

**Reciclaje de Datos, basado en interpolación
numérica, para una reducción eficiente del
orden de modelos dinámicos.**

Tesis que presenta
Aarón Delfino De La Concha Gómez

Para obtener el grado de
Maestro en Ciencias
(Matemáticas Aplicadas e Industriales)

Tutor: Dr. Lorenzo Héctor Juárez Valencia. Tutor Externo: Prof. Dr. Angelika Bunse-Gerstner.

© Todos los derechos reservados.



Casa abierta al tiempo

Tesis de Maestría en Matemáticas Aplicadas e Industriales:

**Reciclaje de datos, basado en interpolación numérica,
para una reducción eficiente del orden de modelos dinámicos.**

Por
Aarón Delfino De La Concha Gómez

Abril 2011

Tutor: Dr. Lorenzo Héctor Juárez Valencia. Tutor Externo: Prof. Dr. Angelika Bunse-Gerstner.

Beloved Father, may you be the beacon of light to illuminate my path
and to bright with shine upon my life.

Agradecimientos

Para los que nombrare aquí, deseo rendir mi mejor tributo por haberme encausado en este fascinante mundo de la ciencia, y en donde espero encontrar algún día, algún tipo de respuesta a algo que sea desconocido por la mente humana.

No existe mejor tributo que llevar por siempre en mi mente sus nombres, acompañados de sus mejores frases que alimentaron mi vida, mi alma, simplemente lo mejor en mi.

Del Doctor Joaquín Delgado, agradezco mucho haberme abierto la puerta de su oficina en esta Universidad que es mi casa, y que lo hizo cuando yo desesperadamente necesitaba ayuda para salir del pantano en donde inmóvil yo me encontraba. A ti querido Joaquín, te debo el inicio de mi nuevo camino que tantas alegrías me ha dado.

Al Doctor Alfredo Nicolás Carrizosa le agradezco la bondad de haberme refugiado bajo la sombra de su paciencia, y haberme brindado la confianza necesaria en este difícil mundo de las matemáticas. Estimado Alfredo, te recordare por siempre con el puño en alto.

A mi insustituible amigo el Doctor Martín Celli, siempre le agradeceré la disposición irrestricta y la pasión inherente de sus enseñanzas que son el reflejo fiel de mis anhelos y de mi esperanza de ser como él algún día. Martín, amigo mio, persigo sin descanso en cada día, ese objetivo que me mencionaste.

Finalmente pero igual de importante, infinitamente estoy agradecido con mis tutores quienes fueron los pilares fundamentales de este trabajo de tesis. A la Doctora Angelika Bunse-Gerstner por haberme ofrecido la oportunidad de verdaderamente soñar despierto fuera de mi suelo nativo. A mi entrañable amigo, el Doctor Héctor Juárez Valencia por haber tomado como suya la batalla que me correspondía pelear para hacer posible mis necesidades genuinas de soñar. A ellos y por ellos, es que estoy en esta altura, con el deseo ferviente de elevarme aún más.

Índice

1. Introducción	1
2. Conceptos Básicos de Teoría de Control	5
2.1. Un ejemplo sencillo, el motor CD	5
2.2. Sistema de control de nivel	8
2.3. Sistemas Micro Electro Mecánicos (SMEM)	10
2.4. Estabilidad	12
2.5. Alcanzabilidad	12
2.6. Observabilidad	14
2.7. Controlabilidad	15
2.8. Comportamiento de Entrada-Salida	15
2.9. Propiedades de la Función de Transferencia	17
2.9.1. Transformaciones similares	17
2.10. Parámetros de Markov	18
2.11. Espacios de Hardy	21
3. Técnicas de reducción. Dos enfoques diferentes	23
3.1. Definición del Problema	23
3.2. Truncamiento Balanceado	27
3.3. Métodos basados en Interpolación	30
3.3.1. Acoplamiento de momentos explícito	31
3.3.2. Proyección del estado-espacio	32
3.3.3. Acoplamiento de momentos implícito	33
3.3.4. Interpolación de Hermite	39
3.3.5. Interpolación Tangencial de Hermite	40
4. Reducción óptima basada en la norma $\ \bullet\ _{\mathcal{H}_2}$	42
4.1. La norma $\ \bullet\ _{\mathcal{H}_2}$	43
4.2. Interpolación óptima en $\ \bullet\ _{\mathcal{H}_2}$	47
4.3. Condiciones necesarias de primer orden	48
4.4. El algoritmo MIRIAM	49
5. Resultados Numéricos	50
5.1. Precondicionamiento	50
5.2. Resultados numéricos	54
6. Conclusiones y Futuras Direcciones	70

7. Apéndice	72
7.1. Subespacios de Krylov	72
Referencias	74

1. Introducción

Existen procesos en las ciencias y en la industria que pueden ser descritos por modelos matemáticos. Algunos de estos modelos están determinados por un sistema de ecuaciones diferenciales ordinarias de la forma

$$\Sigma : \quad \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{cases} \quad (1.1)$$

donde $A \in \mathbb{R}^{N \times N}$ es la matriz de estado, $B \in \mathbb{R}^{N \times m}$ es la matriz de entrada y $C \in \mathbb{R}^{p \times N}$ es la matriz de salida. Los vectores $x(t) \in \mathbb{R}^N$, $u(t) \in \mathbb{R}^m$ e $y(t) \in \mathbb{R}^p$ representan el estado, la entrada y la salida del sistema de ecuaciones (1.1). La solución del anterior sistema, queda definida por el siguiente par de ecuaciones

$$x(t) = e^{At}x_0 + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau, \quad t \geq t_0. \quad (1.2)$$

$$y(t) = Ce^{At}x_0 + C \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau, \quad t \geq t_0. \quad (1.3)$$

que pueden ser fácilmente implementadas en MATLAB[®] por ejemplo. Típicamente, en operaciones industriales de gran envergadura, el tamaño de N al que se le conoce como la dimensión del sistema Σ , llega a ser muy grande, fácilmente puede alcanzar un par de millones. Por lo tanto la complejidad del sistema (1.1) sobrepasa la capacidad de los actuales programas de análisis numérico y las soluciones (1.2) y (1.3) ya no pueden ser implementadas computacionalmente.

Esta dificultad de calcular las soluciones del sistema original de ecuaciones diferenciales ordinarias, hace evidente la necesidad de desarrollar una herramienta, un procedimiento o proceso que al menos nos de como resultado soluciones aproximadas que se asemejen en la mejor medida posible a las soluciones (1.2) y (1.3). Esta herramienta a la que hacemos referencia existe, y se llama Reducción de Orden de Modelos (Model Order Reduction) y tiene como objetivo construir un nuevo sistema de ecuaciones

$$\hat{\Sigma} : \quad \begin{cases} \hat{x}(t) &= \hat{A}\hat{x}(t) + \hat{B}u(t) \\ \hat{y}(t) &= \hat{C}\hat{x}(t) \end{cases} \quad (1.4)$$

de menor dimensión que el sistema original (1.1), donde las matrices $\hat{A} \in \mathbb{R}^{n \times n}$, $\hat{B} \in \mathbb{R}^{n \times m}$, $\hat{C} \in \mathbb{R}^{p \times n}$ y $n < N$, de tal manera que la distancia entre los sistemas (1.1) y (1.4) medida con la siguiente métrica a través de sus correspondientes Funciones de Transferencia $H(s)$ y $\hat{H}(s)$

$$d(H(s), \hat{H}(s)) = \|G(s)\|_{\mathcal{H}_2} \quad (1.5)$$

sea mínima, y donde $G(s) = H(s) - \hat{H}(s)$. La forma explícita de la norma es

$$\|G(s)\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[G(i\omega)]^* [G(i\omega)] d\omega \right)^{\frac{1}{2}}. \quad (1.6)$$

Model Order Reduction cuenta con 2 corrientes o metodologías para construir este sistema reducido (1.4) a partir del sistema (1.1). La primera basada en métodos de Truncamiento que consisten en emplear técnicas de Descomposición en Valores Singulares (DVS) para seleccionar las partes predominantes del sistema y eliminar las restantes. De esta corriente, el Truncamiento Balanceado (primeramente propuesto por Moore[5]) se ha convertido en la elección más popular dentro de la comunidad que se dedica a la Reducción de Orden de Modelos, debido a su sencillez para implementarla en lenguajes de análisis numérico, así mismo como a las elegantes propiedades matemáticas que se derivan de ella, como por ejemplo las cotas del error máximo entre los sistemas Σ y $\hat{\Sigma}$ que están muy bien definidas. Así mismo, se garantiza que el sistema reducido (1.4) construido de esta manera sea estable (todos los valores propios de la matriz \hat{A} están en el lado izquierdo del plano complejo estrictamente). Esta técnica sin embargo es computacionalmente costosa, ya que se requieren de N^3 operaciones para formar el sistema reducido $\hat{\Sigma}$.

Por otro lado, tenemos a la corriente formada por los métodos de interpolación que emplean técnicas de subespacios de Krylov y que pueden manejar eficientemente grandes dimensiones de N . La debilidad de estos métodos radica en la carencia de cotas para el error, y más aún, del hecho de que no se puede garantizar estabilidad del sistema reducido $\hat{\Sigma}$. Un representante de esta corriente es Acoplamiento de Momentos (Moment Matching), que tiene como finalidad el acoplar o hacer coincidir el mayor número posible de momentos (de ahí su nombre) de las Funciones de Transferencia $H(s)$ y $\hat{H}(s)$ de ambos sistemas Σ y $\hat{\Sigma}$ alrededor de un punto $\sigma \in \mathbb{C}$.

Resulta que no solamente se pueden hacer coincidir momentos de $H(s)$ y de $\hat{H}(s)$ en un solo punto $\sigma \in \mathbb{C}$, sino en varios $\sigma_k \in \mathbb{C}$, $k = 1, 2, 3, \dots, n$. Este hecho, que

es la base de la técnica conocida como interpolación de Hermite, es una condición muy difícil de cumplir para sistemas MIMO (abreviación en inglés de Multiple Input Multiple Output). Por esta razón es que se introdujo una generalización de esta técnica, conocida como Interpolación tangencial de Hermite, la cual propone acoplar los momentos de $H(s)$ y de $\hat{H}(s)$ de los sistemas (1.1) y (1.4) en varios puntos de interpolación $\{\sigma_k \in \mathbb{C}\}$, en varias direcciones tangenciales izquierda $\{l_k \in \mathbb{R}^{1 \times p}\}$ y derecha $\{r_k \in \mathbb{R}^{m \times 1}\}$, $k = 1, 2, 3, \dots, n$, de tal manera que se cumpla

$$\begin{aligned} H(\sigma_k)r_k &= \hat{H}(\sigma_k)r_k, \\ l_k H(\sigma_k) &= l_k \hat{H}(\sigma_k), \\ l_k H'(\sigma_k)r_k &= l_k \hat{H}'(\sigma_k)r_k. \end{aligned} \tag{1.7}$$

La desventaja de esta técnica radica en que el conjunto de datos $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$ $k = 1, 2, 3, \dots, n$, no se conoce a priori, forzosamente se tiene que construir iterativamente a partir de una propuesta inicial $\{\{l_k^0\}, \{r_k^0\}, \{\sigma_k^0\}\}$.

En una tesis reciente de Doctorado [4], se presenta un Algoritmo iterativo, denominado Algoritmo MIRIAM, en donde se establece que si para una propuesta inicial de datos $\{\{l_k^0\}, \{r_k^0\}, \{\sigma_k^0\}\}$, $k = 1, 2, 3, \dots, n$, este algoritmo converge al conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$ donde \hat{c}_k^T es el transpuesto de la k-ésima columna de la matriz reducida \hat{C} , \hat{b}_k^T es el transpuesto de la k-ésima fila de la matriz reducida \hat{B} , y $-\hat{\lambda}_k^*$ es la imagen espejo (negativo del conjugado de $\hat{\lambda}_k$) del k-ésimo valor propio de la matriz reducida \hat{A} , entonces el sistema reducido (1.4) construido bajo este esquema satisface las condiciones

$$\begin{aligned} \hat{c}_k^T H(-\hat{\lambda}_k^*) &= \hat{c}_k^T \hat{H}(-\hat{\lambda}_k^*), \\ H(-\hat{\lambda}_k^*)\hat{b}_k^T &= \hat{H}(-\hat{\lambda}_k^*)\hat{b}_k^T, \\ \hat{c}_k^T H'(-\hat{\lambda}_k^*)\hat{b}_k^T &= \hat{c}_k^T \hat{H}'(-\hat{\lambda}_k^*)\hat{b}_k^T, \end{aligned} \tag{1.8}$$

y además minimiza la métrica (1.5). En esta tesis, nos enfocamos a buscar un “buen” conjunto inicial de datos $\{\{l_k^0\}, \{r_k^0\}, \{\sigma_k^0\}\}$ de tal manera que el algoritmo MIRIAM convergiera desde este conjunto inicial al conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$, $k = 1, 2, 3, \dots, n$, y por ende al sistema reducido (1.4) en una sola iteración ó en el menor número posible de iteraciones. Para este fin propusimos un proceso de preconditionamiento basado en este mismo algoritmo que está dividido en tres fases, con la esperanza de obtener al final de este proceso un conjunto inicial $\{\{l_k^0\}, \{r_k^0\}, \{\sigma_k^0\}\}$ con una posible propiedad de rápida convergencia hacia el conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$ cuando el algoritmo MIRIAM es

usado nuevamente para construir (1.4) a partir de (1.1).

El resto del trabajo de tesis en este documento se encuentra distribuido de la siguiente manera. En el Capítulo 2 se proporciona una introducción a la teoría lineal, resaltando aspectos y propiedades importantes de sistemas de la forma (1.1). Se le da una atención especial a la manera de obtener la Función de Transferencia $H(s)$ cuando se le aplica la Transformada de Laplace $\mathcal{L}\{\bullet\}$ al sistema (1.1). Concluimos este capítulo con un resultado muy importante al considerar a $H(s)$ como un elemento de los espacios de Hardy.

En el capítulo 3 hacemos un resumen de las principales corrientes o enfoques para llevar a cabo la Reducción de Orden de Modelos. Sólo se toma en consideración un representante de cada una de las dos corrientes existentes. Para el caso de Truncamiento Balanceado, que es el representante de los métodos de Truncamiento, se explica su funcionamiento básico así como las propiedades relacionadas a la cota del error entre (1.1) y (1.4), y a la formación de sistemas reducidos estables. Para Acoplamiento de Momentos que es el representante de la segunda corriente, se hace un estudio más extenso. Se comienza con la manera de expresar las Funciones de Transferencia $H(s)$ y $\hat{H}(s)$ en la forma de series de Taylor usando para esto la expansión en series de Neumann de $(sI_N - A)$ y de $(sI_n - \hat{A})$. Se hace especial énfasis en dos Teoremas (y en su demostración) que detallan las condiciones bajo las cuales $H(s)$ y $\hat{H}(s)$ pueden ser igualadas (acopladas) hasta un cierto número de momentos.

En el capítulo 4 se explica la Teoría matemática necesaria para minimizar la distancia entre el sistema original (1.1) y el sistema reducido (1.4) representada por la métrica (1.5). En el final de este capítulo, se muestra el Algoritmo MIRIAM al que hemos hecho referencia en párrafos anteriores.

En el capítulo 5, presentamos los resultados numéricos de 4 experimentos que llevamos a cabo con la finalidad de poner a prueba el proceso de preconditionamiento al que hicimos referencia anteriormente. Terminamos este trabajo de tesis en el capítulo 6, donde discutimos las conclusiones y futuras direcciones derivadas de esta tesis.

2. Conceptos Básicos de Teoría de Control

Para mostrar la naturaleza de los sistemas de control lineal, se utilizarán dos ejemplos de sistemas dinámicos, para los cuales se obtienen ecuaciones lineales de la forma (1.1). Estas ecuaciones forman el punto de partida de las técnicas de reducción de orden de modelos dinámicos. Sin embargo, el lector puede estar interesado en profundizar mucho más en lo referente a la teoría de control lineal de sistemas, para ello recomendamos la siguiente bibliografía [1,27,28].

2.1. Un ejemplo sencillo, el motor CD

Comencemos con un modelo simple de un motor CD[1] girando una masa inercial, que muestra el desplazamiento angular $\omega(t)$ de la masa inercial como la variable de salida, y el voltaje aplicado $v_{app}(t)$ como la entrada. El objetivo de este ejemplo es controlar el desplazamiento angular mediante la variación del voltaje aplicado.

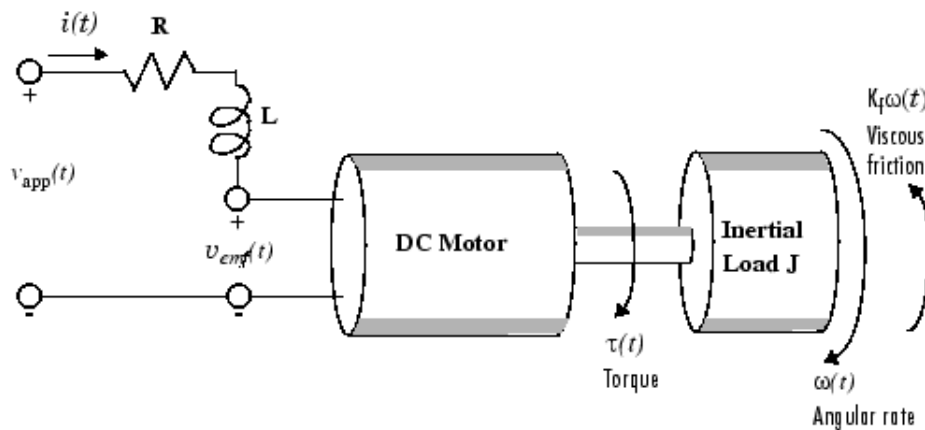


Figura 1: Un modelo simple de un motor CD girando una masa inercial[1].

En este modelo, se idealiza la dinámica del motor; por ejemplo, el campo magnético se supone constante. La resistencia del circuito es denotada por R y la inductancia de la armadura por L . Con este modelo y leyes básicas de la física, es posible obtener ecuaciones diferenciales que describan el comportamiento de este sistema electromecánico. En este ejemplo, la relación entre potencial eléctrico y fuerza

mecánica es la ley de inducción de Faraday, y la ley de Ampère para la fuerza en un conductor que se mueve a través de un campo magnético.

El torque τ visto en la flecha del motor es proporcional a la corriente $i(t)$ inducida por el voltaje aplicado,

$$\tau(t) = K_m i(t)$$

donde K_m , la constante de la armadura, está relacionada a las propiedades físicas del motor, como por ejemplo la intensidad del campo magnético, el número de vueltas del alambre alrededor de la bobina, etc. La fuerza electromotriz (inducida), v_{fem} , es un voltaje proporcional al desplazamiento angular $\omega(t)$ visto en la flecha,

$$v_{fem}(t) = K_b \omega(t)$$

donde K_b , la constante de la fuerza electromecánica, también depende de ciertas propiedades físicas del motor. Las ecuaciones del motor relacionadas a la parte mecánica son derivadas usando la ley de Newton, que establece que la masa inercial J multiplicada por la derivada del desplazamiento angular, es igual a la suma de todos los torques en la flecha del motor,

$$J \frac{d\omega}{dt} = \sum_i \tau_i = -K_f \omega(t) + K_m i(t)$$

donde K_f es una aproximación lineal de la fricción viscosa. Finalmente, las ecuaciones del motor relacionadas a la parte eléctrica pueden ser descritas por

$$v_{app}(t) - v_{emf}(t) = L \frac{di}{dt} + Ri(t)$$

o, resolviendo para el voltaje aplicado y substituyendo para la fuerza electromecánica,

$$v_{app}(t) = L \frac{di}{dt} + Ri(t) + K_b \omega(t)$$

Esta secuencia de ecuaciones conlleva a un conjunto de dos ecuaciones diferenciales que describen el comportamiento del motor, la primera para la corriente inducida,

$$\frac{di}{dt} = -\frac{R}{L}i(t) - \frac{K_b}{L}\omega(t) + \frac{1}{L}v_{app}(t) \quad (2.1)$$

y la segunda para el desplazamiento angular resultante

$$\frac{d\omega}{dt} = -\frac{1}{J}K_f\omega(t) + \frac{1}{J}K_m i(t) \quad (2.2)$$

dadas las dos ecuaciones diferenciales obtenidas previamente, ahora podemos desarrollar una representación en el estado-espacio del motor DC como un sistema dinámico. La corriente $i(t)$ y el desplazamiento angular $\omega(t)$ son los dos estados del sistema. El voltaje aplicado v_{app} es la entrada del sistema, y la velocidad angular $\omega(t)$ es la salida

$$\begin{bmatrix} i'(t) \\ \omega'(t) \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{K_b}{L} \\ \frac{K_m}{J} & -\frac{K_f}{J} \end{bmatrix} \begin{bmatrix} i(t) \\ \omega(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} v_{app}(t)$$

$$y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} i(t) \\ \omega(t) \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} v_{app}(t)$$

que pueden ser representadas como

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \quad (2.3)$$

Estas ecuaciones describen completamente el estado del sistema $x(t) = [i(t), \omega(t)]^T$ y su salida $y(t)$ al tiempo $t \geq 0$. El anterior es un ejemplo de un Single-Input-Single-Output (SISO) sistema dinámico continuo lineal invariante en el tiempo. Se llama SISO porque tiene sólo una entrada y sólo una salida. Es llamado continuo debido a la variable $t \in \{\mathbb{R}_+ \cup 0\}$, y es llamado lineal invariante en el tiempo (LIT) debido a que las matrices A , B y C no cambian en el tiempo.

2.2. Sistema de control de nivel

Considérese el sistema de control de nivel[2] mostrado en la figura 2. En el sistema, \bar{Q}_1 y \bar{Q}_2 son flujos de entrada estables (constantes), y \bar{H}_1 y \bar{H}_2 son alturas fijas. Las cantidades q_{i1} , q_{i2} , h_1 , h_2 , q_1 , y q_0 son consideradas pequeñas. Considerando a q_{i1} y q_{i2} como entradas, y h_1 y h_2 como las salidas, podemos desarrollar el conjunto de ecuaciones que modelen la dinámica del sistema.

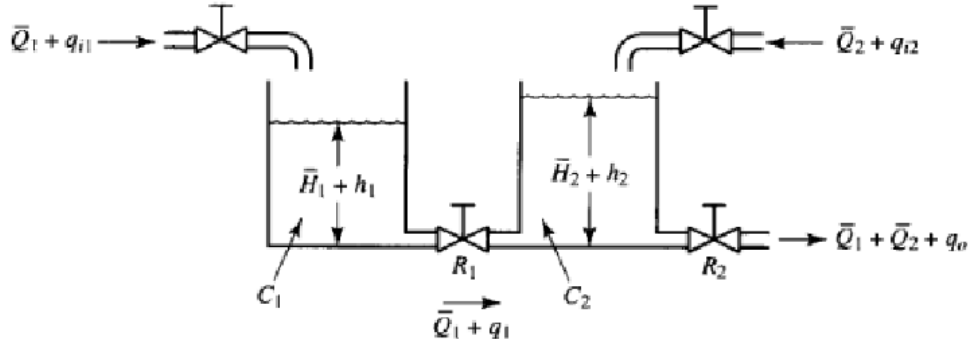


Figura 2: Sistema líquido-nivel[2].

