

# Métodos de Descomposición en Valores Singulares para Tensores

Tesis de Maestría presentada por

**Richard Rios Patiño**

ante la  
Facultad de Ciencias de la  
Universidad Nacional de Colombia  
para acceder al grado académico de

**Magister en Matemáticas**

Director:  
Ph.D, M.Sc Carlos E. Mejía

Codirector:  
Ph.D, M.Sc Jairo J. Espinosa

Medellín, 26 de septiembre de 2009



A mi madre, cuyo apoyo y compromiso durante mi existencia ha permitido que alcance las metas que me he propuesto.



# AGRADECIMIENTOS

Durante estos dos últimos años y medio, mi estancia en la Escuela de Matemáticas ha sido una gran experiencia tanto académica como profesional, y por tanto, quisiera agradecerle a la escuela por brindarme la oportunidad de aprender de profesores de una gran calidad humana y trayectoria académica; e igualmente, por brindarme la oportunidad de dar mis primeros pasos en la docencia.

Al profesor Carlos E. Mejía, por su colaboración y orientación durante la consecución de este trabajo, por su supervisión dentro y fuera del seminario de análisis numérico. Igualmente, quisiera agradecerles a los colegas que durante el último año participamos activamente en el seminario de análisis numérico.

Al profesor Jairo J. Espinosa, por su colaboración y orientación en estos primeros pasos de vida investigativa, por apoyarme en mi decisión de realizar esta maestría, y en orientarme y proponerme el tema de investigación para el trabajo de grado, el cual me ha permitido ser favorecido en dos ocasiones en la convocatoria de jóvenes investigadores de Colciencias.

Al profesor Jorge Cossio, por ser ese puente que permitio establecer una llave de trabajo entre la Facultad de Ciencias y la Facultad de Minas, al abrir espacios en los cuales se pudo discutir el tema de trabajo entre profesores de ambas facultades.

Al Grupo de Automática de la Universidad Nacional de Colombia (GAUNAL) y al grupo de Computación Científica, cuyos aportes y recomendaciones a este trabajo de investigación han sido importantes.

Y por último, quisiera agradecerles a mis compañeros de la maestría, por brindarme su apoyo y amistad durante estos últimos años, a los cuales, les deseo muchos exitos en su vida académica y profesional.



# Índice general

<b>1. INTRODUCCIÓN</b>	<b>9</b>
1.1. Organización del texto de tesis. . . . .	10
<b>2. INTRODUCCIÓN AL ALGEBRA MULTILINEAL</b>	<b>13</b>
2.1. Definición de tensor de orden superior . . . . .	13
2.2. Operaciones Básicas de tensores . . . . .	15
2.3. Producto escalar, ortogonalidad y norma . . . . .	20
2.4. Representación matricial de un tensor de orden superior . . . . .	21
2.5. Rango de un tensor de orden superior . . . . .	23
2.6. Conclusiones . . . . .	24
<b>3. DESCOMPOSICIÓN EN VALORES SINGULARES DE ORDEN SUPERIOR</b>	<b>25</b>
3.1. Introducción . . . . .	25
3.2. Descomposición en Valores Singulares de Orden Superior . . . . .	26
3.3. Propiedades de la HOSVD . . . . .	34
3.4. Cálculo de la HOSVD . . . . .	35
3.5. Conclusiones . . . . .	40
<b>4. MEJOR APROXIMACIÓN DE RANGO-1 Y DE RANGO-<math>(R_1, R_2, \dots, R_N)</math></b>	<b>43</b>
4.1. Método de la potencia y de la iteración ortogonal . . . . .	43
4.1.1. Método de la potencia . . . . .	44
4.1.2. Método de la iteración ortogonal . . . . .	45

---

4.2. Método de la potencia de orden superior . . . . .	45
4.2.1. La mejor aproximación de rango uno . . . . .	46
4.2.2. Generalización del método de la potencia para tensores de orden superior	47
4.3. Iteración ortogonal de orden superior . . . . .	48
4.3.1. Mejor aproximación de <i>rango</i> – $(R_1, \dots, R_N)$ . . . . .	48
4.3.2. Iteración ortogonal de orden superior . . . . .	50
4.4. Proyección por secciones . . . . .	52
4.5. Conclusiones . . . . .	60
<b>5. EL USO DE DESCOMPOSICIÓN DE TENSORES EN REDUCCIÓN DE</b>	
<b>MODELOS</b>	<b>61</b>
5.1. Introducción . . . . .	62
5.2. Reducción de modelos . . . . .	64
5.2.1. Perfil de temperatura de una barra unidimensional . . . . .	64
5.2.2. Perfil de temperatura de una placa . . . . .	67
5.3. SVD para tensores en reducción de modelos . . . . .	68
5.4. Conclusiones . . . . .	72



# Índice de figuras

2.1. Secciones de un tensor de orden tres . . . . .	15
2.2. Ejemplo de producto externo de tres vectores del ejemplo 2.3. . . . .	16
2.3. Producto interno entre un tensor $\mathcal{A}$ y $\mathbf{B}$ del ejemplo 2.5. . . . .	17
2.4. Esquema de producto <i>2-modal</i> entre un tensor $\mathcal{A}$ y una matriz $\mathbf{B}$ . . . . .	18
2.5. Esquema de la multiplicación de un tensor $\mathcal{B} \in \mathbb{C}^{J_1 \times J_2 \times J_3}$ y matrices $\mathbf{U}^{(i)} \in \mathbb{C}^{J_i \times I_i}$ ( $1 \leq i \leq 3$ ). . . . .	20
2.6. <i>Despliega</i> de un tensor $\mathcal{A}$ de orden $I_1 \times I_2 \times I_3$ a una matriz $\mathbf{A}_{(1)}$ de orden $I_1 \times I_2 I_3$ , $\mathbf{A}_{(2)}$ de orden $I_2 \times I_3 I_1$ , $\mathbf{A}_{(3)}$ de orden $I_3 \times I_1 I_2$ . . . . .	22
2.7. Visualización del tensor $\mathcal{A}$ del ejemplo 2.7 . . . . .	22
3.1. Esquema de la SVD matricial . . . . .	27
3.2. Esquema de la HOSVD de un tensor de orden tres . . . . .	31
3.3. Esquema de la descomposición diádica para un tensor de orden tres. . . . .	34
4.1. Evolución de la norma de la aproximación $\hat{\mathcal{A}}$ durante las iteraciones del método de la potencia . . . . .	55
4.2. Evolución de la norma de la aproximación $\hat{\mathcal{A}}$ durante las iteraciones del método de la iteración ortogonal . . . . .	56
4.3. Evolución de la norma de la aproximación $\hat{\mathcal{A}}$ durante las iteraciones del método SP. . . . .	57
4.4. Evolución de la norma de la aproximación $\hat{\mathcal{A}}$ durante las iteraciones del método de la potencia. . . . .	58
4.5. Evolución de la norma de la aproximación $\hat{\mathcal{A}}$ durante las iteraciones del método de la iteración ortogonal . . . . .	59

---

4.6. Evolución de la norma de la aproximación $\widehat{\mathcal{A}}$ durante las iteraciones del método SP. . . . .	59
5.1. Barra unidimensional. Las señales $u_1(t)$ , $u_2(t)$ y $u_3(t)$ son las condiciones de frontera en $x = 0$ , $x = L/2$ y $x = L$ respectivamente. . . . .	65
5.2. Funciones POD Obtenidas para la barra unidimensional . . . . .	66
5.3. Placa bidimensional. Las señales $u_1(t)$ , $u_2(t)$ , $u_3(t)$ y $u_4(t)$ son las condiciones de frontera en $(0, 1/4L)$ , $(0, 1/2L)$ , $(0, 3/4L)$ y $(L, 0)$ respectivamente. . . . .	67
5.4. Representación tensorial de la matriz $Z_{snap}$ del perfil de temperatura de un placa. El tensor $\mathcal{A}$ es de orden 3 y de tamaño $m \times s \times n$ . . . . .	69
5.5. Función de base bidimensional de la placa dada por $U_1^{(1)} \circ U_1^{(3)}$ . . . . .	70
5.6. Función de base bidimensional de la placa dada por $U_2^{(1)} \circ U_3^{(3)}$ . . . . .	70
5.7. Función de base bidimensional de la placa dada por $U_3^{(1)} \circ U_1^{(3)}$ . . . . .	71

# Capítulo 1

## INTRODUCCIÓN

En diversas disciplinas han surgido un número creciente de aplicaciones donde se aprecia la necesidad de manipular datos numéricos descritos en múltiples dimensiones, dichas cantidades u objetos son conocidos como tensores de orden superior. En esta tesis, tensores de orden superior pueden ser pensados de una manera intuitiva como una generalización multilineal de matrices (tensores de orden dos) y de vectores (tensores de orden uno), es decir, se puede considerar como un bloque de números en tres o más direcciones. Formalmente, un tensor es una función multilineal definida sobre un conjunto finito de espacios vectoriales, es decir, una función lineal sobre cada uno de los espacios vectoriales; y, son los objetos de estudio del algebra multilineal, el cual es la generalización del algebra lineal.

La Descomposición en Valores Singulares (SVD-Singular Value Decomposition), ha sido uno de los grandes desarrollos aportados por el algebra lineal moderna con importantes aplicaciones en campos como la visión artificial, recuperación de información en bases de datos, computación lingüística, procesamiento de señales e imágenes, entre otros. Sin embargo, esta herramienta está limitada a arreglos de dos dimensiones lo que limita su aplicación a sistemas descritos por datos en más de dos dimensiones. Durante las últimas décadas varios matemáticos han desarrollado extensiones de SVD capaces de operar en estructuras de orden superior. Entre las generalizaciones desarrolladas hasta el momento se pueden mencionar las siguientes: Higher-Order Singular Value Decomposition (HOSVD) [19], Higher-Order Orthogonal Iteration (HOOI) [20], Slice Projection (SP) [32], Multislice Projection (MP) [28], Tucker Decomposition [29], PARAFAC [14].

Posibles aplicaciones donde se podrían utilizar aproximaciones de bajo rango de tensores de orden superior, se pueden encontrar casi que en cualquier aplicación donde la SVD haya jugado un papel importante; de hecho, muchas aplicaciones reales que actualmente se describen mediante arreglos bidimensionales, se pueden describir a través de una estructura multidimensional. Revisiones bibliográficas sobre posibles aplicaciones, se pueden encontrar en [1], [16] y en [28]; entre tales aplicaciones, se pueden mencionar las siguientes: procesamiento de señales, búsqueda de información en bases de datos, algebra lineal numérica, visión computacional, minería de datos, análisis de imágenes, análisis de redes sociales y reducción de

modelos, entre otros. Dentro de procesamiento de señales, el interés en aproximaciones de bajo rango de tensores, radica en recuperar ó aproximar señales a partir de un número finito de mediciones, usando descomposición espectral de señales ([33],[34]).

Con respecto a reducción de modelos utilizando la técnica de Descomposición Ortogonal Propia (POD-*Proper Orthogonal Decomposition*), se ha despertado un gran interés en descomposición de tensores, dado que este método utiliza descomposición espectral de señales y proyecciones de Galerkin para proyectar la solución de un conjunto de ecuaciones diferenciales ordinarias sobre un conjunto de funciones de base obtenidas a través de datos de simulación o experimentales. Específicamente, el interés radica en reducir el número de funciones de base necesarias para obtener una buena aproximación de una solución a tal conjunto de ecuaciones, almacenando los datos de simulación o experimentales en arreglos multidimensionales ([25]).

En esta tesis, se considerará el problema de obtener aproximaciones de bajo rango de tensores de orden superior, donde en el caso matricial ha sido bien planteado y solucionado a través de la truncación de su descomposición diádica. Específicamente, se estudiaron las descomposiciones HOSVD, HOOI y SP, y adicionalmente, se desarrollaron rutinas numéricas en Matlab, con el fin de emplearlas en reducción de modelos usando el enfoque POD. A través de dos ejemplos, las rutinas fueron validadas con rutinas desarrolladas por Bader y Kolda [7] en el *toolbox* de tensores de Matlab, versión 2.2<sup>1</sup>.

## 1.1. Organización del texto de tesis.

Esta tesis esta organizada en cuatro capítulos, donde tres de ellos están dedicados al problema de obtener aproximaciones de bajo rango de tensores de orden superior, y el último, a una aplicación de descomposición de tensores en reducción de modelos.

En el capítulo dos, se hace una breve revisión sobre conceptos y operaciones básicas de algebra multilineal, donde se da una definición formal del concepto de tensor; e igualmente, se hacen algunos comentarios sobre la notación que se utilizará a lo largo de la tesis, y discuten algunas propiedades geométricas que se desprenden de algunas operaciones básicas que se definen el capítulo.

En el capítulo tres, se presenta una generalización multilineal de la Descomposición en Valores Singulares para tensores de orden superior, desarrollada por De Lathauwer (2000a), el cual es denotada por HOSVD (*Higher Singular Value Decomposition*). Y adicionalmente, se discuten algunas generalizaciones multilineales a propiedades de la SVD matricial.

En el capítulo cuatro, se abordan dos generalizaciones multilineales de la mejor aproxima-

---

<sup>1</sup>La versión más reciente de este toolbox corresponde a la 2.3, y se puede encontrar en la dirección <http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/>.

Cabe mencionar, que durante el desarrollo de las rutinas, se utilizó el servidor volador de la Facultad Ciencias.

---

ción de rango- $R$  de una matriz, llamadas HOOI y SP. La descomposición HOOI, intenta resolver la pregunta de cómo obtener la mejor aproximación de un tensor dado, en el sentido de mínimos cuadrados, por un tensor con rangos *n-modales* pre-especificados; y la descomposición SP, busca generalizar la mejor aproximación de rango- $R$ , al maximizar la mejor aproximación de las *secciones* del tensor original.

Por último en el capítulo cinco, se presenta una aplicación de descomposición de tensores en reducción de modelos usando la técnica de POD. El motivo principal de la introducción de este tipo de descomposiciones, radica en el deseo de reducir el número de funciones de base necesarias para obtener soluciones aproximadas de un modelo de gran dimensión.



## Capítulo 2

# INTRODUCCIÓN AL ALGEBRA MULTILINEAL

A continuación se presenta una breve introducción sobre conceptos y operaciones básicas del algebra multilinear, los cuales fueron obtenidos de De Lathauwer en [18]. Este capítulo esta organizado de la siguiente forma: en la sección 1.1, se da una definición formal de tensor dentro del contexto del algebra multilinear, obtenida de Van Belzen en [35], y el cual generaliza, dentro del algebra lineal, el concepto de que toda transformación lineal define una matriz y viceversa. No obstante, cabe resaltar que el concepto de tensor varía según el dominio científico en que se encuentre, y por lo tanto, no debe confundirse con la definición dada en el campo análisis tensorial, y que usualmente aparece en aplicaciones de la física y de la ingeniería. Igualmente en esta sección, se hacen algunos comentarios sobre la notación que se utilizará a través del texto, con el fin de facilitar la diferenciación entre escalares, vectores, matrices y tensores de orden superior. En la sección 1.2, se definen algunas operaciones básicas de tensores de orden superior, como el producto interno y externo de tensores, y en la sección 1.3, algunos conceptos geométricos relacionados con estas operaciones como son: el producto escalar, ortogonalidad y norma de Frobenius. En la sección 1.4, se establece una convención para representar tensores de orden superior a través de matrices, el cual es necesario para expresar las descomposiciones tensoriales en términos de herramientas matriciales. Por último en la sección 1.5, se aborda el concepto de rango de un tensor de orden superior, donde a la par, se generaliza el concepto de rango columna y rango fila de una matriz.

### 2.1. Definición de tensor de orden superior

**Definición 2.1** (*Tensor de orden  $N$* ) Sean  $V_1, \dots, V_N$  espacios vectoriales con producto interno. Un tensor de orden  $N$  es un funcional lineal

$$T : V_1 \times \dots \times V_N \longrightarrow \mathbb{R} \quad (2.1)$$

definido sobre los espacios vectoriales  $V_1, \dots, V_N$ , el cual, es lineal sobre cada uno de sus  $N$  argumentos. En el caso finito dimensional, donde  $\dim(V_n) = I_n$ ,  $T$  es considerado como un tensor de orden  $I_1 \times \dots \times I_N$  y de dimensión  $n$ -modal  $I_n$ . Considerando una colección de bases ortonormales para  $V_1, \dots, V_N$  dada por

$$\{e_1^{i_1}, i_1 = 1, \dots, I_1\}, \dots, \{e_N^{i_N}, i_N = 1, \dots, I_N\} \quad (2.2)$$

entonces el tensor  $T$  define un arreglo  $N$  dimensional  $[[t_{i_1, \dots, i_N}]] \in \mathbb{R}^{I_1 \times \dots \times I_N}$ , cuyos elementos están dados por  $t_{i_1, \dots, i_N} = T(e_1^{i_1}, \dots, e_N^{i_N})$ . Lo anterior, quiere decir que el tensor  $T$  es representado por sus elementos con respecto a la colección de bases dada en la Ec. 2.2. El espacio generado por todo los tensores de orden  $N$  es llamado el espacio producto tensorial de  $V_1, V_2, \dots, V_N$ .

Para  $V_n = \mathbb{R}^{I_n}(\mathbb{C}^{I_n})$  ( $1 \leq n \leq N$ ), el espacio producto tensorial es llamado el espacio tensorial real valuado (complejo valuado) de dimensión  $I_1 \times I_2 \times \dots \times I_N$ , y es denotado por  $\mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  ( $\mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ ). En el caso finito dimensional, de una manera intuitiva se puede considerar un tensor de orden  $N$  como un arreglo o bloque de datos de  $N$  índices, donde por ejemplo, un tensor de orden uno es un vector, uno de orden dos es una matriz y uno de orden tres un paralelepípedo. Un tensor de orden superior se refiere a un arreglo de tres o más dimensiones. A continuación, con el fin de ilustrar esta definición presentamos los siguientes dos ejemplos.

**Ejemplo 2.1** Sean  $V_1, \dots, V_N$  espacios vectoriales con producto interno. Para una colección fija de elementos  $u^{(n)} \in V_n$  distintos de cero, para  $n = 1, \dots, N$ , el funcional

$$U(v^{(1)}, \dots, v^{(N)}) := \langle u^{(1)}, v^{(1)} \rangle_1 \dots \langle u^{(N)}, v^{(N)} \rangle_N$$

donde  $\langle \cdot, \cdot \rangle_n$  representa el producto interno de  $V_n$ , define un tensor de orden  $N$ , cuyos elementos están dados por

$$u_{i_1 \dots i_N} = u_{i_1}^{(1)} \dots u_{i_N}^{(N)}, \quad \text{donde } u_{i_n}^{(n)} = \langle u^{(n)}, e_{i_n}^{(n)} \rangle_n$$

**Ejemplo 2.2** Sea  $N = 2$ ,  $V_1 = \mathbb{R}^m$  y  $V_2 = \mathbb{R}^n$ . Considere una matriz  $\mathbf{A} \in \mathbb{R}^{m \times n}$  asociada al siguiente funcional bilineal  $T : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}$  definido por

$$T(u, v) := \langle u, \mathbf{A}v \rangle = \langle \mathbf{A}^T u, v \rangle$$

los elementos de  $T$  con respecto a las bases canónicas de  $\mathbb{R}^m$  y  $\mathbb{R}^n$ , están dados por los coeficientes de la matriz  $\mathbf{A}$ , es decir, que  $t_{i_1, i_2} = a_{ij} = \langle e_i, \mathbf{A}e_j \rangle$ .

## Notación

Antes de empezar la siguiente sección, se realizarán algunos comentarios sobre la notación que utilizará a través del texto. Con el fin de facilitar la distinción entre escalares, vectores,



matrices y tensores de orden superior, se identificará el tipo de cantidad según el tipo de letra utilizada<sup>1</sup>: escalares se representan con letra minúscula ( $a, b, \alpha, \lambda, \dots$ ), vectores son representados con letra itálica ( $A, B, \dots$ ), matrices son denotadas con letra en negrilla ( $\mathbf{A}, \mathbf{B}, \dots$ ), tensores de orden superior son denotadas con letra *calligraphic* ( $\mathcal{A}, \mathcal{B}, \dots$ ). De acuerdo con lo anterior, el  $i$ -ésimo elemento del vector  $A$  es denotado por  $(A)_i = a_i$ ; la componente de la  $i$ -ésima fila y  $j$ -ésima columna de la matriz  $\mathbf{A}$  es denotada por  $(\mathbf{A})_{ij} = a_{ij}$ ; la  $i$ -ésima columna de la matriz  $\mathbf{A}$  es denotada por  $\mathbf{A}_i = a_{:i}$  y la  $i$ -ésima fila de la matriz  $\mathbf{A}$  es denotada por  $\mathbf{A}_{i:} = a_{i:}$ .

En tensores de orden superior, los análogos a los vectores columna y fila de una matriz, son llamados *fibras*. En una matriz, las columnas son las fibras *1-modales* (segundo índice fijo) y las filas son las fibras *2-modales* (primer índice fijo). En un tensor de orden tres  $\mathcal{A} \in C^{I_1 \times I_2 \times I_3}$ , las fibras son llamadas columnas, filas y tubos, las cuales son denotadas por  $\mathcal{A}_{:i_2i_3} = a_{:i_2i_3}$ ,  $\mathcal{A}_{i_1:i_3} = a_{i_1:i_3}$  y  $\mathcal{A}_{i_1i_2:} = a_{i_1i_2:}$  respectivamente; no existe nombre especial para las fibras de un tensor de orden cuatro o superior. Igualmente, las matrices de un tensor de orden tres son llamadas *secciones (slices)*, y son denotadas por  $\mathcal{A}_{i_1:}$  (*sección horizontal*),  $\mathcal{A}_{:i_2}$  (*sección lateral*) y  $\mathcal{A}_{::i_3}$  (*sección frontal*). En la Fig. 2.1, se muestran las secciones de un tensor de orden tres.

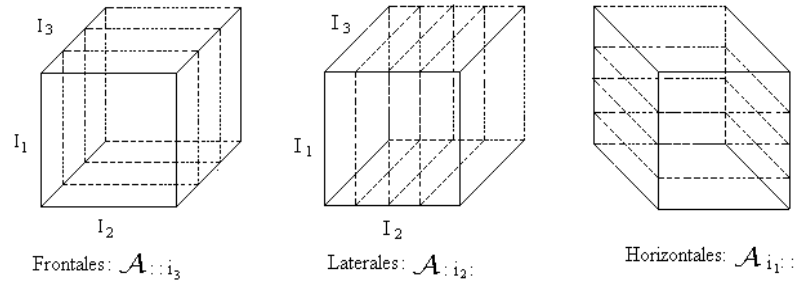


Figura 2.1: Secciones de un tensor de orden tres

## 2.2. Operaciones Básicas de tensores

El espacio producto tensorial real valuado o complejo valuado forman un espacio vectorial lineal, donde la definición de suma y multiplicación por un escalar se extiende naturalmente del caso matricial. A continuación, se definen el producto interno y externo entre tensores, y un caso particular del producto externo, llamado multiplicación entre un tensor y una matriz.

<sup>1</sup>En este texto se sigue la notación introducida por De Lathauwer en [18], [19],[20], anexándole algunas notaciones adicionales definidas por Kolda en [15]

**Definición 2.2 (Producto externo)** El producto externo de un tensor  $\mathcal{A} \in C^{I_1 \times I_2 \times \dots \times I_P}$  y un tensor  $\mathcal{B} \in C^{J_1 \times J_2 \times \dots \times J_Q}$ , es un tensor denotado por  $\mathcal{A} \circ \mathcal{B}$ , y definido componente a componente por

$$(\mathcal{A} \circ \mathcal{B})_{i_1 i_2 \dots i_P j_1 j_2 \dots j_Q} \stackrel{\text{def}}{=} a_{i_1 i_2 \dots i_P} b_{j_1 j_2 \dots j_Q} \quad (2.3)$$

para todos los valores de los índices.

**Ejemplo 2.3** Considere  $U^{(i)} \in \mathbb{R}^{I_i}$  ( $1 \leq i \leq 3$ ), el producto externo entre estos vectores define un tensor  $\mathcal{A}$  de orden tres, cuyas componentes están dadas por

$$a_{i_1 i_2 i_3} = u_{i_1}^{(1)} u_{i_2}^{(2)} u_{i_3}^{(3)}$$

donde  $i_j = 1, \dots, I_j$  ( $j = 1, 2, 3$ ). En la Fig.2.2, se muestra el tensor  $\mathcal{A}$  cuando  $U^{(1)} = \begin{bmatrix} 2 & 1 \end{bmatrix}^T$ ,  $U^{(2)} = \begin{bmatrix} 3 & 2 & 1 \end{bmatrix}^T$ ,  $U^{(3)} = \begin{bmatrix} 1 & -1 \end{bmatrix}^T$ .

