodo $(x,y,z) \in \Omega$, serán las concentraciones iniciales c_{0_y} para la ecuación de difusión en la dimensión y (19). Las concentraciones calculadas al resolver la ecuación de difusión en la dimensión y , serán las concentraciones iniciales c_{0_z} para la ecuación de difusión en la dimensión z (20).

Las concentraciones calculadas en este último paso serán las condiciones iniciales c_{0_R} para el mecanismo de reacciones químicas utilizado en (14); estas concentraciones serán a su vez la concentraciones iniciales para el siguiente paso de tiempo.

La forma de resolver el sistema de ecuaciones diferenciales parciales por medio de un método numérico consiste en reemplazar las derivadas parciales por sus aproximaciones en diferencias finitas, convirtiendo el sistema de ecuaciones diferenciales en un sistema de ecuaciones algebraicas, los detalles pueden verse en [Aguilar,1998].

En nuestra aplicación del FEOM al modelo de difusión/advección de contaminantes, se utilizó el modelo de 11 sustancias propuesto por Seinfeld (16) para resolver la parte de las reacciones químicas. La razón por la que se eligió este modelo de 11 sustancias es que en trabajos previos [Aguilar, 1998] ya se tenía este modelo resuelto con CVODE. De esta manera se pudo comparar la velocidad del FEOM con la que ya se tenía con CVODE.

Los modelos matemáticos con los que se trabajó en esta tesis son idealizados y no incluyen todas las variables que utilizan los investigadores de la calidad del aire. Utilizamos el modelo matemático idealizado para analizar la parte de las reacciones químicas resueltas con FEOM y con CVODE sin tener que incluir datos que no afectan en nuestra investigación.

5.2.- Malla donde se trabaja el modelo de difusión/advección de contaminantes.

El modelo descrito por (14) se trabaja en una malla de 14 puntos sobre el eje de las x (longitud), 16 puntos sobre el eje de las y (latitud) y 5 puntos sobre el eje de las z (altura), sobre esta malla se forman $14*16*5= 1120$ nodos. La distancia entre cada punto de los ejes x , y y z es de 5 km.

A nivel de la troposfera, el ozono se produce por la reacción fotoquímica de óxidos de nitrógeno (NO_x) y compuestos orgánicos volátiles los cuales se denominan precursores del ozono (O_3) [SIMAT, 2006], por lo tanto, las mediciones relevantes en el modelo de Seinfeld de 11 sustancias son; el monóxido de nitrógeno (NO), el dióxido de nitrógeno (NO_2) y la producción de ozono (O_3). En nuestros experimentos mostramos los valores finales (después de una hora) de las concentraciones: $c_1=\text{NO}$, $c_2=\text{NO}_2$ y $c_3=\text{O}_3$. La altura elegida para medir estas concentraciones fué de $z=10$ km, ya que observamos que los valores de c_1 , c_2 y c_3 en los cinco niveles de z dentro de la malla, casi no varían cuando cambia el nivel. De esta manera, podemos decir que el nivel $z=10$ km es un nivel intermedio y proporciona resultados representativos.

5.3. Intervalo de trabajo y elección del punto de referencia del FEOM.

Intervalo de trabajo: Para obtener un sistema *autónomo* (ver 3.2), es necesario considerar constantes los valores de k_1 y k_6 , una hora es un lapso de tiempo en el que estos coeficientes casi no varían, por esto se eligió una hora como intervalo de trabajo del FEOM.

Los experimentos con el modelo en 3 dimensiones de difusión/advección de contaminantes se realizaron para el intervalo de tiempo de una hora.

Muestreando los valores de cada concentración para diferentes tiempos dentro del intervalo de trabajo (una hora) en diferentes puntos de la malla, se determinó que los rangos de las concentraciones son los siguientes:

Concentracion[1]=(0,0.18);
Concentracion[2]=(0,0.12);
Concentracion[3]=(0,0.4);
Concentracion[4]=(0,2e-8);
Concentracion[5]=(0,0.86);
Concentracion[6]=(0,1e-6);
Concentracion[7]=(0,1.5e-5);
Concentracion[8]=(0,0.84);
Concentracion[9]=(0,8e-6);
Concentracion[10]=(0,1.5e-5);
Concentracion[11]=(0,0.03);

Las condiciones iniciales utilizadas fueron las mismas que utilizó [Aguilar, 1998] en sus experimentos con CVODE:

Concentracion[1]= 0.00954;
Concentracion[2]= 0.042;
Concentracion[3]= 0.01951;
Concentracion[4]= 0;
Concentracion[5]= 0.17192;
Concentracion[6]= 0;
Concentracion[7]= 0;
Concentracion[8]= 0.17192;
Concentracion[9]= 0;
Concentracion[10]= 0;
Concentracion[11]= 0;

Los coeficientes que varían con el ángulo del sol se consideraron como constantes en este intervalo de una hora. Con los valores: $k_1=0.533/60.0$ $k_6=1.91e-4/60.0$

Punto de referencia: No es conveniente utilizar la condición inicial c_{0R} (ver 5.1) como punto de referencia ya que la mayoría de las concentraciones se encuentran en cero. Sin embargo, si se deja correr el sistema por 30 minutos, obtendremos unos valores intermedios entre las concentraciones iniciales y las finales, que es precisamente la característica que se requiere para un buen punto de referencia.

Para establecer el punto de referencia de un FEOM hay que buscar un punto real que contenga valores intermedios a todos los valores posibles de cada una de las variables. En nuestro estudio de caso, esto se hizo de manera experimental. El punto de referencia que usamos con los valores en $t=30$ minutos produjo resultados satisfactorios y es el que se utilizó en los experimentos de la próxima sección.

5.4.- Elección de diferentes FEOMs en la malla.

5.4.1- Elección de un solo FEOM para toda la malla.

A continuación se muestra el nivel $z=10$ km de la malla mencionada en 5.2, con 14 puntos sobre el eje de las x (longitud), 16 puntos sobre el eje de las y (latitud) y 5 puntos sobre el eje de las z (altura).

En esta figura se ilustra el punto de referencia establecido en $x=30$ km y $y=35$ km para entrenar al FEOM que resuelve el mecanismo de reacciones químicas del modelo de 11 substancias.

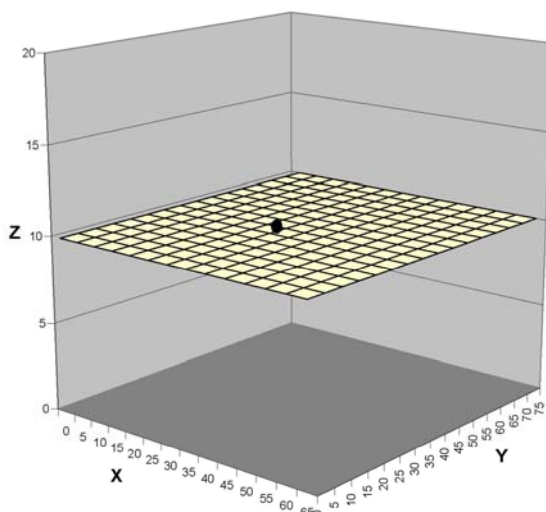


Figura 15. Un solo FEOM en $z=10$ km.

Utilizando un solo FEOM con punto de referencia en $x=30$ km y $y=35$ km con los valores obtenidos después de $t=30$ minutos, se obtienen las gráficas de error entre resultados de FEOM y CVODE en el nivel $z = 10$ km, (figuras 16, 17 y 18).

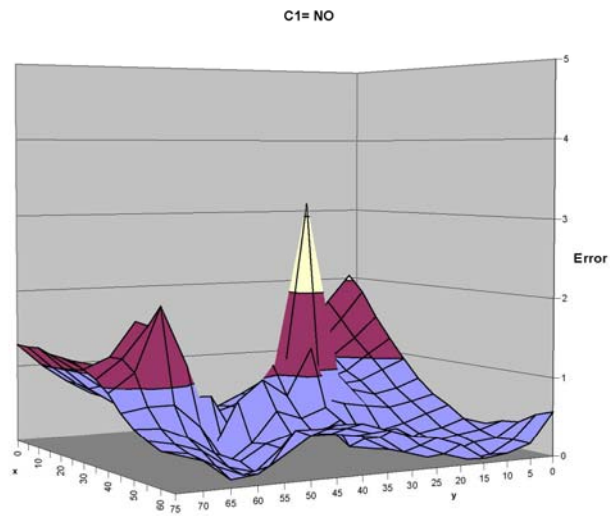


Figura 16: Error de la concentración $c_1 = \text{NO}$.

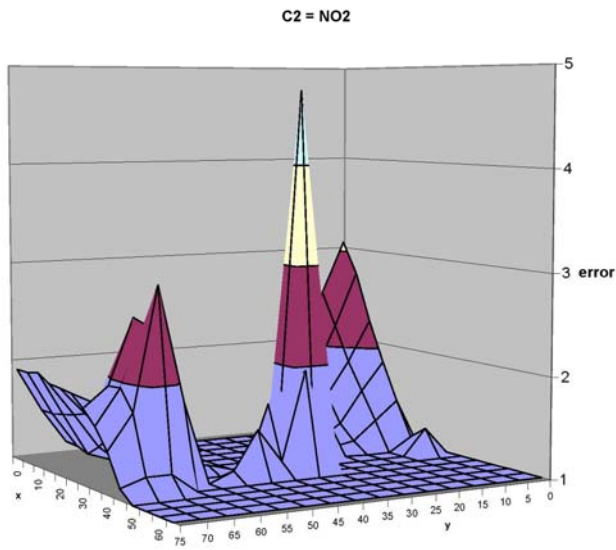


Figura 17: Error de la concentración $c_2 = \text{NO}_2$.

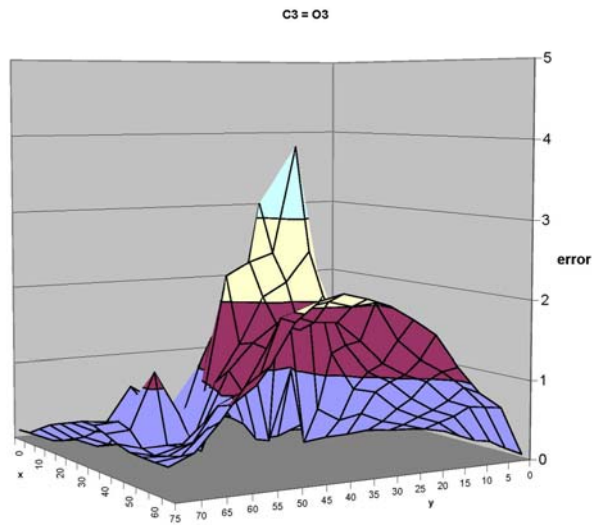


Figura 18: Error de la concentración $c_3 = \text{O}_3$.