Las ecuaciones para el sistema son

$$C_1 \frac{dh_1}{dt} = (q_{i1} - q_1) \quad (2.4)$$

$$\frac{h_1 - h_2}{R_1} = q_1 \quad (2.5)$$

$$C_2 \frac{dh_2}{dt} = (q_1 + q_{i2} - q_0) \quad (2.6)$$

$$\frac{h_2}{R_2} = q_0 \quad (2.7)$$

substituyendo la ecuación (2.5) en la ecuación (2.4) resulta en

$$\frac{dh_1}{dt} = \frac{1}{C_1} \left(q_{i1} - \frac{h_1 - h_2}{R_1} \right) \quad (2.8)$$

eliminando q_1 y q_0 de la ecuación (2.6) mediante el uso de las ecuaciones (2.5) y (2.7) da como resultado

$$\frac{dh_2}{dt} = \frac{1}{C_2} \left(\frac{h_1 - h_2}{R_1} + q_{i2} - \frac{h_2}{R_2} \right) \quad (2.9)$$

definiendo a las variables de estado x_1 y x_2 como

$$\begin{aligned}x_1 &= h_1 \\x_2 &= h_2\end{aligned}$$

las variables de entrada u_1 y u_2 como

$$\begin{aligned}u_1 &= q_{i1} \\u_2 &= q_{i2}\end{aligned}$$

y las variables de salida y_1 y y_2 como

$$\begin{aligned}y_1 &= h_1 = x_1 \\y_2 &= h_2 = x_2\end{aligned}$$

entonces las ecuaciones (2.8) y (2.9) se pueden escribir como

$$\begin{aligned}\dot{x}_1 &= -\frac{1}{R_1 C_1} x_1 + \frac{1}{R_1 C_1} x_2 + \frac{1}{C_1} u_1 \\ \dot{x}_2 &= \frac{1}{R_1 C_2} x_1 - \left(\frac{1}{R_1 C_2} + \frac{1}{R_2 C_2}\right) x_2 + \frac{1}{C_2} u_2.\end{aligned}$$

En la forma estándar de representación matricial, tenemos

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -\frac{1}{R_1 C_1} & \frac{1}{R_1 C_1} \\ \frac{1}{R_1 C_2} & -\left(\frac{1}{R_1 C_2} + \frac{1}{R_2 C_2}\right) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{C_1} & 0 \\ 0 & \frac{1}{C_2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

que es la ecuación de estado, y

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

es la ecuación de salida. Las ecuaciones previas se pueden resumir en

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned}$$

las cuales tienen la misma forma que (2.3). Este es un ejemplo de un Multiple-Input-Multiple-Output (MIMO) sistema dinámico continuo, linealmente invariante en el tiempo con dos entradas y dos salidas.

2.3. Sistemas Micro Electro Mecánicos (SMEM)

Se pueden encontrar aplicaciones más complejas en la vida real en donde surgen sistemas de ecuaciones diferenciales del tipo anterior, por ejemplo en el campo del diseño de SMEM. En [13] se establece que los SMEM son sistemas integrados que combinan componentes eléctricos y mecánicos, usualmente son fabricados empleando técnicas de circuitos integrados (CI) y pueden variar en tamaño, desde los micrómetros hasta los milímetros. En la forma más general, los SMEM consisten de micro estructuras mecánicas, micro sensores y micro actuadores, todos integrados en el mismo chip. Varios ejemplos ilustrativos de modelos SMEM de gran escala, pueden consultarse en la colección de problemas de referencia Oberwolfach[3]. El comportamiento de los SMEM puede ser descrito por un sistema de ecuaciones diferenciales parciales (EDP's) adecuado. Su linearización y discretización por medio de métodos como el de elemento finito o diferencias finitas produce un sistema de ecuaciones diferenciales ordinarias (EDO's) de la misma forma que (2.3). El número resultante de ecuaciones depende directamente de la calidad de la discretización y típicamente es muy grande, puede alcanzar fácilmente el orden de los millones.

En lo que resta del capítulo se presentarán algunos conceptos básicos de teoría de control de sistemas dinámicos lineales, y se pone especial interés en el caso de sistemas continuos. Para el caso de sistemas discretos el lector puede encontrar más información en[4,12]. De acuerdo a [4], un sistema dinámico continuo LIT se puede describir completamente por el siguiente conjunto de ecuaciones algebraicas y diferenciales

$$\Sigma : \quad \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t). \end{cases} \quad (2.10)$$

Aquí $t \in \mathbb{R}_+$ es la variable que corresponde al tiempo, el vector $x(t) \in \mathbb{R}^N$ denota el estado del sistema, $u(t) \in \mathbb{R}^m$ es el vector de entrada y $y(t) \in \mathbb{R}^p$ es el vector de salida. Los arreglos $A \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^{N \times m}$ y $C \in \mathbb{R}^{p \times N}$ son conocidos como la matriz de estado, de entrada y de salida respectivamente. De la matriz $A \in \mathbb{R}^{N \times N}$, el número N es conocido como la dimensión del sistema.

Cuando $m = p = 1$, entonces la ecuación (2.10) es la representación matemática de un sistema dinámico Single-Input-Single-Output (SISO), de otra manera, cuando $m > 1$ y $p > 1$, entonces es la representación matemática de un sistema dinámico Multiple-Input-Multiple-Output (MIMO).

La solución del sistema (2.10) queda definida por el siguiente par de ecuaciones

$$x(t) = e^{At}x_0 + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau, \quad t \geq t_0. \quad (2.11)$$

$$y(t) = Ce^{At}x_0 + C \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau, \quad t \geq t_0. \quad (2.12)$$

En Ingeniería Eléctrica, a la ecuación (2.10) comúnmente se le refiere de la siguiente forma

$$\Sigma = \left[\begin{array}{c|c} A & B \\ \hline C & \end{array} \right].$$

Si $x(t) = T\tilde{x}(t)$ donde T es una matriz no-singular, entonces $\dot{x}(t) = T\dot{\tilde{x}}(t)$. Usando ambas ecuaciones en (2.10) da como resultado

$$\begin{aligned} T\dot{\tilde{x}}(t) &= AT\tilde{x}(t) + Bu(t) \\ y(t) &= CT\tilde{x}(t) \end{aligned}$$

y multiplicando el primer renglón por T^{-1} da como resultado

$$\begin{aligned} \dot{\tilde{x}}(t) &= T^{-1}AT\tilde{x}(t) + T^{-1}Bu(t) \\ y(t) &= CT\tilde{x}(t). \end{aligned} \quad (2.13)$$

Un resultado de álgebra lineal establece que las matrices A y $T^{-1}AT$ tienen exactamente los mismos valores propios, por lo tanto (2.10) y (2.13) se les denomina como algebraicamente equivalentes.

2.4. Estabilidad

Un sistema dinámico de la forma (2.10) se dice que es estable, o más precisamente “asintóticamente estable”, si todos los valores propios de la matriz $A \in \mathbb{R}^{N \times N}$ yacen en el lado izquierdo del plano complejo, i.e., $Re(eig(A)) < 0$.

2.5. Alcanzabilidad

Un estado $x(t) \in \mathbb{R}^N$ al tiempo $t < \infty$, es alcanzable desde el estado inicial cero $x(0) = 0$, si existe una única entrada $u(t) \in \mathbb{R}^m$, tal que $x(t)$ está únicamente determinado por la relación

$$x(t) = \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau. \quad (2.14)$$

En aplicaciones físicas, en general se tiene energía finita y en este caso debe cumplirse que $\|u(t)\|_2^2 = \int_0^t u^T(\tau)u(\tau) d\tau$ es acotada. El subespacio $\mathbb{X}^r \subset \mathbb{R}^N$ que consiste de todos los elementos $x(t)$ que pueden ser alcanzados por (2.14) es llamado el subespacio alcanzable. La matriz infinita

$$\mathcal{R}(A, B) = [B, AB, AB^2, \dots, AB^{N-1}, \dots] \quad (2.15)$$

se conoce como la matriz de alcanzabilidad. Para propósitos computacionales, sólo los primeros N bloques de la matriz de alcanzabilidad son de interés

$$\mathcal{R}_N(A, B) = [B, AB, \dots, AB^{N-1}]. \quad (2.16)$$

De hecho, en [13] se muestra que el rango de (2.15) y la expansión de sus bloques está determinada por los primeros N términos. Además se muestra que

$$\mathbb{X}^r = \text{imagen}(\mathcal{R}(A, B)). \quad (2.17)$$

Definición. Gramiano de Alcanzabilidad. El Gramiano de alcanzabilidad finito $\mathcal{P}(t)$ al tiempo $t < \infty$ de un sistema como (2.10) está dado por

$$\mathcal{P}(t) = \int_0^t e^{A\tau} B B^T e^{A^T \tau} d\tau, \quad t \in \mathbb{R}_+ \quad (2.18)$$

en donde B^T significa la transpuesta de B . Si (2.10) es estable, entonces tiene sentido evaluar (2.18) cuando $t \rightarrow \infty$. En este caso

$$\mathcal{P} = \lim_{t \rightarrow \infty} \mathcal{P}(t) = \int_0^\infty e^{A\tau} B B^T e^{A^T \tau} d\tau \quad (2.19)$$

y es llamado el Gramiano de alcanzabilidad infinito. Éste Gramiano satisface la ecuación de Lyapunov para un sistema estable (2.10)

$$A\mathcal{P} + \mathcal{P}A^T + B B^T = 0. \quad (2.20)$$

Otro concepto importante relacionado con la teoría de control de sistemas es la energía de entrada.

Definición. Menor Energía de Entrada. Sea $\bar{u}(t) \in \mathbb{R}^m$ una entrada que lleva al sistema continuo LIT (2.10) del estado cero $x(0) = 0$ al estado $x(t) = \bar{x}$ al tiempo t . Esto significa

$$\bar{x} = \int_0^t e^{A(t-\tau)} B \bar{u}(\tau) d\tau. \quad (2.21)$$

La menor energía de entrada $\epsilon_t(\bar{x})$ requerida para alcanzar el estado \bar{x} desde el estado cero después de un tiempo t está definida como

$$\epsilon_t(\bar{x}) = \min\{\|\bar{u}\|_2^2 \mid \bar{u} : [0, t] \rightarrow \mathbb{R}^m \text{ y } \text{satisface}(2.21)\}. \quad (2.22)$$

2.6. Observabilidad

Observabilidad es la medida en la cual dos diferentes estados $\bar{x}(t), \tilde{x}(t) \in \mathbb{R}^N$ dada una entrada $u(t) \in \mathbb{R}^m$ dan como resultado diferentes salidas $y(t) \in \mathbb{R}^p$.

Definición. Observabilidad. Sean $\bar{x}(t), \tilde{x}(t) \in \mathbb{R}^N$ dos estados diferentes tales que $\bar{x}(t) - \tilde{x}(t) \neq 0$. Si existe un tiempo t y una entrada $u(t) \in \mathbb{R}^m$ tales que por un lado

$$\bar{y}(t) = Ce^{At}\bar{x}(t) + C \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau, \quad t > 0. \quad (2.23)$$

y por otro lado

$$\tilde{y}(t) = Ce^{At}\tilde{x}(t) + C \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau, \quad t > 0. \quad (2.24)$$

de tal manera que $\bar{y}(t) - \tilde{y}(t) \neq 0$, entonces los estados $\bar{x}(t)$ $\tilde{x}(t)$ son distinguibles u observables.

Una matriz importante es la matriz infinita de observabilidad

$$\mathcal{O}(C, A) = [C^T, A^T C^T, (A^2)^T C^T, \dots, (A^{N-1})^T C^T, \dots]^T \quad (2.25)$$

el rango de la matriz infinita de observabilidad \mathcal{O} y la expansión de sus columnas, están determinados por los primeras N columnas. Entonces para los propósitos computacionales, es de interés la siguiente matriz finita de observabilidad

$$\mathcal{O}_N(C, A) = [C^T, A^T C^T, (A^2)^T C^T, \dots, (A^{N-1})^T C^T]^T. \quad (2.26)$$

Definición. Gramiano de Observabilidad. El Gramiano de observabilidad finito $\mathcal{Q}(t)$ al tiempo $t < \infty$ de un sistema como (2.10) está dado por la siguiente ecuación

$$\mathcal{Q}(t) = \int_0^t e^{A^T \tau} C^T C e^{A \tau} d\tau, \quad t \in \mathbb{R}_+ \quad (2.27)$$

Si (2.10) es estable, entonces

$$\mathcal{Q} = \lim_{t \rightarrow \infty} \mathcal{Q}(t) = \int_0^\infty e^{A^T \tau} C^T C e^{A \tau} d\tau \quad (2.28)$$

y es llamado el Gramiano de observabilidad infinito. Este Gramiano es la solución (única) simétrica definida positiva de la ecuación de Lyapunov

$$A^T Q + Q A + C^T C = 0. \quad (2.29)$$

Un concepto importante relacionado con la observabilidad, es la energía de observación.

Definición. Energía de Observación. La energía de la función de salida $y(t) \in \mathbb{R}^p$ al tiempo t causada por el estado inicial $\bar{x}(t) \in \mathbb{R}^N$ con función de entrada cero $u(t) = 0$, sobre el intervalo $[0, t]$, denotada por $\epsilon_t^{obs}(\bar{x})$ o por $\|y(t)\|_2^2$ es llamada la energía producida por $\bar{x}(t)$ sobre $[0, t]$, y esta expresada como

$$\epsilon_t^{obs}(\bar{x}(t)) = \|y(t)\|_2^2 = \int_0^t y^T(t)y(t)dt = \int_0^t \bar{x}(t)^T e^{A^T t} C^T C e^{A t} \bar{x}(t) dt. \quad (2.30)$$

2.7. Controlabilidad

En Ingeniería, un sistema es llamado controlable si es posible alcanzar los objetivos de control cualquiera que estos sean. Se dice que un sistema es más controlable ó menos controlable de acuerdo a la dificultad de ejercer control. A continuación se da una definición más formal.

Definición. Controlabilidad[14]. Sea un sistema como en (2.10), el sistema será controlable si dado cualquier estado $x(0) = c$, existe un tiempo $t_1 > 0$ y una entrada $u(t)$ definida en $[0, t_1]$ tal que $x(t_1) = 0$.

2.8. Comportamiento de Entrada-Salida

En algunas aplicaciones, la descripción del estado del sistema $x(t)$ no es de mucho interés, sólo el comportamiento de $y(t)$ es de interés cuando $u(t)$ es aplicado a (2.10). Esto es, en lugar del mapeo

$$u(t) \mapsto x(t) = e^{A t} x_0 + \int_0^t e^{A(t-\tau)} B u(\tau) d\tau$$

nuestro interés se centra en el mapeo

$$u(t) \mapsto y(t) = C e^{A t} x_0 + C \int_0^t e^{A(t-\tau)} B u(\tau) d\tau.$$

Este último mapeo describe completamente a $y(t)$ sobre el intervalo de tiempo $[0, t)$. A menudo se está interesado en tal descripción sobre el dominio de frecuencias, la manera de hacer esto es a través de la transformada de Laplace $\mathcal{L}\{\bullet\}$ aplicada a (2.10):

$$\begin{aligned}\mathcal{L}\{\dot{x}(t)\} &= \mathcal{L}\{Ax(t) + Bu(t)\} \\ \mathcal{L}\{y(t)\} &= \mathcal{L}\{Cx(t)\}.\end{aligned}$$

Como $\mathcal{L}\{\bullet\}$ es un operador lineal, y por la regla de la transformada de Laplace para la derivada de primer orden, se obtiene

$$\begin{aligned}sX(s) - x(0) &= AX(s) + BU(s) \\ Y(s) &= CX(s).\end{aligned}$$

Suponiendo el estado inicial $x(0) = 0$, entonces

$$\begin{aligned}sX(s) &= AX(s) + BU(s) \\ Y(s) &= CX(s),\end{aligned}\tag{2.31}$$

y substrayendo $AX(s)$ de la primera línea de (2.31) resulta en

$$sX(s) - AX(s) = BU(s),$$

que puede escribirse como

$$(sI_N - A)X(s) = BU(s).$$

Si $(sI_N - A)$ es no-singular, entonces

$$X(s) = (sI_N - A)^{-1}BU(s),$$

y substituyendo esta última expresión en la segunda línea de (2.31) resulta en

$$Y(s) = \underbrace{C(sI_N - A)^{-1}B}_{H(s)}U(s)$$

donde $H(s) = C(sI_N - A)^{-1}B \in \mathbb{C}^{p \times m}$ es conocida como la función de transferencia del sistema (2.10). Entonces

$$H(s) : U(s) \mapsto Y(s)$$

define un mapeo y éste describe completamente el comportamiento de Entrada-Salida del sistema en el dominio de frecuencias.

2.9. Propiedades de la Función de Transferencia

En esta sección se revisarán algunas propiedades de la función de transferencia $H(s)$, en particular sólo aquellas que serán de importancia en lo que resta de la tesis.

Definición. Polos. Para un sistema como (2.10), los valores propios de la matriz A son llamados polos. En algunas ocasiones también son llamados singularidades.

2.9.1. Transformaciones similares

Es posible cambiar la base en la cual representamos el estado de un sistema de la siguiente forma: Sea $z(t) = Mx(t)$, donde $M \in \mathbb{R}^{N \times N}$ es una matriz invertible, entonces

$$x(t) = M^{-1}z(t), \quad (2.32)$$

y tomando la derivada con respecto al tiempo, se obtiene

$$\dot{x}(t) = M^{-1}\dot{z}(t). \quad (2.33)$$

Entonces, en la representación de estado-espacio

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t), \end{aligned}$$

por lo que, multiplicando la primera línea por M , resulta en

$$\begin{aligned} M\dot{x}(t) &= MAM^{-1}z(t) + MBu(t) \\ y(t) &= CM^{-1}z(t). \end{aligned}$$

Substituyendo (2.32) y (2.33) en el anterior conjunto de ecuaciones, se obtiene

$$\begin{aligned} MM^{-1}\dot{z}(t) &= MAM^{-1}z(t) + MBu(t) \\ y(t) &= CM^{-1}z(t), \end{aligned}$$

y tomando la transformada de Laplace del anterior sistema, resulta

$$\begin{aligned} sZ(s) - z(0) &= MAM^{-1}Z(s) + MBU(s) \\ Y(s) &= CM^{-1}Z(s). \end{aligned} \quad (2.34)$$

si $z(0) = 0$, entonces (2.34) se convierte en

$$sZ(s) = MAM^{-1}Z(s) + MBU(s) \quad (2.35)$$

$$Y(s) = CM^{-1}Z(s). \quad (2.36)$$

y substrayendo $MAM^{-1}Z(s)$ de (2.35) resulta en

$$sZ(s) - MAM^{-1}Z(s) = MBU(s)$$

que puede escribirse como

$$(sI_N - MAM^{-1})Z(s) = MBU(s).$$

Si $(sI_N - MAM^{-1})$ es invertible, entonces

$$Z(s) = (sI_N - MAM^{-1})^{-1}MBU(s),$$

y substituyendo este resultado en (2.36), se obtiene

$$Y(s) = CM^{-1}(sI_N - MAM^{-1})^{-1}MBU(s). \quad (2.37)$$

Como $(sI_N - MAM^{-1}) = M(sI_N - A)M^{-1}$, entonces $(sI_N - MAM^{-1})^{-1} = [M(sI_N - A)M^{-1}]^{-1} = M(sI_N - A)^{-1}M^{-1}$, por lo cual (2.37) se convierte en

$$\begin{aligned} Y(s) &= CM^{-1}[M(sI_N - A)^{-1}M^{-1}]MBU(s) \\ &= CM^{-1}M(sI_N - A)^{-1}M^{-1}MBU(s) \\ &= \underbrace{C(sI_N - A)^{-1}B}_{H(s)}U(s) \\ &= H(s)U(s). \end{aligned}$$

Conclusión: La función de transferencia es invariante bajo la transformación $z(t) = Mx(t)$.

2.10. Parámetros de Markov

Hemos visto que por medio de la transformada de Laplace aplicada a (2.10), podemos obtener la relación

$$H(s) = C(sI_N - A)^{-1}B.$$

Si A es invertible, entonces

$$\begin{aligned} (sI_N - A) &= sAA^{-1} - A \\ &= -A + sAA^{-1} \\ &= -A(I_N - sA^{-1}), \end{aligned}$$

y substituyendo este resultado en $H(s)$ obtenemos lo siguiente

$$\begin{aligned} H(s) &= C[-A(I_N - sA^{-1})]^{-1}B \\ &= C(I_N - sA^{-1})^{-1}(-A)^{-1}B. \end{aligned}$$

Usando la expansión en series de Neumann para $(I_N - sA^{-1})^{-1}$

$$(I_N - sA^{-1})^{-1} = \sum_{k=0}^{\infty} (sA^{-1})^k,$$

entonces $H(s)$ resulta en

$$\begin{aligned} H(s) &= C \sum_{k=0}^{\infty} (sA^{-1})^k (-A)^{-1}B \\ H(s) &= \sum_{k=0}^{\infty} \underbrace{-C(A^{-1})^k (A^{-1}B)}_{h_k(0)} s^k \end{aligned} \quad (2.38)$$

$$H(s) = \sum_{k=0}^{\infty} h_k(0) s^k, \quad (2.39)$$

que es la expansión en series de Taylor de $H(s)$ alrededor de $s_0 = 0, s_0 \in \mathbb{C}$. El término $h_k(0) = -C(A^{-1})^k (A^{-1}B)$ es llamado el k -ésimo momento de $H(s)$ alrededor de cero.

Si $\tilde{A} = A - s_0 I_N$ es también invertible, entonces por un procedimiento similar al anterior, podemos calcular el k -ésimo momento $h_k(s_0)$ en $s_0 \neq 0$ de la función de transferencia

$$\begin{aligned} H(s) &= C[(s - s_0)I_N - \tilde{A}]^{-1}B \\ &= \sum_{k=0}^{\infty} \underbrace{-C(\tilde{A}^{-1})^k (\tilde{A}^{-1}B)}_{h_k(s_0)} (s - s_0)^k \end{aligned} \quad (2.40)$$

donde $h_k(s_0) = -C(\tilde{A}^{-1})^k (\tilde{A}^{-1}B)$ es el k -ésimo momento de $H(s)$ alrededor de $s_0 \neq 0, s_0 \in \mathbb{C}$. Un aspecto importante de (2.40) es averiguar como será su forma cuando $s_0 \rightarrow \infty$, para esto, consideremos nuevamente $(sI_N - A)$

$$(sI_N - A) = (I_N - \frac{1}{s}A)(sI_N)$$

de tal manera que

$$\begin{aligned}
H(s) &= C[(I_N - \frac{1}{s}A)(sI_N)]^{-1} B \\
&= C(sI_N)^{-1}(I_N - \frac{1}{s}A)^{-1} B \\
&= C(\frac{1}{s}I_N)(I_N - \frac{1}{s}A)^{-1} B,
\end{aligned}$$

y por Neumann

$$\begin{aligned}
H(s) &= C(\frac{1}{s}I_N) \sum_{k=0}^{\infty} (\frac{1}{s}A)^k B \\
&= \sum_{k=0}^{\infty} \frac{1}{s^{k+1}} \underbrace{CA^k B}_{m_k}
\end{aligned} \tag{2.41}$$

$$= \sum_{k=0}^{\infty} \frac{1}{s^{k+1}} \cdot m_k, \tag{2.42}$$

donde los m_k 's, $k = 0, 1, 2, \dots, \infty$, son los parámetros de Markov ó momentos de alta frecuencia.