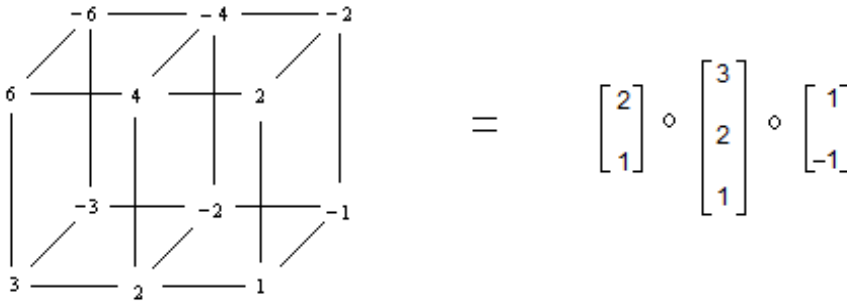


Figura 2.2: Ejemplo de producto externo de tres vectores del ejemplo 2.3.

**Definición 2.3 (Producto interno)** El producto interno  $\langle \mathcal{A}, \mathcal{B} \rangle_{p,q}$  de un tensor  $\mathcal{A} \in C^{I_1 \times \dots \times I_P}$  y un tensor  $\mathcal{B} \in C^{J_1 \times J_2 \times \dots \times J_Q}$ , sobre los índices  $i_p = j_q = u$ , es definido por

$$\left( \langle \mathcal{A}, \mathcal{B} \rangle_{p,q} \right)_{i_1 i_2 \dots i_{p-1} i_{p+1} \dots i_P j_1 j_2 \dots j_{q-1} j_{q+1} \dots j_Q} \stackrel{\text{def}}{=} \sum_u a_{i_1 i_2 \dots i_{p-1} i_{p+1} \dots i_P} b_{j_1 j_2 \dots j_{q-1} j_{q+1} \dots j_Q} \quad (2.4)$$

De una forma general, se puede definir un producto interno sobre todos los posibles valores de los índices. Observe, que una condición implícita para la buena definición del producto es que  $I_p$  e  $I_q$  tengan el mismo valor.

**Ejemplo 2.4** Considere el tensor  $\mathbf{A} \in \mathbb{R}^{I_1 \times I_2} = \mathbb{R}^{2 \times 2}$  y  $\mathbf{B} \in \mathbb{R}^{J_1 \times J_2} = \mathbb{R}^{3 \times 2}$ , el producto interno sobre los índices  $i_1$  y  $j_2$  está dado por

$$\left(\langle \mathcal{A}, \mathcal{B} \rangle_{p,q}\right)_{i_2 j_1} = \sum_u a_{u i_2} b_{j_1 u}$$

observe que para determinados valores de  $i_2$  y  $j_1$  se establece el producto vectorial entre las columnas de  $\mathbf{A}$  y las filas de  $\mathbf{B}$ .

**Ejemplo 2.5** Considere el tensor  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3} = \mathbb{R}^{2 \times 3 \times 2}$  y  $\mathcal{B} \in \mathbb{R}^{J_1 \times J_2} = \mathbb{R}^{3 \times 2}$ , el producto interno sobre los índices  $i_1$  y  $j_2$  cuando  $i_2 = 1, i_3 = 2$  y  $j_1 = 3$ , esta dado por

$$\left(\langle \mathcal{A}, \mathcal{B} \rangle_{1,2}\right)_{123} = \sum_u a_{u 12} b_{3u} = a_{112} b_{31} + a_{212} b_{32}$$

observe que para determinados valores de  $i_2, i_3$  y  $j_1$  se establece el producto vectorial entre las columnas de  $\mathcal{A}$  y las filas de  $\mathcal{B}$ . Por otro lado, el producto interno sobre los índices  $i_2$  y  $j_1$  cuando  $i_1 = 2, i_3 = 1$  y  $j_2 = 1$ , esta dado por

$$\left(\langle \mathcal{A}, \mathcal{B} \rangle_{2,1}\right)_{211} = \sum_u a_{2u1} b_{u1} = a_{211} b_{11} + a_{221} b_{21} + a_{231} b_{31}$$

igualmente, para determinados valores de  $i_1, i_3$  y  $j_2$  se establece el producto vectorial entre las filas de  $\mathcal{A}$  y las columnas de  $\mathcal{B}$ . En la Fig. 2.3, se puede visualizar el tensor  $\mathcal{A}$  y el tensor  $\mathcal{B}$ .

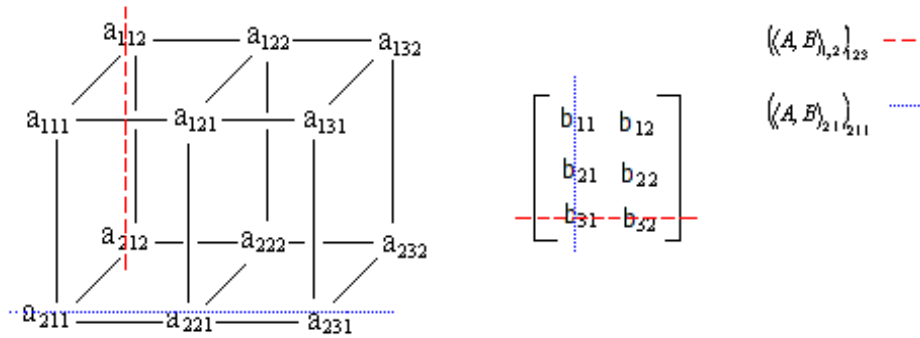


Figura 2.3: Producto interno entre un tensor  $\mathcal{A}$  y  $\mathcal{B}$  del ejemplo 2.5.

**Definición 2.4 (Multiplicación entre un tensor y una matriz)** El producto  $n$ -modal de un tensor  $\mathcal{A} \in C^{I_1 \times I_2 \times \dots \times I_N}$  y una matriz  $\mathbf{U} \in C^{J_n \times I_n}$ , denotado por  $\mathcal{A} \times_n \mathbf{U}$ , es un tensor de dimensión  $I_1 \times I_2 \times \dots \times J_n \times \dots \times I_N$  dado por

$$\mathcal{A} \times_n \mathbf{U} \stackrel{def}{=} \langle \mathcal{A}, \mathbf{U} \rangle_{n,2} \tag{2.5}$$

i.e.

$$(\mathcal{A} \times_n \mathbf{U})_{i_1 i_2 \dots j_n \dots i_N} = \sum_{i_n} a_{i_1 i_2 \dots i_n \dots i_N} b_{j_n i_n} \quad (2.6)$$

para todos los valores de los índices.

**Ejemplo 2.6** Considere el tensor  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3} = \mathbb{R}^{2 \times 3 \times 2}$  y  $\mathbf{B} \in \mathbb{R}^{J_1 \times J_2} = \mathbb{R}^{2 \times 3}$ , el producto 2-modal entre  $\mathcal{A}$  y  $\mathbf{B}$  denotado  $\mathcal{A} \times_2 \mathbf{B}$ , define un tensor  $\mathcal{C} \in \mathbb{R}^{2 \times 2 \times 2}$  cuyas componentes están dadas por

$$c_{i_1 i_2 i_3} = (\mathcal{A} \times_2 \mathbf{B})_{i_1 j_1 i_3} = \sum_{i_2} a_{i_1 i_2 i_3} b_{j_1 i_2}$$

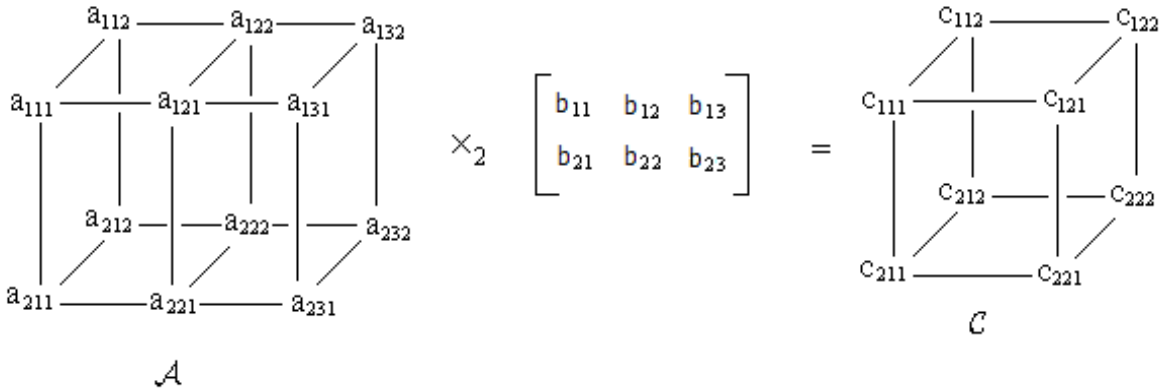


Figura 2.4: Esquema de producto 2-modal entre un tensor  $\mathcal{A}$  y una matriz  $\mathbf{B}$

**Corolario 2.1** Dado un tensor  $\mathcal{A} \in C^{I_1 \times I_2 \times \dots \times I_N}$  y la matriz  $\mathbf{F} \in C^{J_n \times I_n}$ ,  $\mathbf{G} \in C^{J_m \times I_m}$ , entonces

$$(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G} = (\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F} = \mathcal{A} \times_n \mathbf{F} \times_m \mathbf{G} \quad (2.7)$$

**Prueba.** Sea  $\mathcal{A} \in C^{I_1 \times I_2 \times \dots \times I_N}$ ,  $\mathbf{F} \in C^{J_n \times I_n}$  y  $\mathbf{G} \in C^{J_m \times I_m}$ , entonces claramente se cumple que  $(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G}$ ,  $(\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F}$ ,  $\mathcal{A} \times_n \mathbf{F} \times_m \mathbf{G} \in C^{I_1 \times \dots \times J_n \times \dots \times J_m \times \dots \times I_N}$ . Ahora sea

$c_{i_1 \dots i_n \dots j_m \dots i_N}$  una componente de  $(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G}$ , entonces se tiene que

$$c_{i_1 \dots i_n \dots j_m \dots i_N} = ((\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G})_{i_1 \dots j_n \dots j_m \dots i_N} \quad (2.8)$$

$$= \left( \langle \mathcal{A} \times_n \mathbf{F}, \mathbf{G} \rangle_{m,2} \right)_{i_1 \dots j_n \dots i_{m-1} i_{m+1} \dots i_N j_m} \quad (2.9)$$

$$= \sum_{i_m} (\mathcal{A} \times_n \mathbf{F})_{i_1 \dots j_n \dots i_m \dots i_N} \cdot g_{j_m i_m} \quad (2.10)$$

$$= \sum_{i_m} \left( \sum_{i_n} a_{i_1 \dots i_n \dots i_m \dots i_N} \cdot f_{j_n i_n} \right) \cdot g_{j_m i_m} \quad (2.11)$$

$$= \sum_{i_n} \left( \sum_{i_m} a_{i_1 \dots i_n \dots i_m \dots i_N} \cdot g_{j_m i_m} \right) \cdot f_{j_n i_n} \quad (2.12)$$

$$= \sum_{i_n} (\mathcal{A} \times_m \mathbf{G})_{i_1 \dots i_n \dots j_m \dots i_N} \cdot f_{j_n i_n} \quad (2.13)$$

$$= ((\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F})_{i_1 \dots j_n \dots j_m \dots i_N} \quad (2.14)$$

Luego se cumple que  $(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G} = (\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F}$ . De igual forma se muestra que  $\mathcal{A} \times_n \mathbf{F} \times_m \mathbf{G} = (\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F}$  ■

**Corolario 2.2** Dado un tensor  $\mathcal{A} \in C^{I_1 \times I_2 \times \dots \times I_N}$  y la matriz  $\mathbf{F} \in C^{J_n \times I_n}$ ,  $\mathbf{G} \in C^{K_n \times J_n}$ , entonces

$$(\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{G} = \mathcal{A} \times_n (\mathbf{G} \cdot \mathbf{F}) \quad (2.15)$$

**Prueba.** Sea  $\mathcal{A} \in C^{I_1 \times I_2 \times \dots \times I_N}$ ,  $\mathbf{F} \in C^{J_n \times I_n}$  y  $\mathbf{G} \in C^{K_n \times J_n}$ , entonces claramente se cumple que  $(\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{G}$ ,  $\mathcal{A} \times_n (\mathbf{G} \cdot \mathbf{F}) \in C^{I_1 \times \dots \times K_n \times \dots \times I_N}$ . Ahora sea  $c_{i_1 \dots k_n \dots i_N}$  una componente de  $(\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{G}$ , entonces se tiene que

$$c_{i_1 \dots k_n \dots i_N} = ((\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{G})_{i_1 \dots k_n \dots i_N} \quad (2.16)$$

$$= \sum_{j_n} (\mathcal{A} \times_n \mathbf{F})_{i_1 \dots j_n \dots i_N} \cdot g_{k_n j_n} = \sum_{j_n} \left( \sum_{i_n} a_{i_1 \dots i_n \dots i_N} \cdot f_{j_n i_n} \right) \cdot g_{k_n j_n}$$

$$= \sum_{i_n} a_{i_1 \dots i_n \dots i_N} \left( \sum_{j_n} g_{k_n j_n} \cdot f_{j_n i_n} \right) = \sum_{i_n} a_{i_1 \dots i_n \dots i_N} (\mathbf{G} \cdot \mathbf{F})_{k_n i_n}$$

$$= (\mathcal{A} \times_n (\mathbf{G} \cdot \mathbf{F}))_{i_1 \dots k_n \dots i_N}$$

Luego queda demostrado. ■

**Observación 2.1** La notación  $\times_n$  evita la necesidad de generalizar el concepto de transpuesta en el caso multilinear, es decir, considere las matrices  $\mathbf{F} \in \mathbb{R}^{I_1 \times I_2}$ ,  $\mathbf{U} \in \mathbb{R}^{J_1 \times I_1}$  y  $\mathbf{V} \in \mathbb{R}^{J_2 \times I_2}$ , el producto  $\mathbf{U} \cdot \mathbf{F} \cdot \mathbf{V}^T$  puede ser escrito como  $\mathbf{F} \times_1 \mathbf{U} \times_2 \mathbf{V}$ ; de esta forma, las columnas de  $\mathbf{U}$  están asociadas al espacio 1 – modal de  $\mathbf{F}$ , y las columnas de  $\mathbf{V}$  están asociadas al espacio 2 – modal de  $\mathbf{F}$ .

En la Fig. 2.5 para un tensor  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ ,  $\mathcal{B} \in \mathbb{C}^{J_1 \times J_2 \times J_3}$  y  $\mathbf{U}^{(i)} \in \mathbb{C}^{J_i \times I_i}$  ( $1 \leq i \leq 3$ ), se visualiza la ecuación  $\mathcal{A} = \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$ .

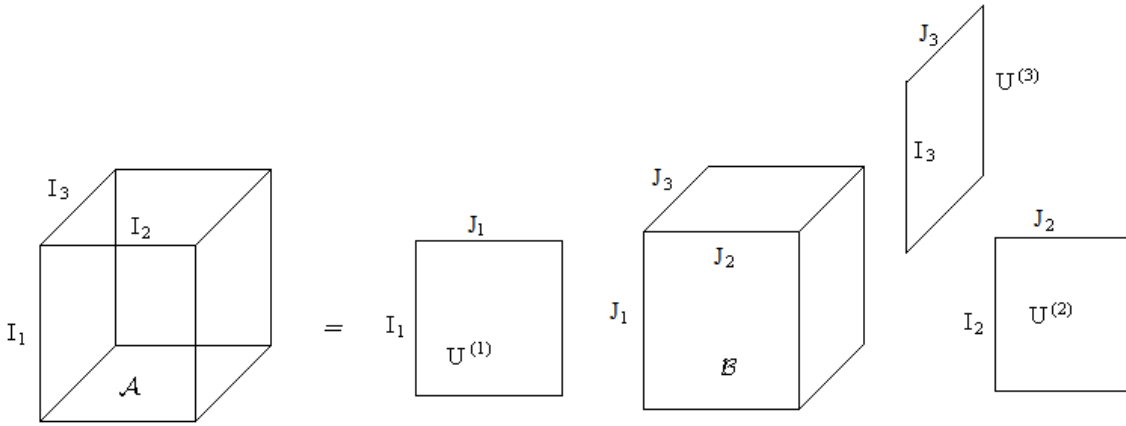


Figura 2.5: Esquema de la multiplicación de un tensor  $\mathcal{B} \in \mathbb{C}^{J_1 \times J_2 \times J_3}$  y matrices  $\mathbf{U}^{(i)} \in \mathbb{C}^{J_i \times I_i}$  ( $1 \leq i \leq 3$ ).

Observe que la multiplicación del tensor  $\mathcal{B}$  y la matriz  $\mathbf{U}^{(1)}$  implica que cada columna de  $\mathcal{B}$  tiene que ser multiplicada con cada fila de  $\mathbf{U}^{(1)}$ ; y de igual forma, se establece que cada fila de  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$  tiene que ser multiplicada con cada fila y tubo del tensor  $\mathcal{B}$ , respectivamente.

## 2.3. Producto escalar, ortogonalidad y norma

Con el fin de establecer propiedades geométricas en el espacio producto vectorial, es necesario generalizar definiciones como el producto escalar, ortogonalidad y norma de Frobenius. Se asumen bases ortogonales en  $\mathbb{C}^{I_1}, \mathbb{C}^{I_2}, \dots, \mathbb{C}^{I_N}$ .

**Definición 2.5 (Producto escalar)** El producto escalar  $\langle \mathcal{A}, \mathcal{B} \rangle$  de dos tensores  $\mathcal{A}, \mathcal{B} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  se define como

$$\langle \mathcal{A}, \mathcal{B} \rangle \stackrel{\text{def}}{=} \sum_{i_1} \sum_{i_2} \cdots \sum_{i_N} b_{i_1 i_2 \dots i_N}^* a_{i_1 i_2 \dots i_N} \quad (2.17)$$

Esta definición de producto escalar generaliza el producto clásico entre vectores y matrices, es decir, considere los vectores  $X, Y \in \mathbb{C}^{I_1}$  y las matrices  $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{I_1 \times I_2}$

$$\begin{aligned}\langle X, Y \rangle &= Y^H X \\ \langle \mathbf{A}, \mathbf{B} \rangle &= \sum_{i_1} \sum_{i_2} b_{i_1 i_2}^* a_{i_1 i_2} = \text{trace}(\mathbf{B}^H \mathbf{A})\end{aligned}\tag{2.18}$$

**Definición 2.6 (Ortogonalidad)** Dos tensores se dicen que son ortogonales si su producto escalar es igual a cero.

**Definición 2.7 (Norma de Frobenius)** La norma de Frobenius de un tensor  $\mathcal{A}$  es definida como

$$\|\mathcal{A}\| = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}\tag{2.19}$$

Según De Lathauwer, esta puede ser interpretada como la medida del "tamaño" del tensor.

## 2.4. Representación matricial de un tensor de orden superior

El concepto de representación matricial se definirá como *despliegue matricial* (*matriz unfolding*) [18], aunque algunos autores, como Kolda [15] lo definen como "*matricization*". Ahora, la importancia de introducir este concepto radica en que muchos de los problemas tensoriales son expresados en términos matriciales y solucionados en ese lenguaje, con el fin de aprovechar las herramientas y software matriciales desarrollados hasta el momento.

**Definición 2.8 (Despliegue matricial)** Supongamos que  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ , su matriz *despliegue*  $\mathbf{A}_{(n)} \in \mathbb{C}^{I_n \times I_{n+1} \dots I_N I_1 \dots I_{n-1}}$  es una matriz cuyas columnas son formadas al variar el índice  $i_n$  y al dejar los otros índices fijos, es decir, es una matriz cuyas columnas están formadas por las fibras ***n*-modales** del tensor  $\mathcal{A}$ . Las columnas de matriz  $\mathbf{A}_{(n)}$  son llamadas vectores *n*-modales y generalizan el concepto de vector fila y vector columna de una matriz.

Para un tensor de orden tres, esta operación se ilustra en la Fig. 2.6. En el caso de un tensor de orden tres, para evitar ambigüedades en la definición se utilizará una forma particular en el ordenamiento de las fibras del tensor; observe, que la definición de la matriz despliegue involucra de una manera cíclica las dimensiones del tensor  $I_1, I_2$  y  $I_3$ , por lo tanto, cuando se trata de obtener una matriz *despliegue* de dimensión  $I_c \times I_a \times I_b$ , se supone que el índice  $i_a$  varía más lentamente que  $i_b$ .

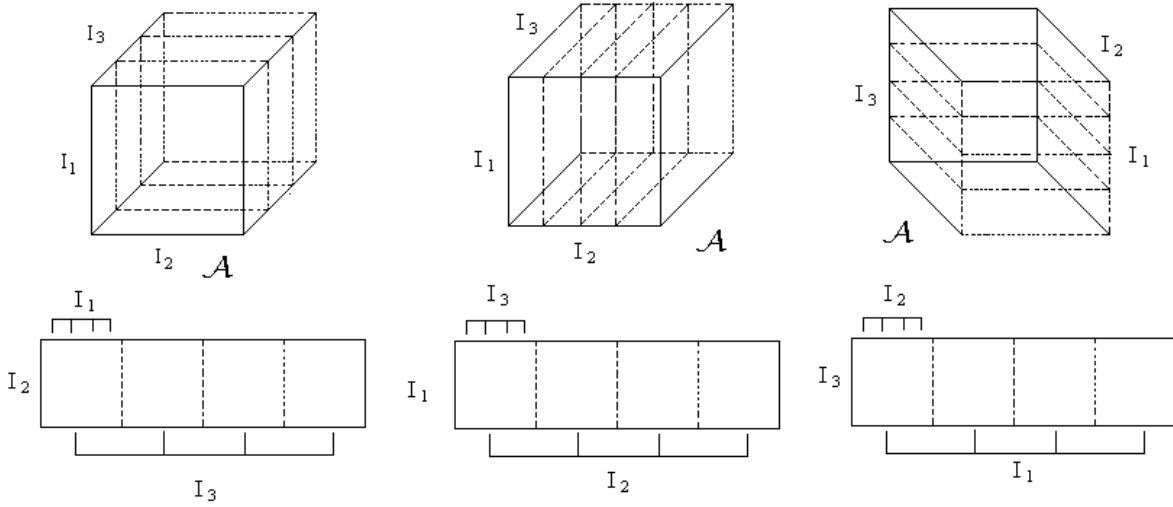


Figura 2.6: Despliegue de un tensor  $\mathcal{A}$  de orden  $I_1 \times I_2 \times I_3$  a una matriz  $\mathbf{A}_{(1)}$  de orden  $I_1 \times I_2 I_3$ ,  $\mathbf{A}_{(2)}$  de orden  $I_2 \times I_3 I_1$ ,  $\mathbf{A}_{(3)}$  de orden  $I_3 \times I_1 I_2$ .

**Ejemplo 2.7** Considere el tensor  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$  cuyas componentes están dadas por  $a_{111} = -1$ ,  $a_{121} = a_{211} = a_{231} = a_{222} = a_{232} = 2$ ,  $a_{131} = a_{112} = a_{132} = 1$ ,  $a_{221} = a_{212} = 0$  y  $a_{122} = 3$ . Las matrices despliegue  $\mathbf{A}_{(1)}$ ,  $\mathbf{A}_{(2)}$  y  $\mathbf{A}_{(3)}$  están dadas por

$$\mathbf{A}_{(1)} = \begin{bmatrix} -1 & 1 & \vdots & 2 & 3 & \vdots & 1 & 1 \\ 2 & 0 & \vdots & 0 & 2 & \vdots & 2 & 2 \end{bmatrix} \quad \mathbf{A}_{(2)} = \begin{bmatrix} -1 & 2 & \vdots & 1 & 0 \\ 2 & 0 & \vdots & 3 & 2 \\ 1 & 2 & \vdots & 1 & 2 \end{bmatrix}$$

$$\mathbf{A}_{(3)} = \begin{bmatrix} -1 & 2 & 1 & \vdots & 2 & 0 & 2 \\ 1 & 3 & 1 & \vdots & 0 & 2 & 2 \end{bmatrix}$$

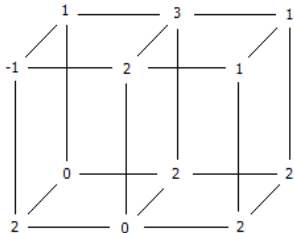


Figura 2.7: Visualización del tensor  $\mathcal{A}$  del ejemplo 2.7



## 2.5. Rango de un tensor de orden superior

Existen grandes diferencias entre matrices y tensores de orden superior cuando se refiere al rango. Estas diferencias, marcarán fuertemente las diferencias entre la descomposición de un tensor de orden superior y una matriz.