En estas gráficas se puede apreciar que en la mayor parte del plano xy , el error relativo es menor al 2% y solo se presentan algunos picos en donde el error sube a valores menores al 4%. Hay una región muy pequeña en la que el error en c_2 es mayor a 4% (figura 17).

5.4.2- Elección de cuatro FEOMs para toda la malla.

A continuación se muestra el nivel $z=10$ km de la malla mencionada en 5.2. En esta figura se ilustran los cuatro puntos de referencia establecidos para entrenar los cuatro FEOMs que resuelven el mecanismo de reacciones químicas del modelo de 11 sustancias.

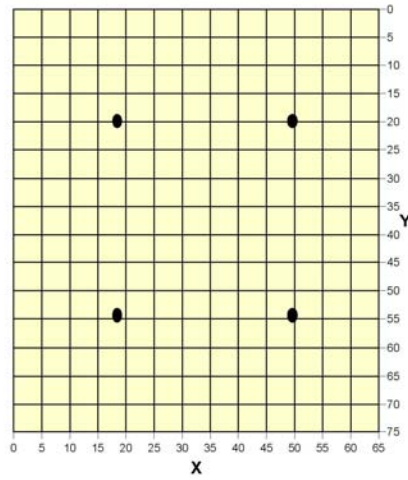


Figura 19: Cuatro FEOMs en $z=10$ km.

Utilizando cuatro FEOMs con puntos de referencia en los puntos de la figura 19 con los valores obtenidos después de $t=30$ minutos, se obtienen las siguientes gráficas de error entre resultados de FEOM y CVODE (figuras 20, 21 y 22).

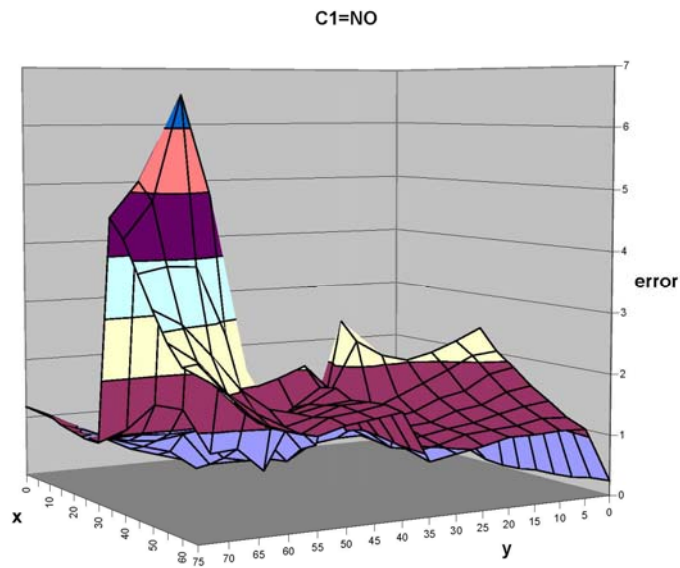


Figura 20: Error de la concentración $c_1 = \text{NO}$.

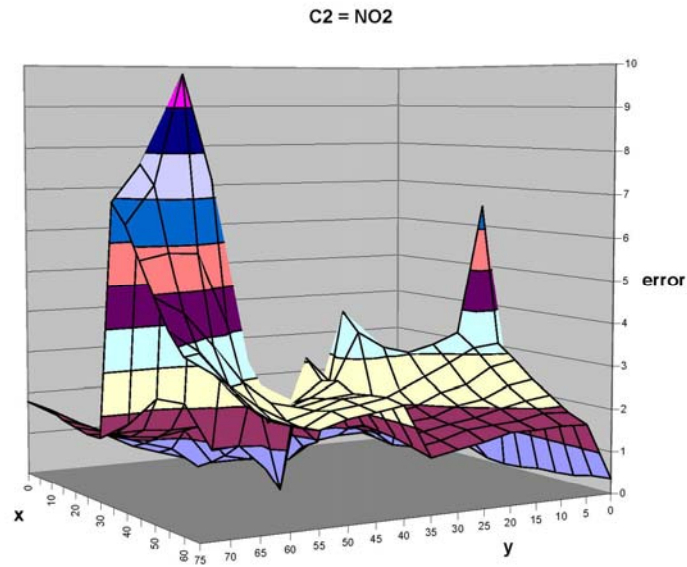


Figura 21: Error de la concentración $c_2 = \text{NO}_2$.

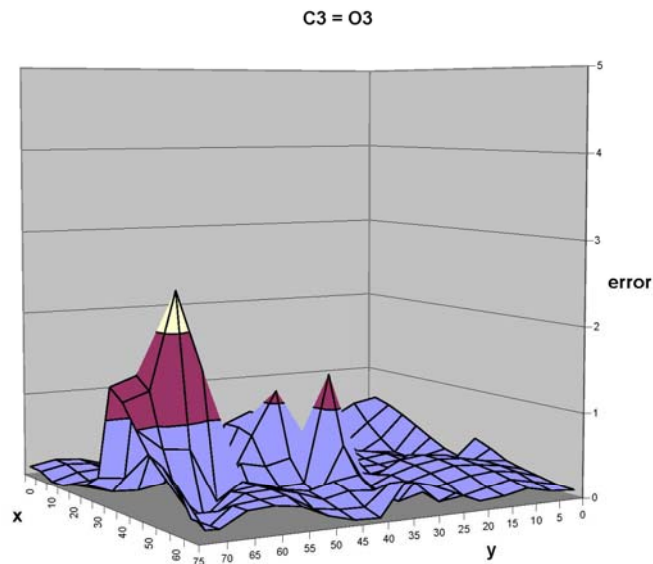


Figura 22: Error de la concentración $c_3 = \text{O}_3$.

En las figuras 20, 21 y 22 se puede apreciar que aunque para c_3 el error se reduce, para c_1 y c_2 el error aumenta.

5.4.3- Elección de nueve FEOMs para toda la malla.

En la figura 23 se muestran los nueve puntos de referencia establecidos para entrenar los 9 FEOMs que resuelven el mecanismo de reacciones químicas del modelo de 11 sustancias en el nivel $z=10$ km.

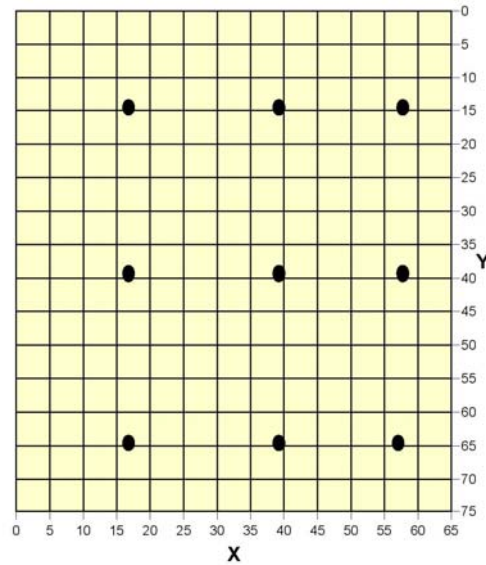


Figura 23: Nueve FEOMs en $Z=3$.

Utilizando nueve FEOMs con puntos de referencia en los puntos de la figura 23 con los valores obtenidos después de $t=30$ minutos, se obtienen las gráficas de error entre resultados de FEOM y CVODE de las figuras 24, 25 y 26.

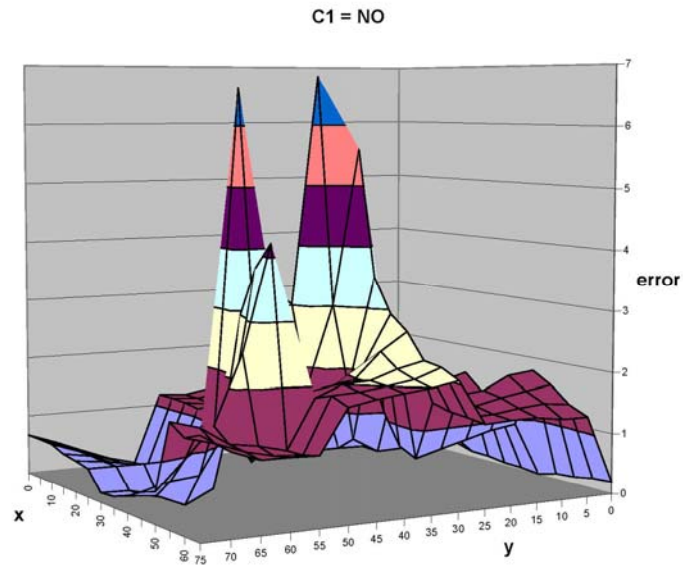


Figura 24: Error de la concentración $c_1 = \text{NO}$.

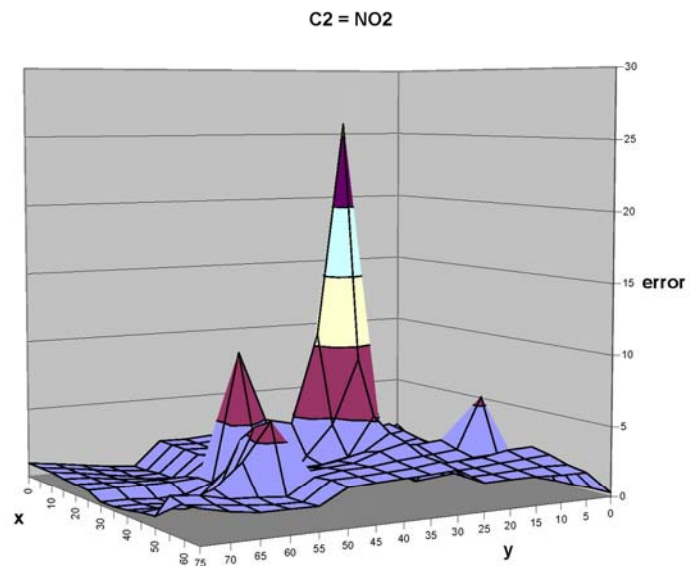


Figura 25: Error de la concentración $c_2 = \text{NO}_2$.