Definición. Matriz de Hankel. El arreglo \mathcal{H} de parámetros de Markov

$$\mathcal{H} = \begin{bmatrix} m_0 & m_1 & m_2 & m_3 & \cdots \\ m_1 & m_2 & m_3 & m_4 & \cdots \\ m_2 & m_3 & m_4 & m_5 & \cdots \\ m_3 & m_4 & m_5 & m_6 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

es llamado la Matriz de Hankel y tiene un número infinito de renglones y de columnas.

No es difícil de ver que \mathcal{H} puede ser factorizada como el producto de $\mathcal{O}(C, A)$ y $\mathcal{R}(A, B)$

$$\mathcal{H} = \begin{bmatrix} CB & CAB & CA^2B & CA^3B & \cdots \\ CAB & CA^2B & CA^3B & CA^4B & \cdots \\ CA^2B & CA^3B & CA^4B & CA^5B & \cdots \\ CA^3B & CA^4B & CA^5B & CA^6B & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$\mathcal{H} = \underbrace{\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ \vdots \end{bmatrix}}_{\mathcal{O}(C,A)} \underbrace{[B \ AB \ A^2B \ A^3B \ \dots]}_{\mathcal{R}(A,B)}.$$

Debido a que ambas matrices $\mathcal{O}(C, A)$ y $\mathcal{R}(A, B)$ son de rango N , también lo es \mathcal{H} bajo la condición de que Σ sea alcanzable y observable.

2.11. Espacios de Hardy

Los espacios de Hardy juegan un papel importante en Teoría de Control. Estos espacios suministran una plataforma para el estudio de funciones analíticas en $\mathbb{C}_\alpha^+ = \{s \in \mathbb{C} : \text{Re}(s) > \alpha\}, \alpha \in \mathbb{R}$. Un ejemplo de tales funciones y que son de interés, son las funciones de transferencia de sistemas continuos LIT. En [4], después de una explicación completa considerando a $H(s)$ como un elemento de los espacios de Hardy, se concluye que

$$\|H(\bullet)\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[H(i\omega)]^*[H(i\omega)]d\omega \right)^{\frac{1}{2}}. \quad (2.43)$$

Éste es un resultado muy importante en los capítulos por venir. Para profundizar más en el estudio del análisis complejo y particularmente en espacios de Hardy se pueden consultar las siguientes referencias [15,16,17,18,19].

3. Técnicas de reducción. Dos enfoques diferentes

Las técnicas de reducción de orden de modelos se pueden agrupar en dos grandes grupos. Por una parte tenemos al grupo de métodos basados en el Gramiano, que utilizan la descomposición en valores singulares (SVD, por sus siglas en Inglés), que retienen las partes más dominantes del sistema y descartan el resto. Una técnica representativa de este grupo se conoce como Truncamiento Balanceado. Y por otra parte tenemos a los métodos basados en interpolación, que utilizan técnicas de subespacios de Krylov, donde una técnica representativa de este grupo es conocida como Acoplamiento de Momentos.

Para el Truncamiento Balanceado se explicarán solamente los conceptos básicos, sin embargo sí se dará una explicación exhaustiva del segundo método. Antes de abordar estas descripciones, primero precisaremos matemáticamente en que consiste el problema objeto de estudio, es decir precisaremos en que consiste la Reducción de Orden del Modelo Σ .

3.1. Definición del Problema

El problema puede establecerse de la siguiente manera:

Dado el sistema continuo asintóticamente estable LIT

$$\Sigma : \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{cases} \quad (3.1)$$

con función de transferencia

$$H(s) = C(sI_N - A)^{-1}B \quad (3.2)$$

donde $A \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^{N \times m}$, $C \in \mathbb{R}^{p \times N}$ y $s \in \mathbb{C}$, **construir un nuevo sistema continuo LIT**

$$\hat{\Sigma} : \begin{cases} \hat{\dot{x}}(t) &= \hat{A}\hat{x}(t) + \hat{B}u(t) \\ \hat{y}(t) &= \hat{C}\hat{x}(t) \end{cases} \quad (3.3)$$

con función de transferencia

$$\hat{H}(s) = \hat{C}(sI_n - \hat{A})^{-1}\hat{B} \quad (3.4)$$

de menor dimensión que el original, esto es, **calcular las matrices** $\hat{A} \in \mathbb{R}^{n \times n}$, $\hat{B} \in \mathbb{R}^{n \times m}$ y $\hat{C} \in \mathbb{R}^{p \times n}$, donde $n < N$ y $s \in \mathbb{C}$, de tal manera que la distancia entre las funciones de transferencia (3.2) y (3.4) se pueda medir con la métrica

$$d(H(s), \hat{H}(s)) = \|G(s)\|_{\mathcal{H}_2}, \quad (3.5)$$

siendo $G(s) = H(s) - \hat{H}(s)$, y (3.5) mínima. La forma explícita de la norma es

$$\|G(s)\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[G(i\omega)]^* [G(i\omega)] d\omega \right)^{\frac{1}{2}}.$$

Ejemplo: Del siguiente sistema dinámico original de dimensión $N = 6$, con $m = 3$ entradas y $p = 3$ salidas

$$\dot{x}(t) = \begin{bmatrix} -1,3412 & 6,8456 & 9,6754 & -1,4679 & -1,3287 & -4,8147 \\ -6,9877 & -0,9857 & 4,4471 & 4,1546 & -2,6167 & 2,7727 \\ -9,5690 & -4,6612 & -1,3615 & 7,3580 & -2,6902 & 6,1452 \\ 1,9799 & -4,4482 & -6,8801 & -1,0269 & 1,9911 & 2,3206 \\ 1,8662 & 2,8061 & 2,75 & -1,3474 & -1,4048 & -2,2738 \\ 4,4474 & -2,0371 & -6,6444 & -2,2407 & 2,4679 & -1,3837 \end{bmatrix} x(t)$$

$$+ \begin{bmatrix} 0,3005 & 0,4128 & 0 \\ 0 & -0,9870 & -0,1318 \\ 0,8155 & 0,7596 & 0,5954 \\ 0 & -0,6572 & 1,0468 \\ 0 & -0,6039 & -0,1980 \\ 0 & 0,1769 & 0,3277 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} -0,2383 & -0,6169 & 0,0923 & -0,7371 & 0,8671 & 0,1837 \\ 0,2296 & 0,2748 & 1,7298 & -1,7499 & -0,0799 & 0,2908 \\ 0,44 & 0 & -0,6086 & 0,9105 & 0,8985 & 0,1129 \end{bmatrix} x(t),$$

donde su Función de Transferencia es

$$H(s) = \begin{bmatrix} 0,0988 - 0,0448s & 0,279 - 0,2509s & -0,2407 + 0,1862s \\ 0,2539 + 0,0107s & 0,9542 - 0,6451s & -0,1892 + 0,4905s \\ 0,0328 - 0,1757s & -0,6225 + 0,6014s & 0,2978 - 0,6160s \end{bmatrix}$$

el sistema reducido de dimensión $n = 3$ con las mismas $m = 3$ entradas y $p = 3$ salidas

$$\hat{\hat{x}}(t) = \begin{bmatrix} -0,6159 & -0,1073 & 0,3217 \\ -0,1044 & -0,8471 & 0,0507 \\ 0,1481 & 0,1350 & -1,6393 \end{bmatrix} \hat{x}(t) + \begin{bmatrix} -0,5250 & -0,6399 & -0,7172 \\ -0,0889 & -1,2980 & 0,3461 \\ 0,1262 & 0,3105 & -0,4594 \end{bmatrix} u(t)$$

$$\hat{y}(t) = \begin{bmatrix} -0,1244 & -0,3067 & 1,2015 \\ 0,0402 & -0,9855 & 1,3199 \\ -0,5336 & 1,1245 & -0,0215 \end{bmatrix} \hat{x}(t)$$

con su Función de Transferencia

$$H(s) = \begin{bmatrix} 0,1084 - 0,0486s & 0,3785 - 0,2432s & -0,2477 + 0,1866s \\ 0,1136 - 0,007s & 0,7818 - 0,6694s & -0,4217 + 0,4699s \\ 0,0594 - 0,1738s & -0,5715 + 0,6169s & 0,3318 - 0,6015s \end{bmatrix}$$

difiere del original a una distancia

$$d(H(s), \hat{H}(s)) = \| H(s) - \hat{H}(s) \|_{\mathcal{H}_2}$$

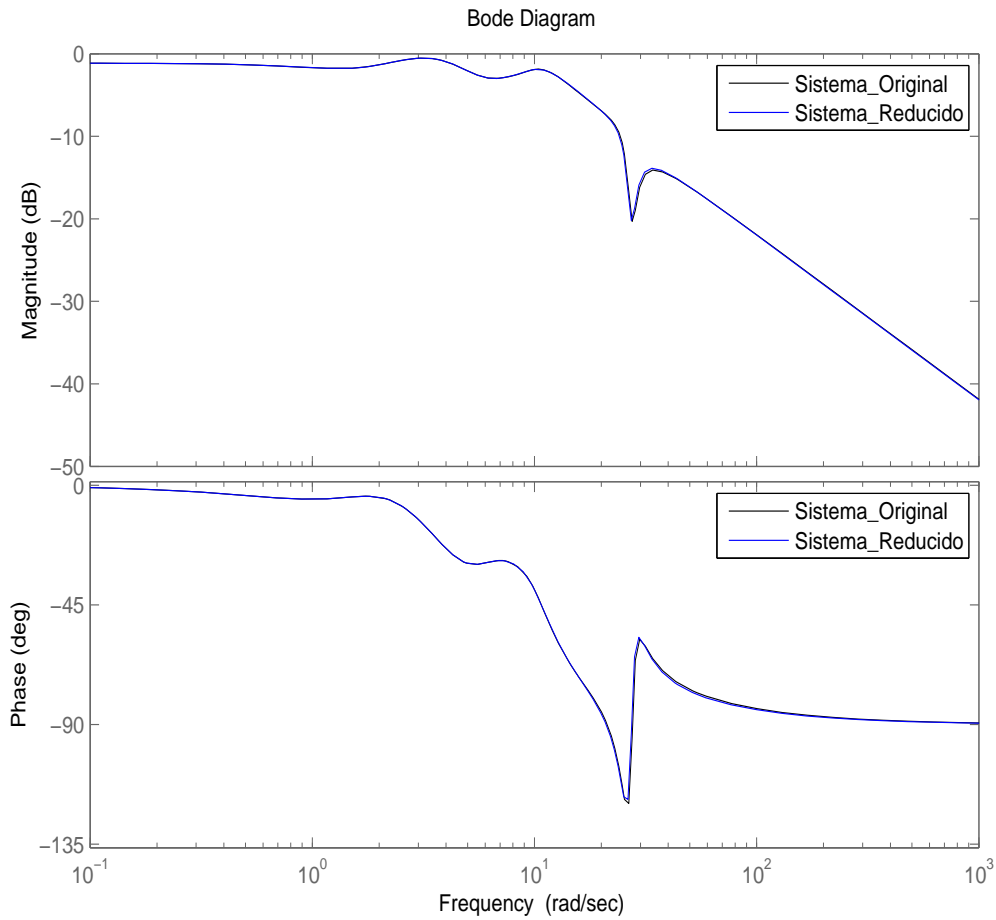
$$d(H(s), \hat{H}(s)) = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[H(i\omega) - \hat{H}(i\omega)]^* [H(i\omega) - \hat{H}(i\omega)] d\omega \right)^{\frac{1}{2}}$$

$$d(H(s), \hat{H}(s)) = 0,9382$$

El anterior es un ejemplo muy sencillo que tiene como objetivo el mostrar numéricamente que es lo que buscamos cuando hablamos de reducción de orden de modelos dinámicos, sin embargo no es comparable en tamaño o dimensión con aquellos sistemas derivados de aplicaciones reales en las ciencias o en la industria en donde N puede llegar al orden de millones como se dijo en la sección 2.3. Únicamente usamos este ejemplo para que el lector tenga una idea más clara de lo que se persigue cuando deseamos reducir un sistema dinámico original como en (3.1) a uno como en (3.3).

Una herramienta muy útil que sirve para visualizar la cercanía del sistema reducido $\hat{\Sigma}$ al sistema original Σ en un rango de frecuencias para $s \in \mathbb{C}$, se provee en MATLAB[®]. Esta herramienta es conocida como BODE PLOT, y es un gráfico que muestra la respuesta de ambas Funciones de Transferencia $H(s)$ y $\hat{H}(s)$ en el mismo rango de frecuencias para $s \in \mathbb{C}$. Por ejemplo, para un sistema dinámico

original (línea negra) de dimensión $N = 100$ con $m = 1$ entrada y $p = 1$ salida, donde el sistema reducido (línea azul) de dimensión $N = 31$ con $m = 1$ entrada y $p = 1$ salida que lo aproxima, tienen el siguiente comportamiento gráfico en el dominio de frecuencias



Para este caso en particular, el comportamiento del sistema reducido es mucho muy cercano al del sistema original en este rango de frecuencia en específico. Podemos concluir entonces que el sistema reducido de dimensión $N = 31$ es una muy buena aproximación del sistema original de dimensión $N = 100$, y por lo tanto puede ser usado como su sustituto en aquellas aplicaciones computacionales que así lo requieran.

3.2. Truncamiento Balanceado

Este método fue primero propuesto por Moore[5], y de entre las técnicas basadas en el Gramiano, el Truncamiento Balanceado se ha convertido en el método más popular de reducción de orden debido a su simplicidad y a las elegantes propiedades matemáticas que se pueden derivar de él. La idea principal de este método, radica en llevar al sistema (3.1) a una forma específica en donde el Gramiano de alcanzabilidad y el Gramiano de observabilidad son idénticos y diagonales. A esta forma en específico se le llama Realización Balanceada

$$\mathcal{P} = \mathcal{Q} = \mathcal{S} = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_N \end{bmatrix} \quad (3.6)$$

donde

$$\underbrace{\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \dots \geq \sigma_N}_{\text{Valores Singulares de Hankel}} > 0.$$

donde σ_k , $k = 1, 2, 3, \dots, N$, son los Valores Singulares de Hankel del sistema Σ . Para llevar a cabo tal realización, primeramente tenemos que calcular la matriz de transformación T que simultáneamente diagonalice el Gramiano \mathcal{P} y \mathcal{Q} . Para ello, consideremos la representación equivalente en el espacio-estado del sistema Σ

$$\left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \end{array} \right] = \left[\begin{array}{c|c} TAT^{-1} & TB \\ \hline CT^{-1} & \end{array} \right] \quad (3.7)$$

Si \mathcal{P} y \mathcal{Q} son las soluciones únicas, simétricas y definidas positivas, de las ecuaciones de Lyapunov (2.20) y (2.29), entonces las transformaciones $\tilde{\mathcal{P}} = T\mathcal{P}T^T$ y $\tilde{\mathcal{Q}} = T^{-T}\mathcal{Q}T^{-1}$ son las soluciones únicas, simétricas y definidas positivas, de las ecuaciones de Lyapunov

$$\begin{aligned} \tilde{A}\tilde{\mathcal{P}} + \tilde{\mathcal{P}}\tilde{A}^T + \tilde{B}\tilde{B}^T &= 0 \\ \tilde{A}^T\tilde{\mathcal{Q}} + \tilde{\mathcal{Q}}\tilde{A} + \tilde{C}^T\tilde{C} &= 0, \end{aligned} \quad (3.8)$$

y el siguiente Lema se cumple

Lema 3.2.1. Dado el sistema estable continuo LIT (3.1) que es alcanzable y observable, con sus correspondientes Gramiano \mathcal{P} y Gramiano \mathcal{Q} , una transformación Balanceada T está dada por

$$T = \mathcal{S}^{1/2}K^TU^{-1} \quad y \quad T^{-1} = UK\mathcal{S}^{-1/2} \quad (3.9)$$

donde U es el factor de Cholesky de \mathcal{P} :

$$\mathcal{P} = UU^T$$

y las matrices K y \mathcal{S} , donde $\mathcal{S} = \text{diag}(\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_N)$, resultan de la descomposición espectral de la matriz positiva definida UQU^T :

$$UQU^T = K\mathcal{S}^2K^T.$$

Demostración: Ver en [20].

El anterior Lema proporciona en forma implícita un procedimiento para calcular la transformación T , y se puede programar fácilmente. De hecho, MATLAB cuenta con sus propios comandos para calcular la transformación T tomando como referencia las ecuaciones (3.8). Más aún, MATLAB tiene una rutina integrada que calcula directamente la realización balanceada del sistema (3.1).

Ahora, supongamos que hemos calculado la realización balanceada T y es tal que

$$\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \dots \geq \sigma_n \geq \sigma_{n+1} \geq \dots > \sigma_N > 0$$

de tal manera que podemos llevar a cabo la siguiente partición

$$S = \left[\begin{array}{c} \left[\begin{array}{ccc} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{array} \right] \\ \left[\begin{array}{ccc} \sigma_{n+1} & & \\ & \ddots & \\ & & \sigma_N \end{array} \right] \end{array} \right] = \left[\begin{array}{cc} S_1 & 0 \\ 0 & S_2 \end{array} \right]$$

donde $S_1 \in \mathbb{R}^{n \times n}$ agrupa los valores singulares de Hankel más grandes y significativos, i.e., los valores propios del producto $\mathcal{P}\mathcal{Q}$ que corresponden a un sistema dominante, donde los estados que son fáciles de alcanzar, son al mismo tiempo fáciles de observar, y S_2 contiene el resto de los valores que corresponden a un subsistema débil.

Si (3.7) se representa como

$$\left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \end{array} \right] = \left[\begin{array}{c|c} TAT^{-1} & TB \\ \hline CT^{-1} & \end{array} \right] = \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & \end{array} \right] \quad (3.10)$$

con $A_{11} \in \mathbb{R}^{n \times n}$, $B_1 \in \mathbb{R}^{n \times m}$, y $C_1 \in \mathbb{R}^{p \times n}$, entonces el Truncamiento Balanceado tiene como objetivo construir un sistema reducido como en (3.3) truncando las partes débiles del subsistema y reteniendo las partes dominantes en las matrices anteriores. Una definición formal para Truncamiento Balanceado es como sigue

Definición. Truncamiento Balanceado. Si (3.10) es la representación estable balanceada en el estado-espacio del sistema Σ , entonces

$$\hat{\Sigma} = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \end{array} \right] = \left[\begin{array}{c|c} A_{11} & B_1 \\ \hline C_1 & \end{array} \right], \quad (3.11)$$

es llamado el n -ésimo truncamiento de Σ .

El sistema reducido $\hat{\Sigma}$ construido en esta manera, tiene un par de propiedades muy ilustrativas:

Teorema 3.2.1. Dado el sistema dinámico LIT Σ , el cual es alcanzable, observable y asintóticamente estable, entonces el sistema reducido $\hat{\Sigma}$, construido mediante Truncamiento Balanceado, tiene las siguientes propiedades:

1. $\hat{\Sigma}$ es balanceado y asintóticamente estable.
2. Sean σ_i , $i = 1, 2, 3, \dots, N$, distintos Valores Singulares de Hankel de Σ , y sea $\hat{\Sigma}$ tal que tenga Valores Singulares de Hankel σ_j , $j = 1, 2, 3, \dots, n$, $n < N$. La norma \mathcal{H}_∞ de la diferencia entre el sistema original Σ y el sistema reducido $\hat{\Sigma}$ está acotada superiormente por el doble de la suma de los valores singulares de Hankel descartados:

$$\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_\infty} \leq 2(\sigma_{n+1} + \dots + \sigma_N). \quad (3.12)$$

Demostración: Ver, por ejemplo [5,8,13].

El Truncamiento Balanceado no es recomendable para sistemas originales que sean muy grandes, ya que requiere la solución de dos ecuaciones de Lyapunov y, adicionalmente, una descomposición en valores singulares (SVD). Esto requiere de hasta $O(N^3)$ operaciones, que es mucho muy costoso computacionalmente para sistemas muy grandes.

3.3. Métodos basados en Interpolación

La ecuación (2.39) provee una manera de expresar $H(s)$ como una serie infinita de momentos $h_k(0)$ alrededor de cero. Si desarrollamos (2.39) tenemos lo siguiente

$$H(s) = -C(A^{-1}B) - C(A^{-1})(A^{-1}B)s - C(A^{-1})^2(A^{-1}B)s^2 - \dots \\ - C(A^{-1})^n(A^{-1}B)s^n - \dots \quad (3.13)$$

$$H(s) = h_0(0) + h_1(0)s + h_2(0)s^2 + \dots + h_n(0)s^n + \dots \quad (3.14)$$

Si suponemos que (3.3) existe y es medible a través de (3.5), entonces (3.4) también puede ser expresada como una serie infinita de momentos $\hat{h}_k(0)$ alrededor de cero

$$\hat{H}(s) = -\hat{C}(\hat{A}^{-1}\hat{B}) - \hat{C}(\hat{A}^{-1})(\hat{A}^{-1}\hat{B})s - \hat{C}(\hat{A}^{-1})^2(\hat{A}^{-1}\hat{B})s^2 - \dots \\ - \hat{C}(\hat{A}^{-1})^n(\hat{A}^{-1}\hat{B})s^n - \dots \quad (3.15)$$

$$\hat{H}(s) = \hat{h}_0(0) + \hat{h}_1(0)s + \hat{h}_2(0)s^2 + \dots + \hat{h}_n(0)s^n + \dots \quad (3.16)$$

Los métodos basados en interpolación, tienen como objetivo construir el sistema reducido (3.3) con su respectiva Función de Transferencia (3.4), de tal manera que se puedan hacer coincidir el mayor número posible de momentos $h_k(0)$ y $\hat{h}_k(0)$ en (3.14) y (3.16) respectivamente.

Existen 2 métodos basados en Interpolación mediante los cuales se pueden formar este sistema reducido (3.3) a partir del sistema original (3.1). A saber son el Acoplamiento de Momentos Explícito y el Acoplamiento de Momentos Implícito. Explicaremos con detalle en que consiste cada uno de ellos.

3.3.1. Acoplamiento de momentos explícito

Es posible expresar la ecuación (3.14) de la siguiente manera

$$H(s) = \underbrace{h_0(0) + h_1(0)s + h_2(0)s^2 + \dots + h_n(0)s^n}_{\hat{H}(s)} + \underbrace{h_{n+1}(0)s^{n+1} + \dots}_{R(s)}$$

$$H(s) = \hat{H}(s) + R(s) \quad (3.17)$$

en donde $\hat{H}(s)$ es llamado el aproximante de Padé de $H(s)$, que es la Función de Transferencia del sistema reducido $\hat{\Sigma}$, y $R(s)$ es el residual de $H(s)$. El objetivo de Acoplamiento de momentos explícito es construir este aproximante $\hat{H}(s)$ directamente de cada $\hat{h}_j(0)$, $j = 0, 1, 2, \dots, n$, de la siguiente manera

$$\begin{aligned} \hat{h}_0(0) &= h_0(0) = -C(A^{-1}B) \\ \hat{h}_1(0) &= h_1(0) = -C(A^{-1})(A^{-1}B) \\ \hat{h}_2(0) &= h_2(0) = -C(A^{-1})^2(A^{-1}B) \\ \hat{h}_3(0) &= h_3(0) = -C(A^{-1})^3(A^{-1}B) \\ &\vdots \\ \hat{h}_n(0) &= h_n(0) = -C(A^{-1})^n(A^{-1}B). \end{aligned}$$

Pero resulta que este método no es muy confiable debido a las inestabilidades numéricas que se presentan durante el cálculo de $\hat{h}_n(0) = h_n(0)$ a medida que n crece.