**Definición 2.9 (Tensor de rango 1)** Un tensor  $\mathcal{A}$  de orden  $N$  tiene rango 1 si es igual al producto externo de  $N$  vectores  $U^{(1)}, U^{(2)}, \dots, U^{(N)}$

$$\mathcal{A} = U^{(1)} \circ U^{(2)} \circ \dots \circ U^{(N)}$$

**Definición 2.10 (Rango de un tensor)** El rango de un tensor  $\mathcal{A}$  de orden  $N$ , representado por  $R = \text{rank}(\mathcal{A})$ , es el mínimo número de tensores de rango 1 que permiten expresar a  $\mathcal{A}$  como una combinación lineal.

**Definición 2.11 (Rango  $n$ -modal de un tensor)** El rango  $n$ -modal de un tensor  $\mathcal{A}$  de orden  $N$ , representado por  $R_n = \text{rank}_n(\mathcal{A})$ , es la dimensión del espacio vectorial generado por los vectores  $n$ -modales.

De igual forma a los vectores  $n$ -modales, esta definición generaliza los conceptos de rango fila y columna de una matriz. Por otra parte, a diferencia del caso matricial, los rango  $n$ -modales de un tensor de orden superior no son necesariamente iguales, y en el caso de que sean iguales, estos pueden ser diferentes al rango del tensor.

**Proposición 2.1** Sea  $\mathcal{A}$  un tensor de orden  $N$  entonces se cumple que  $R_n \leq R$  ( $1 \leq n \leq N$ ).

**Prueba.** Sea  $\mathcal{A}$  un tensor de orden  $N$  de rango  $R$ , entonces se cumple que:

$$\mathcal{A} = \sum_{i=1}^R \mathcal{B}_i = \sum_{i=1}^R U^{(1),i} \circ U^{(2),i} \circ \dots \circ U^{(N),i}$$

donde  $\mathcal{B}_i = U^{(1),i} \circ U^{(2),i} \circ \dots \circ U^{(N),i}$ , con  $U^{(n),i} \in \mathbb{R}^{I_n}$  ( $1 \leq n \leq N$ ) ( $1 \leq i \leq R$ ). Para demostrar la hipótesis, basta mostrar que  $R_k \leq R$  para  $k$  arbitrario. Ahora sea  $\mathbf{B}_{(k),i}$  la matriz *despliega  $k$ -modal* del tensor  $\mathcal{B}_i$ , entonces se tiene que

$$\mathbf{B}_{(k),i} = [ c_{1,i}U^{(k),i} \quad \dots \quad c_{s,i}U^{(k),i} \quad \dots \quad c_{m,i}U^{(k),i} ]$$

donde  $c_{s,i} = u_{j_1} \dots u_{j_{k-1}} u_{j_{k+1}} \dots u_{j_N}$  para  $j_1 \in I_1, j_2 \in I_2, \dots, j_N \in I_N$  y  $m = I_1 \cdots I_{k-1} I_{k+1} \cdots I_N$ . Luego, se cumple que

$$\mathbf{A}_{(\mathbf{k})} = \sum_{i=1}^R \mathbf{B}_{(k),i}$$

Por tanto, la  $s$ -ésima columna de  $\mathbf{A}_{(\mathbf{k})}$  está dada por

$$c_{s,1}U^{(k),1} + \dots + c_{s,R}U^{(k),R} \in \text{span}(U^{(k),1}, U^{(k),2}, \dots, U^{(k),R})$$

Luego se cumple que  $\mathbf{R}(\mathbf{A}_{(k)}) \leq \text{span}(U^{(k),1}, U^{(k),2}, \dots, U^{(k),R})$ . Por lo tanto,

$$R_k = \dim(\mathbf{R}(\mathbf{A}_{(k)})) \leq \dim(\mathbf{R}(\text{span}(U^{(k),1}, U^{(k),2}, \dots, U^{(k),R}))) \leq R$$

■

## 2.6. Conclusiones

En este capítulo se abordaron algunos conceptos y operaciones básicas del algebra multilinear, los cuales son esenciales para desarrollo de esta tesis. Al comienzo del capítulo, se presentó una definición del concepto de tensor de orden superior, propia del algebra multilinear, con el fin de establecer el enfoque con el cual se utilizará este concepto; también, se definió una notación para el trabajo, el cual fue tomada casi en su totalidad por la convención utilizada en [18], [19],[20], pero se le agregaron algunas notaciones adicionales definidas por Kolda en [15], las cuales son necesarias para abordar en el capítulo 4, la descomposición *Proyección por Secciones (Slice Projection-SP)*.

Igualmente, se presentó una forma de representar matricialmente un tensor de orden superior, con el fin de aprovechar las ventajas que ofrecen las herramientas y software matriciales desarrollados hasta el momento. Para un tensor de orden  $N$ , tal formato matricial, es simplemente un conjunto de  $N$  matrices formadas al desplegar de  $N$  formas distintas los vectores  $n$ -modales del tensor; en este texto, se siguió el ordenamiento adoptado por De Lathauwer para apilar tales vectores, y así, evitar confusiones al momento de formar dichas representaciones matriciales. Dado lo anterior, cabe resaltar que algunos autores, como por ejemplo Kolda ([15]), siguen una forma de ordenamiento distinta a la adoptada en este texto, lo cual repercute en la no unicidad en la descomposición de un tensor dado, tal como se mostrará en algunos resultados obtenidos en el siguiente capítulo.

Por último, se generalizaron los conceptos de vector columna, vector fila, rango columna y fila de una matriz, a través de la definición de vectores (rango)  $n$ -modales; e igualmente, se abordó el concepto de rango de un tensor, y se demostró, que éste es siempre mayor ó igual a los rangos  $n$ -modales del tensor.

## Capítulo 3

# DESCOMPOSICIÓN EN VALORES SINGULARES DE ORDEN SUPERIOR

En este capítulo se presenta una generalización multilineal de la Descomposición en Valores Singulares para tensores de orden superior, desarrollada por De Lathauwer (2000a), el cual es denotada por HOSVD (Higher Singular Value Decomposition).

El capítulo está organizado de la siguiente forma: en la sección 3.1 se hace una breve introducción sobre la generalización multilineal de la SVD. En la sección 3.2, se introduce la HOSVD; y posteriormente, en la sección 3.3, se discuten algunas generalizaciones multilineales a propiedades de la SVD matricial. Finalmente, en la sección 3.4, se ilustra como se calcula la HOSVD para un tensor dado, y se presentan algunos resultados obtenidos al aplicarla sobre algunos tensores de orden superior.

### 3.1. Introducción

La Descomposición en Valores Singulares (SVD-Singular Value Decomposition), ha sido uno de los grandes desarrollos aportados por el álgebra lineal moderna con importantes aplicaciones en campos como la visión artificial, recuperación de información en bases de datos, computación lingüística, procesamiento de señales e imágenes, entre otros. Esta herramienta aunque poderosa, está limitada a arreglos de dos dimensiones lo que limita su aplicación a sistemas descritos por datos en más de dos dimensiones.

La generalización multilineal presentada en este capítulo fue inicialmente desarrollada por Tucker ([29]) en el campo de la psicometría, con el fin de encontrar posibles relaciones en tablas con datos de tres dimensiones. La descomposición de Tucker, es un modelo donde el enfoque y el lenguaje utilizados, eran propios del campo de la psicometría, careciendo así, de la

existencia de operadores que permitieran denotar las operaciones multilineales que se definían entre matrices y vectores. A partir de la descomposición anterior, De Lathauer en [19] obtuvo una descomposición tensorial con un enfoque propio del álgebra lineal, cuya notación es una extensión natural del lenguaje matricial. Dado que la SVD matricial residen importantes propiedades del álgebra matricial, De Lathauwer muestra generalizaciones multilineales de estas propiedades y establece un análisis completo de ellas en términos de sus contrapartes lineales.

## 3.2. Descomposición en Valores Singulares de Orden Superior

Antes de presentar el modelo de descomposición en valores singulares para un tensor de orden  $N$ , se mostrará la SVD matricial en la notación inducida por el producto  $n$ -modal entre un tensor y una matriz.

### Teorema 3.1 (*Descomposición en Valores Singulares*)

Toda matriz  $A \in \mathbb{C}^{I_1 \times I_2}$  puede ser escrita como el producto

$$\mathbf{A} = \mathbf{U}^{(1)} \cdot \mathbf{S} \cdot \mathbf{V}^{(2)H} = \mathbf{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{V}^{(2)*} = \mathbf{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \quad (3.1)$$

donde:

- $\mathbf{U}^{(1)} = [U_1^{(1)} U_2^{(1)} \dots U_{I_1}^{(1)}]$  es una matriz unitaria de orden  $I_1 \times I_1$ .
- $\mathbf{U}^{(2)} = [U_1^{(2)} U_2^{(2)} \dots U_{I_2}^{(2)}] = V^{(2)*}$  es una matriz unitaria de orden  $I_2 \times I_2$ .
- $\mathbf{S}$  es una matriz de orden  $I_1 \times I_2$  con las siguientes propiedades:
  - $\mathbf{S} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{\min(I_1 \times I_2)})$
  - $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(I_1 \times I_2)} \geq 0$

Los  $\sigma_i$  son los valores singulares de  $\mathbf{A}$  y los vectores  $\mathbf{U}_i^{(1)}$  y  $\mathbf{U}_i^{(2)}$  son el  $i$ -ésimo vector singular izquierdo y derecho, respectivamente. Un esquema de la descomposición se visualiza en la Fig. 3.1.

**Proposición 3.1** Sean  $\mathcal{A}, \mathcal{S} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  y  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times I_n}$  para  $n = 1, 2, 3$ , relacionados por la siguiente ecuación:

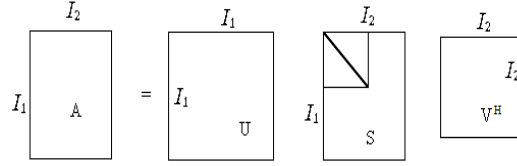


Figura 3.1: Esquema de la SVD matricial

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \quad (3.2)$$

entonces, para todo  $n$  se cumple que<sup>1</sup>

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \cdot \mathbf{S}_{(n)} \cdot [\mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)}]^T \quad (3.3)$$

**Prueba.** Supongamos sin pérdida de generalidad que  $n = 1$ , entonces se cumple que:

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} = (\mathcal{S} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}) \times_1 \mathbf{U}^{(1)} \quad (3.4)$$

$$\mathbf{A}_{(1)} = \mathbf{U}^{(1)} \cdot (\mathcal{S} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)})_{(1)} \quad (3.5)$$

Ahora, se tiene que

$$(\mathcal{S} \times_2 \mathbf{U}^{(2)})_{ijk} = \sum_u s_{iuk} u_{ju}^{(2)} = s_{i1k} u_{j1}^{(2)} + s_{i2k} u_{j2}^{(2)} + \dots + s_{iI_2k} u_{jI_2}^{(2)} \quad (3.6)$$

$$((\mathcal{S} \times_2 \mathbf{U}^{(2)}) \times_3 \mathbf{U}^{(3)})_{rst} = \sum_u (\mathcal{S} \times_2 \mathbf{U}^{(2)})_{rsu} u_{tu}^{(3)} \quad (3.7)$$

$$= (\mathcal{S} \times_2 \mathbf{U}^{(2)})_{rs1} u_{t1}^{(3)} + \dots + (\mathcal{S} \times_2 \mathbf{U}^{(2)})_{rsI_3} u_{tI_3}^{(3)} \quad (3.8)$$

Reemplazando la Ec.3.6 en la Ec. 3.7, se tiene que

$$\begin{aligned} ((\mathcal{S} \times_2 \mathbf{U}^{(2)}) \times_3 \mathbf{U}^{(3)})_{rst} &= \left( s_{r11} u_{s1}^{(2)} + s_{r21} u_{s2}^{(2)} + \dots + s_{rI_21} u_{sI_2}^{(2)} \right) u_{t1}^{(3)} \\ &+ \left( s_{r12} u_{s1}^{(2)} + s_{r22} u_{s2}^{(2)} + \dots + s_{rI_22} u_{sI_2}^{(2)} \right) u_{t2}^{(3)} \\ &+ \dots + \left( s_{r1I_3} u_{s1}^{(2)} + s_{r2I_3} u_{s2}^{(2)} + \dots + s_{rI_2I_3} u_{sI_2}^{(2)} \right) u_{tI_3}^{(3)} \end{aligned} \quad (3.9)$$

<sup>1</sup>El símbolo  $\otimes$  denota la operación producto Kronecker entre dos matrices. Para saber detalles de su definición y propiedades, véase [21].

Luego

$$\begin{aligned}
 ((\mathcal{S} \times_2 \mathbf{U}^{(2)}) \times_3 \mathbf{U}^{(3)})_{rst} &= \left( s_{r11} u_{t1}^{(3)} + s_{r12} u_{t2}^{(3)} + \cdots + s_{r1I_3} u_{tI_3}^{(3)} \right) u_{s1}^{(2)} & (3.10) \\
 &+ \left( s_{r21} u_{t1}^{(3)} + s_{r22} u_{t2}^{(3)} + \cdots + s_{r2I_3} u_{tI_3}^{(3)} \right) u_{s2}^{(2)} \\
 &+ \cdots + \left( s_{rI_21} u_{t1}^{(3)} + s_{rI_22} u_{t2}^{(3)} + \cdots + s_{rI_2I_3} u_{tI_3}^{(3)} \right) u_{sI_2}^{(2)}
 \end{aligned}$$

Luego

$$\begin{aligned}
 ((\mathcal{S} \times_2 \mathbf{U}^{(2)}) \times_3 \mathbf{U}^{(3)})_{rst} &= \left[ s_{r11} \ \cdots \ s_{r1I_3} \mid \cdots \mid s_{rI_21} \ \cdots \ s_{rI_2I_3} \right] \cdot & (3.11) \\
 &\left[ u_{s1}^{(2)} u_{t1}^{(3)} \ \cdots \ u_{s1}^{(2)} u_{tI_3}^{(3)} \mid \cdots \mid u_{sI_2}^{(2)} u_{t1}^{(3)} \ \cdots \ u_{sI_2}^{(2)} u_{tI_3}^{(3)} \right]^T
 \end{aligned}$$

$$((\mathcal{S} \times_2 \mathbf{U}^{(2)}) \times_3 \mathbf{U}^{(3)})_{rst} = [\mathbf{S}_{(1)}]^r \cdot \left[ u_{s1}^{(2)} (\mathbf{U}^{(3)})^t \ \cdots \ u_{sI_2}^{(2)} (\mathbf{U}^{(3)})^t \right]^T \quad (3.12)$$

$$= [\mathbf{S}_{(1)}]^r \cdot \left[ (\mathbf{U}^{(2)} \otimes \mathbf{U}^{(3)})^{s+t+I_3(s-1)} \right]^T \quad (3.13)$$

Luego

$$(\mathcal{S} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)})_{(1)} = \mathbf{S}_{(1)} \cdot [\mathbf{U}^{(2)} \otimes \mathbf{U}^{(3)}]^T \quad (3.14)$$

Reemplazando la Ec. 3.14 en la Ec. 3.5, se obtiene que:

$$\mathbf{A}_{(1)} = \mathbf{U}^{(1)} \cdot \mathbf{S}_{(1)} \cdot [\mathbf{U}^{(2)} \otimes \mathbf{U}^{(3)}]^T \quad (3.15)$$

■

**Proposición 3.2** Sea  $\mathbf{A} \in \mathbb{R}_r^{m \times n}$ , entonces existe una matriz ortogonal  $\mathbf{U} \in \mathbb{R}^{m \times m}$  y una matriz  $\mathbf{V} \in \mathbb{R}^{n \times m}$  con columnas ortogonales tales que

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \quad (3.16)$$

donde  $\mathbf{\Sigma} = \mathbf{diag} \{ \sigma_1, \sigma_2, \dots, \sigma_n \}$ , y  $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ .

**Prueba.** Sea  $\mathbf{A} \in \mathbb{R}_r^{m \times n}$ , entonces existen matrices ortogonales  $\mathbf{U} \in \mathbb{R}^{m \times m}$  y  $\mathbf{V} \in \mathbb{R}^{n \times n}$  tales que

$$\mathbf{A} = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T \quad (3.17)$$

donde  $\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{n \times m}$ ,  $\mathbf{S} = \mathbf{diag}\{\sigma_1, \sigma_2, \dots, \sigma_r\}$ , y  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$ . Ahora, analizemos los siguientes casos:

- Si  $m = n$ , claramente se cumple el resultado.
- Si  $m < n$ , entonces se cumple que:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_m & & \\ & & & & \\ & & & & \end{bmatrix} \quad \mathbf{V} = \begin{bmatrix} v_1 & \cdots & v_m & \cdots & v_n \\ \vdots & & \vdots & & \vdots \\ \vdots & & \vdots & & \vdots \end{bmatrix} \quad (3.18)$$

Dado que las columnas  $m + 1, \dots, n$  son nulas se tiene que

$$\boldsymbol{\Sigma} \cdot \mathbf{V}^T = \underbrace{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_m \end{bmatrix}}_{\boldsymbol{\Sigma}' } \underbrace{\begin{bmatrix} v_1 & \cdots & v_m \\ \vdots & & \vdots \\ \vdots & & \vdots \end{bmatrix}^T}_{\mathbf{V}'} \quad (3.19)$$

Luego redefiniendo  $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}'$  y  $\mathbf{V} = \mathbf{V}'$  se obtiene el resultado.

- Si  $m > n$ , entonces se cumple que:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_n & & \\ & & & & \\ & & & & \end{bmatrix} \quad \mathbf{V}' = \begin{bmatrix} v_1 & \cdots & v_n \\ \vdots & & \vdots \\ \vdots & & \vdots \end{bmatrix} \quad (3.20)$$

Ahora sea

$$\boldsymbol{\Sigma}' = \begin{bmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_n & & & \\ & & & \sigma_{n+1} & & \\ & & & & \ddots & \\ & & & & & \sigma_m \end{bmatrix} \quad \mathbf{V} = \begin{bmatrix} v_1 & \cdots & v_n & v_{n+1} & \cdots & v_m \\ \vdots & & \vdots & \vdots & & \vdots \\ \vdots & & \vdots & \vdots & & \vdots \end{bmatrix} \quad (3.21)$$

Donde  $\sigma_{n+1} = \dots = \sigma_m = 0$  y  $v_{n+1} = \dots = v_m = v_1$ . Luego se cumple que

$$\Sigma \cdot \mathbf{V}^T = \Sigma' \cdot \mathbf{V}'^T \quad (3.22)$$

redefiniendo  $\Sigma = \Sigma'$  y  $\mathbf{V} = \mathbf{V}'$  se obtiene el resultado. ■

**Teorema 3.2** (*Descomposición en Valores Singulares de orden  $N$* )

Todo tensor  $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  puede ser escrito como el producto

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_N \mathbf{U}^{(N)} \quad (3.23)$$

en donde:

- $\mathbf{U}^{(n)} = \left[ U_1^{(n)} U_2^{(n)} \dots U_{I_n}^{(n)} \right]$  es una matriz unitaria de orden  $I_n \times I_n$
- $\mathcal{S}$  es un tensor de orden  $I_1 \times I_2 \times \dots \times I_N$  cuyos subtensores  $\mathcal{S}_{i_n=\alpha}$ , obtenidos al fijar el  $n$ -ésimo índice a  $\alpha$ , tienen las siguientes propiedades:
  - *Ortogonalidad de subtensores:* para todos los posibles valores de  $n$ ,  $\alpha$  y  $\beta$  con  $\alpha \neq \beta$ , se cumple que:

$$\langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0 \quad \text{si } \alpha \neq \beta \quad (3.24)$$

- *Ordenamiento:* para todos los posibles valores de  $n$  se cumple que:

$$\|\mathcal{S}_{i_n=1}\| \geq \|\mathcal{S}_{i_n=2}\| \geq \dots \geq \|\mathcal{S}_{i_n=I_n}\| \geq 0 \quad (3.25)$$

La norma de Frobenius  $\|\mathcal{S}_{i_n=i}\|$ , denotada por  $\sigma_i^{(n)}$ , es el  $i$ -ésimo valor singular  $n$ -modal de  $\mathcal{A}$  y el vector  $U_i^{(n)}$  es el  $i$ -ésimo vector singular  $n$ -modal. Para un tensor de tercer orden, un esquema de tal descomposición se muestra en la Fig. 3.2.

**Prueba.** (Los tensores considerados en esta demostración son reales valuados). Considere dos tensores  $\mathcal{A}$  y  $\mathcal{S}$  de orden  $I_1 \times I_1 \times \dots \times I_N$ , relacionados por la siguiente ecuación:

$$\mathcal{S} = \mathcal{A} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \dots \times_N \mathbf{U}^{(N)T} \quad (3.26)$$

En donde  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$  son matrices ortogonales. La ecuación puede ser expresada en forma matricial como

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \cdot \mathcal{S}_{(n)} \cdot \left[ \mathbf{U}^{(n+1)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \dots \otimes \mathbf{U}^{(n-1)} \right]^T \quad (3.27)$$



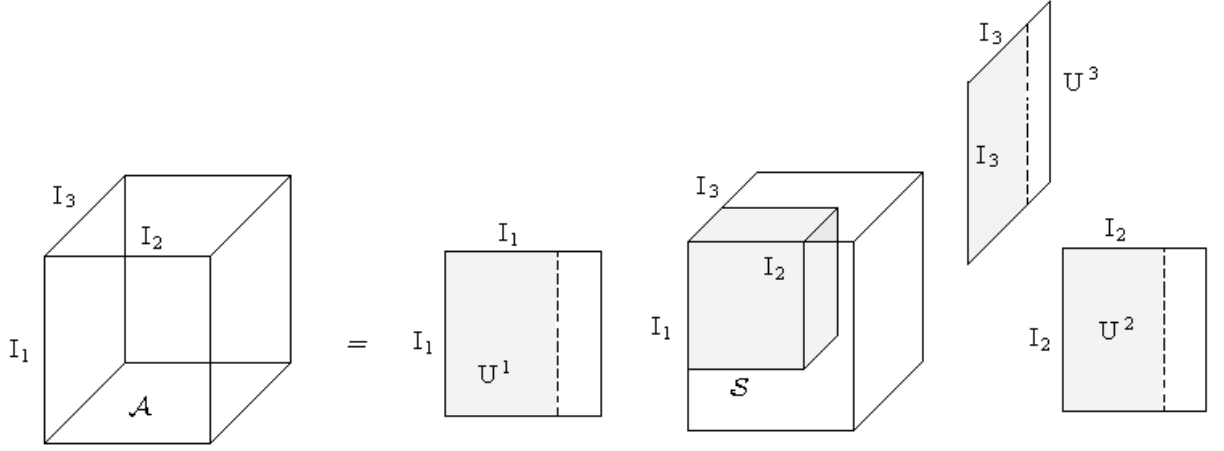


Figura 3.2: Esquema de la HOSVD de un tensor de orden tres

Luego, suponga que  $\mathbf{U}^{(n)}$  es obtenida a través de la SVD de  $\mathbf{A}_{(n)}$  como se observa a continuación:

$$\mathbf{A}_{(n)} = \mathbf{U}^{(n)} \boldsymbol{\Sigma}^{(n)} \mathbf{V}^{(n)T} \quad (3.28)$$

en el cual  $\mathbf{V}^{(n)}$  es una matriz de orden  $I_{n+1}I_{n+2} \cdots I_N I_1 \cdots I_{n-1} \times I_n$  y cuyas columnas son ortonormales;  $\boldsymbol{\Sigma}^{(n)} = \text{diag}(\sigma_1^{(n)}, \sigma_2^{(n)}, \dots, \sigma_{I_n}^{(n)})$ , y donde se cumple:

$$\sigma_1^{(n)} \geq \sigma_2^{(n)} \geq \cdots \geq \sigma_{I_n}^{(n)} \geq 0 \quad (3.29)$$

Ahora sea  $r_n$  el índice más grande que cumple que  $\sigma_{r_n}^{(n)} > 0$ . Teniendo presente que el producto Kronecker de la Ec. 3.27 es ortogonal, se obtiene al comparar las Ec. 3.27 y Ec. 3.28 que:

$$\mathbf{S}_{(n)} = \boldsymbol{\Sigma}^{(n)} \mathbf{V}^{(n)T} \left[ \mathbf{U}^{(n+1)} \otimes \cdots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \cdots \otimes \mathbf{U}^{(n-1)} \right] \quad (3.30)$$

Para matrices ortogonales arbitrarias  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \dots, \mathbf{U}^{(N)}$ , la ecuación anterior implica que:

$$\langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0 \quad \text{si } \alpha \neq \beta \quad (3.31)$$

$$\|\mathcal{S}_{i_n=1}\| = \sigma_1^{(n)} \geq \|\mathcal{S}_{i_n=2}\| = \sigma_2^{(n)} \geq \dots \geq \|\mathcal{S}_{i_n=I_n}\| = \sigma_{I_n}^{(n)} \geq 0 \quad (3.32)$$

además, si  $r_n < I$

$$\|\mathcal{S}_{i_n=r_n+1}\| = \sigma_{r_n+1}^{(n)} = \dots = \|\mathcal{S}_{i_n=I_n}\| = \sigma_{I_n}^{(n)} = 0 \quad (3.33)$$

Construyendo  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \dots, \mathbf{U}^{(N)}$  de forma similar a  $\mathbf{U}^{(n)}$ ,  $\mathcal{S}$  se tiene que satisface todas las condiciones establecidas en el teorema. Además, se observa que a través de las ecuaciones 4.8 - 4.14, las matrices  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$  y el tensor  $\mathcal{S}$  se pueden encontrar a través de la SVD matricial de  $\mathbf{A}_{(1)}, \mathbf{A}_{(2)}, \dots, \mathbf{A}_{(N)}$ . ■

### Discusiones sobre la HOSVD.

Analizando ambas descomposiciones claramente se puede observar que existen analogías entre ellas, verificándose que la descomposición matricial es un caso particular del teorema 2.2 cuando se restringe a tensores de orden dos. De Lathauwer en [19] resalta las siguientes analogías:

- Los vectores singulares izquierdos y derechos son generalizados por los vectores singulares *n-modales*.
- En la descomposición de orden superior la matriz  $\mathbf{S}$  es reemplazada por el tensor  $\mathcal{S}$ .
- El papel de los valores singulares son tomados por la norma de Frobenius de los subtensores de orden  $N - 1$  del tensor  $\mathcal{S}$ ; y que en el caso matricial, los valores singulares corresponden a la norma de Frobenius de las filas y columnas de la matriz  $\mathbf{S}$ .
- Para un tensor de orden  $N$ , existen un conjunto de  $N$  valores propios *n-modales* (posiblemente distintos) positivos y reales, donde cada  $r_n$  (índice más grande que cumple que  $\sigma_{r_n}^{(n)} > 0$ ) es igual al rango *n-modal* del tensor  $\mathcal{A}$ .
- Si el tensor  $\mathcal{A}$  es complejo, entonces el tensor  $\mathcal{S}$  puede tomar valores complejos.
- La mayor diferencia entre ambas descomposiciones radica en que generalmente  $\mathcal{S}$  es un tensor completo, en lugar de ser un tensor pseudodiagonal, es decir, que los únicos elementos distintos de ceros solo pueden ocurrir cuando  $i_1 = i_2 = \dots = i_N$ . No obstante, el tensor  $\mathcal{S}$  cumple la condición de ortogonalidad tal como la cumple la matriz  $\mathbf{S}$ .