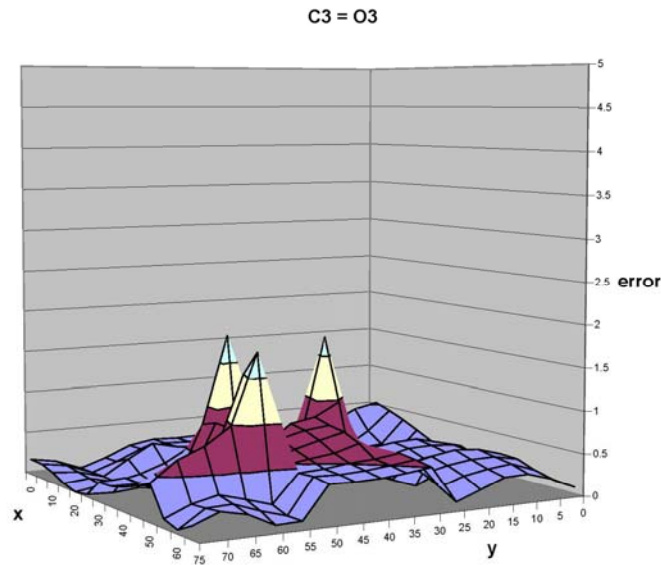


Figura 26: Error de la concentración $c_3 = \text{O}_3$.

En las figuras 24, 25 y 26 se puede apreciar lo mismo que sucede para 4 FEOMs, en la figura 26 se puede ver que el error en la concentración de ozono (c_3) es todavía mas pequeño que en resultados con uno y cuatro FEOMs, pero esto es a costa del incremento del error en los otros resultados.

5.5.- Resultado final del modelo de difusión/advección al aplicar el FEOM para resolver el sistema de reacciones químicas.

Eligiendo finalmente un solo FEOM para trabajar en la malla mencionada en 5.2, podemos comparar la forma de las gráficas de las concentraciones c_1 , c_2 y c_3 integradas con el FEOM y con CVODE para el nivel $z=10$ km, se puede apreciar en las figuras 27-32 que las formas son bastante similares.

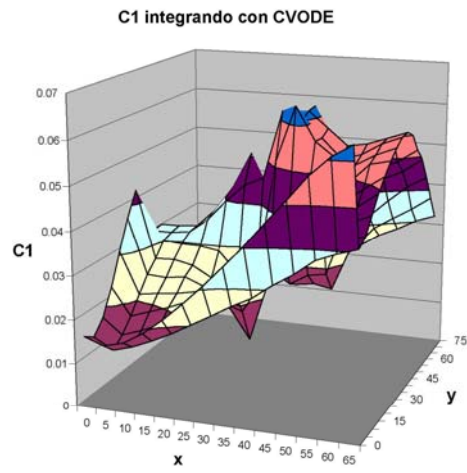


Figura 27: $c_1 = NO$ después de $t= 1$ hora (integrado con CVODE).

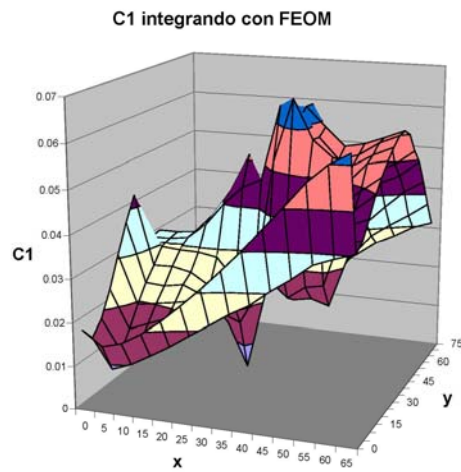


Figura 28: $c_1 = NO$ después de $t= 1$ hora (integrado con FEOM).

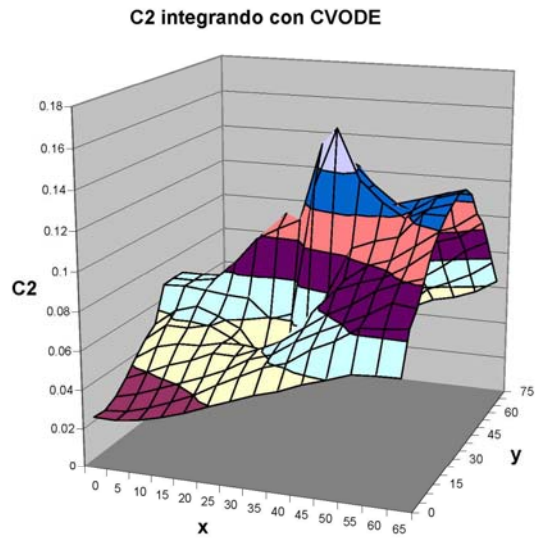


Figura 29: $c_2 = \text{NO}_2$ después de $t= 1$ hora (integrado con CVODE).

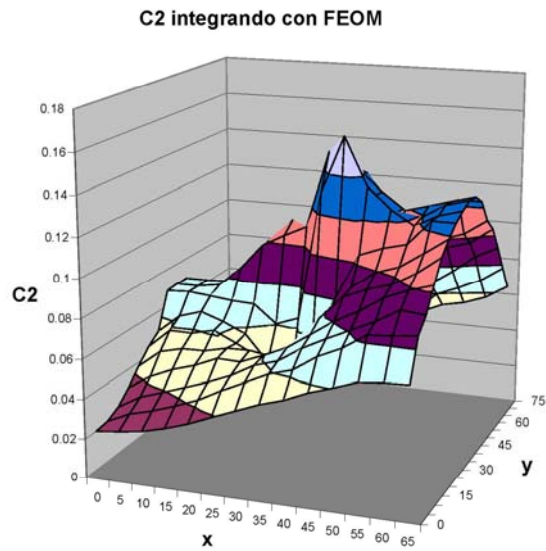


Figura 30: $c_2 = \text{NO}_2$ después de $t= 1$ hora (integrado con FEOM).

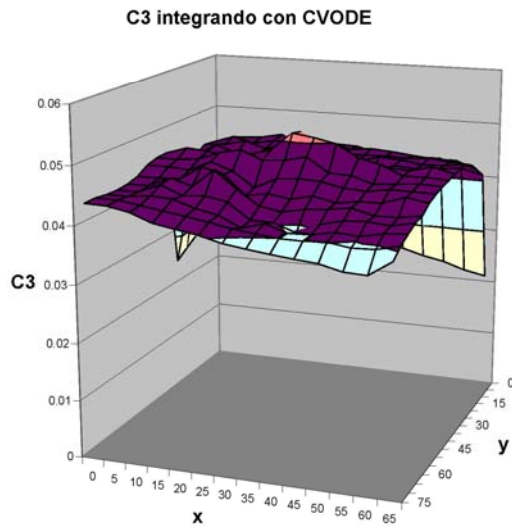


Figura 31: $c_3 = \text{O}_3$ después de $t=1$ hora (integrado con CVODE).

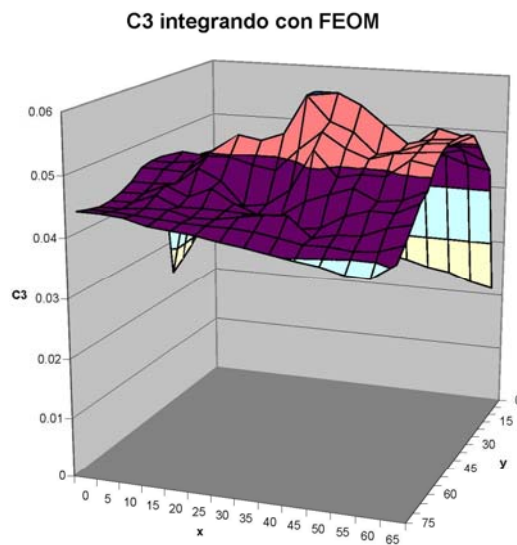


Figura 32: $c_3 = \text{O}_3$ después de $t=1$ hora (integrado con FEOM).

El modelo en 3 dimensiones de difusión/advección de contaminantes con un FEOM para resolver las reacciones químicas fue alrededor de 2.7 veces más rápido que cuando usa CVODE para resolver estas reacciones. El FEOM también fue más rápido que CVODE tanto para el mecanismo básico de cuatro sustancias (ver 4.9.1) como para el

mecanismo de Seinfeld de 27 sustancias (ver 4.9.2). Observamos que el FEOM incrementa la velocidad de solución de los mecanismos de reacciones químicas en comparación con CVODE, haciendo más rápido al modelo en 3 dimensiones de difusión/advección de contaminantes

5.6.- Conclusiones del capítulo.

En nuestra aplicación del FEOM al modelo de difusión/advección de contaminantes, se utilizó el modelo de 11 sustancias propuesto por Seinfeld (16) para resolver la parte de las reacciones químicas. La razón por la que se eligió este modelo de 11 sustancias es que en trabajos previos [Aguilar, 1998] ya se tenía este modelo resuelto con CVODE. De esta manera se pudo comparar la velocidad del FEOM con la que ya se tenía con CVODE.

Aunque el uso de cuatro y nueve FEOM's reduce aún más la diferencia entre los resultados del FEOM y CVODE para la producción de ozono ($c_3 = \text{O}_3$), esta diferencia en los otros dos indicadores ($c_1 = \text{NO}$) y ($c_2 = \text{NO}_2$) aumenta. Además, el esfuerzo necesario para obtener cuatro o nueve FEOM's es mayor que el necesario para trabajar con un solo FEOM.

Debido a que con un solo FEOM se obtuvieron diferencias menores a 4% para los tres indicadores c_1 , c_2 y c_3 (excepto un ligero pico para c_2) consideramos suficiente el empleo de un solo FEOM con grado de cooperatividad $l=1$ para cada intervalo de tiempo de una hora. Con este FEOM que sustituye a CVODE en el modelo en 3 dimensiones de difusión/advección de contaminantes, se obtuvo un tiempo de ejecución alrededor de 2.7 veces más rápido que el obtenido con CVODE.

CONCLUSIONES FINALES DE LA TESIS.