Una manera de superar este problema, es emplear Acoplamiento de Momentos Implícito, un procedimiento que usa técnicas basadas en los subespacios de Krylov[21] para imponer condiciones en el aproximante $\hat{H}(s)$ sin tener que calcular explícitamente $\hat{h}_j(0)$, evitando las inestabilidades anteriormente referidas. Antes de entrar de lleno a esta técnica de reducción, definamos primeramente la proyección del espacio-estado de un sistema Σ , que también es conocida como una técnica de reducción de orden.

3.3.2. Proyección del estado-espacio

Sean V y W dos matrices en $\mathbb{R}^{N \times n}$ tales que

$$\text{colspan}\{V\} = \mathcal{V} \subset \mathbb{R}^N \quad (3.18)$$

$$\text{colspan}\{W\} = \mathcal{W} \subset \mathbb{R}^N. \quad (3.19)$$

Sea $Vz(t)$ una aproximación de $x(t)$ en \mathbb{R}^N , donde $z(t) \in \mathbb{R}^n$. Suponiendo que V y W son ambas de rango completo, tenemos que

$$x(t) \cong Vz(t)$$

$$\dot{x}(t) \cong V\dot{z}(t)$$

y substituyendo lo anterior en (3.1) da como resultado

$$V\dot{z}(t) \cong AVz(t) + Bu(t) \quad (3.20)$$

$$y(t) \cong CVz(t). \quad (3.21)$$

De la ecuación (3.20) el residual, res se define de la siguiente manera

$$res = V\dot{z}(t) - AVz(t) - Bu(t) \quad (3.22)$$

Si existe una base $\{w_1, w_2, w_3, \dots, w_n\}$ para el subespacio \mathcal{W} tal que la matriz W se pueda escribir como $W \equiv [w_1|w_2|w_3|\dots|w_n]$, y además, si forzamos a (3.22) a ser ortogonal a cada $\{w_1, w_2, w_3, \dots, w_n\}$, entonces como $Vz(t)$ aproxima $x(t)$, se debe pedir

$$res \perp w_i, \quad i = 1, 2, 3, \dots, n$$

o en notación de producto interno

$$\langle res, w_i \rangle = 0, \quad i = 1, 2, 3, \dots, n$$

$$w_i^T \cdot res = 0, \quad i = 1, 2, 3, \dots, n.$$

Por lo tanto

$$W^T \cdot res = 0$$

$$W^T[V\dot{z}(t) - AVz(t) - Bu(t)] = 0$$

$$W^TV\dot{z}(t) - W^TAVz(t) - W^TBu(t) = 0$$

$$W^TV\dot{z}(t) = W^TAVz(t) + W^TBu(t). \quad (3.23)$$

Si $W^T V = E$ en la ecuación (3.23) y junto con la ecuación (3.21), tenemos el siguiente sistema dinámico continuo LIT, algebraicamente similar

$$\begin{aligned} E\dot{z}(t) &= \hat{A}z(t) + \hat{B}u(t) \\ \hat{y}(t) &= \hat{C}z(t) \end{aligned} \quad (3.24)$$

donde $\hat{A} = W^T A V \in \mathbb{R}^{n \times n}$, $\hat{B} = W^T B \in \mathbb{R}^{n \times m}$, $\hat{C} = C V \in \mathbb{R}^{p \times n}$, y $n < N$. Usualmente, en la practica $W^T V = I$ por lo cual (3.23) tiene una forma mucho más simple. Al sistema reducido (3.24) se le conoce como la proyección en el espacio-estado del sistema (3.1). Nótese que $z(t)$ se identifica con $\hat{x}(t)$.

Ahora que hemos desarrollado la proyección en el espacio-estado (3.24), podemos proceder a la siguiente técnica de reducción.

3.3.3. Acoplamiento de momentos implícito

Regresemos por un momento a la ecuación (2.38)

$$H(s) = -C(A^{-1}B) - C(A^{-1})(A^{-1}B)s - C(A^{-1})^2(A^{-1}B)s^2 - \dots$$

donde el conjunto

$$\{(A^{-1}B), (A^{-1})(A^{-1}B), (A^{-1})^2(A^{-1}B), \dots\}$$

tiene la propiedad de que la expansión de sus columnas forma el siguiente subespacio de Krylov

$$\mathcal{K}\{(A^{-1}), (A^{-1}B)\} = \text{colspan}\{(A^{-1}B), (A^{-1})(A^{-1}B), (A^{-1})^2(A^{-1}B), \dots\}.$$

Una relación particular entre la matriz V usada en (3.24), el anterior subespacio de Krylov $\mathcal{K}\{(A^{-1}), (A^{-1}B)\}$, y el número posible de momentos a acoplar entre $H(s)$ y $\hat{H}(s)$, se explica en el siguiente teorema.

Teorema 3.3.1.[22]. *Si las columnas de la matriz V , usada en (3.24), forman una base del subespacio de Krylov $\mathcal{K}\{(A^{-1}), (A^{-1}B)\}$ con rango q (donde q es un múltiplo de m) y la matriz W es tal que la matriz \hat{A} es no-singular, entonces los primeros $\frac{q}{m}$ momentos del sistema original y del reducido coinciden. Se supone que A es invertible.*

Antes de proceder con la demostración del teorema, conviene mencionar que es práctica común escoger $W = V$ cuando se toma en consideración este subespacio de Krylov.

Demostración: Sea $\hat{H}(s) = \hat{C}(sI_n - \hat{A})^{-1}\hat{B}$, la función de transferencia del sistema reducido con expansión en series de Taylor alrededor de cero

$$\hat{H}(s) = - \sum_{k=0}^{\infty} \underbrace{\hat{C}(\hat{A}^{-1})^k(\hat{A}^{-1}\hat{B})}_{\hat{h}_k(0)} s^k, \quad (3.25)$$

donde $\hat{h}_k(0)$ es el k-ésimo momento de $\hat{H}(s)$ alrededor de cero. Por simplicidad, usaremos la notación h_k y \hat{h}_k para substituir a $h_k(0)$ y $\hat{h}_k(0)$. Usando inducción

i) Si $k = 0$, entonces

$$\begin{aligned} h_0 &= C(A^{-1})^0(A^{-1}B) \\ h_0 &= CA^{-1}B. \end{aligned}$$

por otro lado

$$\begin{aligned} \hat{h}_0 &= \hat{C}(\hat{A}^{-1})^0(\hat{A}^{-1}\hat{B}) \\ \hat{h}_0 &= \hat{C}(\hat{A}^{-1}\hat{B}) \\ \hat{h}_0 &= CV(W^T AV)^{-1}(W^T B) \\ \hat{h}_0 &= CV(W^T AV)^{-1}(W^T AA^{-1}B) \end{aligned}$$

pero $A^{-1}B \in \text{colspan}\{V\} \therefore \exists r_0 \in \mathbb{R}^{q \times m} : Vr_0 = A^{-1}B$, por lo tanto

$$\begin{aligned} \hat{h}_0 &= CV(W^T AV)^{-1}(W^T AV r_0) \\ \hat{h}_0 &= CV \underbrace{(W^T AV)^{-1}(W^T AV)}_I r_0 \\ \hat{h}_0 &= CV r_0 \\ \hat{h}_0 &= CA^{-1}B \\ \hat{h}_0 &= h_0. \end{aligned}$$

ii) En seguida, si suponemos que $h_k = \hat{h}_k$ cuando $k = \frac{q}{m} - 1$, esto significa que

$$\begin{aligned} C(A^{-1})^k(A^{-1}B) &= \hat{C}(\hat{A}^{-1})^k(\hat{A}^{-1}\hat{B}) \\ &= CV[(W^T AV)^{-1}]^k (W^T AV)^{-1}(W^T B) \\ &= CV[(W^T AV)^{-1}]^{k+1}(W^T B). \end{aligned}$$

y, por lo tanto

$$(A^{-1})^k(A^{-1}B) = V[(W^T AV)^{-1}]^{k+1}(W^T B)$$

que será nuestra hipótesis de inducción.

iii) Finalmente, tenemos que probar que $h_{k+1} = \hat{h}_{k+1}$ cuando $k + 1 = \frac{q}{m}$

$$\begin{aligned} C(A^{-1})^{k+1}(A^{-1}B) &= \hat{C}(\hat{A}^{-1})^{k+1}(\hat{A}^{-1}\hat{B}) \\ &= CV[(W^T AV)^{-1}]^{k+1}(W^T AV)^{-1}(W^T B) \\ &= CV(W^T AV)^{-1}[(W^T AV)^{-1}]^{k+1}(W^T B) \\ &= CV(W^T AV)^{-1}W^T \underbrace{V[(W^T AV)^{-1}]^{k+1}(W^T B)}_{(A^{-1})^k(A^{-1}B)} \\ &= CV(W^T AV)^{-1}W^T(A^{-1})^k(A^{-1}B) \\ &= CV(W^T AV)^{-1}W^T AA^{-1}(A^{-1})^k(A^{-1}B) \\ &= CV(W^T AV)^{-1}W^T A(A^{-1})^{k+1}(A^{-1}B) \end{aligned}$$

pero $(A^{-1})^{k+1}(A^{-1}B) \in \text{colspan}\{V\} \therefore \exists r_{k+1} \in \mathbb{R}^{q \times m} : Vr_{k+1} = (A^{-1})^{k+1}(A^{-1}B)$
por lo tanto

$$\begin{aligned} C(A^{-1})^{k+1}(A^{-1}B) &= CV(W^T AV)^{-1}W^T AVr_{k+1} \\ C(A^{-1})^{k+1}(A^{-1}B) &= CVr_{k+1} \\ C(A^{-1})^{k+1}(A^{-1}B) &= C(A^{-1})^{k+1}(A^{-1}B) \\ C(A^{-1})^{\frac{q}{m}}(A^{-1}B) &= C(A^{-1})^{\frac{q}{m}}(A^{-1}B) \end{aligned}$$

□

Para el momento $h_k = C(A^{-1})^k(A^{-1}B)$, podemos desarrollar una expresión equivalente

$$\begin{aligned} h_k &= C(A^{-1})^k(A^{-1}B) \\ h_k &= C(A^{-1})(A^{-1})^{k-1}(A^{-1}B) \\ h_k &= C(A^{-1})(A^{-1})^k B \\ h_k &= \left\{ [C(A^{-1})(A^{-1})^k]^T \right\}^T B \\ h_k &= [(A^{-T})^k (A^{-T} C^T)]^T B \end{aligned} \tag{3.26}$$

substituyendo (3.26) en (2.38) da como resultado

$$H(s) = \sum_{k=0}^{\infty} -[(A^{-T})^k (A^{-T} C^T)]^T B s^k \quad (3.27)$$

$$H(s) = - [(A^{-T} C^T)]^T B - [(A^{-T})(A^{-T} C^T)]^T B s - [(A^{-T})^2 (A^{-T} C^T)]^T B s^2 - \dots \quad (3.28)$$

donde el conjunto

$$\{(A^{-T} C^T), (A^{-T})(A^{-T} C^T), (A^{-T})^2 (A^{-T} C^T), \dots\}$$

tiene la propiedad de que la expansión de sus columnas forma el siguiente subespacio de Krylov

$$\mathcal{K}\{(A^{-T}), (A^{-T} C^T)\} = \text{colspan}\{(A^{-T} C^T), (A^{-T})(A^{-T} C^T), (A^{-T})^2 (A^{-T} C^T), \dots\}.$$

Resulta que, además de usar V como la base del subespacio de Krylov $\mathcal{K}\{(A^{-1}), (A^{-1} B)\}$, si seleccionamos a W en (3.24) como la base del subespacio de Krylov $\mathcal{K}\{(A^{-T}), (A^{-T} C^T)\}$ de tal manera que $W^T V = I$ (biortogonales), entonces ahora un total de $\frac{q}{m} + \frac{q}{p}$ momentos de las funciones de transferencia $H(s)$ y $\hat{H}(s)$ coinciden. El siguiente Teorema detalla esta circunstancia.

Teorema 3.3.2.[22]. *Si las columnas de las matrices V y W usadas en (3.24), son las bases para los subespacios de Krylov $\mathcal{K}\{(A^{-1}), (A^{-1} B)\}$ y $\mathcal{K}\{(A^{-T}), (A^{-T} C^T)\}$ respectivamente, ambas con rango q (donde q es múltiplo de m y p), entonces los primeros $\frac{q}{m} + \frac{q}{p}$ momentos del sistema original y reducido coinciden. Se supone que A y \hat{A} son invertibles.*

Demostración: Sea h_k como en (3.26), similarmente \hat{h}_k tendrá la misma forma pero con matrices \hat{A} , \hat{B} , \hat{C} . De nueva cuenta, se usara inducción en la demostración.

i) Si $k = 0$, entonces

$$h_0 = [(A^{-T})^0 (A^{-T} C^T)]^T B$$

$$h_0 = (A^{-T} C^T)^T B.$$

Por otro lado

$$\hat{h}_0 = [(\hat{A}^{-T})^0 (\hat{A}^{-T} \hat{C}^T)]^T \hat{B}$$

$$\hat{h}_0 = [(\hat{A}^{-T} \hat{C}^T)]^T \hat{B}$$

$$\hat{h}_0 = [((W^T AV)^{-T} (CV)^T)]^T W^T B$$

$$\hat{h}_0 = [((W^T AV)^{-T} V^T C^T)]^T W^T B$$

$$\hat{h}_0 = [((W^T AV)^{-T} V^T A^T A^{-T} C^T)]^T W^T B,$$

pero $A^{-T} C^T \in \text{colspan}\{W\} : \exists l_0 \in \mathbb{R}^{q \times p} : W l_0 = A^{-T} C^T$, por lo tanto

$$\hat{h}_0 = [(W^T AV)^{-T} V^T A^T W l_0]^T W^T B$$

$$\hat{h}_0 = [(W^T AV)^{-T} (V^T A^T W) l_0]^T W^T B$$

$$\hat{h}_0 = l_0^T W^T B$$

$$\hat{h}_0 = (W l_0)^T B$$

$$\hat{h}_0 = (A^{-T} C^T)^T B$$

$$\hat{h}_0 = h_0.$$

ii) En seguida, si suponemos que $h_k = \hat{h}_k$ cuando $k = (\frac{q}{p} + \frac{q}{m}) - 1$, significa que

$$[(A^{-T})^k (A^{-T} C^T)]^T B = [(A^{-T})^k (\hat{A}^{-T} \hat{C}^T)]^T \hat{B}$$

$$= [((W^T AV)^{-T})^k (W^T AV)^{-T} (CV)^T]^T W^T B.$$

Podemos ver que

$$[(A^{-T})^k (A^{-T} C^T)]^T = [((W^T AV)^{-T})^k (W^T AV)^{-T} (CV)^T]^T W^T$$

$$= (CV) (W^T AV)^{-1} ((W^T AV)^{-1})^k W^T,$$

la cual será nuestra hipótesis de inducción.

iii) Finalmente, debemos probar que $h_{k+1} = \hat{h}_{k+1}$ cuando $k + 1 = \frac{q}{p} + \frac{q}{m}$

$$\begin{aligned}
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= [(\hat{A}^{-T})^{(k+1)}(\hat{A}^{-T}\hat{C}^T)]^T \hat{B} \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= [((W^T AV)^{-T})^{(k+1)}(W^T AV)^{-T}(CV)^T]^T W^T B \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= [((W^T AV)^{-T})^{(k+1)}(W^T AV)^{-T}(CV)^T]^T W^T B \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (CV)(W^T AV)^{-1}((W^T AV)^{-1})^{k+1} W^T B \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (CV)(W^T AV)^{-1}((W^T AV)^{-1})^{k+1} W^T \underbrace{A^{-1}B}_{Vr_0} \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (CV)(W^T AV)^{-1}((W^T AV)^{-1})^{k+1} W^T AV r_0 \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (CV)((W^T AV)^{-1})^{k+1} (W^T AV)^{-1} W^T AV r_0 \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (CV)((W^T AV)^{-1})^{k+1} r_0 \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (CV)(W^T AV)^{-1}((W^T AV)^{-1})^k r_0 \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= \underbrace{(CV)(W^T AV)^{-1}((W^T AV)^{-1})^k}_{[(A^{-T})^k(A^{-T}C^T)]^T} W^T V r_0 \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= [(A^{-T})^k(A^{-T}C^T)]^T V r_0 \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (A^{-T}C^T)^T (A^{-1})^k (A^{-1}B) \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= (A^{-T}C^T)^T (A^{-1})^{k+1} B \\
[(A^{-T})^{k+1}(A^{-T}C^T)]^T B &= [(A^{-T})^{k+1}(A^{-T}C^T)]^T B \\
[(A^{-T})^{\left(\frac{q}{p} + \frac{q}{m}\right)}(A^{-T}C^T)]^T B &= [(A^{-T})^{\left(\frac{q}{p} + \frac{q}{m}\right)}(A^{-T}C^T)]^T B.
\end{aligned}$$

□

Como los elementos en ambos subespacios de Krylov $\mathcal{K}_{q_1}\{(A^{-1}), (A^{-1}B)\}$ y $\mathcal{K}_{q_2}\{(A^{-*}), (A^{-*}C^*)\}$ son matrices, se emplea una generalización del algoritmo de Arnoldi para vectores, llamado algoritmo de Arnoldi para bloques[21], para construir V y W .

Arnoldi para bloques, en efecto construye V y W para cada subespacio de Krylov, sólo que para alcanzar la biortogonalidad se requiere de la siguiente operación

$$W = W(W^T V)^{-T} \quad (3.29)$$

y se confirma la biortogonalidad.

ALGORITHM Block Arnoldi

1. Choose a unitary matrix V_1 of size $n \times p$
 2. For $j = 1, 2, \dots, m$ Do:
 3. Compute $W_j := AV_j$
 4. For $i = 1, 2, \dots, j$ do:
 5. $H_{ij} := V_i^T W_j$
 6. $W_j := W_j - V_i H_{ij}$
 7. EndDo
 8. Compute the Q-R decomposition $W_j = V_{j+1} H_{j+1,j}$
 9. EndDo
-

Figura 3: El algoritmo de Arnoldi para bloques[21].

3.3.4. Interpolación de Hermite

Resulta que la técnica anterior puede ser extendida para acoplar momentos de $H(s)$ y de $\hat{H}(s)$ no solamente alrededor de cero, sino en una multitud de puntos $\sigma_j \neq 0$, $j = 1, 2, 3, \dots, n$. Para hacer esto, debemos considerar nuevamente la ecuación (2.49) con una pequeña modificación

$$H(s) = - \sum_{k=0}^{\infty} C(\tilde{A}_j^{-1})^k (\tilde{A}_j^{-1} B)(s - \sigma_j)^k \quad j = 1, 2, 3, \dots, n. \quad (3.30)$$

donde $\tilde{A}_j = A - \sigma_j I_N$. Para cada $j = 1, 2, 3, \dots, n$., la ecuación (3.28) da surgimiento a cada uno de los siguientes subespacios de Krylov

$$\mathcal{K}\{(\tilde{A}_j^{-1}), (\tilde{A}_j^{-1} B)\} = \text{colspan}\{(\tilde{A}_j^{-1} B), (\tilde{A}_j^{-1})(\tilde{A}_j^{-1} B), (\tilde{A}_j^{-1})^2(\tilde{A}_j^{-1} B), \dots\}$$

o

$$\mathcal{K}\{(\tilde{A}_j^{-T}), (\tilde{A}_j^{-T} C^T)\} = \text{colspan}\{(\tilde{A}_j^{-T} C^T), (\tilde{A}_j^{-T})(\tilde{A}_j^{-T} C^T), (\tilde{A}_j^{-T})^2(\tilde{A}_j^{-T} C^T), \dots\}.$$

Si las matrices V y W usadas en (3.24) son tales que

$$\text{colspan}\{V\} \supseteq \bigcup_{j=1}^n \mathcal{K}\{(\tilde{A}_j^{-1}), (\tilde{A}_j^{-1} B)\} \quad (3.31)$$

$$\text{colspan}\{W\} \supseteq \bigcup_{j=1}^n \mathcal{K}\{(\tilde{A}_j^{-T}), (\tilde{A}_j^{-T} C^T)\} \quad (3.32)$$

entonces el sistema reducido mediante proyección del estado-espacio cuya función de transferencia es $\hat{H}(s)$, interpola a $H(s)$ y a un cierto número μ_j de sus derivadas en todos los puntos $\{\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n\}$, esto es

$$\begin{aligned} H(\sigma_j) &= \hat{H}(\sigma_j), & \text{for } j = 1, 2, 3, \dots, n. \\ \frac{d^l H}{ds^l}(\sigma_j) &= \frac{d^l \hat{H}}{ds^l}(\sigma_j), & \text{for } l = 1, 2, 3, \dots, \mu_k - 1. \end{aligned} \quad (3.33)$$

Esta extensión de la técnica de Acoplamiento de momentos implícito, es conocida como interpolación racional de Hermite, también conocida como Acoplamiento de momentos multipuntuales ó Aproximación multipuntual de Padé[26].

3.3.5. Interpolación Tangencial de Hermite

Para sistemas MIMO, el acoplamiento de momentos es una condición muy complicada de cumplir. Por lo tanto, Vanderdorpe et al.[23] propuso una generalización llamada interpolación tangencial de Hermite.

Esta técnica establece que además de la multitud de puntos de interpolación $\{\sigma_k\}$, $k = 1, 2, 3, \dots, n$, también es necesario contar con adecuadas direcciones tangenciales izquierdas $\{l_k \in \mathbb{R}^{1 \times p}\}$, y derechas $\{r_k \in \mathbb{R}^{m \times 1}\}$, para que el sistema reducido $\hat{H}(s)$ aproxime al original $H(s)$ de mejor manera en los puntos de interpolación $\{\sigma_k\}$, i.e.

$$\begin{aligned} H(\sigma_k)r_k &= \hat{H}(\sigma_k)r_k, \\ l_k H(\sigma_k) &= l_k \hat{H}(\sigma_k), \\ l_k H'(\sigma_k)r_k &= l_k \hat{H}'(\sigma_k)r_k \end{aligned} \quad (3.34)$$

donde las condiciones (3.34) son más débiles que las de acoplamiento de momentos. Los siguientes dos Lemas[4] proporcionan información acerca de las matrices V y W usadas en (3.24)

Lema. Sean $V \in \mathbb{R}^{N \times n}$ y $W \in \mathbb{R}^{N \times n}$ dos matrices de rango completo n de tal modo que $W^T V = I$. Sean $\sigma_k \in \mathbb{C}$, $l_k \in \mathbb{R}^{1 \times p}$ y $r_k \in \mathbb{R}^{m \times 1}$ para $k = 1, 2, 3, \dots, n$ el conjunto de puntos de interpolación y las direcciones tangenciales izquierda y derecha respectivamente. Suponer que todos los puntos σ_k son seleccionados tales que las matrices $(\sigma_k I_N - A)$ son invertibles. Si para todo $k \in 1, 2, 3, \dots, n$

$$\begin{aligned} (\sigma_k I_N - A)^{-1} B r_k &\in \text{columnspan}(V) \\ (\sigma_k^* I_N - A^T)^{-1} C^T l_k^T &\in \text{columnspan}(W) \end{aligned} \quad (3.35)$$

entonces el sistema reducido $\hat{\Sigma} = (\hat{A}, \hat{B}, \hat{C}) = (W_n^T AV_n, W_n^T B, CV_n)$ satisface las condiciones (3.34).