Facilmente, a través de un análisis sobre el rango del tensor  $\mathcal{A}$  y sus rangos *n-modales* se puede observar que un tensor  $\mathcal{S}$  pseudo-diagonal no puede ser una apropiada generalización de la

matriz pseudo-diagonal  $\mathbf{S}$ , dado que se verificaría que  $rank(\mathcal{A}) = rank_n(\mathcal{A})$  para  $i = 1, \dots, N$ . Para demostrar lo anterior, considere un tensor  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  y suponga que el tensor  $\mathcal{S}$  es un tensor pseudo-diagonal, con  $I = \min(I_1, \dots, I_N)$  elementos distintos de cero en la diagonal principal.

Ahora, dada la forma del tensor  $\mathcal{S}$  se puede escribir como

$$\mathcal{S} = \sum_{i=1}^I c_i U_i^{(1)} \circ U_i^{(2)} \circ \dots \circ U_i^{(N)} \quad (3.34)$$

Donde

$$c_i = s_{i_1 i_2 \dots i_N}, \quad \text{con } i_1 = i_2 = \dots = i_N = i \quad (3.35)$$

$$U_i^{(n)} = \begin{bmatrix} 0 \\ \vdots \\ u_{i_1}^{(n)} = 1 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{I_n}, \quad \text{para } n = 1, \dots, N \quad (3.36)$$

Sea  $i \in \{1, \dots, I\}$ , entonces se cumple que  $c_i U_i^{(1)} \circ U_i^{(2)} \circ \dots \circ U_i^{(N)} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  donde

$$\left( c_i U_i^{(1)} \circ U_i^{(2)} \circ \dots \circ U_i^{(N)} \right)_{i_1 i_2 \dots i_N} = \begin{cases} c_i, & \text{si } i_1 = i_2 = \dots = i_N = i \\ 0, & \text{de lo contrario} \end{cases} \quad (3.37)$$

Obteniéndose así, que  $rank_n(\mathcal{S}) = I$  para  $n = 1, \dots, N$ . Ahora como  $rank_n(\mathcal{S}) \leq rank(\mathcal{S}) \leq I$ , entonces

$$rank_n(\mathcal{S}) = rank(\mathcal{S}) = I, \quad \text{para } n = 1, \dots, N \quad (3.38)$$

Luego, por la Ec. 3.26, se tiene que

$$rank(\mathcal{A}) = I = rank(\mathcal{S}) \quad (3.39)$$

y dado que  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$  son matrices ortogonales, por la Ec. 3.27 se cumple que

$$rank_n(\mathcal{A}) = rank_n(\mathcal{S}), \quad \text{para } n = 1, \dots, N \quad (3.40)$$

Por tanto, se concluye que  $rank(\mathcal{A}) = rank_n(\mathcal{A})$  para  $i = 1, \dots, N$ ; lo cual no siempre se cumple.

La descomposición diádica puede ser generalizada expresando la Ec. 3.26 a través de tensores de rango-1, tal como se muestra a continuación:

$$\mathcal{A} = \sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \cdots \sum_{i_N}^{I_N} s_{i_1 i_2 \dots i_N} U_{i_1}^{(1)} \circ U_{i_2}^{(2)} \circ \cdots \circ U_{i_N}^{(N)} \quad (3.41)$$

En la Fig.3.3 para un tensor de orden tres, se visualiza la descomposición diádica.

$$\mathcal{A} = \cdots + s_{i_1 i_2 i_3} U_{i_1}^{(1)} \circ U_{i_2}^{(2)} \circ U_{i_3}^{(3)} + \cdots$$

Figura 3.3: Esquema de la descomposición diádica para un tensor de orden tres.

### 3.3. Propiedades de la HOSVD

En esta sección se mencionan algunas generalizaciones que establece De Lathauwer en [19], sobre ciertas propiedades de la SVD matricial a la HOSVD. Para mayor información puede referirse al texto citado.

**Propiedad 3.3.1 (Generalización)** La HOSVD de un tensor de orden dos se reduce a la SVD matricial.

**Propiedad 3.3.2 (Rango-n)** Considere un tensor  $\mathcal{A}$  con HOSVD dada por el teorema 2.2, y sea  $r_n$  el índice más grande para el cual se cumple que  $\|\mathcal{S}_{i_n=r_n}\| > 0$  en la Ec. 3.25, entonces se cumple que

$$R_n = rank_n(\mathcal{A}) = r_n \quad (3.42)$$

**Propiedad 3.3.3 (Estructura)** Considere un tensor  $\mathcal{A}$  con HOSVD dada por el teorema 2.2, entonces se cumple que:

- El espacio vectorial  $n$ -modal  $R(\mathbf{A}_{(n)}) = span \{U_1^{(n)}, \dots, U_{r_n}^{(n)}\}$
- El complemento ortogonal de  $R(\mathbf{A}_{(n)})$ , es el espacio nulo izquierdo  $N(\mathbf{A}_{(n)}^H) = span \{U_{r_n+1}^{(n)}, \dots, U_{I_n}^{(n)}\}$

**Propiedad 3.3.4 (Norma)** Considere un tensor  $\mathcal{A}$  con HOSVD dada por el teorema 2.2, entonces se cumple

$$\|\mathcal{A}\|^2 = \sum_{i=1}^{R_1} (\sigma_i^{(1)})^2 = \dots = \sum_{i=1}^{R_N} (\sigma_i^{(N)})^2 = \|\mathcal{S}\|^2 \quad (3.43)$$

**Propiedad 3.3.5 (Aproximación)** Considere un tensor  $\mathcal{A}$  con HOSVD dada por el teorema 2.2, y defina el tensor  $\widehat{\mathcal{A}}$  al despreciar los valores singulares  $n$ -modales más pequeños  $\sigma_{I'_n+1}^{(n)}, \sigma_{I'_n+2}^{(n)}, \dots, \sigma_{R_n}^{(n)}$  para un valor dado  $I'_n (1 \leq n \leq N)$ , es decir, al fijar las componentes correspondientes de  $\mathcal{S}$  iguales a cero, entonces se cumple que

$$\|\mathcal{A} - \widehat{\mathcal{A}}\|^2 \leq \sum_{i_1=I'_1+1}^{R_1} (\sigma_{i_1}^{(1)})^2 + \sum_{i_2=I'_2+1}^{R_2} (\sigma_{i_2}^{(2)})^2 + \dots + \sum_{i_N=I'_N+1}^{R_N} (\sigma_{i_N}^{(N)})^2$$

Esta propiedad generaliza el concepto de mejor aproximación de una matriz, en el sentido de mínimos cuadrados, por una matriz de menor rango; donde,  $\widehat{\mathcal{A}}$  es un tensor donde el rango  $n$ -modal  $R_n$  es igual a  $I'_n$ . Sin embargo a diferencia del caso matricial, el tensor  $\widehat{\mathcal{A}}$  no siempre es la mejor aproximación bajo las restricciones de rango  $n$ -modales dadas, pero si  $\sigma_{I'_n}^{(n)} \gg \sigma_{I'_n+1}^{(n)}$ , se considera que sigue siendo una buena aproximación de  $\mathcal{A}$ . En el siguiente capítulo, se introduce una descomposición que en la mayoría de los casos permite obtener la mejor aproximación de bajo rango de un tensor  $\mathcal{A}$  en el sentido de mínimos cuadrados.

### 3.4. Cálculo de la HOSVD

A través de las Ecs. 3.26 y 3.28 se ilustra como obtener la HOSVD de un tensor  $\mathcal{A}$ : la matriz singular  $n$ -modal  $\mathbf{U}^{(n)}$  puede encontrarse directamente a través de los vectores singulares izquierdos de la matriz *despliegue*  $n$ -modal del tensor  $\mathcal{A}$  (ó cualquier matriz cuyas columnas están dadas por los vectores  $n$ -modales)<sup>2</sup>; por lo tanto, obtener la HOSVD de un tensor de orden  $N$  es necesario calcular  $N$  distintas SVDs matriciales de matrices con tamaños  $I_n \times I_1 I_2 \dots I_{n-1} I_{n+1} \dots I_{N-1} I_N$  ( $1 \leq n \leq N$ ). Por último, el tensor  $\mathcal{S}$  puede ser calculado a través de la Ec. 3.26, cuya versión matricial está dado por:

$$\mathbf{S}_{(n)} = \mathbf{U}^{(n)T} \cdot \mathbf{A}_{(n)} \cdot (\mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n+2)} \otimes \dots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \mathbf{U}^{(2)} \otimes \dots \otimes \mathbf{U}^{(n-1)}) \quad (3.44)$$

En el algoritmo 3.1, se describen los pasos para obtener la HOSVD de un tensor de orden tres; el algoritmo, fue programado en Matlab en la rutina *hosvd.m*.

<sup>2</sup>Dado que la matriz *despliegue*  $n$ -modal  $\mathbf{A}_{(n)}$  es una matriz cuyas columnas estan dadas por lo vectores  $n$ -modales, cualquier matriz *despliegue* con un ordenamiento distinto al usado en esta tesis, podra ser utilizada para encontrar la matriz singular  $n$ -modal  $\mathbf{U}^{(n)}$ .

**Algoritmo 3.1**

*Descomposición en Valores Singulares de Orden Superior*

**Entrada:** Tensor  $\mathcal{A}$  de orden  $I_1 \times I_2 \times I_3$  y rango deseado- $(R_1, R_2, R_3)$

$\mathbf{A}_{(n)} \leftarrow$  Asigne la matriz despliegue  $n$ -modal de  $\mathcal{A}$  ( $1 \leq n \leq 3$ )

$\mathbf{U}^{(n)} \leftarrow$  Asigne los primeros  $R_n$  vectores propios de  $\mathbf{A}_{(n)}\mathbf{A}_{(n)}^T$

$\mathcal{S} \leftarrow \mathcal{A} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$

**Resultado:** Matrices ortogonales  $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n}$  ( $1 \leq n \leq 3$ ) y tensor  $\mathcal{S}$  de orden  $R_1 \times R_2 \times R_3$ .

**Ejemplo 3.1** Considere el tensor  $\mathcal{A}$  de orden  $6 \times 4 \times 3$  definido por una matriz despliegue  $\mathbf{A}_{(1)}$  dada por

$$\left[ \begin{array}{ccc|ccc|ccc|ccc} 0,36 & 0,43 & 0,42 & 0,75 & 0,67 & 0,86 & 0,60 & 0,40 & 0,18 & 0,55 & 0,82 & 0,17 \\ 0,54 & 0,56 & 0,37 & 0,74 & 0,20 & 0,76 & 0,25 & 0,47 & 0,49 & 0,29 & 0,95 & 0,62 \\ 0,26 & 0,61 & 0,16 & 0,43 & 0,27 & 0,44 & 0,87 & 0,90 & 0,40 & 0,85 & 0,59 & 0,24 \\ 0,59 & 0,11 & 0,83 & 0,63 & 0,62 & 0,62 & 0,51 & 0,59 & 0,46 & 0,33 & 0,02 & 0,58 \\ 0,04 & 0,89 & 0,83 & 0,8 & 0,53 & 0,95 & 0,73 & 0,32 & 0,61 & 0,68 & 0,81 & 0,50 \\ 0,57 & 0,75 & 0,45 & 0,08 & 0,05 & 0,64 & 0,42 & 0,47 & 0,07 & 0,05 & 0,61 & 0,46 \end{array} \right]$$

Los vectores singulares 1-modales, 2-modales y 3-modales se muestran a continuación:

$$\mathbf{U}^{(1)} = \left[ \begin{array}{cccccc} 0,4202 & -0,0286 & -0,1461 & -0,3275 & -0,5127 & 0,6566 \\ 0,4136 & -0,1325 & 0,4874 & -0,0725 & -0,4853 & -0,5771 \\ 0,3954 & -0,3875 & -0,6219 & 0,5212 & -0,0432 & -0,1821 \\ 0,3657 & 0,8738 & -0,0400 & 0,3157 & 0,0218 & -0,0303 \\ 0,5162 & -0,0382 & -0,1161 & -0,5647 & 0,6111 & -0,1623 \\ 0,3096 & -0,2577 & 0,5824 & 0,4442 & 0,3548 & 0,4188 \end{array} \right]$$

$$\mathbf{U}^{(2)} = \left[ \begin{array}{cccc} 0,4752 & 0,3277 & 0,7660 & 0,2830 \\ 0,5376 & -0,8238 & 0,0786 & -0,1617 \\ 0,4698 & 0,1232 & -0,5843 & 0,6501 \\ 0,5142 & 0,4458 & -0,2562 & -0,6864 \end{array} \right], \quad \mathbf{U}^{(3)} = \left[ \begin{array}{ccc} 0,5674 & 0,1148 & -0,8154 \\ 0,6009 & -0,7348 & 0,3147 \\ 0,5630 & 0,6685 & 0,4859 \end{array} \right]$$

El tensor central de la HOSVD del tensor  $\mathcal{A}$  es obtenido a través de la 3.44, y su matriz

despliegue  $\mathbf{S}_{(1)}$  se muestra a continuación:

$$\mathbf{S}_{(1)} = \begin{bmatrix} 4,4139 & 0,0655 & 0,0367 & | & -0,0758 & -0,7027 & 0,1071 & | \\ 0,0880 & 0,5157 & 0,0110 & | & -0,3998 & 0,6327 & 0,0370 & | \\ -0,0192 & 0,1699 & 0,4094 & | & 0,1399 & -0,2409 & 0,1797 & | \\ 0,0397 & -0,0804 & -0,1166 & | & 0,3763 & 0,0689 & -0,2010 & | \\ -0,0197 & 0,1527 & 0,2170 & | & 0,1981 & 0,0644 & -0,0065 & | \\ -0,0221 & -0,1662 & -0,0433 & | & -0,1492 & -0,0197 & -0,2570 & | \\ -0,0122 & 0,1593 & 0,3998 & | & 0,0260 & 0,0259 & -0,1211 & | \\ 0,1678 & 0,2920 & -0,1374 & | & 0,1733 & -0,0587 & 0,0391 & | \\ 0,4968 & 0,0324 & -0,4055 & | & -0,1295 & -0,0056 & -0,3876 & | \\ -0,1049 & 0,1558 & -0,2695 & | & 0,3959 & -0,4243 & -0,1263 & | \\ 0,1400 & -0,1629 & 0,3933 & | & 0,1851 & -0,0784 & 0,0193 & | \\ 0,1903 & 0,1583 & 0,1628 & | & 0,0171 & 0,0098 & -0,1972 & | \end{bmatrix}$$

Los valores singulares de orden superior están dados por:

$$\begin{aligned} \text{modo 1} & : & 4,4946 & 1,0013 & 0,9413 & 0,8179 & 0,5965 & 0,4957 \\ \text{modo 2} & : & 4,4813 & 1,2253 & 1,0614 & 0,8068 & & \\ \text{modo 3} & : & 4,5236 & 1,3023 & 1,0963 & & & \end{aligned}$$

Según los resultados anteriores, se infiere que:

- Las matrices  $\mathbf{U}^{(1)}$ ,  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$  son aproximadamente ortogonales, dado que  $\|\mathbf{I} - \mathbf{U}^{(i)T} \cdot \mathbf{U}^{(i)}\|_2 < 10^{-4}$  para  $i = 1, 2, 3$ .
- Todos los subtensores de  $\mathcal{S}$ , es decir, las matrices laterales, horizontales y frontales del tensor  $\mathcal{S}$  son mutuamente ortogonales, dado que las matrices formadas por las columnas  $\{1, 2, 3\}$ ,  $\{4, 5, 6\}$ ,  $\{7, 8, 9\}$  y  $\{10, 11, 12\}$  son mutuamente ortogonales; y al igual, que las filas de la matriz  $\mathbf{S}_{(1)}$ , y las matrices formadas por las columnas  $\{1, 4, 7, 10\}$ ,  $\{2, 5, 8, 11\}$  y  $\{3, 6, 9, 12\}$ , respectivamente. Cabe resaltar, que en las filas de las matrices  $\mathbf{S}_{(1)}$ ,  $\mathbf{S}_{(2)}$  y  $\mathbf{S}_{(3)}$  están las componentes de las matrices horizontales, laterales y frontales, respectivamente.
- El tensor central  $\mathcal{S}$  es ordenado, es decir, las matrices laterales, frontales y horizontales son ordenadas, según su norma de Frobenius, de manera decreciente, dado que la norma de Frobenius de cada una de las filas de las matrices  $\mathbf{S}_{(1)}$ ,  $\mathbf{S}_{(2)}$  y  $\mathbf{S}_{(3)}$  son iguales a cada uno de los valores propios  $n$ -modales, es decir,

$$\|\mathbf{S}_{(n)}(j, :)\| = \|\mathcal{S}_{i_n=j}\| = \sigma_j^{(n)}, \quad \text{para } j = 1, \dots, I_n \text{ y } n = 1, 2, 3$$

- Para  $n = 1, 2, 3$  se cumple que  $\text{rank}(\mathbf{A}_{(n)}) = R_n$ , y además,  $\|\mathcal{A}\|^2 = \sum_{i_1=1}^{R_1} (\sigma_{i_1}^{(1)})^2 = \sum_{i_2=1}^{R_2} (\sigma_{i_2}^{(1)})^2 = \sum_{i_3=1}^{R_3} (\sigma_{i_3}^{(1)})^2 = 23,3604$ .
- Definiendo el tensor  $\widehat{\mathcal{S}}$  al despreciar los valores singulares  $n$ -modales más pequeños  $\sigma_5^{(1)}, \sigma_6^{(1)}, \sigma_3^{(2)}, \sigma_4^{(2)}$  y  $\sigma_3^{(3)}$ , i.e, al fijar las componentes correspondientes de  $\mathcal{S}$  iguales a cero, se obtiene que

$$\widehat{\mathcal{S}}_{(1)} = \left[ \begin{array}{ccc|ccc|ccc|ccc} 4,4139 & 0,0655 & 0 & -0,0758 & -0,7027 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,0880 & 0,5157 & 0 & -0,3998 & 0,6327 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0,0192 & 0,1699 & 0 & 0,1399 & -0,2409 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,0397 & -0,0804 & 0 & 0,3763 & 0,0689 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right]$$

el tensor  $\widehat{\mathcal{S}}$  permite obtener una aproximación  $\widehat{\mathcal{A}}$  para el cual

$$\|\mathcal{A} - \widehat{\mathcal{A}}\|^2 = 1,5095 \leq \sum_{i_1=5}^6 (\sigma_{i_1}^{(1)})^2 + \sum_{i_2=3}^4 (\sigma_{i_2}^{(2)})^2 + \sum_{i_N=3}^3 (\sigma_{i_N}^{(3)})^2 = 3,5808$$

Aunque la verificación de las propiedades anteriores permite asegurar una buena calidad de la HOSVD encontrada para el tensor  $\mathcal{A}$ , se realizó una confrontación con una descomposición encontrada a través de la rutina "Tucker Alternating Least Squares" desarrollada por Bader y Kolda [7] en el toolbox de tensores de MATLAB, las componentes de la HOSVD encontrada se muestra a continuación:

$$\mathbf{U}^{(1)} = \left[ \begin{array}{cccccc} 0,4202 & -0,0286 & 0,1461 & 0,3275 & -0,5127 & 0,6566 \\ 0,4136 & -0,1325 & -0,4874 & 0,0725 & -0,4853 & -0,5771 \\ 0,3954 & -0,3875 & 0,6219 & -0,5212 & -0,0432 & -0,1821 \\ 0,3657 & 0,8738 & 0,0400 & -0,3157 & 0,0218 & -0,0303 \\ 0,5162 & -0,0382 & 0,1161 & 0,5647 & 0,6111 & -0,1623 \\ 0,3096 & -0,2577 & -0,5824 & -0,4442 & 0,3548 & 0,4188 \end{array} \right]$$

$$\mathbf{U}^{(2)} = \left[ \begin{array}{cccc} 0,4752 & -0,3277 & 0,7660 & -0,2830 \\ 0,5376 & 0,8238 & 0,0786 & 0,1617 \\ 0,4698 & -0,1232 & -0,5843 & -0,6501 \\ 0,5142 & -0,4458 & -0,2562 & 0,6864 \end{array} \right], \quad \mathbf{U}^{(3)} = \left[ \begin{array}{ccc} 0,5674 & -0,1148 & 0,8154 \\ 0,6009 & 0,7348 & -0,3147 \\ 0,5630 & -0,6685 & -0,4859 \end{array} \right]$$





$$\mathbf{U}^{(1)} = \begin{bmatrix} -0,0725 & -0,0239 & -0,6311 & -0,4578 & -0,6215 \\ -0,4495 & -0,7567 & 0,2735 & -0,3790 & 0,0831 \\ -0,4207 & 0,4079 & 0,5893 & -0,0121 & -0,5561 \\ -0,6691 & 0,4264 & -0,3064 & -0,1696 & 0,4978 \\ 0,4098 & 0,2805 & 0,2928 & -0,7860 & 0,2231 \end{bmatrix}, \quad \mathbf{U}^{(3)} = \begin{bmatrix} -0,0660 & -0,9978 \\ 0,9978 & -0,0660 \end{bmatrix}$$

El tensor central de la HOSVD del tensor  $\mathcal{A}$  es obtenido a través de la 3.44, y su matriz despliegue  $\mathbf{S}_{(1)}$  se muestra a continuación:

$$\mathbf{S}_{(1)} = \begin{bmatrix} -15,140 & -0,6270 & | & 0,2208 & -3,0107 & | & 0,0393 & 1,0472 & | \\ -0,1296 & 4,1903 & | & 10,5244 & -1,2505 & | & -0,8477 & -2,2264 & | \\ 0,9382 & 0,5464 & | & 1,1583 & -0,2870 & | & 7,1683 & 2,2450 & | \\ 0,9893 & -3,6417 & | & 1,6828 & -0,5919 & | & -0,6623 & -0,0950 & | \\ 0,2418 & -1,8578 & | & 0,2508 & -1,7482 & | & 0,8550 & -1,4110 & | \\ & 0,9922 & -4,2404 & | & -0,1087 & 0,4455 & | & -0,8433 & -1,0343 \\ & -1,5322 & 0,8542 & | & -0,1429 & 0,7290 & | & -0,1930 & -1,5286 \\ & 1,4666 & -2,0977 & | & 0,9887 & -1,3160 & | & -0,1771 & -0,9572 \\ & 0,6943 & -2,2845 & | & 1,7595 & 1,8989 & | & -1,3435 & 1,4476 \\ & -0,0930 & 0,3376 & | & -3,1987 & -0,6998 & | & 0,0332 & -0,1324 \end{bmatrix}$$

Los valores singulares de orden superior están dados por:

$$\begin{aligned} \text{modo 1} & : & 16,1492 & 11,8988 & 8,3208 & 5,8458 & 4,4960 \\ \text{modo 2} & : & 16,3136 & 11,3651 & 8,1467 & 5,8666 & 4,5710 & 3,0035 \\ \text{modo 3} & : & 20,5521 & 10,1789 & & & & \end{aligned}$$

Dado los resultados anteriores, se tiene que  $\mathbf{U}^{(1)}$ ,  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$  son matrices ortogonales; el tensor  $\mathcal{S}$  es mutuamente ortogonal y ordenado; y por último, que la suma al cuadrado de los valores singulares de orden superior es 526, el cual es igual a la norma de Frobenius del tensor  $\mathcal{A}$ . Cabe mencionar, que se obtuvieron resultados similares al utilizar la rutina "Tucker Alternating Least Squares".