Las expansiones HDMR son herramientas para modelar sistemas físicos matemáticamente. Una de las aplicaciones principales de las HDMR es el uso de un FEOM (Modelo Operacional Completamente Equivalente) para modelar sistemas descritos por medio de un conjunto de ecuaciones diferenciales, los resultados obtenidos con el FEOM son exactos y se obtienen considerablemente más rápido que con los métodos numéricos convencionales.

Esta tesis estudia la factibilidad de la aplicación de un FEOM a la solución de los sistemas rígidos de ecuaciones diferenciales ordinarias que describen los mecanismos de reacciones químicas del modelo de difusión/advección de contaminantes en tres dimensiones. Hasta ahora, los problemas de valor inicial para los sistemas rígidos de ecuaciones diferenciales ordinarias, se habían resuelto normalmente usando integradores numéricos convencionales, entre ellos el paquete CVODE.

Aunque ya se ha aplicado el FEOM a la cinética química a nivel global en la troposfera [Wang et al., 1999; Wang et al., 2005] y en la estratosfera [Shorter et al., 1999] para períodos largos de tiempo, nunca se habían hecho investigaciones de los efectos del FEOM en modelos locales para períodos cortos de tiempo. Además, en estos trabajos previos nunca se especificó que tipo de integrador numérico se utilizó para hacer las comparaciones con el FEOM. En este trabajo se especifica claramente el uso de CVODE, que es un paquete sofisticado y poderoso, diseñado específicamente para resolver sistemas rígidos de ecuaciones diferenciales.

Si bien [Shorter et al., 1999; Wang et al., 2005] reportaron ganancias mucho mayores a las obtenidas en esta investigación, es necesario considerar dos aspectos. Primero, ellos utilizaron un paso de tiempo de un día, y forzaron al integrador numérico que utilizaron a emplear alrededor de 2000 pasos para obtener los resultados de un día, de esta forma obtuvieron una ganancia en velocidad de 1 paso de FEOM por cada 2000 de integrador numérico. Al usar CVODE se pierde esta ventaja, ya que éste es auto-ajustable, es decir, optimiza su paso de tiempo de manera automática, así es que no es posible ajustar manualmente el paso de tiempo de CVODE para que el FEOM tenga una mayor ventaja. El segundo aspecto que hay que considerar es el hecho de que la ganancia en velocidad que se reporta en estos dos trabajos, no se debe únicamente al FEOM, sino a la reducción de una importante cantidad de variables. Cuando se toma un paso de tiempo de un día, muchas variables no cambian su valor al inicio y final de cada día, por lo que se les considera como constantes. Con un análisis de sensibilidad, [Shorter et al., 1999; Wang et al., 2005] eliminaron una cantidad substancial de variables del FEOM y lo siguieron comparando contra la velocidad del integrador numérico que trabaja con todas las

variables, esto dio al FEOM aún mas ventaja. En nuestro caso, trabajamos con períodos cortos de tiempo, esto implica que todas las variables son importantes y no es posible eliminar ninguna. Aún sin las ventajas de estos dos aspectos, en esta tesis se logró reducir significativamente la carga computacional mediante el uso del FEOM.

El FEOM se aplicó a varios modelos diferentes de reacciones químicas. Se investigó en detalle la influencia de diferentes parámetros del FEOM sobre su exactitud y eficiencia computacional y se obtuvieron nuevos resultados desconocidos hasta la fecha. Con la experiencia obtenida se construyó un FEOM que opera de manera rápida y exacta para resolver el sistema rígido de EDO's planteado por el modelo de 11 variables de [Seinfeld, 1986] para los mecanismos de reacciones químicas que están dentro del modelo en 3 dimensiones de difusión/advección de contaminantes.

Durante las investigaciones de este proyecto se compararon las velocidades del FEOM con las de CVODE para 3 diferentes mecanismos de reacciones químicas de interés para la región de la ciudad de México y se encontró que el FEOM es entre 2.5 y 11 veces más rápido que CVODE. Este es un resultado interesante, considerando que CVODE es una herramienta muy eficiente desarrollada específicamente para este tipo de sistemas.

En esta tesis se demostró que si los mecanismos de reacciones químicas del modelo en 3 dimensiones de difusión/advección de contaminantes se modelan con un FEOM, se pueden obtener resultados rápidos y exactos. Esto puede brindar un ahorro de tiempo importante a los investigadores de la calidad del aire.

La robustez del FEOM queda pendiente para futuras investigaciones.

APÉNDICES.

APÉNDICE A: Algoritmos de búsqueda de proyección.

Este procedimiento modela una función como una suma de funciones suaves de combinaciones lineales de las variables de entrada de una manera iterativa.

Consiste en un refinamiento sucesivo de funciones base, a las cuales se les va aumentando el grado de complejidad hasta que los puntos muestreados se ajusten de la mejor manera posible al modelo.

Los algoritmos de búsqueda de proyección aproximan la función multivariable $f(\mathbf{x})$ de la siguiente forma:

$$f(\mathbf{x}) \equiv f(x_1, x_2, \dots, x_n) \approx \mu + \sum_{i=1}^k f_i \left(\sum_{k=1}^n \beta_{ik} x_k \right)$$

donde $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{in}]$ representa las direcciones de proyección y μ es el promedio de la función. β_i y f_i se estiman con los datos.

Desgraciadamente, las funciones que se presentan en la práctica con frecuencia no se pueden modelar bien con funciones de orden bajo (por ejemplo las superficies con asíntotas). El uso de polinomios de orden alto está limitado a que el número de muestras sea manejable y a que la carga computacional sea factible.

En este tipo de algoritmos se tienen puntos en el espacio de “ N ” dimensiones. Los valores de la función se conocen en dichos puntos. Se propone un esquema que permita, dados los valores en un conjunto de puntos, calcular el valor de la función en cualquier punto.

Si bien este algoritmo proporciona un modelo matemático del sistema para predecir sus valores de salida, no se tienen los recursos para apreciar la influencia de cada variable de entrada y así determinar cuales son las variables mas importantes y su orden de importancia [Friedman & Stuetzle, 1981].

APÉNDICE B: Funciones de base radial.

Es uno de los métodos que sirve para hacer interpolaciones de funciones de varias variables. Éste consiste en elegir una función F de la siguiente forma:

$$F(\mathbf{x}) = \sum_{i=1}^N c_i h(\|\mathbf{x} - x_i\|) + \sum_{i=1}^m d_i p_i(x) \quad m \leq n$$

Donde:

h es una función continua de R^+ a R llamada *función de base radial*.

$\| \cdot \|$ es la norma euclidiana de R^n .

$\{p_i \mid i = 1, \dots, m\}$ es la base del espacio lineal $\Pi_{k-1}(R^n)$ de los polinomios algebraicos de grado $k-1$ como máximo de R^n a R . K esta dado.

Las condiciones de interpolación proporcionan N ecuaciones lineales para los N coeficientes c_i y d_i , los grados de libertad restantes se fijan con la siguiente restricción:

$$\sum_{i=1}^N c_i p_i(x_i) = 0 \quad i = 1, 2, \dots, m$$

Según [Micchelli, 1986] en la práctica puede usarse la siguiente lista de funciones para interpolar por medio del método de funciones de base radial:

$$\begin{aligned} h(r) &= e^{-\left(\frac{r}{2}\right)^2} && \text{(Gaussiana)} \\ h(r) &= \frac{1}{(c^2 + r^2)^\alpha} && \alpha > 0 \\ h(r) &= (c^2 + r^2)^\beta && 0 < \beta < 1 \\ h(r) &= r && \text{(lineal)} \end{aligned}$$

Casi todas estas funciones tienen la desventaja de depender de un parámetro, el cual generalmente está en función de la distribución de los datos.

Para aplicar el método de las funciones de base radial se utilizan redes neuronales de 3 capas, en donde existe una neurona en la capa de entrada llamada *capa de regularización* por cada variable de entrada.

La segunda capa está compuesta por neuronas con funciones no-lineales conectadas completamente a la primera capa de la siguiente forma:

En la 2ª capa existe una neurona por cada dato $f(x_{1i}, x_{2i}, \dots, x_{ni})$ y los pesos que conectan a la neurona i de la 2ª capa, con las neuronas de la 1ª capa, están dados por los

valores del punto $(x_{1i}, x_{2i}, \dots, x_{ni})$. La *función de activación* de las neuronas de la 2ª capa está dada por la función de Green [Poggio & Girosi, 1990].

La 3era capa es una combinación lineal (suma) de todas las funciones de Green de la 2ª capa.

Para resolver en la práctica problemas de interpolación de muchas variables utilizando redes de regularización, debe proporcionarse la mayor cantidad posible de *información a priori* para que el entrenamiento de la red sea manejable y se pueda contrarrestar la *maldición de la dimensionalidad*.

Este método es adecuado para las funciones que sean suaves.

La *suavidad* de la función es una restricción muy general y “débil”, para lograr una buena aproximación con este método son necesarias otras *restricciones a priori* mas “fuertes”, como: que la función a aproximar es lineal, o que tiene un rango positivo, o que tiene un dominio limitado, o que no varía con algún grupo de transformaciones.

Si no se tiene ninguna de las restricciones “fuertes” a priori, la única manera de resolver el problema es asumir que la función tiene un alto grado de *suavidad*, ya que de otra forma, el número de muestras que se requerirían sería totalmente impráctico.

El punto clave para poder comparar el método de interpolación de base radial con el de las expansiones HDMR es determinar: ¿Cuántas muestras se necesitan para conseguir un cierto grado de exactitud deseado?

La respuesta depende de la cantidad de variables de entrada n y del grado de suavidad de las funciones a aproximar p . [Stone , 1982] estudió a fondo este problema y obtuvo el siguiente resultado fundamental:

“El número de muestras que se necesitan para aproximar de manera razonable a una función, crece de forma exponencial con la dimensión de la entrada n y con el grado de suavidad de la función p .” [Stone , 1982].

Una de las razones de mas peso para optar por las expansiones HDMR es que mediante esta técnica es posible reducir el problema de complejidad exponencial a complejidad polinomial. Por otra parte, las redes neuronales requieren de un buen conocimiento a priori del sistema para que el problema sea manejable.

Además, no existen procedimientos generales para determinar el número de capas de la red, el número de funciones a utilizar, etc.

APÉNDICE C: Interpolación multidimensional.