Dada una selección de puntos de interpolación y direcciones tangenciales, las matrices de proyección V_n y W_n que satisfacen (3.35) se pueden obtener mediante la solución de las ecuaciones de Sylvester.

Lema. Sean α_k, r_k y l_k para $k = 1, 2, 3, \dots, n$ puntos de interpolación, direcciones tangenciales derecha e izquierda, respectivamente. Sean W y V las soluciones de las ecuaciones de Sylvester

$$\begin{aligned} AV_n - V_n\Lambda &= -B[r_1 | \dots | r_n] \\ A^T W_n - W_n\Lambda^T &= -C^T[l_1^T | \dots | l_n^T] \end{aligned} \quad (3.36)$$

donde $\Lambda = \text{diag}(\sigma_1, \dots, \sigma_n)$, y asignando $W \leftarrow W(W^T V = I)^{-T}$ (para obtener $W^T V = I$). Entonces el sistema continuo reducido LIT

$$\hat{\Sigma} = (\hat{A}, \hat{B}, \hat{C}) = (W^T AV, W^T B, CV)$$

satisface las condiciones (3.34).

La prueba de estos Lemas, así como más teoremas que vinculan a las ecuaciones de Sylvester con los métodos de interpolación de Hermite e interpolación racional de Hermite, pueden encontrarse por ejemplo en [22,23] y [24].

4. Reducción óptima basada en la norma $\|\bullet\|_{\mathcal{H}_2}$

Dedicaremos este capítulo en desarrollar las condiciones necesarias para minimizar la norma \mathcal{H}_2 de la diferencia entre las funciones de transferencia $H(s)$ y $\hat{H}(s)$.

En la sección 3.1, propusimos la métrica $d(H(s), \hat{H}(s))$ que mide la distancia entre ambas funciones de transferencia, de tal manera que el problema aquí es minimizar

$$d(H(s), \hat{H}(s)) = \|G(s)\|_{\mathcal{H}_2},$$

donde $G(s) = H(s) - \hat{H}(s)$, y la forma explícita de la norma es

$$\|G(s)\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[G(i\omega)]^*[G(i\omega)]d\omega \right)^{\frac{1}{2}}. \quad (4.1)$$

El problema de minimizar la norma \mathcal{H}_2 es uno de los problemas más antiguos dentro del ámbito de reducción de orden de modelos dinámicos. Este problema tiene alrededor de 40 años, y tiene aspectos que aún no han sido resueltos; por ejemplo, no se ha resuelto el problema de la existencia de un óptimo global.

En lugar de encontrar el óptimo global, la mayoría de los métodos de reducción se enfocan en satisfacer solamente las condiciones de primer orden para (4.1) con respecto al conjunto $\{\{\sigma_k\}, \{l_k\}, \{r_k\}\}$, $k = 1, 2, 3, \dots, n.$, esto es, resolver simultáneamente el conjunto de ecuaciones

$$\begin{aligned} \frac{\partial}{\partial \sigma_k} \|G(s)\|_{\mathcal{H}_2} &= 0 \\ \frac{\partial}{\partial l_k} \|G(s)\|_{\mathcal{H}_2} &= 0 \\ \frac{\partial}{\partial r_k} \|G(s)\|_{\mathcal{H}_2} &= 0. \end{aligned} \quad (4.2)$$

En [4], se introduce un algoritmo llamado MIRIAM (MIMO Iterative Rational Interpolation Algorithm). Si este algoritmo converge, es decir, si el algoritmo encuentra el conjunto óptimo $\{\{\sigma_k^{opt}\}, \{l_k^{opt}\}, \{r_k^{opt}\}\}$, $k = 1, 2, 3, \dots, n.$, entonces proporciona un sistema reducido $\hat{\Sigma}$ que satisface el anterior conjunto de ecuaciones.

4.1. La norma $\|\bullet\|_{\mathcal{H}_2}$

En esta sección, se desarrolla una nueva expresión para la norma \mathcal{H}_2 . Antes de proceder, necesitaremos algunos resultados del análisis complejo.

Teorema 4.1.1. Teorema del residuo de Cauchy. Sea D un dominio simplemente conexo, y sea C un contorno simple positivamente orientado que yace dentro de D . Si $F(s)$ es analítica dentro de C y sobre C , excepto en los puntos $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_N$ que yacen dentro de C entonces

$$\int_C F(s)ds = 2\pi i \sum_{k=1}^N Res[F(s), s = \lambda_k], \quad k = 1, 2, 3, \dots, N. \quad (4.3)$$

Demostración: La demostración de este teorema se puede encontrar en [26], ó en otro libro estándar de análisis complejo.

Una consecuencia muy conocida del anterior teorema es

$$\frac{1}{2\pi i} \int_{-i\infty}^{i\infty} F(s)ds = \sum_{k=1}^N Res[F(s), s = \lambda_k], \quad k = 1, 2, 3, \dots, N. \quad (4.4)$$

Ahora podemos proceder con el desarrollo de la nueva forma de la norma \mathcal{H}_2 . Comencemos con el siguiente teorema para sistemas SISO.

Teorema 4.1.2.[4]. Sea $H(s)$ la función de transferencia de un sistema SISO continuo LIT cuyos polos $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_N$ son distintos por pares y pertenecen a $\mathbb{C}^- = \{s \in \mathbb{C} : Re(s) < 0\}$. Sean $\phi_k = Res[H(s), s = \lambda_k]$, $k = 1, 2, 3, \dots, N$ los correspondientes residuos. La norma \mathcal{H}_2 de $H(s)$ está dada por

$$\|H(s)\|_{\mathcal{H}_2}^2 = \sum_{k=1}^N \phi_k^* H(-\lambda_k^*). \quad (4.5)$$

Demostración: Para sistemas SISO, $H(s)$ es una función racional, i.e.

$$H(s) = \sum_{k=1}^N \frac{\phi_k}{s - \lambda_k} \quad (4.6)$$

donde $\phi_k = \text{Res}[H(s), s = \lambda_k]$, $k = 1, 2, 3, \dots, N$, son los residuos, y λ_k son los polos de $H(s)$ respectivamente. Entonces

$$[H(s)]^* = \left[\sum_{k=1}^N \frac{\phi_k}{s - \lambda_k} \right]^* = \left[\sum_{k=1}^N \frac{\phi_k^*}{(s - \lambda_k)^*} \right]$$

de tal manera que la norma de $H(s)$

$$\|H(s)\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} [H(i\omega)]^* [H(i\omega)] d\omega \quad (4.7)$$

$$= \frac{1}{2\pi i} \int_{-i\infty}^{i\infty} [H(i\omega)]^* [H(i\omega)] d\omega \quad (4.8)$$

pero como el producto $[H(s)]^* [H(s)]$ tiene los mismos polos que $H(s)$, usando (4.4), la ecuación (4.8) se convierte en

$$\begin{aligned} \|H(s)\|_{\mathcal{H}_2}^2 &= \sum_{k=1}^N \text{Res} \{ [H(s)]^* [H(s)], s = \lambda_k \} \\ &= \sum_{k=1}^N \text{Res}[H(s), s = \lambda_k] \cdot \lim_{s \rightarrow \lambda_k} [H(s)]^* \\ &= \sum_{k=1}^N \text{Res}[H(s), s = \lambda_k] \cdot \lim_{s \rightarrow \lambda_k} \sum_{k=1}^N \frac{\phi_k^*}{(s - \lambda_k)^*} \\ &= \sum_{k=1}^N \text{Res}[H(s), s = \lambda_k] \cdot \lim_{s \rightarrow \lambda_k} \sum_{k=1}^N \frac{\phi_k^*}{-s - \lambda_k^*} \\ &= \sum_{k=1}^N \text{Res}[H(s), s = \lambda_k] \cdot \sum_{k=1}^N \frac{\phi_k^*}{-\lambda_k^* - \lambda_k} \\ &= \sum_{k=1}^N \phi_k \cdot [H(-\lambda_k^*)]^* \end{aligned}$$

además, como $\|\bullet\|_{\mathcal{H}_2}^2$ es una función con valores reales

$$\|H(s)\|_{\mathcal{H}_2}^2 = \sum_{k=1}^N \phi_k^* H(-\lambda_k^*)$$

□

Ahora estamos en condiciones de generalizar el teorema anterior para el caso de sistemas MIMO.

Teorema 4.1.3.[4]. Sea $H(s)$ la función de transferencia de un sistema continuo en el tiempo MIMO (2.10). Entonces la norma \mathcal{H}_2 de $H(s)$ está dada por

$$\|H(s)\|_2^2 = \text{traza} \left\{ \sum_{k=1}^N H(-\lambda_k^*) b_k^T c_k^T \right\} \quad (4.9)$$

donde $b_k = [b_{k1}, \dots, b_{km}]$ es el k -ésimo vector renglón de $B = [b_1^T, \dots, b_N^T]$, y $c_k = [c_{1k}, \dots, c_{pk}]^T$ es el k -ésimo vector columna de $C = [c_1, \dots, c_N]$.

Demostación: Sea $H(s) = [H_{ql}]$ para $q = 1, 2, 3, \dots, p$, y $l = 1, 2, 3, \dots, m$, entonces

$$P(s) = [H(s)]^* [H(s)] = \begin{bmatrix} H_{11}^* & H_{21}^* & \cdots & H_{p1}^* \\ H_{12}^* & H_{22}^* & \cdots & H_{p2}^* \\ \vdots & \vdots & \ddots & \vdots \\ H_{1m}^* & H_{2m}^* & \cdots & H_{pm}^* \end{bmatrix} \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1m} \\ H_{21} & H_{22} & \cdots & H_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ H_{p1} & H_{p2} & \cdots & H_{pm} \end{bmatrix}.$$

Nuestro interés aquí, se centra en la diagonal principal de $P(s)$

$$P(s) = \begin{bmatrix} \sum_{q=1}^p H_{q1}^* H_{q1} & & & & \\ & \vdots & & & \\ & & \sum_{q=1}^p H_{q2}^* H_{q2} & & \\ & & & \ddots & \vdots \\ & & & & \sum_{q=1}^p H_{qm}^* H_{qm} \end{bmatrix}$$

de tal manera que la *traza* de $P(s)$ es

$$\begin{aligned} \text{traza}\{P(s)\} &= \sum_{q=1}^p H_{q1}^* H_{q1} + \sum_{q=1}^p H_{q2}^* H_{q2} + \cdots + \sum_{q=1}^p H_{qm}^* H_{qm} \\ \text{traza}\{P(s)\} &= \sum_{l=1}^m \sum_{q=1}^p H_{ql}^* H_{ql} \end{aligned}$$

y la norma \mathcal{H}_2 de $H(s)$ será

$$\begin{aligned}
\|H(s)\|_{\mathcal{H}_2}^2 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[H(s)]^*[H(s)]d\omega \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}\{P(s)\}d\omega \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} \sum_{l=1}^m \sum_{q=1}^p H_{ql}^* H_{ql} d\omega \\
&= \sum_{l=1}^m \sum_{q=1}^p \cdot \frac{1}{2\pi} \int_{-\infty}^{\infty} H_{ql}^* H_{ql} d\omega \\
&= \sum_{l=1}^m \sum_{q=1}^p \|H_{ql}(s)\|_{\mathcal{H}_2}^2
\end{aligned} \tag{4.10}$$

donde $H_{ql}(s)$ es la función de transferencia entre la l-ésima entrada y la q-ésima salida. Ahora usando (4.5) en (4.10)

$$\|H(s)\|_{\mathcal{H}_2}^2 = \sum_{l=1}^m \sum_{q=1}^p \sum_{k=1}^N \phi_{k_{ql}}^* H_{ql}(-\lambda_k^*)$$

pero $\phi_{k_{ql}} = c_{qk} b_{kl}$, por lo que tenemos $\phi_{k_{ql}}^* = (c_{qk} b_{kl})^* = b_{kl}^* c_{qk}^*$

$$\begin{aligned}
\|H(s)\|_{\mathcal{H}_2}^2 &= \sum_{k=1}^N \sum_{l=1}^m \sum_{q=1}^p \underbrace{b_{kl}^* c_{qk}^* H_{ql}(-\lambda_k^*)}_{\text{traza}\{H(-\lambda_k^*) b_k^* c_k^*\}} \\
&= \sum_{k=1}^N \text{traza}\{H(-\lambda_k^*) b_k^* c_k^*\} \\
&= \text{traza}\left\{ \sum_{k=1}^N H(-\lambda_k^*) b_k^* c_k^* \right\}
\end{aligned}$$

□

Ahora podemos desarrollar completamente la forma de la métrica en términos de la norma \mathcal{H}_2 . Para simplificar, nos referiremos a $d(H(s), \hat{H}(s)) = \|G(s)\|_{\mathcal{H}_2}$ simplemente como d . También vale la pena mencionar que la matriz $H(s)$ tiene relacionados los polos $\{\lambda_1, \dots, \lambda_N\}$ y los residuos $\{\phi_1, \dots, \phi_N\}$, y la matriz $\hat{H}(s)$ tiene relacionados los polos $\{\hat{\lambda}_1, \dots, \hat{\lambda}_n\}$ y los residuos $\{\hat{\phi}_1, \dots, \hat{\phi}_n\}$

$$\begin{aligned}
d^2 &= \|G(s)\|_{\mathcal{H}_2}^2 \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traza}[G(i\omega)]^* [G(i\omega)] d\omega \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} [H(i\omega) - \hat{H}(i\omega)]^* [H(i\omega) - \hat{H}(i\omega)] d\omega \\
&= \text{traza} \left\{ \sum_{k=1}^N [H(-\lambda_k^*) - \hat{H}(-\lambda_k^*)] b_k^T c_k^T + \sum_{k=1}^n [H(-\hat{\lambda}_k^*) - \hat{H}(-\hat{\lambda}_k^*)] \hat{b}_k^T \hat{c}_k^T \right\} \quad (4.11)
\end{aligned}$$

4.2. Interpolación óptima en $\|\bullet\|_{\mathcal{H}_2}$

En esta sección estableceremos matemáticamente el problema de minimizar la distancia entre las funciones de transferencia $H(s)$ y $\hat{H}(s)$ usando (4.11). Veamos la definición formal del problema.

Definición del Problema. Dado el sistema dinámico LIT asintóticamente estable (3.1), construir un sistema dinámico LIT asintóticamente estable (3.3) de dimensión fija n que minimice la norma \mathcal{H}_2 de la diferencia entre ellos, i.e.

$$\min d^2 = \|G(s)\|_{\mathcal{H}_2}^2 \quad (4.12)$$

con respecto a la variable de optimización $\tilde{v} = [Re(v^T), Im(v^T)]^T \in \mathbb{R}^{2(m+p+1)n}$, donde

$$v = [\hat{\lambda}_1, \dots, \hat{\lambda}_n, \hat{b}_1, \dots, \hat{b}_n, \hat{c}_1^T, \dots, \hat{c}_n^T]^T \quad (4.13)$$

con los vectores renglón $\hat{b}_k = [\hat{b}_{k1}, \dots, \hat{b}_{km}]$ y los vectores columna $\hat{c}_k = [\hat{c}_{1k}, \dots, \hat{c}_{pk}]^T$ para $k = 1, 2, 3, \dots, n$.

4.3. Condiciones necesarias de primer orden

Basados en (4.11) podemos señalar las condiciones necesarias de primer orden para que d^2 sea mínima. El siguiente teorema señala tales condiciones.

Teorema 4.3.1. Considérese el sistema dinámico original de la forma (3.1) con función de transferencia $H(s)$. Sea $\hat{H}(s)$ la función de transferencia del sistema reducido (3.3) dada en una base vectorial propia tal que $\hat{A} = \text{diag}(\hat{\lambda}_1, \dots, \hat{\lambda}_n)$, $\hat{B} = [\hat{b}_1^T, \dots, \hat{b}_n^T]^T$ y $\hat{C} = [\hat{c}_1, \dots, \hat{c}_n]$. Si $\hat{H}(s)$ resuelve el problema de optimización (4.12), entonces se satisfacen las siguientes condiciones

$$\begin{aligned} \hat{c}_k^* H(-\hat{\lambda}_k^*) &= \hat{c}_k^* \hat{H}(-\hat{\lambda}_k^*), \\ H(-\hat{\lambda}_k^*) \hat{b}_k^* &= \hat{H}(-\hat{\lambda}_k^*) \hat{b}_k^*, \\ \hat{c}_k^* H'(-\hat{\lambda}_k^*) \hat{b}_k^* &= \hat{c}_k^* \hat{H}'(-\hat{\lambda}_k^*) \hat{b}_k^* \end{aligned} \tag{4.14}$$

para $k = 1, 2, 3, \dots, n.$, donde $\hat{\lambda}_k \in \mathbb{C}^-$ son los polos de $\hat{\Sigma}$, \hat{b}_k es el k -ésimo renglón de \hat{B} y \hat{c}_k es la k -ésima columna \hat{C} .

Demostración: La demostración de este teorema es algo extensa, y puede encontrarse en [4], tiene como objetivo resolver el conjunto de ecuaciones

$$\begin{aligned} \frac{\partial d^2}{\partial \text{Re}(\hat{\lambda}_k)} &= \frac{\partial \|G(s)\|_{\mathcal{H}_2}^2}{\partial \text{Re}(\hat{\lambda}_k)} = 0, \\ \frac{\partial d^2}{\partial \text{Im}(\hat{\lambda}_k)} &= \frac{\partial \|G(s)\|_{\mathcal{H}_2}^2}{\partial \text{Im}(\hat{\lambda}_k)} = 0, \\ \frac{\partial d^2}{\partial \text{Re}(\hat{b}_{kl})} &= \frac{\partial \|G(s)\|_{\mathcal{H}_2}^2}{\partial \text{Re}(\hat{b}_{kl})} = 0, \\ \frac{\partial d^2}{\partial \text{Im}(\hat{b}_{kl})} &= \frac{\partial \|G(s)\|_{\mathcal{H}_2}^2}{\partial \text{Im}(\hat{b}_{kl})} = 0, \\ \frac{\partial d^2}{\partial \text{Re}(\hat{c}_{qk})} &= \frac{\partial \|G(s)\|_{\mathcal{H}_2}^2}{\partial \text{Re}(\hat{c}_{qk})} = 0, \\ \frac{\partial d^2}{\partial \text{Im}(\hat{c}_{qk})} &= \frac{\partial \|G(s)\|_{\mathcal{H}_2}^2}{\partial \text{Im}(\hat{c}_{qk})} = 0. \end{aligned}$$

4.4. El algoritmo MIRIAM

Las condiciones (4.14) involucran el acoplamiento de un cierto número de momentos de ambas funciones de transferencia $H(s)$ y $\hat{H}(s)$ en los puntos $\{-\hat{\lambda}_1, \dots, -\hat{\lambda}_n\}$, es decir, en las imágenes espejo de los polos del sistema reducido $\hat{\Sigma}$. Calcularlos explícitamente conlleva a inestabilidades numéricas, el remedio para esto es el uso de técnicas de subespacios de Krylov que imponen condiciones de acoplamiento de momentos, evitando el procedimiento explícito. Más aún, el conjunto de datos $\{-\hat{\lambda}_k\}, \{\hat{b}_k\}, \{\hat{c}_k\}$ no es conocido a-priori, de tal manera que debe ser construido iterativamente.

Como ya se mencionó anteriormente, en [4] se presenta el algoritmo MIRIAM: Dado un conjunto inicial de puntos de interpolación $\{\sigma_k^0\}$, así como direcciones tangenciales izquierdas y derechas $\{l_k^0\}, \{r_k^0\}$, si este algoritmo converge, entonces encuentra el conjunto óptimo $\{-\hat{\lambda}_k\}, \{\hat{b}_k\}, \{\hat{c}_k\}$ que satisface las condiciones (4.14), de tal manera que el sistema $\hat{\Sigma}$ construido minimiza $d(H(s), \hat{H}(s))$. A continuación mostramos el algoritmo

MIMO Iterative Rational Interpolation Algorithm (MIRIAM)

1. Seleccionar un conjunto inicial de puntos de interpolación $\sigma = \{\sigma_1, \dots, \sigma_n\}$.
2. Seleccionar un conjunto inicial de direcciones tangenciales $l = \{l_1, \dots, l_n\}$ y $r = \{r_1, \dots, r_n\}$.
3. Hacer, hasta obtener convergencia:
 - a) calcular V_n vía:

$$v_k = (-A + \sigma_k I_N)^{-1} B r_k^*, \text{ para } k = 1, \dots, n.$$
 - b) calcular W_n vía:

$$w_k = (-A^* + \sigma_k^* I_N)^{-1} C^* l_k, \text{ para } k = 1, \dots, n.$$
 - c) $W_n = W_n (W_n^* V_n)^{-*}$ (para obtener $W_n^* V_n = I_n$).
 - d) $\hat{A} = W_n^* A V_n, \hat{B} = W_n^* B, \hat{C} = C V_n$.
 - e) calcular la descomposición en valores propios de $\hat{A} = X \Omega X^{-1}$, donde:

$$\Omega = \text{diag}(\omega_1, \dots, \omega_n).$$
 - f) asignar $\sigma_k = -\omega_k^*, [l_1^*, \dots, l_n^*] = \hat{C} X, [r_1, \dots, r_n] = (X^{-1} \hat{B})^*$.
4. $\hat{H}(s) = \hat{C}(sI_n - \hat{A})^{-1} \hat{B}$.