### 3.5. Conclusiones

En este capítulo, se abordó el problema de la generalización de la SVD matricial a tensores de orden superior, donde se mostraron fuertes analogías tanto a nivel estructural como a nivel de propiedades. Estructuralmente, se mostró que la matriz pseudo-diagonal de valores singulares es reemplazada por un tensor que cumple las propiedades de ortogonalidad y ordenamiento; donde, la propiedad de ortogonalidad generaliza el hecho de que los vectores filas y columnas de la matriz pseudo-diagonal sean ortogonales, y donde la propiedad de ordenamiento, generaliza el hecho de que los valores singulares en la matriz pseudo-diagonal estén organizados en forma decreciente. Sin embargo, se mostró que la generalización de la

---

matriz de valores singulares no puede ser un tensor pseudo-diagonal, dado que se cumpliría que los rangos *n-modales* serían iguales al rango del tensor, hecho que no siempre se cumple. A nivel de propiedades, se mostraron contrapartes de orden superior a algunas de propiedades de la SVD matricial, dada la estrecha relación entre un tensor de orden superior y sus representaciones matriciales.

Al final del capítulo se discutió cómo obtener la HOSVD de un tensor de orden  $N$ , el cual simplemente se reduce al cálculo de la SVD de cada una de las  $N$  matrices *despliega* del tensor dado. Lo anterior, permite concluir que la HOSVD es una descomposición que se caracteriza por optimizar cada *modo*  $n$  de manera independiente. Por último, se mostraron algunos resultados obtenidos con la implementación numérica desarrollada, examinada a través de dos ejemplos, y cuyos resultados, fueron validados a través de la verificación de su estructura y de algunas propiedades básicas mencionadas sobre la HOSVD; y a través, de la confrontación con descomposiciones obtenidas con el *toolbox* desarrollado por Bader y Kolda en [7]; constatando así, que la HOSVD de una tensor dado no es única al igual que en el caso matricial.



## Capítulo 4

# MEJOR APROXIMACIÓN DE RANGO-1 Y DE RANGO- $(R_1, R_2, \dots, R_N)$

En este capítulo, se introducen dos generalizaciones multilineales de la mejor aproximación de rango- $R$  de una matriz. Una de ellas, intenta resolver la pregunta de cómo obtener la mejor aproximación de un tensor dado, en el sentido de mínimos cuadrados, por un tensor con rangos  $n$ -modales preespecificados; y la otra, busca obtener la mejor aproximación de rango- $R$ , al maximizar la mejor aproximación de las *secciones* del tensor original. Cabe resaltar, que la mejor aproximación de rango  $R$  fue inicialmente abordada en el capítulo anterior, a través de la HOSVD, donde un tensor  $\mathcal{A}$  se podía aproximar de manera adecuada por un tensor con rangos  $n$ -modales iguales a los preespecificados, al despreciar ciertos valores singulares  $n$ -modales del tensor  $\mathcal{A}$ .

El capítulo está organizado de la siguiente forma: en la sección 4.1 se discuten brevemente los métodos de la potencia y de la iteración ortogonal. En la sección 4.2, se aborda el problema de la mejor aproximación de rango 1 de un tensor de orden superior, y se discute una generalización del método de la potencia matricial. En las secciones 4.3 y 4.4, se aborda la generalización multilineal de la mejor aproximación de rango  $R$  de una matriz, y se introducen dos descomposiciones que son la iteración ortogonal de orden superior y la *Proyección por Secciones (Slice Projection-SP)*, respectivamente.

### 4.1. Método de la potencia y de la iteración ortogonal

En esta sección se discutirá el método de la potencia y el método de la iteración ortogonal, donde el primero es utilizado para calcular de manera aproximada el valor singular y vectores singulares dominantes de una matriz; y el segundo, es utilizado para calcular los subespacios

fila y columnas dominantes de dimensión  $R$  de una matriz ([11], véase sección 7.3).

### 4.1.1. Método de la potencia

Suponga que  $\mathbf{A} \in \mathbb{C}^{n \times n}$  es diagonalizable, con Descomposición en Valores Propios (Eigenvalue Decomposition - EVD) dada por  $\mathbf{X}^{-1} \mathbf{A} \mathbf{X} = \mathbf{diag}(\lambda_1, \dots, \lambda_n)$  con  $\mathbf{X} = [X_1, \dots, X_n]$ , y  $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$ . Dado un vector norma-2 unitario  $U^{(0)} = \sum_i \alpha_i X_i$  con  $\alpha_1 \neq 0$ , el método de la potencia consiste en la siguiente iteración:

$$\begin{aligned}\tilde{U}^{(k)} &= \mathbf{A} U^{(k-1)} \\ U^{(k)} &= \tilde{U}^{(k)} / \left\| \tilde{U}^{(k)} \right\| \\ \lambda^{(k)} &= [U^{(k)}]^H \mathbf{A} U^{(k)}\end{aligned}$$

Según Golub, la efectividad del método depende primordialmente en que la matriz  $\mathbf{A}$  sea diagonalizable y que  $|\lambda_1| > |\lambda_2|$ ; la condición de que  $U^{(0)}$  es un vector unitario con  $\alpha_1 \neq 0$ , es menos importante dado que errores de redondeo sostenidos durante las iteraciones aseguran que los  $U^{(k)}$  subsecuentes tengan una componente en esta dirección.

El anterior algoritmo converge al valor y vector propio dominante de una matriz diagonalizable, por lo que puede ser usado para calcular el valor singular y vector singular izquierdo y derecho de una matriz, tal como se muestra a continuación:

$$\begin{aligned}\tilde{U}^{(k)} &= \mathbf{A} \tilde{V}^{(k-1)} \\ U^{(k)} &= \tilde{U}^{(k)} / \left\| \tilde{U}^{(k)} \right\| \\ \lambda^{(k)} &= [U^{(k)}]^H \mathbf{A} \tilde{U}^{(k)} \\ \tilde{V}^{(k)} &= \mathbf{A}^H U^{(k)} \\ V^{(k)} &= \tilde{V}^{(k)} / \left\| \tilde{V}^{(k)} \right\| \\ \mu^{(k)} &= [\tilde{V}^{(k)}]^H \mathbf{A}^H V^{(k)}\end{aligned}$$

Donde  $\mathbf{A} \in \mathbb{C}^{m \times n}$ ,  $U^{(0)} = e_1 \in \mathbb{C}^m$  y  $V^{(0)} = e_1 \in \mathbb{C}^n$  ( $e_1$  vector canónico)<sup>1</sup>; adicionalmente, se cumple que  $\lambda^{(k)}$  y  $\mu^{(k)}$  convergen al valor singular dominante. Ahora, existe otra variante del método de la potencia para calcular la tripleta dominante de una matriz, introducido por De Lathauwer en [18]. La importancia de esta variante radica en que ésta permite una directa generalización del método de la potencia matricial al método de la potencia de orden superior, tal como se mostrará en la siguiente sección. El algoritmo se visualiza a continuación:

<sup>1</sup> $U^{(0)}$  y  $V^{(0)}$  pueden ser cualquier vector de norma-2 unitario de  $\mathbb{C}^m$  y  $\mathbb{C}^n$  respectivamente.

$$\begin{aligned}
\tilde{V}^{(k)} &= \mathbf{A} U^{(k)} \\
\lambda^{(k)} &= \left\| \tilde{V}^{(k)} \right\| \\
V^{(k)} &= \tilde{V}^{(k)} / \lambda^{(k)} \\
\tilde{U}^{(k)} &= \mathbf{A}^H V^{(k-1)} \\
\mu^{(k)} &= \left\| \tilde{U}^{(k)} \right\| \\
U^{(k)} &= \tilde{U}^{(k)} / \mu^{(k)}
\end{aligned}$$

Los vectores  $U^{(k)}$  y  $V^{(k)}$  convergen al vector singular derecho e izquierdo dominantes; adicionalmente,  $\lambda^{(k)}$  y  $\mu^{(k)}$  convergen al valor singular dominante.

### 4.1.2. Método de la iteración ortogonal

Una generalización directa del método de la potencia puede ser usada para calcular subespacios invariantes de mayor dimensión. Considere  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , con EVD dada por  $\mathbf{X}^{-1} \mathbf{A} \mathbf{X} = \mathbf{diag}(\lambda_1, \dots, \lambda_n)$ ,  $\mathbf{X} = [X_1, \dots, X_n]$ , y  $|\lambda_1| \geq \dots \geq |\lambda_r| > |\lambda_{r+1}| \geq \dots \geq |\lambda_n|$ . Sea  $\mathbf{Q}_0 \in \mathbb{R}^{m \times r}$  una matriz con columnas ortogonales, el método de la potencia consiste en la siguiente iteración:

$$\begin{aligned}
\mathbf{Z}_k &= \mathbf{A} \mathbf{Q}_{k-1} \\
\mathbf{Q}_k \mathbf{R}_k &= \mathbf{Z}_k
\end{aligned}$$

Donde el último paso es la factorización QR. Observe que si  $r = 1$ , se obtiene el método de la potencia. Reemplazando  $\mathbf{A}$  por  $\mathbf{A} \mathbf{A}^T$  el algoritmo anterior puede ser utilizado para calcular el subespacio invariante dominante de dimensión  $r$  del espacio columna de la matriz  $\mathbf{A}$ .

## 4.2. Método de la potencia de orden superior

En esta sección, se busca obtener la mejor aproximación de un tensor dado, en el sentido de mínimos cuadrados, por un tensor de rango uno. Este problema, fue inicialmente abordado por De Lathauwer en [18], donde formula el problema utilizando la técnica de multiplicadores de Lagrange. En la sección 4.2.2., se mostrará que una aproximación puede ser obtenida a través de una generalización del método de la potencia para tensores de orden superior; esta generalización, utiliza como aproximación inicial la truncación de la HOSVD, el cual corresponde a una muy buena aproximación de bajo rango.

### 4.2.1. La mejor aproximación de rango uno

El problema que se desea resolver, es formulado de la siguiente forma

Dado un tensor real  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ , encuentre un escalar  $\lambda$  y vectores unitarios  $U^{(1)}, U^{(2)}, \dots, U^{(N)}$  tales que el tensor de rango uno  $\widehat{\mathcal{A}} \stackrel{\text{def}}{=} \lambda U^{(1)} \circ U^{(2)} \circ \dots \circ U^{(N)}$  minimice la siguiente función cuadrática

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A} - \widehat{\mathcal{A}}\|^2 \quad (4.1)$$

sobre el variedad de tensores de rango uno.

Este problema de optimización es analizado por De Lathauwer en [18], utilizando la técnica de los multiplicadores de Langrange, concluyendo que la minimización de la función  $f$  es equivalente a la maximización de la siguiente función:

$$g(U^{(1)}, U^{(2)}, \dots, U^{(N)}) = \left| \mathcal{A} \times_1 U^{(1)T} \times_2 U^{(2)T} \dots \times_N U^{(N)T} \right|^2 \quad (4.2)$$

sobre los vectores unitarios  $U^{(1)}, U^{(2)}, \dots, U^{(N)}$ , si el escalar  $\lambda$  es escogido de acuerdo a las siguientes ecuaciones

$$\mathcal{A} \times_1 U^{(1)T} \dots \times_{n-1} U^{(n-1)T} \times_{n+1} U^{(n+1)T} \dots \times_N U^{(N)T} = \lambda U^{(n)} \quad (4.3)$$

$$\mathcal{A} \times_1 U^{(1)T} \times_2 U^{(2)T} \dots \times_N U^{(N)T} = \lambda \quad (4.4)$$

$$\|U^{(n)}\| = 1 \quad (4.5)$$

para  $(1 \leq n \leq N)$ , llamadas ecuaciones de *Lagrange*. Teniendo presente lo anterior, las funciones de las Ecs. 4.1 y 4.2 están relacionadas por

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A}\|^2 - g$$

dado que  $U^{(1)}, U^{(2)}, \dots, U^{(N)}$  son vectores de norma uno,  $\|\widehat{\mathcal{A}}\|^2 = \lambda^2$  y

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A} - \widehat{\mathcal{A}}\|^2 = \|\mathcal{A}\|^2 - 2 \langle \mathcal{A}, \widehat{\mathcal{A}} \rangle + \|\widehat{\mathcal{A}}\|^2$$



### 4.2.2. Generalización del método de la potencia para tensores de orden superior

Suponiendo que los vectores  $U^{(1)}, \dots, U^{(n-1)}, U^{(n+1)}, \dots, U^{(N)}$  son fijos y que la función  $f$  en la Ec. 4.1 es solamente una función cuadrática del vector desconocido  $\lambda U^{(n)}$ , entonces las ecuaciones de Lagrange pueden ser enlazadas en un algoritmo de mínimos cuadrados alternantes (Alternating Least Squares-ALS) para la minimización local de la función  $f$ , tal como se describe en el algoritmo 4.1.

---

**Algoritmo 4.1**  
*Método de la Potencia de Orden Superior*

---

**Entrada:** Tensor  $\mathcal{A}$  de orden  $I_1 \times I_2 \times I_3$ —tensor  
Inicializar  $U_0^{(n)} \in \mathbb{R}^{I_n}$  ( $1 \leq n \leq 3$ )  
mientras  $k \leq 50$

$$\begin{aligned} \tilde{U}_{k+1}^{(1)} &= \mathbf{A}_{(1)} \cdot (U_k^{(2)} \otimes U_k^{(3)} \dots \otimes U_k^{(N)}) \\ \lambda_{k+1}^{(1)} &= \|\tilde{U}_{k+1}^{(1)}\| \\ U_{k+1}^{(1)} &= \tilde{U}_{k+1}^{(1)} / \lambda_{k+1}^{(1)} \\ \tilde{U}_{k+1}^{(2)} &= \mathbf{A}_{(2)} \cdot (U_k^{(3)} \otimes \dots \otimes U_k^{(N)} \otimes U_k^{(1)}) \\ \lambda_{k+1}^{(2)} &= \|\tilde{U}_{k+1}^{(2)}\| \\ U_{k+1}^{(2)} &= \tilde{U}_{k+1}^{(2)} / \lambda_{k+1}^{(2)} \\ &\dots \\ \tilde{U}_{k+1}^{(N)} &= \mathbf{A}_{(N)} \cdot (U_k^{(1)} \otimes U_k^{(2)} \dots \otimes U_k^{(N-1)}) \\ \lambda_{k+1}^{(N)} &= \|\tilde{U}_{k+1}^{(N)}\| \\ U_{k+1}^{(N)} &= \tilde{U}_{k+1}^{(N)} / \lambda_{k+1}^{(N)} \\ \lambda &= U_{k+1}^{(1)T} \cdot \mathbf{A}_{(1)} \cdot (U_{k+1}^{(2)} \otimes U_{k+1}^{(3)} \dots \otimes U_{k+1}^{(N)}) \end{aligned}$$

si ( $\Delta fit < \epsilon$ ) termine  
fin

---

**Resultado:** Vectores unitarios  $U^{(1)}, U^{(2)}, \dots, U^{(N)}$  y  $\lambda$ .

---

En cada paso del algoritmo 4.1, la estimación del escalar  $\lambda$  y la estimación de uno de los vectores  $U^{(1)}, \dots, U^{(N)}$  es optimizada, mientras los otros vectores son tomados constantes. Cabe resaltar que la expresión

$$\tilde{U}_{k+1}^{(n)} = \mathcal{A} \times_1 U_{k+1}^{(1)T} \dots \times_{n-1} U_{k+1}^{(n-1)T} \times_{n+1} U_k^{(n+1)T} \dots \times_N U_k^{(N)T}$$

puede ser escrita en forma matricial como

$$\tilde{U}_{k+1}^{(n)} = \mathbf{A}_{(n)} \cdot (U_{k+1}^{(n+1)} \otimes \dots \otimes U_{k+1}^{(N)} \otimes U_k^{(1)} \otimes \dots \otimes U_k^{(n-1)})$$

La anterior ecuación fácilmente se puede demostrar siguiendo un procedimiento similar al realizado en la demostración de la proposición 3.1. Igualmente cabe mencionar, que el algoritmo 4.1, programado en la rutina *hopm.m*, es una generalización de orden superior del método de la potencia matricial, si se toma en cuenta la versión matricial descrita por De Lathauwer.

**Observación 4.1** *La función de la Ec. 4.2 es la generalización de la función  $|U^T \mathbf{A} V|$ , donde encontrar el máximo de dicha función sobre los vectores unitarios  $U$  y  $V$ , es equivalente a encontrar la mejor aproximación de rango uno de una matriz  $\mathbf{A}$ . En el caso matricial, las ecuaciones de Lagrange (Ec.4.3-4.5) poseen una única solución estable, el cual, es obtenida con la truncación de la SVD después del primer valor singular.*

**Observación 4.2** *A diferencia de su contraparte matricial, en tensores de orden superior, las ecuaciones de Lagrange pueden tener varias soluciones estables, por tanto, cualquier algoritmo descendente, como el método de la potencia de orden superior, solo logrará descubrir el máximo global si el punto inicial está dentro de la región de atracción del máximo global.*

*En el capítulo anterior, se mencionó que la propiedad 2.3.5 generaliza el concepto de mejor aproximación de bajo rango de una matriz, en el sentido de mínimos cuadrados, donde a diferencia del caso matricial, la truncación de la HOSVD después del primer valor singular  $n$ -modal no necesariamente conlleva a la mejor aproximación de rango uno. Sin embargo, De Lathauwer menciona que usualmente la estimación proporcionada por la truncación de la HOSVD pertenece a la región de atracción de la mejor aproximación de rango uno, por lo que recomienda usar la HOSVD como valor inicial para el método de la potencia descrito anteriormente. De igual forma, menciona que pueden existir casos donde eventualmente se puede conseguir un óptimo global aparente, pero aun así, diferente al óptimo global.*

### 4.3. Iteración ortogonal de orden superior

En esta sección, se busca obtener la mejor aproximación de un tensor dado, en el sentido de mínimos cuadrados, por un tensor con rangos  $n$ -modales preespecificados, el cual será llamado tensor  $rank - (R_1, R_2, \dots, R_N)$ . Este problema es abordado siguiendo un esquema similar al tomado en la sección anterior, donde se obtiene un algoritmo descendente el cual es considerado una generalización al método de la iteración ortogonal.

#### 4.3.1. Mejor aproximación de rango $-(R_1, \dots, R_N)$

El problema que se desea resolver, es planteado de la siguiente forma:

*Dado un tensor real  $A \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ , encuentre un tensor  $\hat{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  con  $rank_1(\hat{\mathcal{A}}) = R_1, rank_2(\hat{\mathcal{A}}) = R_2, \dots, rank_N(\hat{\mathcal{A}}) = R_N$ , que minimice la siguiente función cuadrática*

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A} - \widehat{\mathcal{A}}\|^2 \quad (4.6)$$

El tensor  $\widehat{\mathcal{A}}$  deseado puede ser descompuesto como

$$\widehat{\mathcal{A}} = \mathbf{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_N \mathbf{U}^{(N)} \quad (4.7)$$

donde  $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}$ ,  $\mathbf{U}^{(2)} \in \mathbb{R}^{I_2 \times R_2}$ ,  $\dots$ ,  $\mathbf{U}^{(N)} \in \mathbb{R}^{I_N \times R_N}$  son matrices con columnas ortonormales y  $\mathbf{B} \in \mathbb{R}^{R_1 \times R_2 \times \dots \times R_N}$ .

Reemplazando la expresión 4.7 en la función  $f$ , se tiene que

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A} - \mathbf{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_N \mathbf{U}^{(N)}\|^2 \quad (4.8)$$

Por tanto, dadas matrices  $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$ , la función  $f$  es simplemente un problema clásico lineal de mínimos cuadrados en función de las componentes del tensor  $\mathbf{B}$ , donde su optimización, corresponde a resolver el siguiente conjunto de ecuaciones lineales

$$\mathcal{A} = \mathbf{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_N \mathbf{U}^{(N)} \quad (4.9)$$

Teniendo presente que las columnas de las matrices  $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$  son ortonormales se cumple

$$\mathbf{B} = \mathcal{A} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \dots \times_N \mathbf{U}^{(N)T} \quad (4.10)$$

Dado lo anterior, solo es suficiente determinar las matrices  $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(N)}$  para optimizar la función  $f$ . De manera análoga a la sección anterior, la minimización de la función  $f$  equivale a la maximización de la siguiente función

$$g(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}) = \left\| \mathcal{A} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \dots \times_N \mathbf{U}^{(N)T} \right\|^2 \quad (4.11)$$

sobre las matrices con columnas ortonormales  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$ , si el tensor  $\mathcal{A}$  es escogido de acuerdo a la Ec. 4.10. Las funciones  $f$  y  $g$  están relacionadas por

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A}\|^2 - g$$

dado que  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$  son matrices con columnas ortonormales,  $\|\widehat{\mathcal{A}}\|^2 = \|\mathcal{B}\|^2$  y

$$f(\widehat{\mathcal{A}}) = \|\mathcal{A} - \widehat{\mathcal{A}}\|^2 = \|\mathcal{A}\|^2 - 2 \langle \mathcal{A}, \widehat{\mathcal{A}} \rangle + \|\widehat{\mathcal{A}}\|^2$$

$$\begin{aligned} \langle \mathcal{A}, \widehat{\mathcal{A}} \rangle &= \langle \mathcal{A}, \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \dots \times_N \mathbf{U}^{(N)} \rangle \\ &= \langle \mathcal{A} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \dots \times_N \mathbf{U}^{(N)T}, \mathcal{B} \rangle = \|\mathcal{B}\|^2 \end{aligned}$$

**Observación 4.3** *Básicamente, el problema de la mejor aproximación de rango-( $R_1, \dots, R_N$ ) busca determinar que tanta energía del tensor  $\mathcal{A}$  se puede incorporar en un tensor  $\widehat{\mathcal{A}}$  de rango  $n$ -modal reducido bajo las restricciones de rango  $n$ -modales dadas. Por otra parte, dadas las Ec. 4.4 y 4.10, se puede observar que el tensor  $\mathcal{B}$  es el equivalente multidimensional del escalar  $\lambda$ .*

### 4.3.2. Iteración ortogonal de orden superior

Suponiendo que las matrices  $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \dots, \mathbf{U}^{(N)}$  son fijas y que la función  $g$  en la Ec. 4.11 es solamente una función cuadrática de las componentes desconocidas de la matriz  $\mathbf{U}^{(n)}$ , entonces

$$g(\mathbf{U}^{(n)}) = \left\| \mathbf{U}^{(n)T} \cdot \tilde{\mathbf{U}}^{(n)} \right\|^2 \quad (4.12)$$

en donde

$$\tilde{\mathbf{U}}^{(n)} = \mathcal{A} \times_1 \mathbf{U}^{(1)T} \dots \times_{n-1} \mathbf{U}^{(n-1)T} \times_{n+1} \mathbf{U}^{(n+1)T} \dots \times_N \mathbf{U}^{(N)T}$$

Por lo tanto, los vectores columna de la matriz  $\mathbf{U}^{(n)}$  puede encontrarse como una base ortogonal para el subespacio dominante del espacio columna de la matriz  $\tilde{\mathbf{U}}^{(n)}$ . Repitiendo este procedimiento por cada modo se obtiene un algoritmo ALS para la minimización local de  $f$ , véase algoritmo 4.2, donde en cada paso se optimiza una de las matrices  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$ , mientras se dejan las demas constantes. Cabe mencionar, que la expresión

$$\tilde{\mathbf{U}}_{k+1}^{(n)} = \mathcal{A} \times_1 \mathbf{U}_{k+1}^{(1)T} \dots \times_{n-1} \mathbf{U}_{k+1}^{(n-1)T} \times_{n+1} \mathbf{U}_k^{(n+1)T} \dots \times_N \mathbf{U}_k^{(N)T}$$

puede ser escrita en forma matricial como

$$\tilde{\mathbf{U}}_{k+1}^{(n)} = \mathbf{A}_{(n)} \cdot (\mathbf{U}_{k+1}^{(n+1)} \otimes \dots \otimes \mathbf{U}_{k+1}^{(N)} \otimes \mathbf{U}_k^{(1)} \otimes \dots \otimes \mathbf{U}_k^{(n-1)})$$

El algoritmo 4.2, es una generalización de orden superior del método de la potencia matricial, si se toma en cuenta la versión matricial descrita por De Lathauver. Este algoritmo fue programado en la rutina *hooi.m*.