El método seleccionado para hacer esta clase de interpolación [Press et al., 1999] tiene mas precisión que la interpolación bilinear y no requiere de la obtención de gradientes ni de derivadas.

La idea fundamental del método es dividir el problema en una sucesión de interpolaciones de una sola dimensión. Por ejemplo, para la interpolación de un punto (x_1, x_2) en una función de dos variables $y(x_1, x_2)$ se tiene una matriz de tamaño $m \times n$:

$x_1 \setminus x_2$	x_{21}	x_{22}	x_{2n}
x_{11}	y_{11}	y_{12}	y_{1n}
x_{12}	y_{21}	y_{22}	y_{2n}
x_{13}	y_{31}	y_{32}	y_{3n}
\vdots
x_{1m}	y_{m1}	y_{m2}	y_{mn}

Suponiendo que se desea interpolar para encontrar el valor de $y(x_1, x_2)$ se hacen n interpolaciones en dimensión-1 para x_1 y se obtiene un vector de valores interpolados \mathbf{z} , es decir:

Con x_{21} fija:

x_1	$y(x_1, x_{21})$
x_{11}	y_{11}
x_{12}	y_{21}
x_{13}	y_{31}
\vdots	...
x_{1m}	y_{m1}

$x_1 >$ $< \mathbf{z}_1$

Con x_{22} fija:

x_1	$y(x_1, x_{22})$
x_{11}	y_{12}
x_{12}	y_{22}
x_{13}	y_{32}
\vdots	...
x_{1m}	y_{m2}

$x_1 >$ $< \mathbf{z}_2$

\vdots
 \vdots

\vdots

Con x_{2n} fija:

x_1	$y(x_1, x_{2n})$	$< \mathbf{z}_n$
x_{11}	y_{1n}	
x_{12}	y_{2n}	
x_{13}	y_{3n}	
\vdots	\dots	
x_{1m}	y_{mn}	

Finalmente, para encontrar $y(x_1, x_2)$ se hace otra interpolación en dimensión-1 sobre el vector \mathbf{z} obtenido.

x_2	$\mathbf{z} = y(x_{1i}, x_2)$	$< y(x_1, x_2)$
x_{21}	z_1	
x_{22}	z_2	
x_{23}	z_3	
\vdots	\dots	
x_{2n}	z_n	

Puede usarse el método de interpolación de Lagrange [Smith, 1998] de orden $m-1$ para obtener el vector \mathbf{z} y de orden $n-1$ para obtener $y(x_1, x_2)$, de esta forma no es necesario que el espaciamiento de las funciones tabulares sea constante.

Usar el método de interpolación de Lagrange es útil cuando se tienen como datos los valores exactos de la función $y(x_1, x_2)$.

La interpolación es mas rápida se si usa interpolación lineal para obtener el vector \mathbf{z} .

Extendiendo el problema a una función de 3 variables $y(x_1, x_2, x_3)$, el vector \mathbf{z} se construye de la siguiente manera:

Para cada valor de alguna de las variables, digamos x_3 , se hace una interpolación en dimensión-2 con las variables restantes. Finalmente se hace una interpolación en dimensión-1 con el vector \mathbf{z} variando x_3 .

Si se requiere una interpolación en k dimensiones, para cada valor de x_l $1 \leq l \leq k$ se interpola sobre las demás variables en dimensión $k-1$. De los valores obtenidos se interpola en dimensión-1.

Se puede observar que el problema de interpolación en una función de k variables crece exponencialmente con el número de variables de la función. Afortunadamente, gracias a la conjetura fundamental de las HDMR las funciones que se presentan son de una, de dos variables y en muy raros casos hasta de 3 variables.

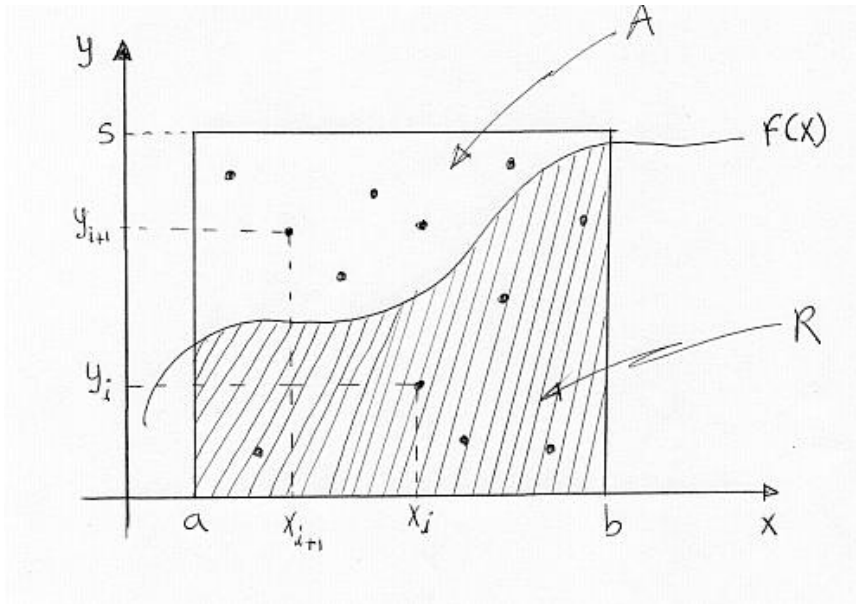
APÉNDICE D: El método de Monte Carlo.

Este método es una forma probabilística de aproximar las integrales. Sirve tanto para integrales de una variable, como para integrales de “n” variables. Para integrar una función de una variable, primero se delimita un área que contenga a la función y se tiran N puntos al azar sobre esta área. En el caso de una función de una variable, el principio del método de Monte Carlo es obtener la fracción de los puntos que caen bajo la curva a integrar, con esto se tiene un estimado de la proporción entre el área bajo la curva y el área del cuadrado.

La base del Método de Monte Carlo es la generación de números aleatorios. Por fortuna, casi todos los lenguajes de programación cuentan con un generador de números aleatorios.

Algoritmo para integrar una función de una variable por el método de Monte Carlo.

(Ejemplo de una función de una variable)



Se escoge un área **A** delimitada por :

- 1.- Una cota superior **S** sobre el eje de las y .
- 2.- Un intervalo (a,b) sobre el eje de las x .

3.- Se tira un punto (x_i, y_i) al azar dentro del área **A**.

4.- Para esa x_i obtenida se calcula $f(x_i)$.

5.- Se compara y_i con $f(x_i)$ y se registra si fue menor o mayor a $f(x_i)$.

6.- Se repiten los pasos 3 al 5 para N puntos al azar que caen dentro del área **A**.

7.- Se calcula el porcentaje de y_i que resultaron menores que la $f(x_i)$, es decir, que cayeron

dentro de la región **R**.

8.- Si N_R es el número de puntos que cayeron dentro de la región **R**, donde **R** es el área

bajo la curva $f(x)$ y N es el número total de puntos tirados al azar sobre el área A , entonces:

$$\frac{N_R}{N} \approx \frac{Area(R)}{Area(A)} = \frac{\int_a^b f(x)dx}{Area(A)}$$

Cuando se emplea un cubo unitario entonces $Area(A) = 1$, por lo tanto:

$$\int_0^1 f(x)dx = \frac{N_R}{N}$$

El método de Monte Carlo puede ampliarse a mas variables, por ejemplo, para 2 variables, si se escoge un cubo unitario:

$$\frac{N_R}{N} \approx \frac{Vol(W)}{Vol(C)} = Vol(w) = \int_0^1 \int_0^1 f(x_1, x_2)dx_1dx_2$$

Donde W es la región bajo $f(x_1, x_2)$ y C es el cubo unitario.

Cuando se use el método de Monte Carlo, es importante seleccionar una pequeña caja C que contenga totalmente a la región W . Por intuición, cuanto mejor sea el ajuste entre los volúmenes, se necesitan menos puntos aleatorios para obtener una aproximación razonable. De hecho, el problema mas grande del método de Monte Carlo es encontrar una caja rectangular lo mas pequeña posible pero de tamaño suficiente como para que encierre al volumen que se desea obtener.

En el caso en el que la función no esté acotada en el intervalo de integración (por ejemplo:

$\int_0^1 \int_0^1 x^{-y} dx dy$), no es posible utilizar el método anteriormente mencionado. Existe otro

método de Monte Carlo [Hughes et al., 1998] en el que no es necesario tener una cota superior de la función y consiste en lo siguiente:

Puede calcularse el valor promedio de $f(x)$ si se escogen N puntos (x_1, x_2, \dots, x_N) al azar en la región " R ". Se suman los valores de $f(x)$ en estos puntos y se divide entre N , con lo que resulta:

$$A = \frac{1}{N} \sum_{i=1}^N f(x_1, x_2, \dots, x_n)$$

Para una función de 2 variables:

$$A = \frac{1}{N} \sum_{i=1}^N f(x_{1_i}, x_{2_i})$$

$$\int_R f(x_1, x_2) dA \approx \text{Area}(R) * A$$

Para una función de 3 variables:

$$A = \frac{1}{N} \sum_{i=1}^N f(x_{1_i}, x_{2_i}, x_{3_i})$$

$$\int_R f(x_1, x_2, x_3) dA \approx \text{Volumen}(R) * A$$

Para cualquier dimensión: si se tienen N puntos aleatorios distribuidos uniformemente en un volumen multidimensional V llamados $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$, entonces el teorema básico de la integración de Monte Carlo sobre un volumen multidimensional es:

$$\int f(\mathbf{x}) dv = V \left(\frac{1}{N} \sum_{s=1}^N f(\mathbf{x}_s) \right) \pm \text{error}$$

En el caso de RS_HDMR, como las variables x_i del vector \mathbf{x} están normalizadas de modo que $0 \leq x_i \leq 1$ para todas las i , entonces:

$$\int f(\mathbf{x}) dv \approx \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}_s)$$

El éxito del método de Monte Carlo está en las cualidades del generador de números aleatorios. Un buen generador de números aleatorios debe lanzar los puntos *lo mas uniformemente posible*.

APÉNDICE E: “El algoritmo de interpolación de Neville para interpolación de Lagrange”.

El algoritmo de Neville es una mejora al algoritmo de Lagrange, consiste en elaborar un polinomio único de interpolación de la siguiente forma:

Sea P_1 el valor en x de un polinomio de grado cero, es decir, una constante que pasa por el punto (x_1, y_1) , entonces $P_1 = y_1$. De esta manera se definen los polinomios P_2, P_3, \dots, P_N .