5. Resultados Numéricos

5.1. Precondicionamiento

En el capítulo anterior vimos que para que $d^2 = \|G(s)\|_{\mathcal{H}_2}^2$ sea mínima, entonces tenemos que construir el sistema dinámico reducido $\hat{H}(s)$ de tal manera que el conjunto de condiciones

$$\begin{aligned}\hat{c}_k^T H(-\hat{\lambda}_k^*) &= \hat{c}_k^T \hat{H}(-\hat{\lambda}_k^*), \\ H(-\hat{\lambda}_k^*) \hat{b}_k^T &= \hat{H}(-\hat{\lambda}_k^*) \hat{b}_k^T, \\ \hat{c}_k^T H'(-\hat{\lambda}_k^*) \hat{b}_k^T &= \hat{c}_k^T \hat{H}'(-\hat{\lambda}_k^*) \hat{b}_k^T,\end{aligned}$$

se satisfagan. Tales condiciones involucran conocer el conjunto de direcciones tangenciales $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}\}$ y los puntos de interpolación $\{-\hat{\lambda}_k^*\}$. En total debemos conocer al conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$ $k = 1, 2, 3, \dots, n$, para hacer que efectivamente d^2 sea mínima. La principal desventaja que enfrentamos, es que no conocemos en primera instancia al conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$ donde \hat{c}_k^T es el transpuesto de la k-ésima columna de la matriz reducida \hat{C} , \hat{b}_k^T es el transpuesto de la k-ésima fila de la matriz reducida \hat{B} , y $-\hat{\lambda}_k^*$ es la imagen espejo (negativo del conjugado de $\hat{\lambda}_k$) del k-ésimo valor propio de la matriz reducida \hat{A} . Estas matrices \hat{A} , \hat{B} , \hat{C} son elementos constitutivos del sistema dinámico reducido $\hat{\Sigma}$ que deseamos construir. Para este fin, proponemos construir iterativamente el conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}$, a partir de un conjunto inicial dado $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$, usando el algoritmo MIRIAM que al final de cada iteración construye un nuevo sistema dinámico reducido $\hat{\Sigma} = (\hat{A}, \hat{B}, \hat{C})$, para que en la iteración subsecuente utilice las imágenes espejo de los valores propios de la matriz \hat{A} como los nuevos puntos de interpolación $\{\sigma_k^{next} = -\text{eig}(\hat{A})^*\}$, las columnas de la transpuesta de \hat{B} como las direcciones tangenciales derecha $\{r_k^{next} = \hat{b}_k^T\}$ y las filas de la transpuesta de \hat{C} como las direcciones tangenciales izquierda $\{l_k^{next} = \hat{c}_k^T\}$.

El algoritmo MIRIAM que presentamos al final del capítulo anterior tiene la siguiente propiedad: si para un conjunto inicial de datos $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$ $k = 1, 2, 3, \dots, n$, el algoritmo converge al conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$, y por ende al sistema reducido $\hat{H}(s)$, entonces resuelve el problema de minimizar la norma $\|G(s)\|_{\mathcal{H}_2}^2$. En este trabajo de tesis nos propusimos buscar un “buen” conjunto inicial de datos $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$ con el objetivo de que MIRIAM convergiera al sistema reducido óptimo $\hat{H}(s)$ en una sola iteración ó al menos en el menor número de iteraciones posibles. Para esto propusimos un proceso de “precondicionamiento” con la idea de equipar al conjunto inicial de datos $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$ con una posible propiedad de rápida convergencia hacia el conjunto óptimo $\{\{\hat{c}_k^T\}, \{\hat{b}_k^T\}, \{-\hat{\lambda}_k^*\}\}$. Seamos claros al respecto, aquí “precondicionamiento” no se refiere al concepto

usualmente encontrado en el álgebra lineal numérica, sino a esta posible propiedad de rápida convergencia que aún debe ser explicada en detalle. Lo cual hacemos a continuación.

Dicho proceso de preconditionamiento, al que hacemos referencia, está dividido en tres fases o etapas como hemos decidido llamarles. La primera etapa (figura 4), tiene como objetivo formar un sistema reducido, llamado sistema reducido nivel 1 de dimensión n_1 con $n_1 < N$, empleando para este fin el algoritmo MIRIAM usando un primer conjunto de datos $\{\{l_k^1\}, \{r_k^1\}, \{\sigma_k^1\}, k = 1, 2, 3, \dots, n_1\}$, provenientes de un sistema reducido llamado Host A de la misma dimensión n_1 construido mediante la proyección del estado-espacio del sistema original Σ . Las matrices V y W usadas

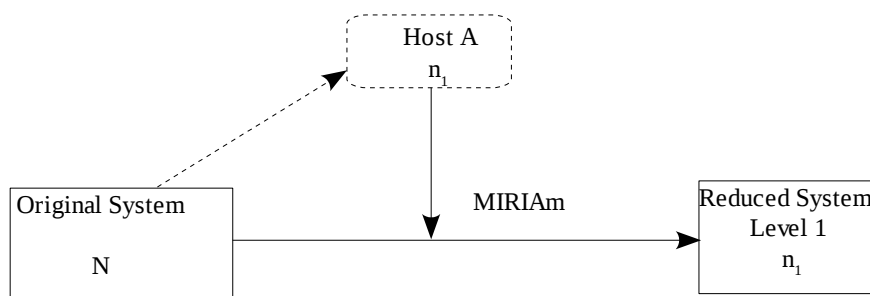


Figura 4: Etapa 1 del proceso de preconditionamiento.

para calcular las matrices reducidas $\hat{A} = W^T A V$, $\hat{B} = W^T B$ y $\hat{C} = C V$ de este Host A, deben ser construidas mediante un procedimiento de Arnoldi para bloques (figura 3, pagina 36) en conformidad con el teorema 3.3.1., donde V es una base del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $W = V$, ó en conformidad con el teorema 3.3.2., donde V es una base del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y W es una base del subespacio de Krylov $K_q\{(A^{-T}), (A^{-T}C^T)\}$. Si fuera el caso de escoger a V y W en base a la descripción del teorema 3.3.2., entonces se debe hacer la siguiente reasignación a la matriz W

$$W \leftarrow W(W^T V)^{-T} \quad (5.1)$$

para obtener $W^T V = I$ (biortogonalidad) y así entonces poder calcular las matrices reducidas \hat{A} , \hat{B} y \hat{C} del Host A.

En la segunda etapa del proceso de preconditionamiento (figura 5), el sistema reducido nivel 1 de dimensión n_1 que construimos en la etapa anterior, es ahora usado

como el punto de partida para formar un conjunto de sistemas reducidos nivel 2 de dimensión $n = 1$ hasta una dimensión fija $n = z$ determinada por la precisión del computador para calcular las matrices V y W en relación al teorema 3.3.1. ó al teorema 3.3.2. Previamente a la construcción de cada uno de estos sistemas reducidos nivel 2, es necesaria la construcción de su respectivo Host B mediante la proyección en el estado-espacio del sistema reducido nivel 1 desde donde se les extraerá un segundo conjunto de datos $\{\{l_k^2\}, \{r_k^2\}, \{\sigma_k^2\} \mid k = 1, 2, 3, \dots, n\}$, para que el algoritmo MIRIAM trabaje hasta la convergencia a cada uno de estos sistemas reducidos nivel 2.

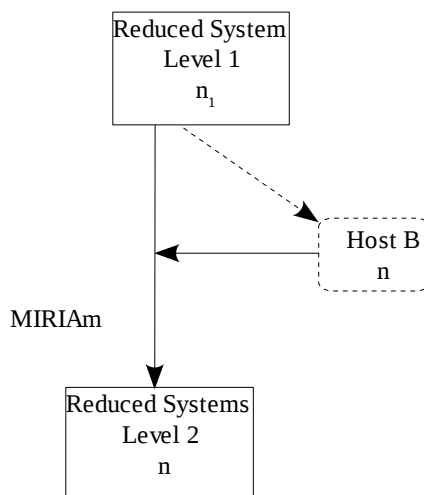


Figura 5: Etapa 2 del proceso de preconditionamiento.

Creemos que la convergencia es un requisito indispensable para hacer de cada uno de estos sistemas reducidos nivel 2 excelentes proveedores de conjuntos de datos “precondicionados” $\{\{l_k\}, \{r_k\}, \{\sigma_k\} \mid k = 1, 2, 3, \dots, z\}$, para ser usados en la siguiente etapa.

En la tercera y última etapa del proceso de preconditionamiento (figura 6) cada uno de los sistemas reducidos nivel 2 calculados en la anterior etapa es usado como fuente de alimentación para que el algoritmo MIRIAM converja desde el sistema dinámico original Σ hacia cada uno de los sistemas reducidos objetivo de dimensión $n = 1$ hasta $n = z$ en una sola iteración, ó al menos en el menor número de iteraciones posibles. La idea aquí es usar de cada uno de los sistemas reducidos nivel 2 las imágenes espejo de los valores propios de \hat{A} como los puntos de interpolación precondicionados $\{\sigma_k\}$, las columnas de la matriz transpuesta de \hat{B} como las direcciones tangenciales precondicionadas derecha $\{r_k\}$, y las filas de la matriz

transpuesta de \hat{C} como las direcciones tangenciales preconditionadas izquierda $\{l_k\}$, $k = 1, 2, 3, \dots, z$. Se espera que con utilizar este conjunto preconditionado $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$, $k = 1, 2, 3, \dots, z$, podemos hacer que MIRIAM converja a cada uno de los sistemas reducidos objetivo $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, z$, en una sola iteración ó en pocas iteraciones, de tal manera que cada una de las distancias entre el sistema original Σ y cada $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, z$, sea la mínima.

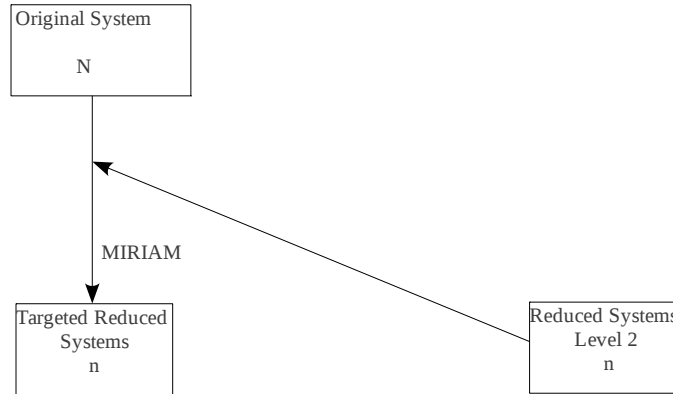


Figura 6: Etapa 3 del proceso de preconditionamiento.

El proceso de preconditionamiento en conjunto puede verse en la siguiente figura

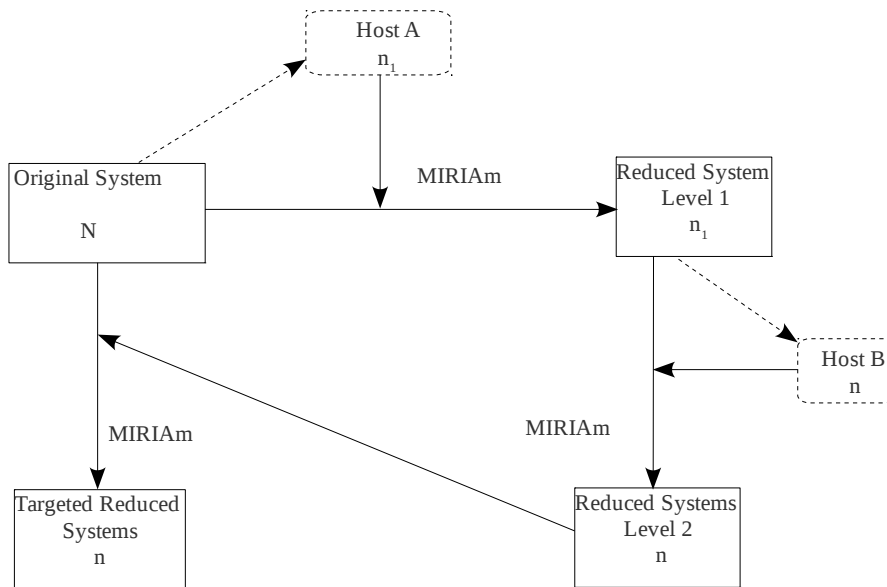


Figura 7: El proceso de preconditionamiento.

Para probar la efectividad de este proceso de preconditionamiento, llevamos a cabo 4 experimentos usando 2 problemas de referencia generados aleatoriamente en MATLAB. El primer problema de referencia consistió en la generación de un sistema dinámico original de dimensión $N = 100$ con $m = 4$ entradas y $p = 5$ salidas, i.e. $A \in \mathbb{R}^{100 \times 100}$, $B \in \mathbb{R}^{100 \times 4}$, y $C \in \mathbb{R}^{5 \times 100}$, y el segundo problema de referencia consistió en la generación de un sistema dinámico original de dimensión $N = 300$ con $m = 3$ entradas y $p = 2$ salidas, i.e. $A \in \mathbb{R}^{300 \times 300}$, $B \in \mathbb{R}^{300 \times 3}$ y $C \in \mathbb{R}^{2 \times 300}$.

El primer problema de referencia fue usado en los primeros 2 experimentos: Model Order Reduction 1 y Model Order Reduction 2. El segundo problema de referencia fue usado en los restantes 2 experimentos: Model Order Reduction 3 y Model Order Reduction 4. En Model Order Reduction 1 y Model Order Reduction 3, los Host A y Host B fueron calculados en conformidad con el Teorema 3.3.1, con V como la base del subespacio de Krylov $K\{(A^{-1}), (A^{-1}B)\}$ y $W = V$. A esta forma en particular de usar a la matriz V para calcular estos Host A y Host B se le conoce como 1-sided Arnoldi. Por otra parte en Model Order Reduction 2 y Model Order Reduction 4, los Host A y Host B fueron calculados en conformidad con el Teorema 3.3.2, con V como la base del subespacio de Krylov $K\{(A^{-1}), (A^{-1}B)\}$ y W como la base del subespacio de Krylov $K\{(A^{-T}), (A^{-T}C^T)\}$. A esta forma en particular de usar las matrices V y W se le conoce como 2-sided Arnoldi.


5.2. Resultados numéricos

Model Order Reduction 1. En la primera etapa de este experimento, calculamos el Host A mediante la proyección del estado-espacio del sistema original Σ usando para esto a la matriz $V \in \mathbb{R}^{100 \times 68}$ en conformidad con el teorema 3.3.1. (1-sided Arnoldi) donde el rango de V es $q = 68$. Este rango en específico quedo determinado por la precisión a la que decidimos formar a la matriz V . Para esto usamos una rutina en MATLAB, basada en el algoritmo de Arnoldi para bloques, que en cada iteración calcula la norma del producto $V^T V$ y la compara contra la tolerancia $tol=1.00000000000001$, de tal manera que cuando $\|V^T V\|_2 \geq tol$ entonces se paran las iteraciones. La matriz V construida en esta forma es una base muy precisa del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$. El Host A producido de esta manera, es entonces un sistema reducido de dimensión $n_1 = 68$, con las mismas $m = 4$ entradas y $p = 5$ salidas, de donde usamos información proveniente de sus matrices reducidas $\hat{A}, \hat{B}, \hat{C}$ para formar iterativamente (80 iteraciones) con el algoritmo MIRIAM el sistema reducido nivel 1 de la misma dimensión $n_1 = 68$. Cabe mencionar que a este número de iteraciones no observamos convergencia hacia el sistema reducido nivel 1. El esquema de este experimento esta en la página 53.

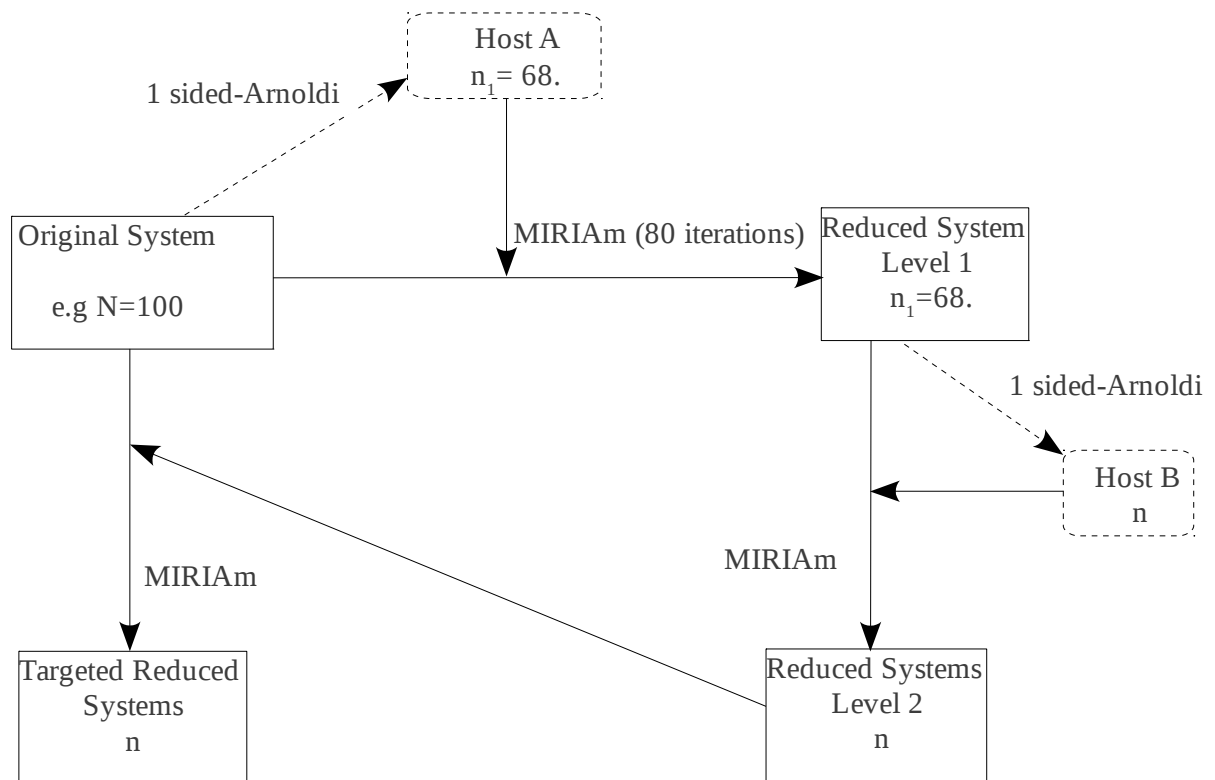
En la segunda etapa de este experimento, tomamos como inicio el sistema reducido nivel 1 de dimensión $n_1 = 68$ calculado en la etapa anterior, para el cual se le calcularon un conjunto de 60 sistemas reducidos Host B desde la dimensión $n = 1$ hasta $n = 60$ mediante su proyección en el estado-espacio. Esto debido a la precisión con la cual ejecutamos la misma rutina basada en el algoritmo de Arnoldi para bloques y que decidimos detener cuando $\|V^T V\|_2 \geq tol$ con la misma tolerancia $tol=1.00000000000001$, de esta forma supimos que la matriz $V \in \mathbb{R}^{68 \times 60}$ formada mediante esta rutina, era una base muy precisa del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ donde $A \in \mathbb{R}^{68 \times 68}$ y $B \in \mathbb{R}^{68 \times 4}$ son las matrices del sistema reducido nivel 1. De cada uno de estos sistemas reducidos Host B, al igual que en la etapa anterior, usamos información proveniente de sus matrices reducidas $\hat{A}, \hat{B}, \hat{C}$ buscando formar iterativamente hasta la convergencia empleando el algoritmo MIRIAM cada uno de los sistemas reducidos nivel 2 desde la dimensión $n = 1$ hasta la dimensión $n = 60$. Los resultados numéricos de esta etapa se encuentran en la pagina 54 bajo la etiqueta “Convergence Analysis for the Reduced Systems Level 2”. Como dijimos anteriormente, estuvimos interesados en formar hasta la convergencia cada uno de los sistemas reducidos nivel 2, y además registrar cuantas iteraciones le llevaba al algoritmo MIRIAM en converger a cada uno de estos, por ejemplo: para el sistema reducido nivel 2 de dimensión $n = 26$, MIRIAM si convergió (por ello de la marca (•) bajo la etiqueta “yes”) a este sistema reducido en 3 iteraciones solamente. El criterio de convergencia que usamos en todos aquellos casos en donde MIRIAM si convergió al sistema reducido nivel 2, fue haber medido la distancia mínima verdadera que separaba el sistema reducido nivel 1 y su correspondiente nivel 2 y buscar en iteraciones previas un valor de la distancia que no se alejara de la mínima verdadera en más de 10^{-3} , por ejemplo en el mismo caso del sistema reducido nivel 2 de dimensión $n = 26$, en la iteración 3 la distancia que separaba al sistema reducido nivel 1 y el correspondiente sistema reducido que MIRIAM había formado hasta esa misma iteración, no estaba alejada más allá de 10^{-3} de la verdadera distancia mínima entre el sistema reducido nivel 1 y el sistema reducido nivel 2 al cual MIRIAM convergió en iteraciones posteriores. Esta precisión en la convergencia fue aplicada en toda esta segunda etapa del proceso de preconditionamiento.

En la etapa 3, usamos de cada sistema reducido nivel 2, información proveniente de las matrices $\hat{A}, \hat{B}, \hat{C}$ para formar el conjunto “precondicionado” $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$, $k = 1, 2, 3, \dots, 60$, con la finalidad de que MIRIAM convergiera hacia cada uno de los sistemas reducidos objetivo de dimensión $n = 1$ hasta $n = 60$ en una sola iteración ó en el menor número de iteraciones posibles. En esta misma etapa también estuvimos interesados en observar cuales eran las dimensiones de los sistemas reducidos objetivo que si convergían y en cuantas iteraciones lo hacían bajo la

misma precisión de 10^{-3} como en la etapa anterior. Los resultados de esta etapa se encuentran en la página 54 bajo la etiqueta “Convergence Analysis for the Targeted Reduced Systems”.

 Casa abierta al tiempo	Project Name: Model Order Reduction 1.
	Autors: Aaron De La Concha.
	Date: 15.07.2010
	Data Source: MATLAB.
	Remarks: Host A was formed by 1 sided-Arnoldi procedure. Reduced System Level 1 was formed using MIRIAM with 80 iterations. Every Host B was formed by 1 sided-Arnoldi. Reduced Systems Level 2 formed using MIRIAM with 300 iterations (when convergence was confirmed), and 500 iterations when doubts about convergence. Targeted Reduced Systems were formed using MIRIAM with only 50 iterations.