**Algoritmo 4.2***Iteración Ortogonal de Orden Superior*


---

**Entrada:** Tensor  $\mathcal{A}$  de orden  $I_1 \times I_2 \times I_3$ -tensor y rango deseado- $(R_1, R_2, R_3)$

Inicializar  $\mathbf{U}_0^{(n)} \in \mathbb{R}^{I_n \times R_n} (1 \leq n \leq 3)$

mientras  $k \leq 50$

$\tilde{\mathbf{U}}_{k+1}^{(1)} = \mathbf{A}_{(1)} \cdot (\mathbf{U}_k^{(2)} \otimes \mathbf{U}_k^{(3)} \cdots \otimes \mathbf{U}_k^{(N)})$   
 $\mathbf{U}_{k+1}^{(1)} \leftarrow$  Asigne los primeros  $R_1$  VSD v.s izquierdos de  $\tilde{\mathbf{U}}_{k+1}^{(1)}$   
 $\tilde{\mathbf{U}}_{k+1}^{(2)} = \mathbf{A}_{(2)} \cdot (\mathbf{U}_k^{(3)} \otimes \cdots \otimes \mathbf{U}_k^{(N)} \otimes \mathbf{U}_{k+1}^{(1)})$   
 $\mathbf{U}_{k+1}^{(2)} \leftarrow$  Asigne los primeros  $R_2$  VSD v.s izquierdos de  $\tilde{\mathbf{U}}_{k+1}^{(2)}$   
 $\cdots$   
 $\tilde{\mathbf{U}}_{k+1}^{(N)} = \mathbf{A}_{(N)} \cdot (\mathbf{U}_{k+1}^{(1)} \otimes \mathbf{U}_{k+1}^{(2)} \cdots \otimes \mathbf{U}_{k+1}^{(N-1)})$   
 $\mathbf{U}_{k+1}^{(N)} \leftarrow$  Asigne los primeros  $R_N$  VSD v.s izquierdos de  $\tilde{\mathbf{U}}_{k+1}^{(N)}$   
 $\mathbf{B}_{(1)} = \mathbf{U}_{k+1}^{(1)T} \cdot \mathbf{A}_{(1)} \cdot (\mathbf{U}_{k+1}^{(2)} \otimes \mathbf{U}_{k+1}^{(3)} \cdots \otimes \mathbf{U}_{k+1}^{(N)})$   
 si  $(\Delta fit < \epsilon)$  termine

fin

**Resultado:** Matrices con vectores ortonormales  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$  y tensor  $\mathcal{B}$ .

---

**Observación 4.4** En el algoritmo 4.2, no es necesario que las columnas de  $\mathbf{U}_k^{(n)}$  ( $k \geq 0$  y  $1 \leq n \leq N$ ) coincidan con los vectores singulares izquierdos de la matriz  $\tilde{\mathbf{U}}_k^{(n)}$ , solo es suficiente que éstas sean una base ortonormal arbitraria para el subespacio dominante de dimensión  $R_n$  del espacio columna de  $\tilde{\mathbf{U}}_k^{(n)}$ .

**Observación 4.5** Aunque De Lathauwer en [18] no menciona ningún criterio de parada cuando introduce el HOOI, se utilizó un criterio desarrollado por Bader y Kolda en [7], el cual está basado en la siguiente función de ajuste entre el tensor  $\hat{\mathcal{A}}$  y  $\mathcal{A}$

$$fit(\mathcal{A}, \hat{\mathcal{A}}) = 1 - \frac{\|\mathcal{A} - \hat{\mathcal{A}}\|}{\|\mathcal{A}\|} \quad (4.13)$$

El lazo principal en el algoritmo 4.2, se termina una vez que se alcance el número máximo de iteraciones ó cuando  $\Delta fit(k) = fit(\mathcal{A}, \hat{\mathcal{A}}_k) - fit(\mathcal{A}, \hat{\mathcal{A}}_{k-1})$  sea menor que  $10^{-5}$ . Usualmente, el lazo principal termina con un número de iteraciones menor que el umbral y con un  $\Delta fit(k) < 10^{-5}$ .

**Observación 4.6** Básicamente dado un tensor  $\mathcal{A}$ , la idea del algoritmo HOOI es calcular cada matriz  $\mathbf{U}^{(n)}$  a través del tensor  $\mathcal{A}$  y de las matrices  $\mathbf{U}^{(1)}, \dots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \dots, \mathbf{U}^{(N)}$ ,

es decir, optimizar un modo utilizando la interacción de los otros, al desplegar los vectores  $n$ -modales en la matriz  $\mathbf{A}_n$  y proyectandola posteriormente sobre el espacio de dimensión  $R_1, \dots, R_{n-1}, R_{n+1}, \dots, R_N$ .

## 4.4. Proyección por secciones

En esta sección, se introduce un nuevo algoritmo para obtener aproximaciones de *rango*  $(R_1, R_2, \dots, R_N)$  para tensores de orden tres, llamado *Proyección por Secciones* (*Slice Projection - SP*) ([32]). El algoritmo es abordado siguiendo un esquema totalmente distinto al utilizado por De Lathauwer en HOOI, y se obtiene, un algoritmo más eficiente en uso de memoria y en tiempos de cómputo, pero con el defecto de que solo trabaja con tensores de orden tres. En el algoritmo HOOI, el tensor es desplegado a través de los diferentes ejes coordenados para formular matrices formadas por los vectores  $n$ -modales, mientras que en el algoritmo SP el tensor es representado a través de *secciones* a lo largo de los diferentes ejes coordenados, donde cada *sección*, es representada por una matriz ortogonal en esa dirección, véase sección 2.1.1,  $\mathbf{A}_{i::}$ ,  $\mathbf{A}_{:j}$  y  $\mathbf{A}_{::k}$ , donde  $1 \leq i \leq I_1$ ,  $1 \leq j \leq I_2$  y  $1 \leq k \leq I_3$ .

En este enfoque, fijada una dirección, se proyecta cada *sección* del tensor original sobre los otros dos ejes coordenados restantes, con el fin de aproximarlos a través de sus *secciones* originales; y por ende, se tiene que la maximización de la norma de la mejor aproximación del tensor original sea establecida como la maximización de la suma de las normas de las *secciones* proyectadas en las tres direcciones.

### Formulación del problema

El problema que se desea resolver, es formulado de la siguiente forma:

Dado un tensor real  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ , encuentre matrices  $\mathbf{U}^{(1)}$ ,  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$  que son solución de

$$\max_{\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}} \sum_{i=1}^{I_1} \left\| \mathbf{U}^{(2)T} \mathbf{A}_{i::} \mathbf{U}^{(3)} \right\|^2 + \sum_{j=1}^{I_2} \left\| \mathbf{U}^{(1)T} \mathbf{A}_{:j} \mathbf{U}^{(3)} \right\|^2 + \sum_{k=1}^{I_3} \left\| \mathbf{U}^{(1)T} \mathbf{A}_{::k} \mathbf{U}^{(2)} \right\|^2 \quad (4.14)$$

donde  $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}$ ,  $\mathbf{U}^{(2)} \in \mathbb{R}^{I_2 \times R_2}$  y  $\mathbf{U}^{(3)} \in \mathbb{R}^{I_3 \times R_3}$  tienen columnas ortonormales.

El siguiente teorema, describe como encontrar localmente una solución óptima usando un procedimiento iterativo (ALS).

**Teorema 4.1** Sean  $\mathbf{U}^{(1)}$ ,  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$  la solución óptima al problema de maximización dado en la Ec. 4.14, entonces

- Dados  $\mathbf{U}^{(1)}$  y  $\mathbf{U}^{(2)}$ ,  $\mathbf{U}^{(3)}$  consiste de los primeros  $R_3$  vectores propios de la matriz  $\mathbf{M}_{31} = \sum_{j=1}^{I_2} \mathbf{A}_{:j}^T \mathbf{U}^{(1)} \mathbf{U}^{(1)T} \mathbf{A}_{:j}$  (y  $\mathbf{M}_{32} = \sum_{i=1}^{I_1} \mathbf{A}_{i::}^T \mathbf{U}^{(2)} \mathbf{U}^{(2)T} \mathbf{A}_{i::}$ ).
- Dados  $\mathbf{U}^{(1)}$  y  $\mathbf{U}^{(3)}$ ,  $\mathbf{U}^{(2)}$  consiste de los primeros  $R_2$  vectores propios de la matriz  $\mathbf{M}_{23} = \sum_{i=1}^{I_1} \mathbf{A}_{i::} \mathbf{U}^{(3)} \mathbf{U}^{(3)T} \mathbf{A}_{i::}^T$  (y  $\mathbf{M}_{21} = \sum_{k=1}^{I_3} \mathbf{A}_{::k}^T \mathbf{U}^{(1)} \mathbf{U}^{(1)T} \mathbf{A}_{::k}$ ).
- Dados  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$ ,  $\mathbf{U}^{(1)}$  consiste de los primeros  $R_1$  vectores propios de la matriz  $\mathbf{M}_{12} = \sum_{k=1}^{I_3} \mathbf{A}_{::k} \mathbf{U}^{(2)} \mathbf{U}^{(2)T} \mathbf{A}_{::k}^T$  (y  $\mathbf{M}_{13} = \sum_{j=1}^{I_2} \mathbf{A}_{:j} \mathbf{U}^{(3)} \mathbf{U}^{(3)T} \mathbf{A}_{:j}^T$ ).

Dados  $\mathbf{U}^{(1)}$ ,  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$ , la proyección de  $\mathcal{A}$  sobre el espacio de ejes coordenados esta representada por:  $\mathcal{B} = \mathcal{A} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \times_3 \mathbf{U}^{(3)T}$ .

**Prueba.** Dados  $\mathbf{U}^{(1)}$  y  $\mathbf{U}^{(2)}$ , se desea encontrar  $\mathbf{U}^{(3)}$  que solucione

$$\max_{\mathbf{U}^{(3)}} \sum_{j=1}^{I_2} \left\| \mathbf{U}^{(1)T} \mathbf{A}_{:j} \mathbf{U}^{(3)} \right\|^2 \quad \text{y} \quad \max_{\mathbf{U}^{(3)}} \sum_{i=1}^{I_1} \left\| \mathbf{U}^{(2)T} \mathbf{A}_{i::} \mathbf{U}^{(3)} \right\|^2 \quad (4.15)$$

El primer termino de la Ec. 4.15 se escribir como

$$\begin{aligned} \sum_{j=1}^{I_2} \left\| \mathbf{U}^{(1)T} \mathbf{A}_{:j} \mathbf{U}^{(3)} \right\|^2 &= \sum_{j=1}^{I_2} \text{traza} \left( \mathbf{U}^{(3)T} \mathbf{A}_{:j}^T \mathbf{U}^{(1)} \mathbf{U}^{(1)T} \mathbf{A}_{:j} \mathbf{U}^{(3)} \right) \\ &= \text{traza} \left( \mathbf{U}^{(3)T} \left( \sum_{j=1}^{I_2} \mathbf{A}_{:j}^T \mathbf{U}^{(1)} \mathbf{U}^{(1)T} \mathbf{A}_{:j} \right) \mathbf{U}^{(3)} \right) \\ &= \text{traza} \left( \mathbf{U}^{(3)T} \mathbf{M}_{31} \mathbf{U}^{(3)} \right) \end{aligned}$$

El cual se tiene que es máxima para  $\mathbf{U}^{(1)}$  solo si  $\mathbf{U}^{(3)} \in \mathbb{R}^{I_3 \times R_3}$  consiste de los primeros  $R_3$  vectores propios de la matriz  $\mathbf{M}_{31}$ . Siguiendo un procedimiento similiar, en el segundo termino de la Ec. 4.15 se obtiene que  $\mathbf{U}^{(3)}$  consiste de los primeros  $R_3$  vectores propios de la matriz  $\mathbf{M}_{32}$ . En cualquier caso,  $\mathbf{U}^{(3)}$  es localmente optimizado por la maximización de la Ec. 4.15. De igual forma, se pueden probar las otras partes del teorema. ■

Este teorema suministra un procedimiento iterativo para encontrar  $\mathbf{U}^{(1)}$ ,  $\mathbf{U}^{(2)}$  y  $\mathbf{U}^{(3)}$ , el cual converge al máximo local de la Ec. 4.14; este procedimiento, descrito en el algoritmo 4.3, fue programado en la rutina *sp.m*.

**Algoritmo 4.3***Proyección por secciones***Entrada:** Tensor  $\mathcal{A}$  de orden  $I_1 \times I_2 \times I_3$ -tensor y rango deseado- $(R_1, R_2, R_3)$ Inicializar  $\mathbf{U}_0^{(n)} \in \mathbb{R}^{I_n \times R_n} (1 \leq n \leq 3)$ **mientras**  $k \leq 50$ 

$$\mathbf{M}_{21} = \sum_{k=1}^{I_3} \mathbf{A}_{::k}^{(T)} \mathbf{U}_s^{(1)} \mathbf{U}_s^{(1)T} \mathbf{A}_{::k}$$

$$U_{s+1}^{(2)} \leftarrow \text{Asigne los primeros } R_2 \text{ v.s izquierdos de } \mathbf{M}_{21}$$

$$\mathbf{M}_{32} = \sum_{i=1}^{I_1} \mathbf{A}_{i::}^{(T)} \mathbf{U}_{s+1}^{(2)} \mathbf{U}_{s+1}^{(2)T} \mathbf{A}_{i::}$$

$$U_{s+1}^{(3)} \leftarrow \text{Asigne los primeros } R_3 \text{ v.s izquierdos de } \mathbf{M}_{32}$$

...

$$\mathbf{M}_{13} = \sum_{j=1}^{I_2} \mathbf{A}_{:j}^{(T)} \mathbf{U}_{s+1}^{(3)} \mathbf{U}_{s+1}^{(3)T} \mathbf{A}_{:j}$$

$$U_{s+1}^{(1)} \leftarrow \text{Asigne los primeros } R_1 \text{ v.s izquierdos de } \mathbf{M}_{13}$$

$$\mathbf{B}_{(1)} = U_{s+1}^{(1)T} \cdot \mathbf{A}_{(1)} \cdot (U_{s+1}^{(2)} \otimes U_{s+1}^{(3)})$$

$$\text{si } (\Delta fit < \epsilon) \text{ termine}$$

**fin****Resultado:** Matrices con vectores ortonormales  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}$  y tensor  $\mathcal{B}$ .

**Observación 4.7** Al igual que el algoritmo HOOI, el SP aplica un procedimiento ALS para encontrar una solución óptima local a la Ec. 4.14, donde en cada paso, se optimiza solamente una matriz a través del tensor  $\mathcal{A}$  y una de las otras dos matrices. De esta forma, SP se ubica entre la HOSVD y HOOI.

**Observación 4.8** El criterio de convergencia utilizado por Wang y Ahuja en el SP, es distinto al utilizado en el algoritmo 4.3, el cual es dado por

$$(\|\mathcal{B}_{j+1}\|^2 - \|\mathcal{B}_j\|^2) < \varepsilon \quad (4.16)$$

para cierto  $\varepsilon$  suficientemente pequeño.

**Observación 4.9** Con respecto a la inicialización del algoritmo 4.3, Wang y Ahuja en [32] utilizan una inicialización dada por  $\mathbf{U}_0^{(n)} = [I_{R_n} \ 0]$  ó  $\mathbf{U}_0^{(n)}$  igual a una distribución uniforme de números aleatorios (con columnas no necesariamente ortonormales). Wang y Ahuja, establecen que en comparación con los otros métodos desarrollados, los resultados obtenidos, utilizando estas inicializaciones, no marcan una gran diferencia con respecto a los otros algoritmos.

**Observación 4.10** Aunque Wang y Ahuja en [32] solo discuten tensores de orden tres, Turney en [28] discute una generalización del algoritmo SP para tensores de orden cuatro, y



además, establece la necesidad de obtener una mejor notación, con el fin de generalizar *SP* a tensores de orden arbitrario. Igualmente, Turney en [28] realizó una mejora al algoritmo *SP* e introduce un nuevo algoritmo llamado *Multislice Projection-MP*, el cual toma la misma estructura del anterior, pero tomó tres ideas del *HOOI* que son: utilizar la *HOSVD* para inicializar el algoritmo, utilizar el criterio de convergencia descrito en la observación 4.5, y por último, utilizar todas las matrices restantes para optimizar una matriz dada; es decir, por ejemplo en la matriz  $\mathbf{U}^{(1)}$  se encuentra como los primeros  $R_1$  vectores propios de la siguiente matriz:

$$\mathbf{M}_1 = \sum_{k=1}^{I_3} \mathbf{A}_{::k} \mathbf{U}^{(2)} \mathbf{U}^{(2)T} \mathbf{A}_{::k}^T + \sum_{j=1}^{I_2} \mathbf{A}_{:j} \mathbf{U}^{(3)} \mathbf{U}^{(3)T} \mathbf{A}_{:j}^T$$

Lo anterior, permite que el algoritmo *MP*, según Turney, obtenga una mejor aproximación que el algoritmo *SP*. Turney en [28], establece con respecto a la precisión de los cuatro algoritmos que *HOOI* obtiene la mejor aproximación, seguido por *MP*, después por *SP* y por último *HOSVD*. Dado que la *HOSVD* y *HOOI* exceden fácilmente el uso de memoria *RAM* disponible con tensores de gran dimensión, Turney recomienda utilizar *HOOI* para tensores con dimensiones que no exceden la memoria *RAM* disponible y *MP* para tensores de gran dimensión.

**Ejemplo 4.1** Considere el tensor  $\mathcal{A}$  de orden  $6 \times 4 \times 3$  del ejemplo 3.1. La truncación de la *HOSVD* después del primer valor singular en cada modo, da un tensor de rango-1 cuya norma de Frobenius es igual a 4,4139. Recuerde que la norma del tensor  $\mathcal{A}$  está dada por 4,8333. La norma de la aproximación  $\hat{\mathcal{A}}$ , igual a  $\lambda$ , es incrementada hasta un valor de 4,4171 por el método de la potencia de orden superior. La evolución de la norma durante cada iteración en el algoritmo 4.1 se muestra en la Fig. 4.1, y las componentes de la aproximación de rango – 1 obtenidas con el método de la potencia son descritas a continuación:

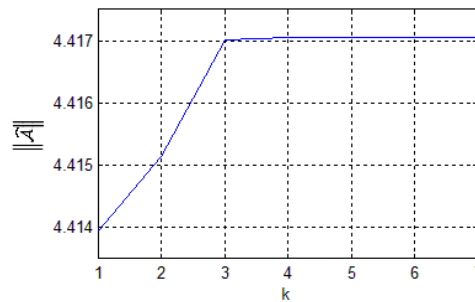


Figura 4.1: Evolución de la norma de la aproximación  $\hat{\mathcal{A}}$  durante las iteraciones del método de la potencia

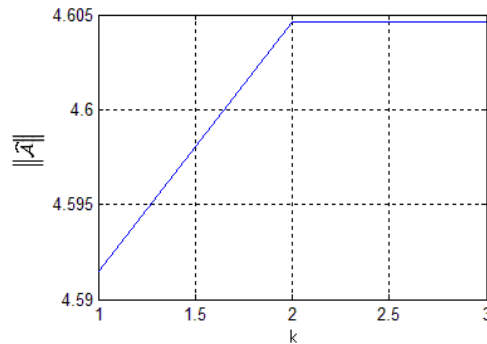


Figura 4.2: Evolución de la norma de la aproximación  $\hat{A}$  durante las iteraciones del método de la iteración ortogonal

$$U^{(1)} = \begin{bmatrix} 0,4163 \\ 0,4127 \\ 0,3931 \\ 0,3895 \\ 0,5097 \\ 0,3004 \end{bmatrix}, \quad U^{(2)} = \begin{bmatrix} 0,4689 \\ 0,5542 \\ 0,4726 \\ 0,4996 \end{bmatrix}, \quad U^{(3)} = \begin{bmatrix} 0,5642 \\ 0,5875 \\ 0,5875 \end{bmatrix}$$

Ahora aproximación de rango  $(4, 2, 2)$  obtenida por HOOI, representada por  $\mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$ , se muestra a continuación :

$$\mathbf{U}^{(1)} = \begin{bmatrix} -0,4182 & 0,0247 & 0,8261 & 0,0394 \\ -0,4207 & -0,1813 & -0,1063 & -0,4974 \\ -0,3960 & -0,3733 & -0,0816 & 0,7634 \\ -0,3677 & 0,7939 & -0,3237 & 0,2335 \\ -0,5096 & 0,1086 & -0,0247 & -0,3218 \\ -0,3105 & -0,4301 & -0,4407 & -0,1010 \end{bmatrix}, \quad \mathbf{U}^{(3)} = \begin{bmatrix} -0,5636 & 0,0960 \\ -0,5972 & -0,7336 \\ -0,5707 & 0,6728 \end{bmatrix}$$

$$\mathbf{U}^{(2)} = \begin{bmatrix} -0,4773 & 0,3507 \\ -0,5414 & -0,8198 \\ -0,4720 & 0,1168 \\ -0,5063 & 0,4373 \end{bmatrix}, \quad \mathbf{B}_{(1)} = \begin{bmatrix} -4,4150 & 0,0380 & | & -0,0535 & 0,6961 \\ 0,0929 & -0,4934 & | & 0,4624 & 0,6511 \\ -0,0169 & 0,2485 & | & 0,3959 & -0,0890 \\ 0,0374 & 0,2184 & | & -0,0865 & 0,2191 \end{bmatrix}$$

La norma de Frobenius de la aproximación es igual a 4,6046. La evolución de la norma durante cada iteración en el algoritmo 4.2 se muestra en la Fig. 4.2.

Por último, la aproximación de rango  $(4, 2, 2)$  obtenida por SP, representada por  $\hat{\mathcal{B}} \times_1$

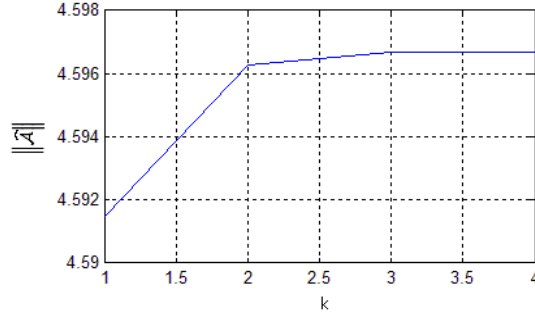


Figura 4.3: Evolución de la norma de la aproximación  $\hat{\mathcal{A}}$  durante las iteraciones del método SP.