Ahora, sea P_{12} el valor en x de un polinomio único de grado 1 que pasa por los puntos (x_1, y_1) y (x_2, y_2) . De la misma manera se definen los polinomios $P_{23}, P_{34}, \dots, P_{(N-1)N}$.

Los polinomios de grado superior (hasta grado $N-1$: $P_{12\dots N}$) se definen de forma similar.

En base al número de puntos “ N ” se construye una tabla en donde los polinomios *ancestros* están del lado izquierdo y los *descendientes* del lado derecho, por ejemplo, para $N=4$:

i		$m=1$	$m=2$	$m=3$
1	$x_1 \quad y_1 = P_1$			
		P_{12}		
2	$x_2 \quad y_2 = P_2$		P_{123}	
		P_{23}		P_{1234}
3	$x_3 \quad y_3 = P_3$		P_{234}	
		P_{34}		
4	$x_4 \quad y_4 = P_4$			

El algoritmo de Neville es una manera recursiva de llenar la tabla anterior. Se obtiene primero la columna del lado izquierdo y después se van obteniendo las columnas sucesivas de izquierda a derecha.

Este algoritmo se basa en la relación que existe entre el polinomio “hijo” y sus dos “padres”:

$$P_{i(i+1)\dots(i+m)} = \frac{(x - x_{i+m})P_{i(i+1)\dots(i+m-1)} + (x_i - x)P_{(i+1)(i+2)\dots(i+m)}}{x_i + x_{i+m}} \quad (1)$$

La recurrencia funciona porque los dos padres coinciden con el mismo hijo en los puntos

$$x_{i+1} \dots x_{i+m-1}$$

Una mejora a la fórmula recurrente (1) es llevar un registro de las pequeñas diferencias que existen entre padres e hijos para después hacer las correcciones pertinentes. Sean los diferentes niveles de los padres $m = 1, 2, \dots, N-1$, si:

$$\begin{aligned}
C_{m,i} &\equiv P_{i\dots(i+m)} - P_{i\dots(i+m-1)} \\
D_{m,i} &\equiv P_{i\dots(i+m)} - P_{(i+1)\dots(i+m)}
\end{aligned}
\tag{2}$$

entonces de la relación (1) se pueden derivar las siguientes relaciones:

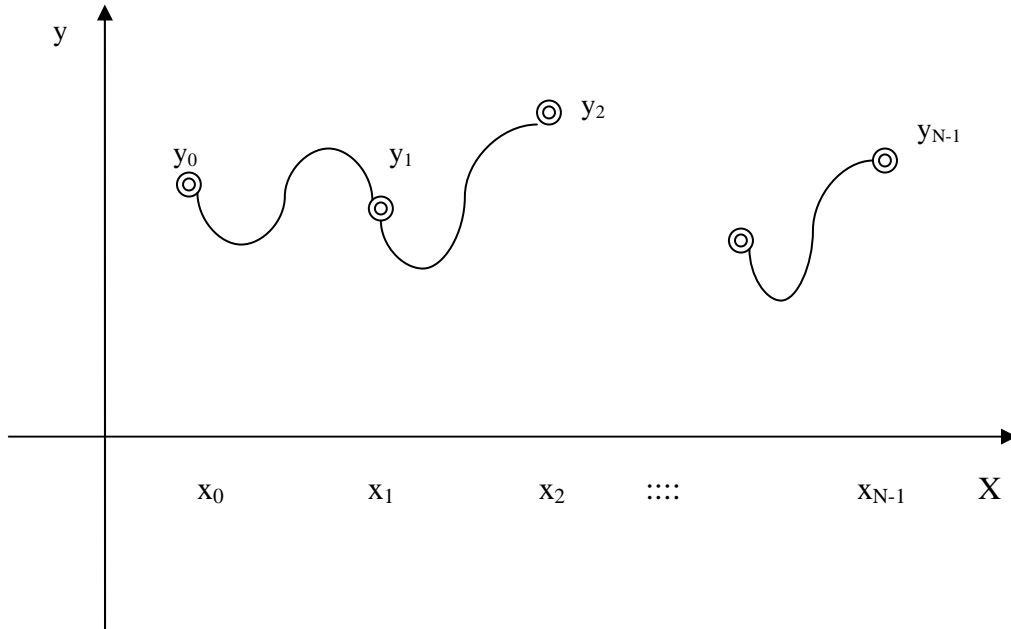
$$\begin{aligned}
D_{m+1,i} &= \frac{(x_{i+m+1} - x)(C_{m,i+1} - D_{m,i})}{x_i - x_{i+m+1}} \\
C_{m+1,i} &= \frac{(x_i - x)(C_{m,i+1} - D_{m,i})}{x_i - x_{i+m+1}}
\end{aligned}
\tag{3}$$

las C's y las D's son las correcciones en cada nivel m para hacer la interpolación de un grado mas alto.

La respuesta final, que es el polinomio de interpolación de grado $N-1$: $P_{123\dots N}$ es igual a la suma de cualquier y_i mas el conjunto de C's o D's que forman el camino para llegar desde y_i hasta el hijo que se encuentre hasta la derecha. [Press et al.; 1999]

APÉNDICE F: “El algoritmo de interpolación spline cúbico”.

Este tipo de interpolación se basa en la construcción de polinomios de 3er grado que pasan por 2 puntos de la función tabular dada por $x_j, y_j \quad j = 0, \dots, N-1$.



Para suavizar la curva en los puntos x_j se requiere que la segunda derivada del polinomio de tercer grado que va de x_{j-1} a x_j sea la misma que la del polinomio de tercer grado que va de x_j a x_{j+1} . Además la variación de la segunda derivada en cualquier intervalo (x_j, x_{j+1}) debe ser lineal.

El polinomio de grado 3 que se obtiene del punto x_j al punto x_{j+1} , está dado por:

$$y = Ay_j + By_{j+1} + Cy_j'' + Dy_{j+1}''$$

Donde:

$$A = \frac{x_{j+1} - x}{x_{j+1} - x_j}$$

$$B = 1 - A = \frac{x - x_j}{x_{j+1} - x_j}$$

$$C = \frac{1}{6}(A^3 - A)(x_{j+1} - x_j)^2$$

$$D = \frac{1}{6}(B^3 - B)(x_{j+1} - x_j)^2$$

Dado que la variación en la 2ª derivada es lineal, en la 1ª derivada la variación es cuadrática. Para que la 1ª derivada sea continua, se requiere que la primera derivada en x_j del polinomio de tercer grado que va de x_{j-1} a x_j sea la misma que la del polinomio de tercer grado que va de x_j a x_{j+1} . Con esto se obtiene un sistema de $N-2$ ecuaciones lineales con N incógnitas. Para encontrar una solución única es necesario especificar dos condiciones más, llamadas condiciones en la frontera, estas pueden ser:

- Que una de las dos o las dos segundas derivadas y_0'' y y_{N-1}'' sean cero, a esto se le llama *spline cúbico natural*.

- Que una de las dos o las dos segundas derivadas y_0'' y y_{N-1}'' tengan un valor determinado.

.. Cuando se especifican las dos condiciones en la frontera, el sistema de ecuaciones lineales que se obtiene es *tridiagonal*, por lo que la solución del sistema es $O(N)$, es decir, es bastante rápida.

Para obtener las segundas derivadas en los puntos (x_j, y_j) , se llama una sola vez a la función *spline*, cuyo prototipo es el siguiente:

```
Void spline(Vec_i_DP &x, Vec_i_DP &y, const DP yp1,  
           Const DP ypn, Vec_O_DP &y2)
```

Los parámetros de **entrada** que se requieren son los siguientes:

Función tabular:

x : arreglo con los valores de x_i : $x[0.....n-1]$

y : arreglo con los valores de $y_i = f(x_i)$: $y[0.....n-1]$

Condiciones a la frontera:

y_{p1} : valor de la primera derivada en x_0 .

y_{pn} : valor de la primera derivada en x_{n-1} .

Para especificar un spline cúbico natural, el valor de y_{p1} y/o y_{pn} debe ser mayor o igual a 1×10^{30} .

Salida: en y_2 se obtienen los valores de las segundas derivadas en cada punto (x_j, y_j) , una vez obtenido el vector de segundas derivadas Y_2 , se hace un llamado a “splint” cada vez que se desee hacer la interpolación de un punto. El prototipo de splint es el siguiente

```
Void splint(Vec_i_DP &Xa, Vec_i_DP &Ya, Vec_i_DP &Y2a,  
           const DP x, DP &y)
```

Los *parámetros de entrada* que “splint” requiere son los siguientes:

- La función tabular:

Xa : arreglo con los valores de x_i : $x[0.....n-1]$

Ya : arreglo con los valores de $y_i = f(x_i)$: $y[0.....n-1]$

- El vector de segundas derivadas obtenido con la función “spline”: $Y2a$.

- El valor a interpolar: x .

La *Salida* es : $y=f(x)$ que es el valor interpolado con un polinomio spline cúbico.
[Press et al.; 1999]

APÉNDICE G: "Valoración del grado global de incertidumbre".

Como en I.3.3.- *Descomposición de la varianza en las expansiones HDMR* : "La varianza total σ_f^2 de $f(\mathbf{x})$ causada por todas las variables de entrada, puede ser descompuesta en la suma de las diferentes contribuciones a la varianza total de cada una de las variables de entrada":

$$\sigma_f^2 = \sum_{i=1}^n \sigma_i^2 + \sum_{1 \leq i < j \leq n} \sigma_{ij}^2 + \dots \quad (1)$$

ya que la ortogonalidad de las funciones componente de la expansión HDMR implica que éstas son independientes y cada una de ellas proporciona información única acerca de cómo afecta la relación entre las variables de entrada a las propiedades de la salida.

Una expresión análoga a la ec. (1) fue dada por [Sobol, 1993] y por [Cukier, 1978].

La descomposición de la función original por medio de la expansión HDMR en un conjunto de funciones de baja dimensión, las cuales pueden ser muestreadas exhaustivamente, permite conocer a fondo la muestra ahorrando carga computacional. Es decir, con un esfuerzo de muestreo modesto, HDMR puede proporcionar información fiable.

Además, utilizando la propiedad de la ec. (1) es factible analizar cuales variables del modelo son las mas importantes y como interactúan unas con otras.