SCHEMATICS:



size	Convergence Analysis For The Reduced Systems Level 2.			Convergence Analysis for The Targeted Reduced Systems			which iteration?
	Convergence		If yes, which iteration?	Convergence		If yes iteration one?	
	yes	no		yes	no		
1	•		11	•		✓	1
2	•		9	•		X	2
3	•		8	•		X	3
4	•		3	•		X	4
5		•			•		
6	•		9	•		X	4
7	•		7	•		X	5
8	•		12	•		X	5
9	•		11	•		X	12
10	•		6	•		X	3
11	•		14	•		X	11
12	•		10	•		X	15
13		•			•		
14		•			•		
15	•		6	•		X	5
16	•		13	•		X	8
17		•			•		
18	•		12	•		X	5
19		•			•		
20	•		35	•		X	5
21	•		40	•		X	3
22	•		14	•		X	16
23	•		10	•		X	8
24		•			•		
25	•		5	•		X	13
26	•		3	•		X	9
27	•		5	•		X	4
28	•		9	•		X	4
29	•		5	•		X	8
30	•		6	•		X	11
31	•		3	•		X	2
32	•		4	•		X	3
33	•		10	•		X	2
34	•		5	•		X	2
35	•		5	•		X	20
36	•		2	•		X	5
37	•		4	•		X	2
38	•		3	•		X	2
39	•		4	•		X	5
40	•		2	•		✓	1
41	•		3	•		X	3
42	•		2	•		X	2
43	•		5	•		X	2
44	•		4	•		X	2
45	•		2	•		✓	1
46	•		3	•		✓	1
47	•		2	•		✓	1
48	•		2	•		X	2
49	•		2	•		✓	1
50	•		1	•		X	2
51	•		1	•		X	7
52	•		3	•		✓	1
53	•		1	•		✓	1
54	•		1	•		✓	1
55	•		1	•		X	6
56	•		1	•		✓	1
57	•		2	•		X	2
58	•		1	•		✓	1
59	•		1	•		✓	1
60	•		5	•		✓	1

Model Order Reduction 2. El esquema de este experimento esta en la pagina 57. En la primera etapa de este experimento, construimos las matrices V y W en relación al Teorema 3.3.2., (2-sided Arnoldi) donde V es una base del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y W es una base del subespacio de Krylov $K_q\{(A^{-T}), (A^{-T}C^T)\}$ con una rutina de MATLAB basada en el algoritmo de Arnoldi para bloques y una tolerancia $tol=1.00000000000001$, de tal manera que cuando $\|V^T V\|_2 \geq tol$ y $\|W^T W\|_2 \geq tol$ se paran las iteraciones, teniendo así bases muy precisas para ambos subespacios de Krylov en cuestión. Al termino de esta rutina, obtuvimos que $V \in \mathbb{R}^{100 \times 68}$ y $W \in \mathbb{R}^{100 \times 75}$ por lo cual decidimos tomar de cada una de las anteriores matrices las primeras 60 columnas para estar en concordancia con el valor del rango q (que debe ser el mismo en V y W) al que se hace referencia en el Teorema 3.3.2. Tras haber formado el sistema reducido Host A, de aqui mismo usamos información proveniente de sus matrices reducidas \hat{A} , \hat{B} y \hat{C} como un primer conjunto de datos $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$, $k = 1, 2, 3, \dots, 60$, para hacer que el algoritmo MIRIAM iterara en 80 ocasiones hasta la convergencia hacia el sistema reducido nivel 1 de dimensión $n_1 = 60$ con $m = 4$ entradas y $p = 5$ salidas.

Para la segunda etapa de este experimento, tomamos como inicio el sistema reducido nivel 1 de dimensión $n_1 = 60$ calculado en la etapa anterior, y para el cual se le calcularon un conjunto de 40 sistemas reducidos Host B desde la dimensión $n = 1$ hasta $n = 40$ mediante su proyección en el estado-espacio. Esto fue debido a la precisión con la cual ejecutamos la misma rutina basada en el algoritmo de Arnoldi para bloques y que decidimos detener cuando $\|V^T V\|_2 \geq tol_1$ y $\|W^T W\|_2 \geq tol_1$, donde $tol_1=1.00000000000001$. Escogimos esta tolerancia, ya que a este valor de tol_1 la rutina de Arnoldi para bloques era capaz de construir ambas bases V y W de los correspondientes subespacios de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $K_q\{(A^{-T}), (A^{-T}C^T)\}$ donde el rango q en ambas bases era menor que $n_1 = 60$. Resultó entonces, que a esta tolerancia tol_1 , el rango q en ambas matrices fué de $q = 40$, por ello de la construcción de los 40 sistemas reducidos Host B. Siguiente, de cada uno de estos 40 sistemas reducidos Host B, usamos información proveniente de sus matrices reducidas \hat{A} , \hat{B} y \hat{C} buscando formar iterativamente hasta la convergencia empleando el algoritmo MIRIAM, cada uno de los sistemas reducidos nivel 2 desde la dimensión $n = 1$ hasta la dimensión $n = 40$. Los resultados numéricos de esta etapa se encuentran en la pagina 58 bajo la etiqueta “Convergence Analysis for the Reduced Systems Level 2”. Al igual que en el experimento anterior, nuestro interés se centro en todos aquellos sistemas reducidos para los cuales se observo convergencia, así mismo como anotar cuantas iteraciones necesitaba MIRIAM para llegar a esta. Idénticamente al experimento anterior, nuestro criterio de convergencia se baso en ubicar el número de la iteración en donde la

distancia del sistema reducido en dicha iteración y el sistema reducido nivel 1 no difería en más de 10^{-3} de la verdadera distancia mínima.

En la etapa 3 de este experimento, de cada sistema reducido nivel 2 formado en la etapa anterior, usamos información proveniente de cada matriz \hat{A} , \hat{B} y \hat{C} para formar el conjunto “precondicionado” $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$, $k = 1, 2, 3, \dots, 40$, con la finalidad de que MIRIAM convergiera hacia cada uno de los sistemas reducidos objetivo en el menor número de iteraciones posibles, o si era el caso en una sola iteración. En esta etapa, de todos aquellos sistemas reducidos objetivo que efectivamente convergieron, también lo hicieron sus respectivas contrapartes de la etapa 2, siendo el sistema reducido objetivo de dimensión $n = 38$ el único que no convergió cuando su contraparte si lo había hecho en la etapa anterior. De hecho este fue un caso muy peculiar ya que el algoritmo MIRIAM no iteró ni una sola ocasión e ignoramos por completo el motivo de esto. Para esta misma etapa, se tomó el mismo criterio de convergencia al usar una precisión de 10^{-3} como en la etapa anterior.



Casa abierta al tiempo

Project Name: Model Order Reduction 2.

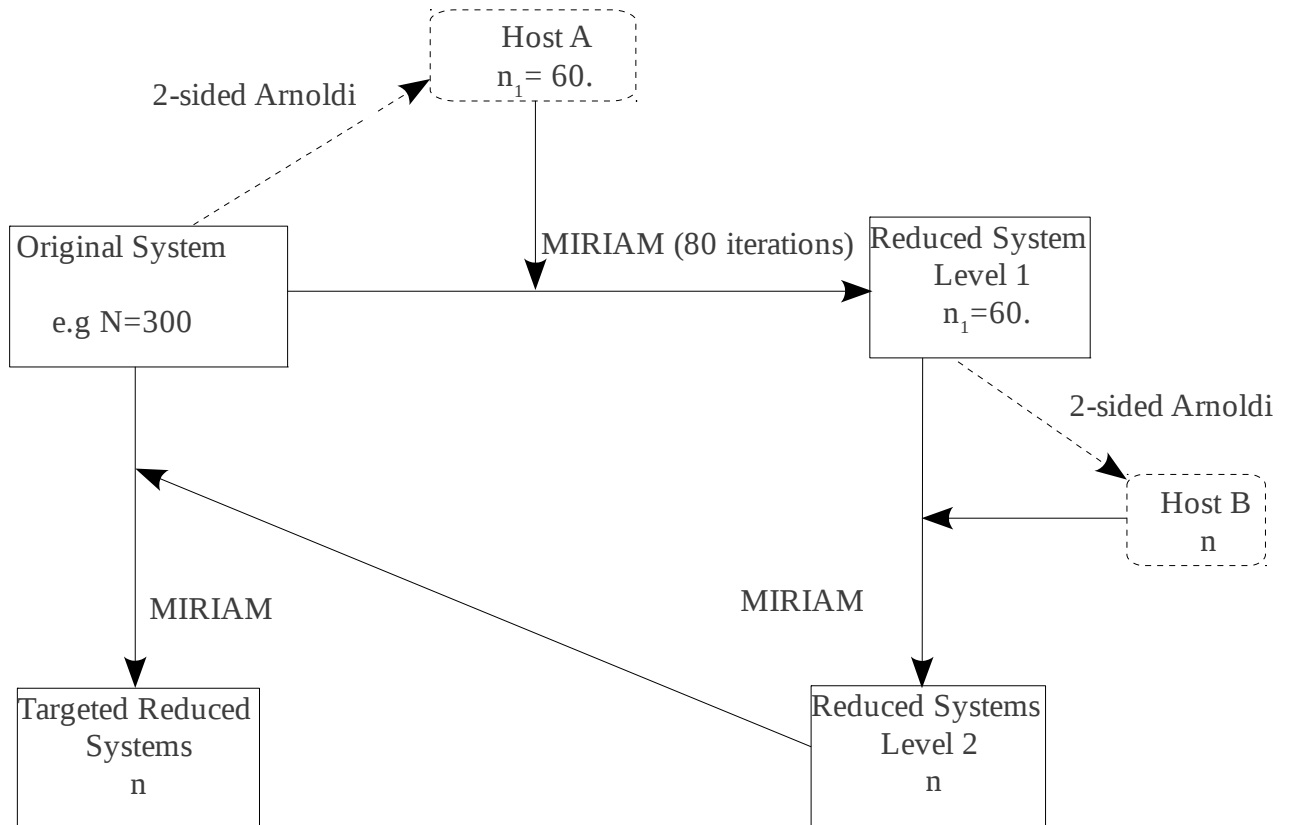
Autors: Aaron De La Concha.

Date: 15.07.2010

Data Source: MATLAB.

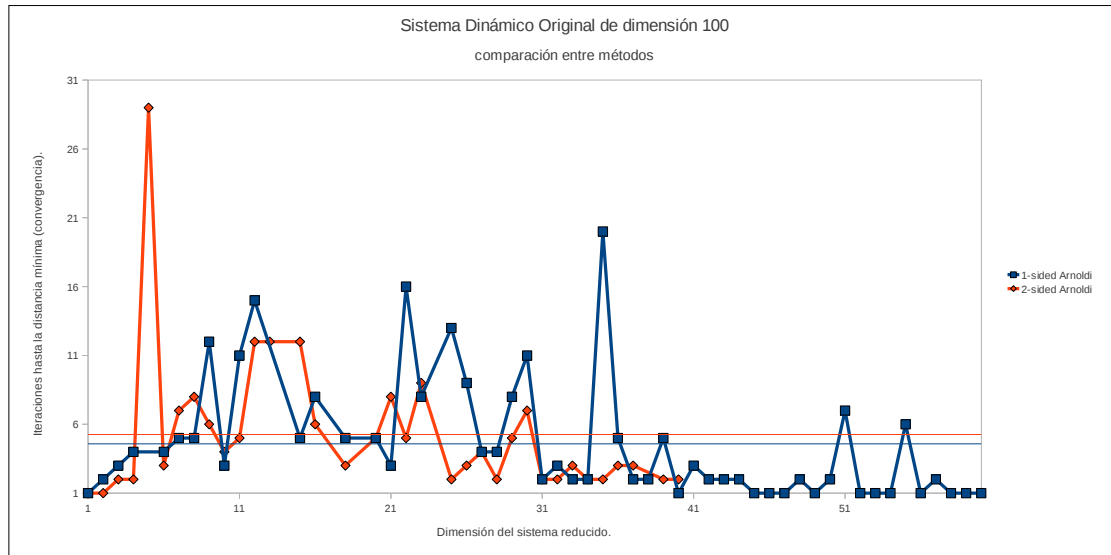
Remarks: Host A formed 2-sided Arnoldi procedure. **Reduced System Level 1** formed using MIRIAM with 80. **Every Host B** was formed 2-sided Arnoldi. **Reduced Systems Level 2** formed using MIRIAM with 300 iterations (when convergence was confirmed), and 500 iterations when doubts about convergence. **Targeted Reduced Systems** were formed using MIRIAM with only 50 iterations.

SCHEMATICS:



Convergence Analysis For The Reduced Systems Level 2.				Convergence Analysis for The Targeted Reduced Systems			
size	Convergence		If yes , which iteration?	Convergence		If yes iteration one?	which iteration?
	yes	no		yes	no		
1	•		12	•		✓	1
2	•		9	•		✓	1
3	•		5	•		X	2
4	•		8	•		X	2
5	•		5	•		X	29
6	•		10	•		X	3
7	•		7	•		X	7
8	•		11	•		X	8
9	•		12	•		X	6
10	•		43	•		X	4
11	•		18	•		X	5
12	•		6	•		X	12
13	•		131	•		X	12
14		•			•		
15	•		12	•		X	12
16	•		38	•		X	6
17		•			•		
18	•		10	•		X	3
19		•			•		
20	•		6	•		X	5
21	•		34	•		X	8
22	•		5	•		X	5
23	•		8	•		X	9
24		•			•		
25	•		5	•		X	2
26	•		4	•		X	3
27	•		7	•		X	4
28	•		6	•		X	2
29	•		5	•		X	5
30	•		6	•		X	7
31	•		6	•		X	2
32	•		4	•		X	2
33	•		4	•		X	3
34	•		3	•		X	2
35	•		2	•		X	2
36	•		7	•		X	3
37	•		4	•		X	3
38	•		11		•		
39	•		2	•		X	2
40	•		4	•		X	2

Este es el gráfico que muestra los resultados numéricos correspondientes a la tercera etapa de los proyectos Model Order Reduction 1 y Model Order Reduction 2.




Se puede ver que en promedio (línea azul horizontal) el proceso de preconditionamiento basado en la técnica de 1-sided Arnoldi, tiende a reducir el número de iteraciones necesarias para la convergencia hacia cada sistema reducido objetivo $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, 60$, en comparación con el promedio del proceso de preconditionamiento (línea naranja horizontal) basado en la técnica 2-sided Arnoldi.

Model Order Reduction 3. El esquema de este experimento esta en la pagina 61. La primera etapa consistió en construir a la matriz V en conformidad con el Teorema 3.3.1. (1-sided Arnoldi) donde V es la base del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $W = V$. La tolerancia que usamos fue de $tol=1.000000000001$ y decidimos parar la rutina de MATLAB cuando $\|V^T V\|_2 \geq tol$, dando como resultado una base $V \in \mathbb{R}^{300 \times 69}$ donde $q = 69$ es el rango de V al que se hace referencia en dicho Teorema. El sistema reducido Host A fue calculado de la proyección del estado-espacio del sistema original Σ , y desde donde se uso información proveniente de sus matrices \hat{A} , \hat{B} y \hat{C} para construir con el algoritmo MIRIAM en 60 iteraciones hasta la convergencia el sistema reducido nivel 1 de dimensión $n_1 = 69$ con $m = 3$ entradas y $p = 2$ salidas.

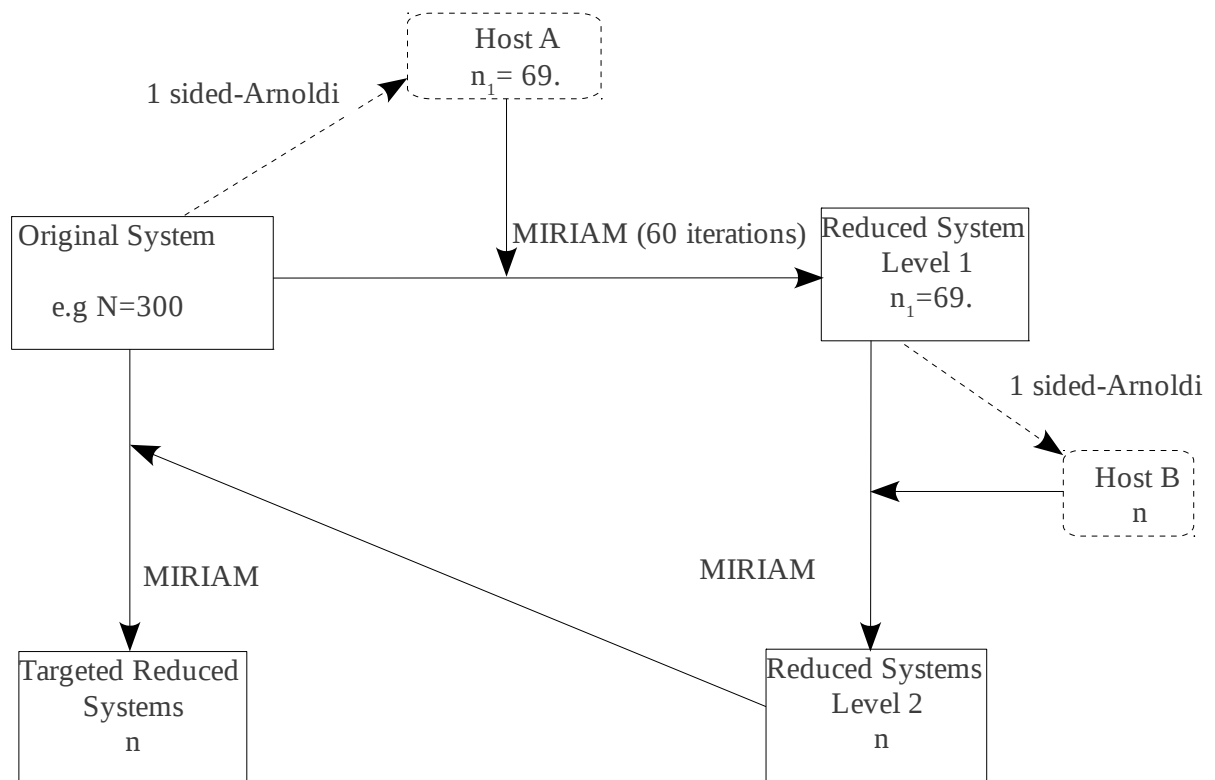
En la segunda etapa, debido a la tolerancia $tol_1=1.000000000001$ que usamos para construir la base V del subespacio de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$, donde A y B son la matrices del sistema reducido nivel 1, obtuvimos a la matriz $V \in \mathbb{R}^{69 \times 45}$ donde $q = 45$ (rango de V) determinaba el número máximo de sistemas reducidos nivel 2 que podíamos formar, en este caso desde la dimensión $n = 1$ hasta la dimensión $n = 45$. Para la formación de cada uno de estos sistemas reducidos nivel 2, era necesario entonces calcular su correspondiente sistema reducido Host B mediante la proyección del estado-espacio del sistema reducido nivel 1. Los resultados numéricos de esta etapa se encuentran en la pagina 62 bajo la etiqueta “Convergence Analysis for the Reduced Systems Level 2”. Al igual que en los experimentos anteriores, mantuvimos un estricto control de aquellas dimensiones que si convergían y en que numero de iteración lo hacían cuando considerábamos una precisión de 10^{-3} como en los casos anteriores. Debemos mencionar una peculiaridad que observamos con el sistema reducido nivel 2 de dimensión $n = 39$, para el cual el algoritmo MIRIAM no itero en ni una sola ocasión cuando se le alimento con el conjunto de datos $\{\{l_k^2\}, \{r_k^2\}, \{\sigma_k^2\}\}$, $k = 1, 2, 3, \dots, 39$, provenientes de su correspondiente sistema reducido Host B de la misma dimensión $n = 39$.

La tercera etapa de este proyecto consistió en usar información proveniente de cada uno de los 39 sistemas reducidos nivel 2 calculados en la etapa anterior para utilizar esta información junto con el algoritmo MIRIAM en un máximo de 200 iteraciones y obtener un registro detallado de cuales fueron los sistemas reducidos objetivo que si convergieron en esta etapa bajo el mismo criterio de convergencia usando una precisión de 10^{-3} . Los resultados pueden observarse en la pagina 62 bajo la etiqueta “Convergence Analysis for the Targeted Reduced Systems”. Un caso particular lo observamos con el sistema reducido objetivo de dimensión $n = 35$ que en esta etapa si convergió aún y cuando el correspondiente sistema reducido nivel 2 de la misma dimensión $n = 35$ no lo había hecho en la etapa anterior. Para

el sistema reducido objetivo de dimensión $n = 39$, no hubo convergencia ya que no obtuvimos datos provenientes de su correspondiente sistema reducido nivel 2 en la etapa anterior por el hecho de que el algoritmo MIRIAM no itero en una sola ocasión como se explico con anterioridad.

 <p>Casa abierta al tiempo</p>	Project Name: Model Order Reduction 3.
	Autors: Aaron De La Concha.
	Date: 15.07.2010
	Data Source: MATLAB.
	Remarks: Host A formed 1 sided-Arnoldi procedure. Reduced System Level 1 was formed using MIRIAM with 60 iterations. Every Host B was formed by 1 sided-Arnoldi. Reduced Systems Level 2 formed using MIRIAM with 300 iterations (when convergence was confirmed), and 500 iterations. Targeted Reduced Systems were formed using MIRIAM with 50 iterations (when convergence was confirmed), and 200 iterations when doubts of convergence.

SCHEMATICS:



size	Convergence Analysis For The Reduced Systems Level 2.			Convergence Analysis for The Targeted Reduced Systems			which iteration?
	Convergence		If yes, which iteration?	Convergence		If yes iteration one?	
	yes	no		yes	no		
1	•		2	•		✓	1
2	•		6	•		X	2
3	•		5	•		X	3
4	•		8	•		X	2
5	•		13	•		X	3
6	•		110	•		X	4
7	•		17	•		X	4
8	•		20	•		X	9
9	•		92	•		X	5
10	•		34	•		X	32
11	•		52	•		X	66
12	•		22	•		X	6
13	•		36	•		X	4
14	•		24	•		X	16
15	•		21	•		X	66
16	•		30	•		X	23
17	•		13	•		X	25
18	•		60	•		X	26
19	•		84	•		X	65
20		•			•		
21	•		14	•		X	11
22		•			•		
23	•		109	•		X	8
24	•		39	•		X	31
25	•		73	•		X	4
26	•		37	•		X	14
27	•		32	•		X	6
28		•			•		
29	•		27	•		X	60
30	•		92	•		X	18
31	•		100	•		X	7
32	•		41	•		X	9
33	•		46	•		X	37
34	•		50	•		X	16
35		•		•		X	37
36	•		38	•		X	14
37	•		57	•		X	11
38	•		51	•		X	12
39		•			•		
40	•		58	•		X	44
41	•		69	•		X	7
42	•		21	•		X	11
43	•		72	•		X	46
44	•		38	•		X	10
45	•		170	•		X	10

Model Order Reduction 4. El esquema de este experimento esta en la pagina 64. En la primera etapa de este experimento construimos a las matrices V y W como las bases de los subespacios de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $K_q\{(A^{-T}), (A^{-T}C^T)\}$ respectivamente. Para esto usamos la rutina de MATLAB en base al algoritmo de Arnoldi para bloques que detuvimos cuando $\|V^T V\| \geq tol$ y $\|W^T W\| \geq tol$ donde $tol=1.000000000001$, teniendo como resultado que a esta tolerancia $V \in \mathbb{R}^{300 \times 69}$ y $W \in \mathbb{R}^{300 \times 90}$. En base a la descripción del Teorema 3.3.2., ambas matrices V y W deben tener el mismo rango q , por ello decidimos que $q = 66$ sería el valor máximo que podía ajustarse. El Host A, fue calculado en base a la proyección del estado-espacio del sistema original Σ . De este Host A, se uso información proveniente de sus matrices reducidas \hat{A} , \hat{B} y \hat{C} para tratar de formar con 200 iteraciones usando el algoritmo MIRIAM el sistema reducido nivel 1 de dimensión $n_1 = 66$ con $m = 3$ entradas y $p = 2$ salidas. Cabe mencionar que a este numero de iteraciones (200) no hubo convergencia.