$\mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$ , se muestra a continuación :

$$\mathbf{U}^{(1)} = \begin{bmatrix} -0,4193 & 0,0064 & -0,4775 & -0,3308 \\ -0,4210 & 0,2194 & -0,2376 & 0,1766 \\ -0,3951 & 0,3457 & 0,6740 & -0,5191 \\ -0,3695 & -0,8690 & 0,2912 & 0,0798 \\ -0,5081 & 0,0065 & -0,3146 & 0,0063 \\ -0,3100 & 0,2777 & 0,2783 & 0,7638 \end{bmatrix}, \quad \mathbf{U}^{(3)} = \begin{bmatrix} -0,5630 & 0,1369 \\ -0,5979 & -0,7483 \\ -0,5706 & 0,6490 \end{bmatrix}$$

$$\mathbf{U}^{(2)} = \begin{bmatrix} -0,4742 & -0,3931 \\ -0,5399 & 0,7958 \\ -0,4676 & -0,0130 \\ -0,5149 & -0,4605 \end{bmatrix}, \quad \hat{\mathbf{B}}_{(1)} = \begin{bmatrix} -4,4148 & 0,0381 & | & 0,0724 & -0,6836 \\ -0,0834 & 0,4962 & | & 0,3995 & 0,6864 \\ 0,0251 & 0,0658 & | & 0,3252 & -0,1749 \\ -0,0208 & -0,1988 & | & 0,2863 & 0,1335 \end{bmatrix}$$

La norma de Frobenius de la aproximación es igual a 4,5967. La evolución de la norma durante cada iteración en el algoritmo 4.2 se muestra en la Fig. 4.3.

**Ejemplo 4.2** Considere el tensor  $\mathcal{A}$  de orden  $5 \times 6 \times 2$  del ejemplo 3.2. La aproximación de rango  $-1$  dada por la HOSVD después de la truncación del primer valor singular en cada modo, tiene norma de Frobenius igual a 15,1405, mientras, la norma del tensor  $\mathcal{A}$  esta dada por 22,9347. La norma de la aproximación, obtenida con el método de la potencia de orden superior, tiene norma de Frobenius dada por 15,2547. La evolución de la norma durante cada iteración en el algoritmo 4.1 se muestra en la Fig. 4.4, y las componentes de la aproximación de rango  $-1$  obtenidas con el método de la potencia son descritas a continuación:

$$U^{(1)} = \begin{bmatrix} -0,0047 \\ 0,4523 \\ 0,4493 \\ 0,6393 \\ -0,4299 \end{bmatrix}, \quad U^{(2)} = \begin{bmatrix} -0,1271 \\ -0,4817 \\ 0,3719 \\ 0,6256 \\ 0,2041 \\ -0,4248 \end{bmatrix}, \quad U^{(3)} = [0,9968]$$

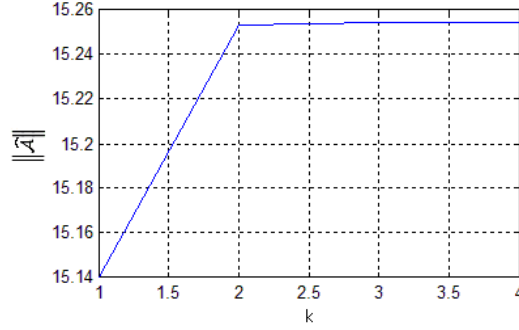


Figura 4.4: Evolución de la norma de la aproximación  $\hat{A}$  durante las iteraciones del método de la potencia.

Ahora aproximación de rango  $(2, 4, 1)$  obtenida por HOOI, representada por  $\mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$ , se muestra a continuación :

$$\mathbf{B}_{(1)} = \begin{bmatrix} -15,2410 & -0,0014 & 0,0005 & -0,0003 \\ 0,0020 & 10,8502 & -0,0002 & 0,0002 \end{bmatrix}, \quad \mathbf{U}^{(3)} = \begin{bmatrix} -0,0261 \\ 0,9997 \end{bmatrix}$$

$$\mathbf{U}^{(1)} = \begin{bmatrix} -0,0088 & -0,1546 \\ 0,4748 & -0,7710 \\ 0,4290 & 0,4382 \\ 0,6263 & 0,4017 \\ -0,4452 & 0,1682 \end{bmatrix}, \quad \mathbf{U}^{(2)} = \begin{bmatrix} -0,1346 & -0,3503 & -0,1356 & -0,4393 \\ -0,4811 & -0,2559 & -0,2870 & 0,6218 \\ 0,4010 & 0,6713 & -0,0762 & 0,2519 \\ 0,6071 & -0,5151 & 0,0384 & 0,4916 \\ 0,2028 & -0,2536 & 0,6659 & -0,1345 \\ -0,4242 & 0,1774 & 0,6698 & 0,3116 \end{bmatrix}$$

La norma de Frobenius de la aproximación es igual a 18,7087. La evolución de la norma durante cada iteración en el algoritmo 4.2 se muestra en la Fig. 4.5.

Por último, la aproximación de rango  $(4, 2, 2)$  obtenida por SP, representada por  $\hat{\mathcal{B}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$ , se muestra a continuación :

$$\hat{\mathbf{B}}_{(1)} = \begin{bmatrix} -15,2410 & -0,0014 & 0,0005 & -0,0003 \\ 0,0020 & 10,8502 & -0,0002 & 0,0002 \end{bmatrix}, \quad \mathbf{U}^{(3)} = \begin{bmatrix} -0,0261 \\ 0,9997 \end{bmatrix}$$

$$\mathbf{U}^{(1)} = \begin{bmatrix} 0,0077 & -0,1529 \\ -0,4672 & -0,7766 \\ -0,4359 & 0,4354 \\ -0,6308 & 0,3927 \\ 0,4402 & 0,1723 \end{bmatrix}, \quad \mathbf{U}^{(2)} = \begin{bmatrix} 0,1117 & 0,3128 & 0,1955 & -0,7389 \\ 0,4908 & 0,1899 & 0,5994 & -0,1122 \\ -0,4150 & 0,6713 & -0,0762 & 0,2519 \\ 0,6071 & -0,5151 & 0,0384 & 0,4916 \\ 0,2028 & -0,2536 & 0,6659 & -0,1345 \\ -0,4242 & 0,1774 & 0,6698 & 0,3116 \end{bmatrix}$$

La norma de Frobenius de la aproximación es igual a 18,7065. La evolución de la norma durante cada iteración en el algoritmo 4.2 se muestra en la Fig. 4.6.

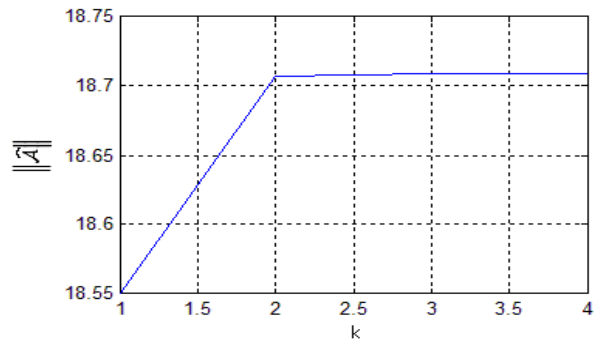


Figura 4.5: Evolución de la norma de la aproximación  $\hat{\mathcal{A}}$  durante las iteraciones del método de la iteración ortogonal

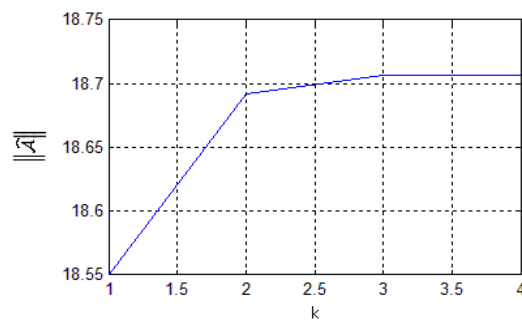


Figura 4.6: Evolución de la norma de la aproximación  $\hat{\mathcal{A}}$  durante las iteraciones del método SP.

## 4.5. Conclusiones

En este capítulo, se abordó el problema de la generalización multilineal de la mejor aproximación de rango  $R$  de una matriz, donde en el caso matricial, es obtenida por la truncación de la SVD. Con respecto a tensores de orden superior, la truncación de la *HOSVD* provee una buena aproximación más no la mejor. Para un tensor de orden tres, el problema fue abordado siguiendo dos enfoques distintos, uno en el sentido de mínimos cuadrados utilizando la técnica de multiplicadores de Lagrange; y el otro, representando el tensor a través de *secciones* y maximizando la suma de las normas de las proyecciones de las *secciones* a lo largo de las tres direcciones. A través de tales enfoques, se desarrollaron dos descomposiciones, las cuales presentan algunas similitudes y diferencias, a continuación se mencionan algunas de ellas:

- Ambos reciben un tensor de orden tres arbitrario con unas restricciones de rango  $n$ -*modales* deseadas, y entregan un tensor central con rangos  $n$ -*modales* iguales a los deseados y tres matrices ortogonales.
- Ambos utilizan ALS para encontrar las matrices ortogonales, lo cual implica, que cada matriz unitaria es optimizada de manera individual, mientras dejan la otras fijas.
- Ambos transforman el tensor en matrices para manipularlo, pero con enfoques distintos: el *HOOI* despliega todo el tensor a lo largo de los diferentes ejes coordenados para formar matrices; mientras, el *SP* representa el tensor como *secciones* a lo largo de diferentes ejes coordenados.

No obstante, a pesar de sus diferencias, ambos enfoques obtuvieron resultados muy similares según los resultados de simulación obtenidos anteriormente. Los resultados de simulación mostraron que en cuanto a precisión, el algoritmo *HOOI* obtiene la mejor aproximación, seguido por el *SP* y *HOSVD*; tales resultados son esperados, si se tiene presente que el *HOOI* optimiza cada matriz unitaria a partir del tensor original y de las otras matrices unitarias restantes; mientras, el *HOSVD* calcula cada matriz unitaria solo a partir del tensor original, y el *SP*, a partir del tensor original y una de las matrices unitarias restantes. Por lo que permite concluir que *SP* reside entre *HOOI* y *HOSVD*.

## Capítulo 5

# EL USO DE DESCOMPOSICIÓN DE TENSORES EN REDUCCIÓN DE MODELOS

En este capítulo se aborda el problema de encontrar espacios de proyección óptimos para reducción de modelos utilizando el enfoque de Descomposición Ortogonal Propia (POD- Proper Orthogonal Reduction). La reducción de modelos es una rama de la teoría de sistemas, la cual estudia las propiedades de los sistemas dinámicos lineales de gran dimensión para reducir su complejidad, mientras preserva la mayor cantidad de información del sistema original. Básicamente, su finalidad es encontrar un sistema de menor dimensión con características similares al sistema original, pero con menores requerimientos de almacenamiento (memoria) y tiempos de cómputo, el cual posteriormente, pueda ser fácilmente integrado como una componente en una simulación de un sistema a gran escala ó en el diseño de sistemas de control adecuados para implementaciones en tiempo real.

En la literatura de teoría de sistemas, las técnicas de reducción de modelos se pueden dividir en dos grupos:

- Métodos de aproximación basados en SVD.
- Métodos basados en momentos coincidentes.

En el primer grupo se encuentran el método de balanceo introducido por Moore ([24]) a sistemas lineales y extendido por Scherpen ([27]) a no lineales; en este método, el sistema es transformado en otro a través de un cambio de base, donde los estados que son difíciles de alcanzar son a la vez difíciles de observar, y obtienen el modelo reducido simplemente por truncar los estados con esta propiedad. Igualmente, en este primer grupo se encuentran el método de aproximación en norma de Hankel ([12]), el método de aproximación por perturbación singular ([17], [22]); y en el segundo grupo, los métodos de proyección de subespacios de Krylov ([4],[13],[31]).

El método de POD, conocido también como el método de descomposición de Karhunen-Loeve, es una técnica de álgebra lineal numérica que proporciona una versión finito dimensional de la expansión en modos de Fourier de espacios abstractos infinito dimensionales. Básicamente, es una técnica que busca obtener representaciones aproximadas de señales a través de su descomposición espectral (secuencia infinita de funciones ortogonales). Por tanto, un tema central en este método de reducción de modelos es la construcción de funciones de base ortonormales (empíricas) que permitan obtener representaciones espectrales de señales y para las cuales su aproximación de bajo rango sea óptima.

Dentro del contexto de reducción de modelos, el método de POD es una técnica que usa la proyección de Galerkin y la descomposición espectral para proyectar un sistema de ecuaciones diferenciales ordinarias sobre un conjunto de funciones de base empíricas obtenidas a través de datos experimentales ó de simulación del sistema. Actualmente, en diversas disciplinas han surgido un número creciente de aplicaciones donde se aprecia la necesidad de manipular datos numéricos descritos en múltiples dimensiones, llamados tensores de orden superior; entre tales aplicaciones, se encuentra reducción de modelos, en donde se cree que a través de la manipulación de estos tipos de objetos se puedan superar las técnicas basadas en valores singulares ([33]).

Existen cinco pasos a seguir para obtener un modelo reducido a partir de un sistema descrito por Ecuaciones Diferenciales Parciales (EDPs) usando el método de POD ([5]). En este capítulo, se introduce una forma de calcular las funciones de base ortonormales, la cual involucra los dos primeros pasos. El uso de descomposición de tensores de orden superior en reducción de modelos está motivado por el deseo de reducir el número de funciones de base necesarias para obtener soluciones aproximadas del sistema EDP ([25]).

Este texto se organiza de la siguiente manera: en la sección 5.2 se desarrollan algunos fundamentos matemáticos del método POD y se discute su aplicación en sistemas EDPs discretizados por métodos numéricos. En la sección 5.3 se introduce el concepto de tensor y se propone la aplicación de SVD para tensores en reducción de modelos. Por último se presentan algunas conclusiones.

## 5.1. Introducción

Actualmente en teoría de sistemas y control, existen herramientas que fácilmente permiten obtener modelos descritos por Ecuaciones Diferenciales Parciales (EDPs) para sistemas físicos. Estos modelos se simulan mediante la discretización del dominio espacial y/o temporal de las variables, usando métodos numéricos y se cumple que entre más fina sea la discretización más precisa será la solución numérica. Sin embargo, una fina discretización conlleva a sistemas aproximados de miles de Ecuaciones Diferenciales Ordinarias (EDOs) los cuales no son adecuados para el diseño de sistemas de control y su implementación en tiempo real.

A partir de tales sistemas EDOs, se pueden obtener modelos de orden reducido aptos para el



diseño de sistemas de control, usando distintas técnicas desarrolladas en la teoría de sistemas. El método de Descomposición Ortogonal Propia (POD-*Proper Orthogonal Decomposition*), técnica de interés en este texto, proporciona una versión finito dimensional de la expansión en modos de Fourier de espacios abstractos infinito dimensionales; su idea principal, es encontrar un conjunto de funciones empíricas de tal forma que el error obtenido al proyectar un conjunto de datos de simulación o experimentales del sistema sobre el espacio generado por estas funciones sea mínimo. Este método, es una técnica que ha sido aplicada en varios sistemas físicos descritos por EDPs, como por ejemplo [5], [2][3][23].

El algoritmo que describe la técnica de reducción de modelos usando POD comienza por almacenar en una matriz llamada *instantánea* (snapshot), un conjunto de medidas de las variables del sistema tanto en el espacio como en el tiempo ante cambios en las señales de entrada del sistema (condiciones de frontera); donde posteriormente, se hace la Descomposición en Valores Singulares (SVD-*Singular Value Decomposition*) de esta matriz para obtener un conjunto de funciones que constituyen una aproximación de una base ortogonal que genera el espacio de soluciones del sistema de EDPs. Esta forma de calcular las funciones de base ortogonales ha funcionado de manera aceptable desde el punto de vista ingenieril en varias aplicaciones ([5], [2], [3], [23]); las cuales incluyen sistemas descritos por PDEs de una o varias variables con dominios espaciales unidimensionales o bidimensionales. Para estas aplicaciones se considera que pueden existir otras funciones de base distintas a las funciones obtenidas a través de la SVD de la matriz *instantánea*, que condensen de mejor forma las dinámicas espaciales del sistema; ésto debido a que los datos de simulación o experimentales usados para calcular las funciones ortogonales, son apilados en la matriz *instantánea* sin respetar totalmente la estructura del dominio espacial. Por tanto, en el texto se propone respetar cualquier estructura cartesiana del dominio espacial al almacenar los datos de simulación o experimentales en tensores de orden tres.

Existen cinco pasos a seguir para obtener un modelo reducido para un sistema descrito por PDEs usando POD ([5]), los cuales son: generación de la matriz *instantánea*, cálculo de las funciones de base, selección de las funciones POD, construcción del modelo dinámico para los coeficientes de las funciones POD y validación del modelo reducido. La idea principal de este texto, es mostrar una posible forma de mejorar el cálculo de las funciones de base en sistemas de  $L$  variables y con dominios espaciales de  $N$  dimensiones, donde  $L$  es un entero positivo y  $N = 2$  ó  $3$ . Cabe resaltar, que mejorar dicho cálculo es importante, ya que con ello se puede reducir el número de funciones de base necesarias para representar una determinada señal, y por ende, el número de ecuaciones ordinarias necesarias para representar el sistema.

En diversas disciplinas ha surgido un número creciente de aplicaciones donde se aprecia la necesidad de manipular datos numéricos descritos en múltiples dimensiones, dichas cantidades u objetos son conocidos como tensores de orden superior. Entre tales aplicaciones, se puede mencionar reducción de modelos [33], dada la necesidad de construir espacios de proyección que permitan obtener una buena aproximación y representación de señales.

## 5.2. Reducción de modelos

El método de POD está fuertemente relacionado con el concepto de espacios de Hilbert separables, ya que si el conjunto de datos o de funciones que se quieren aproximar residen en un espacio de Hilbert separable, entonces se garantiza la existencia de un conjunto contable de funciones ortonormales que permiten obtener representaciones espectrales de estas funciones o datos.

Sea  $\mathbb{T} \subseteq \mathbb{R}$ ,  $\mathbb{X}$  el dominio temporal y espacial respectivamente, y sea  $H = \{f : \mathbb{X} \rightarrow \mathbb{R} / f \text{ es continua}\}$  un espacio de Hilbert separable con base ortogonal  $\{\varphi_i\}_{i \in I}$ . Sea  $t \in \mathbb{T}$  y  $z(\cdot, t) \in H$ , entonces se tiene que:

$$z(x, t) = \sum_{i=1}^{\infty} a_i(t) \varphi_i(x), \quad \text{para todo } x \in \mathbb{X} \quad (5.1)$$

donde  $a_i(t) = \langle z(\cdot, t), \varphi_i \rangle$  son los coeficientes de  $z(\cdot, t)$  en la base de  $H$ . La idea de POD es que la función  $z(\cdot, t)$  pueda ser representada de forma aproximada por una expansión de  $n$  funciones  $\varphi_i$ , tal como se muestra a continuación:

$$z(x, t) \cong z_n(x, t) = \sum_{i=1}^n a_i(t) \varphi_i(x), \quad \text{para todo } x \in \mathbb{X} \quad (5.2)$$

Las funciones ortogonales  $\{\varphi_i\}_{i=1}^n$ , llamadas funciones POD, son seleccionadas de tal forma que condensan las principales correlaciones espaciales. Cabe mencionar, que los coeficientes de estas funciones son obtenidos al aplicar la proyección de Galerkin del sistema original sobre el espacio generado por las funciones POD.

A continuación, se expondrán dos aplicaciones que ilustran la forma como se han abordado los dos primeros pasos para la derivación de un modelo reducido. El primer ejemplo, un sistema de una sola variable con dominio unidimensional, corresponde al caso más sencillo en reducción de modelos; y el segundo, corresponde a una aplicación de un sistema de una sola variable con dominio bidimensional, el cual, constituye un ejemplo para proponer el uso de tensores en reducción de modelos.

### 5.2.1. Perfil de temperatura de una barra unidimensional

En esta aplicación se considera una barra unidimensional, véase Fig.5.1, cuyo perfil de temperatura está descrito por la Ec. 5.3 ([2]). Usando el método de diferencias finitas se puede obtener un modelo lineal del sistema de  $N - 2$  ecuaciones diferenciales ordinarias, al dividir el dominio espacial en  $N$  secciones, véase Ec. 5.3.

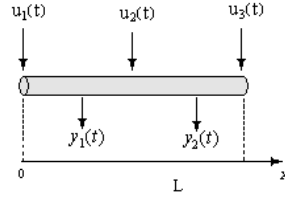


Figura 5.1: Barra unidimensional. Las señales  $u_1(t)$ ,  $u_2(t)$  y  $u_3(t)$  son las condiciones de frontera en  $x = 0$ ,  $x = L/2$  y  $x = L$  respectivamente.

$$\frac{\lambda}{\rho c_p} \frac{\partial^2 T(x, t)}{\partial x^2} = \frac{\partial T(x, t)}{\partial t} \quad (5.3)$$

$$\dot{T}(t) = AT(t) + Bu(t) \quad (5.4)$$

donde  $T$  contiene la temperatura en cada punto de discretización del dominio espacial,  $u(t)$  el vector de señales de condiciones de frontera. La matriz *instantánea* se obtiene usando datos experimentales o de simulación al aplicar de manera simultánea una señal persistentemente excitante sobre las entradas del proceso (condiciones de frontera). La matriz *instantánea*, denotada por  $Z_{snap}$ , es una matriz de orden  $m \times n$ , donde el número de filas corresponde al número de celdas de la discretización espacial de la ecuación diferencial; y el número de columnas corresponde al número de muestras tomadas a cierto periodo de muestreo. Así, la matriz  $Z_{snap}$  para un número  $s$  de muestras a un periodo de muestreo  $\tau$ , tiene la siguiente forma:

$$Z_{snap} = \begin{bmatrix} T_1(k=1) & \cdots & \cdots & T_1(k=s) \\ \vdots & \ddots & & \vdots \\ \vdots & \ddots & & \vdots \\ T_N(k=1) & \cdots & \cdots & T_N(k=s) \end{bmatrix} \quad (5.5)$$

Las funciones ortogonales son obtenidas calculando la SVD de la matriz  $Z_{snap}$ ,

$$Z_{snap} = \Phi \Sigma \Psi^T$$

donde  $\Phi$  y  $\Psi$  son matrices unitarias y  $\Sigma$  es una matriz que contiene los valores singulares de  $Z_{snap}$  en forma decreciente en su diagonal principal. Astrid en [5], presenta una demostración formal del porque las funciones POD aparecen en las columnas de la matriz  $\Phi$ , argumentando que en primera instancia la mejor función de base POD, denotada  $\varphi_1$ , es aquella que maximiza el promedio de las proyecciones individuales de los datos y que además tiene norma uno; y posteriormente, argumentando que la segunda función POD debe igualmente

maximizar el promedio de las proyecciones individuales y además ser ortogonal a la primera función POD. Luego de manera inductiva, se obtienen las demás funciones POD.

Las funciones ortogonales aparecen en las columnas de la matriz  $\Phi$  y el subconjunto finito de funciones ortogonales mencionado en la Ec.5.2, es seleccionado examinando los valores singulares de la matriz  $\Sigma$ . La correlación entre las funciones ortogonales y los datos es atribuido a los valores singulares, donde un valor significativo de éste corresponde a una mayor correlación entre los datos y la respectiva función ortogonal. El criterio para seleccionar el número  $n$  de funciones de base más relevantes está dado por:

$$P_n = \frac{\sum_{i=1}^n \sigma_i}{\sum_{i=1}^r \sigma_i} \cong 0,999, \quad n = 1, \dots, r \quad (5.6)$$

donde  $\{\sigma_i\}_{i=1}^r$  son los valores singulares distintos de cero de la matriz  $Z_{snap}$ . Dado que los valores singulares son positivos se tiene que  $0 \leq P_1 \leq \dots \leq P_r = 1$ . La ecuación anterior es usada para determinar el grado de truncamiento de las funciones ortogonales. El criterio de truncamiento comúnmente usado es tomar  $n$  tal que  $P = 0,999$  (99,9 % de la información contenida en  $Z_{snap}$ ).