La información resultante de este análisis proporciona indicios para modificar el modelo y agregar mas observaciones del laboratorio o del campo con el fin de mejorar su calidad.

Una de las aplicaciones de la representación de las funciones componente por medio de polinomios ortonormales [Li et al., 2002 (28)] es *la valoración del grado global de incertidumbre*.

La determinación de cada una de las varianzas $\sigma_i^2, \sigma_{ij}^2, \dots$ esta dada por la evaluación de las integrales: $\int f_i^2(x_i) dx_i, \int \int f_{ij}^2(x_i, x_j) dx_i dx_j, \dots$ etc. El método de Monte Carlo es muy buen medio para la evaluación de estas integrales.

Si se tienen N muestras de un vector de entrada de dimensión n generadas de manera aleatoria, entonces, usando el método de Monte Carlo:

$$f_0 = \int_{k^n} f(\mathbf{x}) d\mathbf{x} \approx \frac{1}{N} \sum_{s=1}^N f(\mathbf{x}^{(s)}) \quad (2)$$

$$\sigma_f^2 = \int_{k^n} [f(\mathbf{x}) - f_0]^2 d\mathbf{x} \approx \frac{1}{N} \sum_{s=1}^N f^2(\mathbf{x}^{(s)}) - f_0^2 \quad (3)$$

Cuando $N \rightarrow \infty$ se obtienen valores muy precisos de f_0 y $\sigma_{\bar{f}}$, sin embargo, con mucha frecuencia f_0 y $\sigma_{\bar{f}}$ convergen bastante rápido y un valor modesto de N puede dar muy buenos resultados.

Calculando los coeficientes α_k^i y β_{pq}^{ij} de la forma en la que se expone en *Aproximación de funciones componente por medio de polinomios ortonormales*, [Li et al., 2002 (28)] σ_i^2 y σ_{ij}^2 pueden obtenerse usando la propiedad ortonormal de la base $\varphi_k(x_i)$ con $k = 1, 2, 3$:

$$\begin{aligned}
 \sigma_i^2 &= \int_0^1 f_i^2(x_i) dx_i \approx \int_0^1 \left[\sum_{k=1}^3 \alpha_k^i \varphi_k(x_i) \right]^2 dx_i = \\
 &= (\alpha_1^i)^2 \int_0^1 \varphi_1^2(x_i) dx_i + (\alpha_2^i)^2 \int_0^1 \varphi_2^2(x_i) dx_i + (\alpha_3^i)^2 \int_0^1 \varphi_3^2(x_i) dx_i \\
 &= \sum_{k=1}^3 (\alpha_k^i)^2 \\
 \sigma_i^2 &= \sum_{k=1}^3 (\alpha_k^i)^2 \tag{4}
 \end{aligned}$$

De manera similar:

$$\begin{aligned}
 \sigma_{ij}^2 &= \int_0^1 \int_0^1 f_{ij}^2(x_i, x_j) dx_i dx_j \approx \int_0^1 \int_0^1 \left[\sum_{p=1}^3 \sum_{q=1}^3 \beta_{pq}^{ij} \varphi_p(x_i) \varphi_q(x_j) \right]^2 dx_i dx_j = \\
 &= \sum_{p=1}^3 \sum_{q=1}^3 (\beta_{pq}^{ij})^2 \\
 \sigma_{ij}^2 &= \sum_{p=1}^3 \sum_{q=1}^3 (\beta_{pq}^{ij})^2 \tag{5}
 \end{aligned}$$

Como solo es necesario un conjunto de muestras aleatorias de $f(\mathbf{x})$ para determinar f_0 y todos los coeficientes de la expansión $\alpha_k^i, \beta_{pq}^{ij} \dots$ para $f_i(x_i), f_{ij}(x_i, x_j) \dots$, consecuentemente solo es necesario un conjunto de muestras aleatorias para determinar $\sigma_{\bar{f}}^2, \sigma_i^2, \sigma_{ij}^2, \dots$ etc.

Así, RS-HDMR proporciona no solamente valoraciones cuantitativas de $\sigma_{\bar{f}}^2, \sigma_i^2, \sigma_{ij}^2, \dots$ sino también proporciona el comportamiento cualitativo de las acciones

independientes y colectivas de las variables de entrada en la salida por medio de las funciones componente $f_i(x_i), f_{ij}(x_i, x_j), \dots$.

En [Li et al., 2002 (28)] se ilustra la metodología para analizar cual es la importancia de cada una de las variables de entrada y cuales son los efectos cooperativos de primer y segundo orden de dichas variables sobre la salida. La metodología se aplicó a dos ejemplos:

- a) Un modelo de fotoquímica atmosférica.
- b) Un modelo de remediación biológica de rastros de metal.

En este análisis se utilizaron polinomios ortonormales lineales, cuadráticos y cúbicos para aproximar las funciones componente de primer y segundo orden de cooperatividad $f_i(x_i)$ y $f_{ij}(x_i, x_j)$.

Del análisis realizado en este artículo sobre dos problemas de la práctica se puede destacar la siguiente información importante:

1.- Se confirmó que los términos de la expansión HDMR con un alto orden de cooperatividad son despreciables.

2.- Se confirmó que la suma de las contribuciones a la varianza total son menores o iguales a la varianza total:

$$\sum \sigma_i^2 + \sum \sigma_{ij}^2 \leq \sigma_f^2$$

Además la acción de las variables en forma independiente es mucho mayor que la acción de las variables con orden de cooperatividad $l = 2$:

$$\sum \sigma_i^2 \gg \sum \sigma_{ij}^2$$

3.- La variable de entrada cuya desviación estándar sea la mayor será la mas importante. A su vez, la que tenga la menor desviación estándar será la variable de entrada que menos afecte a la salida.

4.- El orden de importancia de las variables de entrada que se obtiene con muestras de diferente tamaño es básicamente el mismo.

5.- Se hizo el mismo análisis utilizando *polinomios ortonormales teóricos* y *polinomios ortonormales óptimos* y se encontró que éstos últimos solo son necesarios cuando se tiene un número de muestras muy pequeño.

APÉNDICE H: “Aseguramiento de la calidad del software (inspecciones)”.

Registro de las inspecciones.

- Fecha: 8 de septiembre 2004.

Resultados:

1.- En el análisis de requerimientos faltaban los siguientes puntos importantes:

- Las posibles salidas deben quedar dentro del área que cubren las entradas seleccionadas.

- Incluir la métrica de los resultados que permita evaluar el desempeño del sistema.

2.- En el diseño de alto nivel se cambió el enfoque, en un principio se contempló un caso sencillo, pero de esta manera luego resulta complicado implantar un caso difícil, sin embargo, si se abarca desde el principio el caso mas complicado, éste contiene al mas sencillo y así se resuelve de una sola vez todo el problema.

Además se determinó bajar los datos de los archivos que contienen las tablas de las $f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{ijk}(x_i, x_j, x_k)$ a unas tablas en RAM, para aumentar así la velocidad de acceso.

- Fecha: 23 de septiembre 2004.

Resultados:

1.- En el diseño de alto nivel, es necesario determinar los límites del sistema, es decir, ¿cuántas variables es capaz de manejar?.

2.- En el diseño detallado se propuso trabajar con extensiones especiales para los archivos del sistema.

- Fecha: 28 de octubre 2004.

Resultados:

Al hacer la lectura de código de la fase 2 del sistema: FEOM.cpp, se encontró que la organización de las tablas de datos no era la adecuada, ya que para hacer cada interpolación había que extraer la columna o la matriz correspondiente a cada función

$f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{ijk}(x_i, x_j, x_k)$ lo que provocaba una ejecución ineficiente.

Para optimizar la ejecución se rediseñaron las tablas $f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{ijk}(x_i, x_j, x_k)$ utilizadas tanto en la primera fase como en la segunda fase.

- Fecha: 3 de Noviembre-2004.

Resultados:

1.- La lectura de código detectó un error al calcular las $f(x^{ij}, \bar{x})$ (al copiar el punto de referencia se borraba un dato).

2.- Se propuso una optimización del código que reduce el número de llamadas al integrador (se dividió el código en dos partes, un conjunto de funciones que obtiene las $f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{ijk}(x_i, x_j, x_k)$ y otro conjunto de funciones que las guarda en archivos en el orden del rediseño.

- Fecha: 17 de Noviembre-2004.

Resultados:

1.- Se rediseñó por segunda vez el sistema FEOM.cpp para hacer las siguientes mejoras:

a.- Los archivos de entrada están como default en el sistema, ya no los selecciona el usuario a menos que después los quiera cambiar.

b.- Se cambió de lugar el código donde se alojan las tablas $f_i(x_i)$, $f_{ij}(x_i, x_j)$, $f_{ijk}(x_i, x_j, x_k)$ a un lugar mas apropiado, ahora se alojan cuando el usuario oprime el botón RUN.

2.- En la lectura de código se encontró un error, se corrigió de tal manera que las condiciones iniciales cambian hasta que se tienen todos los estados finales, así no se alteran los resultados.

- Fecha: 30 de Enero-2006.

Resultados:

1.- Analizando la rutina para encontrar el offset de los arreglos X (valores de la variable) e Y (f(x)), no estaba contemplado el caso en el que el valor a interpolar coincide con uno de los valores de las tablas, se agregó:

```
if (X[m] == p)
    return (Y[m]);
```

REFERENCIAS.