En la segunda etapa de este experimento, usamos una tolerancia $tol_1=1.000000000001$ para generar las bases V y W de los subespacios de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $K_q\{(A^{-T}), (A^{-T}C^T)\}$ respectivamente, donde A, B y C son las matrices reducidas del sistema reducido nivel 1 calculado en la etapa anterior. Con este valor de la tolerancia tol_1 obtuvimos que $V \in \mathbb{R}^{66 \times 42}$ y $W \in \mathbb{R}^{66 \times 42}$, por lo cual decidimos tomar las primeras 40 columnas de ambas matrices para que el rango $q = 40$ estuviera en concordancia con la definición del Teorema 3.3.2. Para esta etapa calculamos un máximo de 40 sistemas reducidos Host B desde donde se uso de sus respectivas matrices reducidas \hat{A} , \hat{B} y \hat{C} información para iterar buscando convergencia a cada uno de los sistemas reducidos nivel 2 desde la dimensión $n = 1$ hasta $n = 40$. Los resultados numéricos de esta etapa están en la pagina 65 bajo la etiqueta “Convergence Analysis for the Reduced Systems Level 2” en donde aplicamos el mismo criterio de convergencia al considerar una precisión de 10^{-3} como en los experimentos anteriores. De esta etapa, solo en 2 casos no hubo convergencia, siendo el de la dimensión $n = 40$ el más peculiar ya que aquí MIRIAM no itero en una sola ocasión. En la etapa 3, aplicamos el mismo criterio de convergencia al considerar una precisión de 10^{-3} como en todos los experimentos anteriores. Los resultados se pueden ver en la pagina 65 bajo la etiqueta “Convergence Analysis for the Targeted Reduced Systems”. En esta etapa capturo nuestra atención los sistemas reducidos objetivo de dimensión $n = 22$ y $n = 28$ que no convergieron cuando sus contrapartes en la etapa 2 si lo hicieron. En el caso particular de la dimensión $n = 40$, no tuvimos convergencia ya que su correspondiente sistema reducido nivel 2 no fue formado por la inhabilidad del algoritmo MIRIAM de efectuar iteraciones como se explico con anterioridad.



Casa abierta al tiempo

Project Name: Model Order Reduction 4.

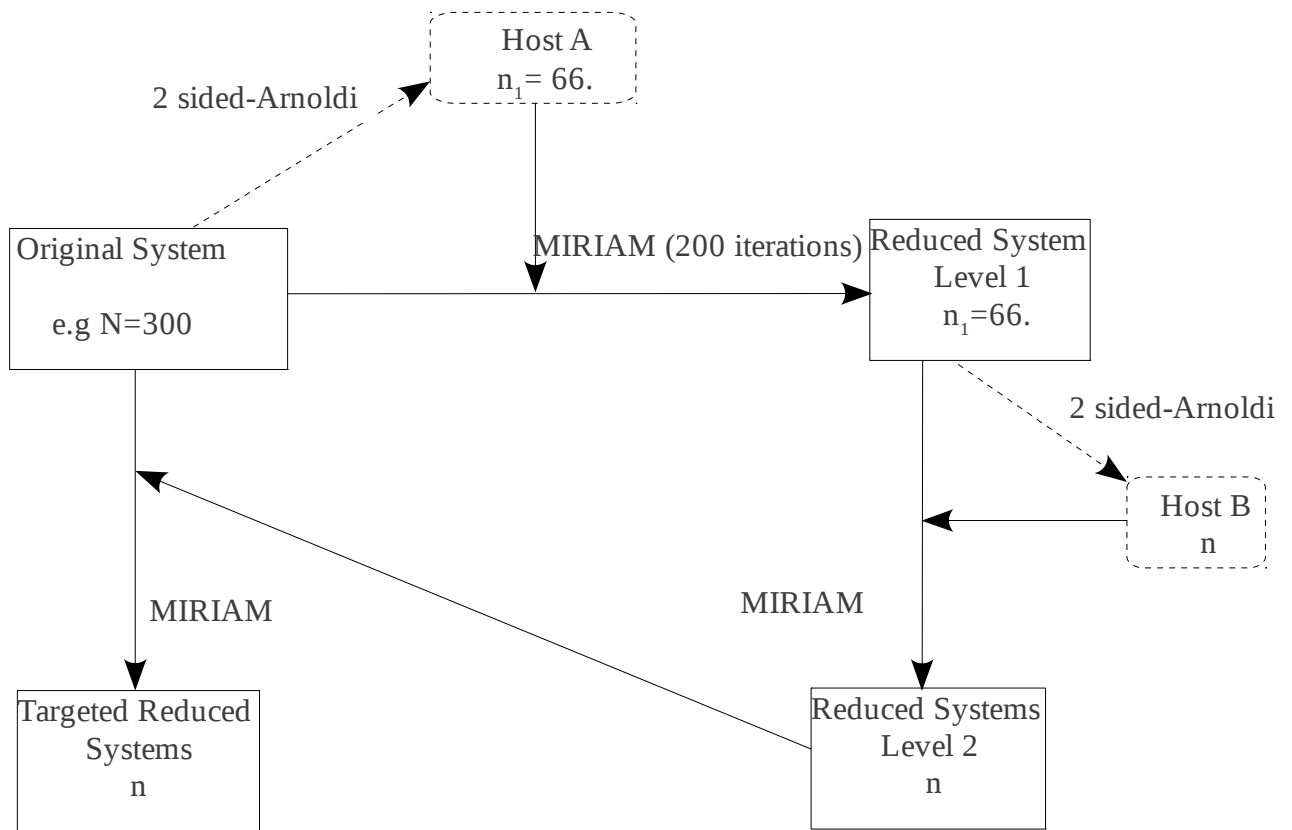
Autors: Aaron De La Concha.

Date: 15.07.2010

Data Source: MATLAB.

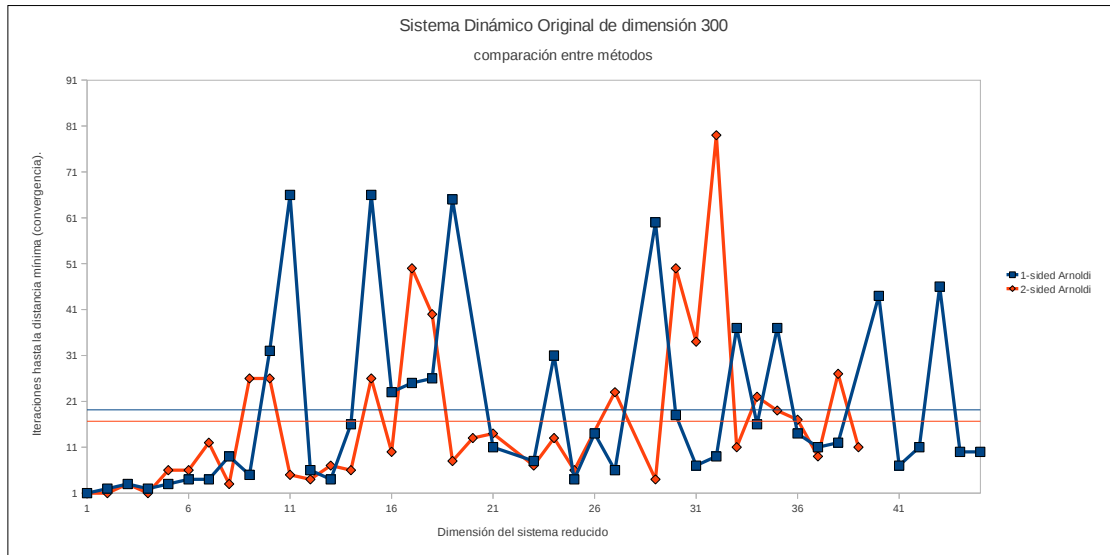
Remarks: Host A formed 2 sided-Arnoldi procedure. **Reduced System Level 1** formed using MIRIAM with 200 iterations. **Every Host B** was formed by 2 sided-Arnoldi. **Reduced Systems Level 2** formed using MIRIAM with 300 iterations (when convergence was confirmed), and 500 iterations. **Targeted Reduced Systems** were formed using MIRIAM with only 50 iterations (when convergence was confirmed), and 200 iterations when doubts of convergence.

SCHEMATICS:



size	Convergence Analysis For The Reduced Systems Level 2.			Convergence Analysis for The Targeted Reduced Systems			which iteration?
	Convergence		If yes , which iteration?	Convergence		If yes iteration one?	
	yes	no		yes	no		
1	•		1	•		✓	1
2	•		6	•		✓	1
3	•		8	•		X	3
4	•		8	•		✓	1
5	•		29	•		X	6
6	•		8	•		X	6
7	•		8	•		X	12
8	•		19	•		X	3
9	•		16	•		X	26
10	•		8	•		X	26
11	•		77	•		X	5
12	•		45	•		X	4
13	•		35	•		X	7
14	•		14	•		X	6
15	•		23	•		X	26
16	•		300	•		X	10
17	•		25	•		X	50
18	•		94	•		X	40
19	•		8	•		X	8
20	•		27	•		X	13
21	•		13	•		X	14
22	•		11		•		
23	•		14	•		X	7
24	•		29	•		X	13
25	•		50	•		X	6
26		•			•		
27	•		34	•		X	23
28	•		33		•		
29	•		33	•		X	4
30	•		103	•		X	50
31	•		15	•		X	34
32	•		50	•		X	79
33	•		37	•		X	11
34	•		57	•		X	22
35	•		22	•		X	19
36	•		19	•		X	17
37	•		30	•		X	9
38	•		23	•		X	27
39	•		24	•		X	11
40		•			•		

Este es el gráfico que muestra los resultados numéricos correspondientes a la tercera etapa de los proyectos Model Order Reduction 3 y Model Order Reduction 4.



Se puede ver que en promedio (línea naranja horizontal), el proceso de preconditionamiento basado en la técnica 2-sided Arnoldi, tiende a reducir el número de iteraciones en que el algoritmo MIRIAM construye cada uno de los sistemas objetivo reducidos $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, 45$, en comparación con el promedio del proceso de preconditionamiento (línea azul horizontal) basado en la técnica 1-sided Arnoldi.

6. Conclusiones y Futuras Direcciones

Los métodos de reducción de orden de modelos dinámicos basados en las técnicas de subespacios de Krylov se han convertido en la actualidad en la elección más popular debido a su capacidad de manejar grandes dimensiones de N para la construcción eficiente del sistema dinámico reducido $\hat{\Sigma}$ a partir del original Σ . De estos, acoplamiento implícito de momentos resulta ser la mejor opción ya que acopla el mayor número de momentos de ambas funciones de transferencia $H(s)$ y $\hat{H}(s)$ alrededor de $\sigma = 0 \in \mathbb{C}$ sin la necesidad de calcularlos explícitamente. La generalización de la anterior técnica es conocida como interpolación racional de Hermite, que también acopla momentos de $H(s)$ y de $\hat{H}(s)$ no solamente en $\sigma = 0 \in \mathbb{C}$, sino en una multitud $\sigma_k \in \mathbb{C}$, $k = 1, 2, 3, \dots, n$, de tal manera que $\hat{H}(s)$ sea una mejor aproximación de $H(s)$ en varios puntos del plano complejo.

Además del conjunto de puntos de interpolación $\{\sigma_k\}$, también es necesario contar con un conjunto de direcciones tangenciales izquierda $\{l_k \in \mathbb{R}^{1 \times p}\}$, y direcciones tangenciales derecha $r_k \in \mathbb{R}^{m \times 1}$ de tal manera que las siguientes condiciones

$$\begin{aligned} l_k H(\sigma_k) &= l_k \hat{H}(\sigma_k), \\ H(\sigma_k) r_k &= \hat{H}(\sigma_k) r_k, \\ l_k H'(\sigma_k) r_k &= l_k \hat{H}'(\sigma_k) r_k \end{aligned}$$

se cumplan para $k = 1, 2, 3, \dots, n$. La desventaja de esta técnica radica en que el conjunto $\{\{l_k\}, \{r_k\}, \{\sigma_k\}\}$, $k = 1, 2, 3, \dots, n$, que hace que se cumplan las anteriores condiciones no se conoce en primera instancia, forzosamente debe ser construido iterativamente. Para este fin se diseñó un algoritmo llamado MIRIAM que si para una propuesta inicial de datos $\{\{l_k^0\}, \{r_k^0\}, \{\sigma_k^0\}\}$, converge al conjunto óptimo $\{\{\hat{c}_k\}, \{\hat{b}_k\}, \{-\hat{\lambda}_k^*\}\}$, $k = 1, 2, 3, \dots, n$, entonces el sistema reducido $\hat{H}(s)$ al que ha convergido minimiza la distancia $d(H(s), \hat{H}(s)) = \|H(s) - \hat{H}(s)\|_{\mathcal{H}_2}$ y hace que el siguiente conjunto de condiciones

$$\begin{aligned} \hat{c}_k^T H(-\hat{\lambda}_k^*) &= \hat{c}_k^T \hat{H}(-\hat{\lambda}_k^*), \\ H(-\hat{\lambda}_k^*) \hat{b}_k^T &= \hat{H}(-\hat{\lambda}_k^*) \hat{b}_k^T, \\ \hat{c}_k^T H'(-\hat{\lambda}_k^*) \hat{b}_k^T &= \hat{c}_k^T \hat{H}'(-\hat{\lambda}_k^*) \hat{b}_k^T \end{aligned}$$

sean validas para $k = 1, 2, 3, \dots, n$. En este trabajo de tesis, quisimos poner a prueba el efecto que tiene un proceso de preconditionamiento para hacer que el conjunto inicial de datos $\{\{l_k^0\}, \{r_k^0\}, \{\sigma_k^0\}\}$ fuera un “buen” conjunto inicial de datos con la propiedad de que el algoritmo MIRIAM convergiera al sistema dinámico reducido $\hat{\Sigma}$ a partir del original Σ en una sola iteración, o al menos en el menor

numero de iteraciones posibles. Para este fin llevamos a cabo 4 experimentos basados en 2 sistemas dinámicos generados aleatoriamente en MATLAB. Los primeros 2 experimentos aplicados al primer sistema dinámico de dimensión $N = 100$, y los restantes experimentos aplicados al segundo sistema dinámico de dimensión $N = 300$. Del primer gráfico comparativo (pagina 59) podemos ver que en promedio, el proceso de preconditionamiento basado en la técnica de 1-sided Arnoldi tiende a reducir el número de iteraciones necesarias en que el algoritmo MIRIAM encuentra el correspondiente sistema reducido óptimo $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, 60$, de tal manera que su correspondiente distancia $d(\Sigma, \hat{\Sigma})$ sea la mínima. Este resultado en particular fue un poco sorprendente, ya que creíamos que al usar información preconditionada proveniente de la técnica 2-sided Arnoldi, haríamos a este proceso basado en esta misma técnica mucho más eficiente que su contraparte 1-sided Arnoldi. Creemos que esto se debió a la limitada cantidad de sistemas reducidos nivel 2 que pudimos formar debido a las restricciones computacionales inherentes que se tienen cuando se forman las bases V y W de los correspondientes subespacios de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $K_q\{(A^{-T}), (A^{-T}C^T)\}$.

En el caso del segundo gráfico comparativo (pagina 66), vemos que cuando usamos datos preconditionados provenientes de la técnica 2-sided Arnoldi, en promedio se necesitan menos iteraciones para formar los sistemas reducidos nivel 2 $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, 40$, en comparación con la técnica 1-sided Arnoldi que aunque permite formar más sistemas reducidos nivel 2 $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, 45$, esta técnica en promedio necesita más iteraciones. Este resultado esta en concordancia con nuestras expectativas, ya que creíamos que al usar a las matrices V y W como las bases de los subespacios de Krylov $K_q\{(A^{-1}), (A^{-1}B)\}$ y $K_q\{(A^{-T}), (A^{-T}C^T)\}$, haríamos del conjunto de datos preconditionados provenientes de cada sistema reducido nivel 2, un conjunto tal que haría que el algoritmo MIRIAM iterara en un menor número de ocasiones en comparación de su contraparte 1-sided Arnoldi.

Aunque los resultados en cada par de experimentos aplicados a cada uno de los 2 problemas son contradictorios, tenemos la convicción de que el proceso de preconditionamiento basado en la técnica 2-sided Arnoldi da mejores resultados en relación al número de iteraciones necesarias para construir el sistema reducido objetivo $\hat{\Sigma}_k$, $k = 1, 2, 3, \dots, n$. Creemos que con este procedimiento en específico, podemos reducir significativamente el número de iteraciones en comparación con el procedimiento basado en la técnica 1-sided Arnoldi. Pero para llegar a una conclusión fundamentada, es necesaria mucha más experimentación considerando ambos casos, con la idea de tener una amplia base de resultados desde donde se pueda llevar a cabo un estudio estadístico con un alto grado de certeza para ser concluyentes.

7. Apéndice

7.1. Subespacios de Krylov

Un subespacio de Krylov $K_n(A, b)$ de dimensión n donde $A \in \mathbb{R}^{N \times N}$ y $b \in \mathbb{R}^N$, es un subconjunto de \mathbb{R}^N , $K_n(A, b) \subset \mathbb{R}^N$, que está expresado de la siguiente forma

$$K_n(A, b) = \text{span}\{b, Ab, A^2b, \dots, A^{n-1}b\}.$$

Existe un algoritmo llamado Arnoldi para construir una base del anterior subespacio evitando las inestabilidades numéricas que se pueden presentar si se calcularan directamente los vectores $A^j b$, $j = 0, 1, 2, \dots, n - 1$, a medida que j crece. Este algoritmo construye iterativamente una base $\{v_1, v_2, v_3, \dots, v_n\}$ para el subespacio $K_n(A, b)$ teniendo como inicio al vector $v_1 = \frac{b}{\|b\|_2}$. El algoritmo se presenta a continuación

ARNOLDI ALGORITHM (A, b)	
1.	Compute $v_1 = \frac{b}{\ b\ _2}$ % normalize b
2.	For $j = 1, \dots, n$, do
3.	Compute $w_j = Av_j$ % calculate the next vector
4.	For $i = 1, \dots, j$, do ↵
5.	$h_{i,j} = v_i^* w_j$
6.	$w_j = w_j - h_{i,j} v_i$ % orthogonalization
7.	End
8.	If $w_j = 0$, STOP, otherwise $h_{j+1,j} = \ w_j\ _2$
9.	$v_{j+1} = \frac{w_j}{h_{j+1,j}}$ % normalization
10.	End

En esta tesis, nuestro interés se centró en sistemas MIMO donde $A \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^{N \times m}$ y $C \in \mathbb{R}^{p \times N}$, por lo cual el correspondiente subespacio de Krylov que obtenemos es

$$K_n(A, B) = \text{span}\{B, AB, A^2B, \dots, A^{n-1}B\}$$

para el cual es posible construir una base mediante el uso del Algoritmo de Arnoldi para bloques, tomando como inicio una matriz unitaria $V_1 \in R^{N \times m}$ que comunemente se obtiene al hacer la factorización Q-R de la matriz AB . El algoritmo de Arnoldi para bloques se presenta a continuación

ALGORITHM **Block Arnoldi**

1. Choose a unitary matrix V_1 .
 2. For $j = 1, 2, \dots, m$ Do:
 3. Compute $W_j := AV_j$
 4. For $i = 1, 2, \dots, j$ do:
 5. $H_{ij} := V_i^T W_j$
 6. $W_j := W_j - V_i H_{ij}$
 7. EndDo
 8. Compute the Q-R decomposition $W_j = V_{j+1} H_{j+1,j}$
 9. EndDo
-

Referencias

- [1] The MATLAB documentation center. Available at:
<http://www.mathworks.com/access/helpdesk/help/toolbox/control/getstart/f1-1010549.html>
- [2] Modern Control Engineering. Third Edition.
Katsuhiko Ogata. Prentice Hall. Page 121-122.
- [3] Oberwolfach model reduction benchmark collection. Available online at:
<http://www.imtek.de/simulation/benchmark/>.
- [4] Dorota Kubalinska. *Optimal interpolation-based model reduction*. PhD Thesis, Universität Bremen, 2008.
- [5] B.C. Moore. Principal component analysis in linear systems: controllability, observability and model reduction. *IEEE Trans. Automatic Control*, 26(1):17–32, 1981.
- [6] P. Astrid. *Reduction of Process Simulation Models: a proper orthogonal decomposition approach*. PhD thesis, Technische Universitaet Eindhoven, 2004.
- [7] E.N. Lorenz. *Empirical orthogonal functions and statistical weather prediction*. Scientific Report No. 1 in Statistical Forecasting Project. MIT Press, Cambridge, MA, 1956.
- [8] K. Glover. *All optimal Hankel-norm approximations of linear multivariable systems and their L^∞ -error bounds*. International Journal of Control, 39:1115–1193, 1984.
- [9] P. Benner, E. S. Quintana-Orti, and G. Quintana-Ort. *Singular perturbation approximation of large, dense linear systems*. In Proceedings of the 2000 IEEE International Symposium on Computer Aided Control System Design, Anchorage, Alaska, USA, September 25-27, 2000, pages 255–260, Omnipress, Madison, WI, 2000.
- [10] S. Gugercin, A.C. Antoulas, and C. Beattie. *Rational Krylov methods for optimal H_2 model reduction*. Technical report, Virginia Tech, 2006. Submitted to SIMAX.
- [11] K. Gallivan, A. Vandendorpe, and P. Van Dooren. *Model reduction of mimo systems via tangential interpolation*. SIAM J. Matrix Anal. Appl., 26(2):328–349, 2005.

- [12] Diederich Hinrichsen, Anthony J. Pritchard. *Mathematical Systems Theory I. Modelling, State Space Analysis, Stability and Robustness*. Springer Verlag. 2000.
- [13] A.C. Antoulas. *Approximation of large-scale dynamical systems*. Number DC-06 in Advances in Design and Control. SIAM, Philadelphia, PA, USA, 1st edition, 2005.
- [14] Roger W. Brockett, H.H. Rosenbrock. *State-space and Multivariable Theory*. John Wiley and Sons. New York. 1970.
- [15] P.L. Duren. *Theory of H^p spaces*. Academic Press, New York, 1970.
- [16] K. Hoffman. *Banach spaces of analytic functions*. Prentice Hall, London, 1962.
- [17] P. Koosis. *Introduction to \mathcal{H}^p spaces*. Cambridge University Press, Cambridge, 1980.
- [18] M. Rosenblum and J. Rovnyak. *Hardy classes and Operator theory*. Oxford University Press, Oxford, 1985.
- [19] M. Rosenblum and J. Rovnyak. *Topics in hardy classes and univalent functions*. 1994.
- [20] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Second edition. 2000.
- [21] Salimbahrami, B. Lohmann, B. *Krylov Subspace Methods in Linear Model Order Reduction: Introduction and Invariances Properties*. Institute of Automation, University of Bremen, Germany. August 2002.
- [22] K. Gallivan, A. Vandendorpe, and P. Van Dooren. *Model reduction of mimo systems via tangential interpolation*. SIAM J. Matrix Anal. Appl., 26(2):328–349, 2005.
- [23] K.A. Gallivan, A. Vandendorpe, and P. Van Dooren. *Sylvester equations and projection-based model reduction*. Journal of Computational and Applied Mathematics, 162:213–229, 2004.
- [24] A. Vandendorpe. *Model reduction of linear systems, an interpolation point of view*. PhD thesis, Univerisité Catholique De Louvain, 2004.
- [25] John H. Mathews, Russel W. Howell. *Complex Analysis for Mathematics and Engineering*. Third edition. Johns and Bartlett publishers. 1997.

- [26] E.J. Grimme. *Krylov projection methods for model reduction*. PhD thesis, ECE Department, University of Illinois, Urbana-Champaign, 1997.
- [27] S.Barnett. *Introduction to Mathematical Control Theory*. Clarendon Press, 1985.
- [28] W.M. Wonham. *Linear Multivariable Control. A Geometric Approach*. Springer-Verlag, 1985.