A manera de ilustración, a continuación se muestran algunas funciones de base obtenidas para la barra unidimensional ([10]):

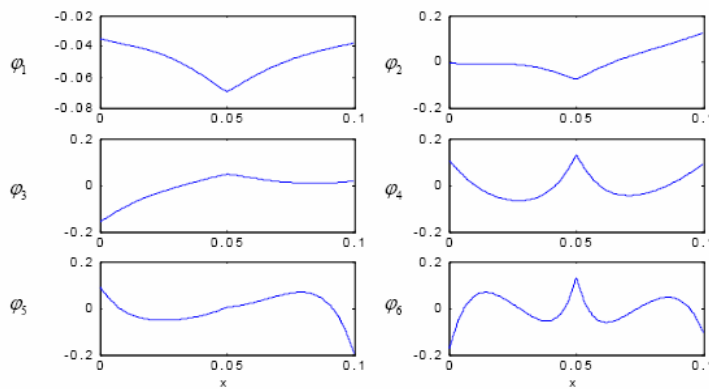


Figura 5.2: Funciones POD Obtenidas para la barra unidimensional

Aunque las funciones anteriores están normalizadas, se puede observar que éstas tienen la forma de posibles perfiles de temperatura de la barra, y por tanto, cualquier perfil de temperatura  $T(x, t)$  puede ser obtenido de manera aproximada al ponderar estas funciones con ciertos coeficientes (variantes en el tiempo), tal como se muestra a continuación:

$$T(x, t) \cong T_n(x, t) = a_1(t)\varphi_1(x) + \dots + a_6(t)\varphi_6(x) \quad (5.7)$$

El paso siguiente para obtener modelo reducido, es construir un modelo dinámico para los

coeficientes  $\{a_i(t)\}_{i=1}^6$  a través de la proyección de Galerkin, en la cual se define una función residual que posteriormente, se proyecta sobre cada una de las funciones POD, tal como se muestra a continuación:

$$\begin{aligned} R(T) &= \dot{T} - AT + Bu \\ \langle R(T_n), \varphi_i \rangle &= 0, \quad i = 1, \dots, 6 \end{aligned} \quad (5.8)$$

de ésta forma se obtiene que

$$\begin{aligned} \dot{a}(t) &= A_r a(t) + B_r u(t) \\ T_n(t) &= [\varphi_1 \ \cdots \ \varphi_6] \cdot a(t) \end{aligned}$$

Obsérvese, que de esta forma se pasa de un modelo EDO de orden  $N$  a un modelo EDO de orden 6, donde por lo general  $N \gg 6$ .

### 5.2.2. Perfil de temperatura de una placa

En esta aplicación considere una placa de dos dimensiones, véase Fig.5.3, cuyo perfil de temperatura  $T(x, y, t)$  es descrito por la Ec. 5.9 ([23]). El dominio espacial es dividido en  $m$  celdas a lo largo del eje  $x$  y en  $n$  celdas a lo largo del eje  $y$ , obteniéndose así, un total de  $mn = N$  celdas, tal como se muestra en la Fig.5.3.

$$\rho c_p \frac{\partial T(x, t)}{\partial t} = k \frac{\partial^2 T(x, t)}{\partial x^2} + k \frac{\partial^2 T(x, t)}{\partial y^2} + h \frac{T_a - T}{\Delta z} \quad (5.9)$$

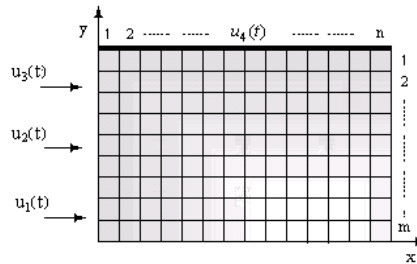


Figura 5.3: Placa bidimensional. Las señales  $u_1(t)$ ,  $u_2(t)$ ,  $u_3(t)$  y  $u_4(t)$  son las condiciones de frontera en  $(0, 1/4L)$ ,  $(0, 1/2L)$ ,  $(0, 3/4L)$  y  $(L, 0)$  respectivamente.

La matriz  $Z_{snap}$  se obtiene de igual forma que en el caso anterior; sin embargo, los datos son almacenados en la matriz  $Z_{snap}$  de una forma especial, dado que el dominio espacial es

bidimensional. El número de filas de la matriz  $Z_{snap}$  corresponderá al número de celdas de las divisiones del dominio espacial, es decir, cada fila representará la evolución temporal de cada celda de la placa. Para efectos prácticos las filas de la matriz  $Z_{snap}$  pueden ser organizadas de la siguiente forma: las primeras  $m$  filas corresponderán a las celdas de la columna 1 de la división espacial, véase Fig.5.3, seguidas de las celdas de la columna 2, y así sucesivamente, tal como se muestra en la Ec.5.10, para un número de  $s$  muestras a un periodo de muestreo  $\tau$ :

$$Z_{snap} = \begin{bmatrix} T_{11}(k=1) & \cdots & \cdots & T_{11}(k=s) \\ \vdots & \ddots & & \vdots \\ T_{m1}(k=1) & \cdots & \cdots & T_{m1}(k=s) \\ T_{12}(k=1) & \cdots & \cdots & T_{12}(k=s) \\ \vdots & \ddots & & \vdots \\ T_{m2}(k=1) & \cdots & \cdots & T_{m2}(k=s) \\ \vdots & \ddots & & \vdots \\ T_{1n}(k=1) & \cdots & \cdots & T_{1n}(k=s) \\ \vdots & \ddots & & \vdots \\ T_{mn}(k=1) & \cdots & \cdots & T_{mn}(k=s) \end{bmatrix} \quad (5.10)$$

Al igual que en el caso anterior, las funciones de base son obtenidas calculando la SVD de la matriz  $Z_{snap}$ . Las funciones ortogonales aparecen en las columnas de la matriz  $\Phi$  y el subconjunto de funciones POD son seleccionadas de igual forma que en el caso anterior. Sin embargo, dado que existen distintas formas de almacenar los datos en la matriz  $Z_{snap}$ , pueden existir funciones de base distintas a las obtenidas anteriormente que condensen de mejor forma las dinámicas espaciales. Observe, que otra posible forma de almacenar las celdas es que las primeras  $n$  filas correspondan a las celdas la fila 1 de la división espacial, (véase Fig 5.3), seguidas de las celdas de la fila 2, y así sucesivamente. El hecho de que existan distintas formas de indexar las celdas implica que se puede obtener distintas matriz  $Z_{snap}$ , y por ende, funciones de base ortogonales que generen distintos subespacios. Fácilmente se puede constatar lo anterior, al observar que para distintas matrices  $Z_{snap}$  no necesariamente las funciones generan el mismo subespacio, dado que estas funciones deben de maximizar las proyecciones promedio de los elementos de la matriz ([5]).

### 5.3. SVD para tensores en reducción de modelos

La idea de introducir tensores en reducción de modelos, viene dada por la hipótesis de que a través de la descomposición de tensores se pueda reducir el número de funciones de base necesarias para obtener una buena representación de las dinámicas espaciales sistema, lo cual es de suma importancia, dado que la idea principal en esta línea de investigación es desarrollar una metodología general que permita obtener modelos reducidos de sistemas

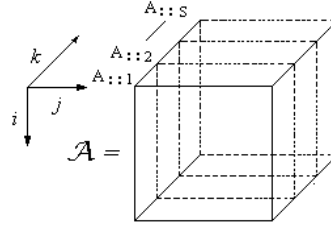


Figura 5.4: Representación tensorial de la matriz  $Z_{snap}$  del perfil de temperatura de un placa. El tensor  $\mathcal{A}$  es de orden 3 y de tamaño  $m \times s \times n$

descritos por EDPs usando la técnica de POD.

Una ilustración de como se pretende incorporar el uso tensores en reducción de modelos se puede observar a través de la aplicación anterior (sección 5.2.2), donde la matriz  $Z_{snap}$  en forma tensorial denotada por  $\mathcal{A}$ , se observa en la Fig. 5.4. El elemento en la fila  $i$ , columna  $j$  y tubo  $k$  representado por  $a_{ijk}$  corresponde a la celda de la fila  $i$  y columna  $k$  en el instante de tiempo  $j$ . Observe que almacenando los datos en forma tensorial se logra respetar la continuidad de las celdas de la placa en cualquier dirección, es decir, que para la celda representada por la componente  $(i, j, k)$  se cumple que las celdas representadas con componente  $(i + 1, j, k)$ ,  $(i - 1, j, k)$ ,  $(i, j, k - 1)$ ,  $(i, j, k + 1)$ ,  $(i + 1, j, k + 1)$ ,  $(i - 1, j, k + 1)$ ,  $(i - 1, j, k - 1)$  y  $(i + 1, j, k - 1)$  son adyacentes en la discretización de la placa. Por tanto, se obtiene que independientemente de la forma en que se indexen las celdas, bien sea en forma horizontal o vertical, siempre se obtendrá el mismo tensor salvo rotaciones de éste.

Una vez almacenados los datos experimentales o de simulación en forma tensorial, el cálculo de las funciones POD se realiza a través del uso de descomposición de tensores, que es simplemente una extensión de la SVD de matrices a arreglos multidimensionales. Ahora, independiente de la descomposición utilizada, el esquema de la descomposición del tensor  $Z_{snap}$  esta dada por la Fig. 3.2; por tanto, dado la forma en que se almacenaron los datos se tiene que en las matrices  $\mathbf{U}^{(1)}$  y  $\mathbf{U}^{(3)}$  se encuentran aproximaciones de funciones de base ortogonales espaciales, y en  $\mathbf{U}^{(2)}$  aproximaciones de funciones de base ortogonales temporales.

Dentro de un dominio espacial finito dimensional, a través del teorema 3.2<sup>1</sup> se puede expresar la temperatura de la placa (de forma aproximada) como una expansión espectral en 2-dimensiones dada por:

$$T_n(x, y, t) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_3} a_{ij}(t) \left( U_i^{(1)}(y) \circ U_j^{(3)}(x) \right) \quad (5.11)$$

donde

<sup>1</sup>En este experimento se utilizó la descomposición HOSVD dado que fue la primera que se abordó, y además, porque para esta primera prueba, solo era necesario obtener una aproximación de bajo rango.

$$U_i^{(1)} \circ U_j^{(3)} = U_i^{(1)} \circ \left( U_j^{(3)} \right)^T$$

La operación  $(\cdot \circ \cdot)$  denota el producto exterior de tensores de orden superior,  $n_1$  y  $n_3$  el número de funciones de base truncadas de  $\mathbf{U}^{(1)}$  y  $\mathbf{U}^{(3)}$  respectivamente. Ahora en el caso de la placa, las funciones de base están dadas por  $U_i^{(1)} \circ U_j^{(3)}$  para  $i = 1, \dots, n_1$  y  $j = 1, \dots, n_3$ , donde a diferencia de las funciones de base de la barra, éstas son funciones bidimensionales.

Con el fin de obtener un conjunto de funciones de base bidimensionales para la temperatura de placa, se obtuvo a través de datos de simulación, un tensor de orden  $30 \times 501 \times 30$  al discretizar el dominio espacial con una malla de  $30 \times 30$  y al almacenar datos de temperatura de cada celda por 501 periodos de muestreo respectivamente. A continuación se muestran algunas funciones de base bidimensionales  $U_i^{(1)} \circ U_j^{(3)}$  obtenidas

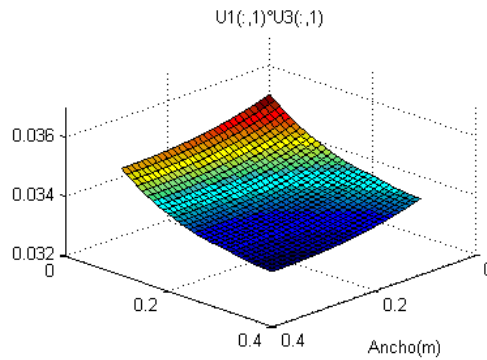


Figura 5.5: Función de base bidimensional de la placa dada por  $U_1^{(1)} \circ U_1^{(3)}$ .

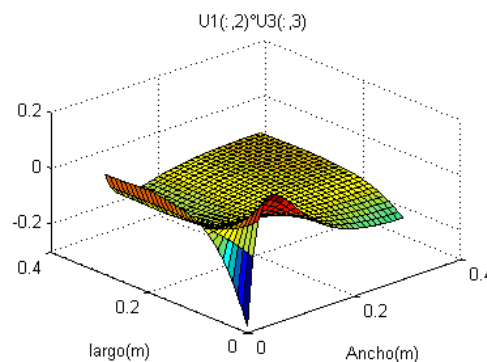


Figura 5.6: Función de base bidimensional de la placa dada por  $U_2^{(1)} \circ U_3^{(3)}$ .

Un vez obtenidas las funciones de base en la expansión en series de Fourier de la temperatura de la placa, véase Ec. 5.11, solo resta obtener un modelo dinámico para los coeficientes  $a_{ij}(t)$ . Siguiendo el mismo procedimiento descrito por Astrid, estos coeficientes se obtienen



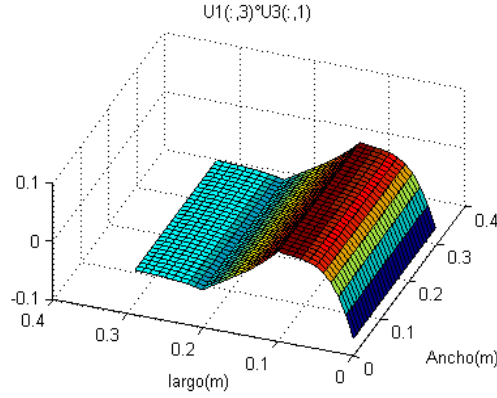


Figura 5.7: Función de base bidimensional de la placa dada por  $U_3^{(1)} \circ U_1^{(3)}$ .

definiendo una función residual, y posteriormente, proyectandola sobre cada una de las funciones de base bidimensionales (proyección de Galerkin).

Dado que las funciones de base obtenidas son bidimensionales, la función residual no puede ser definida de manera análoga a la función de la Ec.5.8, debido a que el modelo que se obtiene después de utilizar técnicas diferencias finitas obedece a un almacenamiento matricial, y por tanto, la función residual residiría en un espacio de dimensión diferente. No obstante, se podría definir una función residual utilizando la ecuación diferencial parcial del modelo original (Ec. 5.9), tal como se muestra a continuación:

$$R(T) = \frac{\partial T}{\partial t} - r \frac{\partial T}{\partial x^2} - r \frac{\partial T}{\partial y^2} - s(T_a - T)$$

donde  $r = k/\rho c_p$  y  $s = h/\Delta z \rho c_p$ . Luego, se tiene que el dominio espacial de la función residual  $R(T)$  es continuo con dimensión infinita. No obstante, dentro de este dominio espacial, la operación  $(\cdot \circ \cdot)$  no se puede definir como el producto exterior de tensores, y por consiguiente, no quedaría muy claro que significa esta operación dentro de este dominio espacial. Dado lo anterior, futuros trabajos deberían ser dedicados a definir claramente que significa esta operación  $(\cdot \circ \cdot)$  dentro de este dominio, con el fin, de establecer correctamente el producto interno entre la función residual  $R(T)$  y las funciones de base.

Por otro lado, futuras investigaciones deberían ser dedicadas en como escoger los coeficientes  $n_1$  y  $n_3$ , dado que la forma tradicional, basada en la Ec. 5.6, no puede ser utilizada para determinar la truncación de las funciones de base en las matrices  $\mathbf{U}^{(1)}$  y  $\mathbf{U}^{(3)}$ ; esto debido, a que la forma tradicional busca determinar la mayor cantidad de información presente en funciones de base unidimensionales, y no tiene en cuenta, la sinergia que ocurre entre las funciones de base unidimensionales cuando forman las bidimensionales. Una posible opción para determinar estos coeficientes, es a través de la cantidad de energía<sup>2</sup> almacenada dentro

<sup>2</sup>Una posible interpretación física de la norma de Frobenius, es la cantidad de energía presente en el

del tensor central ( $\mathcal{B}$  ó  $\mathcal{S}$ ) de la descomposición del tensor  $Z_{snap}$ , es decir, fijado  $n_2 = 1$ , determinar los valores de  $n_1$  y  $n_3$ , a través de la energía de los posibles tensores formados con las distintas combinaciones de estos coeficientes; no obstante, dentro de varias posibles combinaciones de  $n_1$  y  $n_3$ , el como criterio final de selección serian resultados de simulación.

## 5.4. Conclusiones

En este capítulo, se desarrollaron algunos conceptos básicos sobre reducción de modelos, y se expusieron dos ejemplos permiten visualizar cómo actualmente se han abordado los dos primeros pasos, en sistemas de una sola variable, para la derivación de un modelo reducido utilizando el enfoque POD. Adicionalmente, se mostró una aplicación de descomposición de tensores en reducción de modelos, donde principalmente, se mostró la posibilidad de obtener la funciones de base POD para un sistema de una sola variable y dominio espacial bidimensional, a través de la descomposicion de tensores.

Básicamente, se mostró que la idea de introducir descomposición de tensores de orden superior es reducir el número de funciones de base POD necesarias para obtener representaciones razonables de bajo costo de las dinámicas espaciales, en aplicaciones con sistemas de una variable y dominios no unidimensionales, respetando cualquier estructura cartesiana que el dominio espacial pueda presentar. Al almacenar la información de las celdas en arreglos multidimensionales, se busca que éstas al indexarlas en el tensor permanezcan contiguas unas con otras, es decir, que al ubicarse en una componente del tensor las componentes adyacentes representen celdas que son continuas a la celda que representa la componente fijada. De esta manera, se infiere que las funciones POD calculadas condensarán de mejor forma la distribución espacial del sistema, necesitando un menor número de funciones POD para representar la variable del sistema.

Por último, se plantearon dos preguntas que serán tema de futuras investigaciones, que son la obtención de los coeficientes en la truncación de la representación espectral de la variable del sistema; y la obtención, de los coeficientes variantes en el tiempo que multiplican las funciones de base POD y que reflejan las dinámicas temporales del sistema.

# Bibliografía

- [1] Acar, E. and Yener, B. *Unsupervised multiway data analysis: A literature survey*. Technical report, Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY, 2007.
- [2] Agudelo, O.M., Espinosa, J.J. and De Moor, B. *Application of POD and Predictive Control Techniques to the Control of the Temperature Profile of an one-dimensional Bar*. In Proceedings of the XII Latin-American Congress on Automatic Control (CLCA 2006), Salvador, Brazil Oct. 2006, pp. CD-ROM, 2007.
- [3] Agudelo, O.M., Espinosa, J.J. and De Moor, B. *Control of a Tubular Chemical Reactor by means of POD and Predictive Control Techniques*. In Proceedings of the European Control Conferences 2007 (ECC'07), Kos, Grece, pp.1046-1053, Jul. 2007.
- [4] Antoulas, A.C., Sorensen, D.C. and Gugercin, S. *A survey of model reduction methods for large-scale systems*. Structured Matrices in Operator Theory, Numerical Analysis, Control, Signal and Image Processing, Contemporary Mathematics, AMS publications, 280:193-219, 2001.
- [5] Astrid, P. *Reduction of Process Simulation Models: a Proper Orthogonal Decomposition Approach*. PhD thesis, Technische Universiteit Eindhoven, Eindhoven (Netherlands), 2004.
- [6] Bader, B.W and Kolda, T.G. *Efficient MATLAB computations with sparse and factored tensors*. SIAM Journal on Scientific Computing, vol 30, No, 1, pp 205-231, December 2007a.
- [7] Bader, B.W and Kolda, T.G. MATLAB Tensor Toolbox version 2.2. <http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/>, 2007b.
- [8] Brand, M. *Incremental Singular Value Decomposition of uncertain data with missing values*. Proceedings of the European Conference on Computer Vision, pag 707-760,2002.
- [9] Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K. and Harshman, R. *Indexing by latent semantic analysis*. Journal of the American Society for Information Science, vol 41, pp 391-407, 1990.

- 
- [10] Espinosa, J. J. *Seminario de Reducción de Modelos Usando POD*. Documento interno, 2006.
- [11] Golub, G. H. and Van Loan, C.F. *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.
- [12] Glover, K. *All Optimal Hankel-norm Approximations of Linear Multivariable Systems and Their  $L^\infty$ -error Bounds*. Int. J. Control, 39, 1115-1193, 1984.
- [13] Grimme, E.J. *Krylov Projection Methods for Model Reduction*. PhD thesis, ECE Dept., U. of Illinois, Urbana-Champaign, 1997.
- [14] Harshman, R.A. *Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis*. UCLA Working Papers in Phonetics, vol 16, pp 1-94, 1970.
- [15] Kolda, T.G. *Multilinear operators for higher-order decompositions*. Technical Report SAND2006-2081, Sandia National Laboratories, Livermore, CA, 2006. <http://csmr.ca.sandia.gov/~tgkolda/pubs/>.
- [16] Kolda, T. G. and Bader, W. *Tensor Decompositions and applications*. SIAM review, vol 51, No 3, September 2009.
- [17] Kokotovic, P.V, O'Malley, R.E and Sannuti, P. *Singular Perturbations and Order Reduction in Control Theory - an Overview*. Automatica, 12, 123-132, 1976.
- [18] De Lathauwer (1997). *Signal Processing Based on Multilinear Algebra*. Ph.D thesis, K.U. Leuven, E.E. Dept.-ESAT, Belgium.
- [19] De Lathauwer, L., De Moor, B. and Vandewalle, J. *A Multilinear Singular Value Decompositions*. SIAM J. Matrix Analysis and Applications, vol 21, No.4, pp. 1253-1278, 2000.
- [20] De Lathauwer, L., De Moor, B. and Vandewalle, J. *On the best rank-1 and rank approximation of higher order tensors*. SIAM J. Matrix Analysis and Applications, vol 21, No.4, pp. 1324-1342, 2000.
- [21] Laub, A.J. *Matrix Analysis for Scientists & Engineers*. Society for Industrial and Applied Mathematics-SIAM, Philadelphia, 2005.
- [22] Liu, Y and Anderson, B, D, O. *Singular Perturbation Approximation of Balanced Systems*. Int. J. Control, 50, 1379-1405, 1989.
- [23] Marquez, A. *Control Predictivo Basado en Modelo con Reducción de Modelos por el Método de POD Aplicado a un Sistema de Parámetros Distribuidos*. Tesis de grado, Universidad Nacional de Colombia, Medellín, Colombia, 2007.

- 
- [24] B. C. Moore. *Principal Component Analysis in Linear System: Contrability, Observability and Model Reduction*. IEEE Transactions on Automatic Control, AC-26;17-32,1981.
- [25] Rios, R., Espinosa, J.J. and Mejía, C. *El Uso de Descomposición en Valores Singulares para Tensores en Reducción de Modelos*. VII congreso de la Asociación Colombiana de Automática, Cartagena, Colombia, 2009.
- [26] Schutze, H. *Automatic word sense discrimination*. Computational Linguistic, vol 24, pp 97-123, 1998.
- [27] Scherpen, J.M.A. *Balancing For Nonlinear Systems*. Systems & Control Letters, vol 21, 143-153,1993.
- [28] Turney, D. P. *Empirical Evaluation of Four Tensor Decomposition Algorithms*. Technical Report ERB-1152, NRC-49877, National Research Council of Canada, Ottawa, 2007.
- [29] Tucker, L.R. *Some mathematical notes on three-mode factor analysis*. Psychometrika, vol 32, pp 379-416, 1966.
- [30] Vandewalle, J. and De Moor, B. *On the use of the Singular Value Decomposition in Identification and Signal Processing*. NATO ASI Series: Numerical Linear Algebra, Digital Signal Processing and Parallel Algorithms, vol 70, pp 321-360,1966.
- [31] Van Dooren, P. *The Lanczos algorithm and Padé approximations*. Short Course, Benelux Meeting on Systems and Control, 1995.
- [32] Wang, H. and Ahuja, N. *Rank- R approximation of tensors: Using image-as-matrix representation*. Proceeding of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol 2, pp 346-535, 2005.
- [33] Weiland, S. and Van Belzen, F. *Model Reduction Through Multilinear Singular Value Decompositions*. European Conference on Computational Fluid Dynamics, Netherlands, 2006.
- [34] Van Belzen, F. and Weiland, S. *Reconstruction and Approximation of Multidimensional Signals Described by Proper Orthogonal Decompositions*. IEEE Trans. on Signal Processing, vol 56, No. 2, February 2008.
- [35] Van Belzen, F. and Weiland, S. and De Graaf, J. *Singular Value Decompositions and Low Rank Approximations of Multi-linear Functionals*. IEEE Conference on Decision and Control, New Orleans, LA,USA,Dec. 12-14,2007.