- [1] Aguilar A., *Modelo de Calidad del aire para la contaminación por ozono en la zona metropolitana de la ciudad de México*. Ph.D. thesis, Universidad Politécnica de Madrid (1998).
- [2] Aliş, Ö. and Rabitz, H. *General Foundations of High Dimensional Model Representations*, J. Math. Chem., **25**, 197-233 (1999).
- [3] Aliş, Ö. and Rabitz, H. *Efficient implementation of high dimensional model representations*, J. Math. Chem., **29**, 127-142 (2001).
- [4] Aliş Ömer Faruk. *Inverse Problems in Quantum Mechanics*, PhD Dissertation. Princeton University January 2001.
- [5] Anton & Howard. *Introducción al álgebra lineal* . 3era ed., Ed. Limusa, pags 375-384, México (2003).
- [6] Blanchard P, Devaney R. & Hall G. *Ecuaciones Diferenciales* . Ed. International Thomson, México y América Central (1999).
- [7] Borelli R. & Coleman C. *Ecuaciones Diferenciales una perspectiva de modelación*. Ed. Oxford University press, México (2002).
- [8] Calvo H. *Algoritmos de expansión de datos aplicados a la contaminación atmosférica de la Ciudad de México* Tesis Maestria. Universidad Nacional Autónoma de México (2002).
- [9] Cohen, Scott and Hindmarsh, Alan. *CVODE User Guide* . Lawrence Livermore National Laboratory, Numerical Mathematics Group, UCRL-MA-118618, September (1994).
- [10] Cohen S. and Hindmarsh A.. *CVODE, a Stiff/Nonstiff ODE Solver in C*, *Computers in Physics*, **10**, 138-143 (1996).
- [11] Cukier R.I., Levine H.B. & Schuler K.E., *Nonlinear sensivity analysis of multiparameter model systems*. Journal of Computational Physics, **26**, 1-42 (1978).
- [12] Devore J. L. *Probabilidad y estadística para ingeniería y ciencias*. Ed International Thomson editores. 4a ed., México (1998).
- [13] Friedman J. and Stuetzle W. *Projection Pursuit Regression*. Journal of the American Statistical Association, Volume **76**, number 376, 817-823, december (1981).
- [14] Geremia J.M., H Rabitz H and Rosenthal C.. *Constructing global functional maps between molecular potentials and quantum observables*. Journal of Chem. Phys., **114**, 9325-9336(2001).
- [15] Geremia J.M. and Rabitz H.. *Global, nonlinear algorithm for inverting quantum-mechanical observations*. Phys. Rev. A, **64**, 022710-1-13 (2001).

- [16] Gilb T. and Graham D. *Software Inspection*, Ed. Adison-Wesley, 4^a ed., England (1998).
- [17] Gómez M.C., Tchijov V, León F. & Aguilar A. *FEOM: A tool to improve the execution time of the air quality models*. Environmental Modelling & Software, (in revision (minor)), 2007.
- [18] Gómez M.C., Tchijov V, León F. & Aguilar A. *Application of FEOM to chemical reaction mechanisms used in the 3D model of diffusion/advection of pollutants*. Lecture Series on Computer and Computational Science. Volume 7A p.p. 165-168. Greece, (2006).
- [19] Gómez M.C. & Tchijov Vladimir. *Application of an efficient solver for the atmospheric pollution problems*. Ninth PanAmerican Congress of Applied Mechanics(PACAM IX), pag. 72, Mexico January 2-6, (2006).
- [20] Gómez M.C. & Tchijov Vladimir. *Application of the Fully Equivalent Operational Model to the Basic Air Pollution Mathematical Model*. 1st ISSCultArt Meeting on the Classification, Characterization, Analysis, Synthesis, Modeling, Simulation and Optimization of Complex Systems, p.p. 1-9, Mexico January 24-27, (2006).
- [21] Gómez M.C. & Tchijov Vladimir. *An efficient model to solve highdimensional differential equations*. The first international conference on nonlinear, fractional and stochastic dynamics and the first international workshop on complex systems, applied category theory, fractals, fractional calculus and chaos, p.p. 1-8, Mexico January 24-28, (2005).
- [22] Gómez M.C. *Elaboración de un sistema de cómputo integrado que funciona en red, para la enseñanza de métodos numéricos usando el proceso de desarrollo de software: en espiral*. M.D. tesis, Universidad Nacional Autónoma de México (2004).
- [23] Hughes Deborah et. al., *Calculo de varias variables* ,1a ed, Compañía Editorial Continental S.A. de C.V., México (1998).
- [24] Jazcilevich A., *Locally induced surface air confluence by complex terrain and its effects on air pollution in the valley of Mexico* , Atmospheric Environment, **29**, 5481-5489 (2005).
- [25] León Frida, *Análisis Comparativo de los Mecanismos de Reacción y Cinética Química de los Compuestos Orgánicos Volátiles para la Formación de SMOG Fotoquímico en la Vida de México*. Ph.D. tesis, Universidad Politécnica de Madrid (1998).
- [26] Li Genyuan, C. Rosenthal and Hershel Rabitz. *High Dimensional Model Representations*. Journal of Physical Chemistry A, **105**, 7765-7777, (2001).
- [27] Li G., Wang SW, and Rabitz H. *High Dimensional Model Representations(HDMR): Concepts and applications*. Department of Chemistry, Princeton University. <http://silver.ima.umn.edu/talks/workshops/3-15-19.200/li/hdmr.pdf> (2000).

- [28] Li Genyuan, Sheng-Wei Wang and Hershel Rabitz. *Practical Approaches To Construct RS-HDMR Component Functions*. Journal of Physical Chemistry A, **106**, 8721-8733 (2002).
- [29] Li Genyuan, Sheng-Wei Wang , Hershel Rabitz, Sookyun Wang, Peter Jaffé. *Global uncertainty assessments by high dimensional model representations (HDMR)*. Chemical Engineering Science, **57**, 4445-4460 (2002).
- [30] Li Genyuan, Schoendorf J., Tak-san Ho and Hershel Rabitz. *Multicut-HDMR with an Application to an Ionospheric Model.*” Journal of Computational Chemistry, Vol **25**, No. 9, 1149-1156(2004).
- [31] Lorentz, M.V. Golitschek, Y. Makovoz. *Constructive Approximation*. Springer, New York (1996).
- [32] Micchelli C.A., *Interpolation of scattered data: Distance matrices and conditionally positive definite functions*, Constr. Approx., **59**, 11-22. (1986).
- [33] Nagornov O. V., Sokolov E. S. and Tchijov V. E., *Indirect determination of the turbulent diffusion coefficient*, J. Eng. Phys. Thermophys., Vol. 76, pp. 417-423, (2003).
- [34] Parker D., *Learning logic*. Center for Computational Research in Economics and Management Science. Working Paper 47. Massachusetts Institute Technology, (1985).
- [35] Poggio Tomaso and Girosi Federico. *Networks for Approximation and Learning*. Proceedings of the Institute of Electrical and Electronics Engineers (IEEE). Vol.78, No. 9, 1481-1497, september (1990).
- [36] Press, W. H. Teukolsky, S. A. Vetterling, W. T. and Flannery, B. P., *Numerical Recipes in C* ,The art of scientific computing. Cambridge University Press, New York, (1999).
- [37] Pressman, Roger, *Ingeniería del software, un enfoque práctico*, 4ª Edición, Ed. McGraw-Hill., (1998).
- [38] Rabitz Hershel, Genyuan Li, Sheng-Wei Wang. *High Dimensional Model Representations (HDMR): Concepts and applications*. Department of Chemistry, Princeton University. Princeton, New Jersey 2000.
<http://Silver.ima.umn.edu/talks/workshops/3-15-19.2000/li/hdmr.pdf>.
- [39] Rabitz Hershel, Ö F. Alis, J. Shorter and K. Shim. *Efficient input-output model representations*, Computer Physics Communications, **117**, 11-20 (1998).
- [40] Rabitz Hershel and K. Shim. *Multicomponent semiconductor material discovery guided by a generalized correlated function expansion*, Journal of Chem. Phys., **111**, 10640-10651 (1999).

- [41] Saltelli & Sobol I.M., *About the use of rank transformation in sensitivity analysis of model output*. Reliability Engineering and System Safety, **50**, 225-239 (1995).
- [42] Schreider, *The Monte Carlo Method*. Oxford Pergamon Press (1967).
- [43] Schoendorf J., Rabitz H., Genyuan Li, *A fast and accurate operational model of ionospheric electron density*. Geophysical Research Letters, **30**, No. 9, 45-1—45-4 (2003).
- [44] Seinfeld J. H. *Atmospheric chemistry and physics of air pollution*, J. Wiley, N.Y., (1986).
- [45] Shim K. and Rabitz H., *Independent and correlated composition behavior of material properties: Application to energy band gaps for the $Ga_{\alpha}In_{1-\alpha}P_{\beta}As_{1-\beta}$ and $Ga_{\alpha}In_{1-\alpha}P_{\beta}Sb_{\gamma}As_{1-\beta-\gamma}$ alloys*, Phys. Rev. B, **58**, 1940-1946 (1998).
- [46] Shorter J., Percila Ip and Rabitz H. *An efficient Chemical Kinetics Solver using High Dimensional Model Representation*, J. Phys. Chem. A, **103**, 7192-7198 (1999).
- [47] Shorter Jeffrey and Percila Ip, *Radiation transport simulation by means of a fully equivalent operational model*, Geophysical Research Letters, **27**, No. 21, 3485-3488 (2000).
- [48] SIMAT. Sistema de Monitoreo Atmosférico de la Ciudad de México. <http://www.sma.df.gob.mx/simat/pnozon.htm>.
- [49] Smith W. Allen, *Análisis Numérico*, Ed. Prentice-Hall, (1998).
- [50] Sobol I.M., *Sensitivity estimates for nonlinear mathematical models*. Mathematical Modeling and Computational Experiments, **1**, 407-414 (1993).
- [51] Sommerville, Ian, *Software Engineering*, 3^a Edición, Ed. Addison-Wesley, (1989).
- [52] Stone C.J., *Optimal global rates of convergence for non-parametric regression* Annals of Statistics, **10**, 1040-1053 (1982).
- [53] Tunga, M.A., Demiralp, M.. *Hybrid high dimensional model representation (HDMR) on the partitioned data*. Journal of Computational and Applied Mathematics(185) issue 1, 107-132, (2006).
- [54] Wang S.W., Levy H., Li G. and Rabitz H., *Fully Equivalent Operational Models for Atmospheric Chemical Kinetics within Global Chemistry-Transport Models*. J. Geophys. Res., **104**, 30, 417-30, 426 (1999).
- [55] Wang S.W., Georgopoulos P.G. and Rabitz H., *Computationally Efficient Atmospheric Chemical Kinetic modeling by means of High Dimensional Model*

- Representations (HDMR)*. Lecture Notes in Computer Science, Springer Berlin/Heidelberg, ISSN 0302-9743, **2179**, 326-333 (2001).
- [56] Wang, S.W., Balakrishnan, S., Georgopoulos, P.. *Fast Equivalent Operational Model of Tropospheric Alkane Photochemistry*. AIChE Journal **51**, 1297-1303 (2005).
- [57] Yanenko. *The Method of Fractional Steps*. SpringerVerlag 1971.