

Desacoplamiento para Sistemas Continuos en el Tiempo mediante la Técnica Matriz Función de Transferencia

Uncoupling Continuous Time Systems through a Transfer Function Matrix Technique

E. Granizo, P. Ayala y M. Barragán

Abstract— Multivariable control systems have several inputs and outputs; each input controls more than one output, and each output is controlled by more than one input. This coupling feature presents mathematical complications for solving and controlling such systems. This work aims to develop a decoupling technique using a transfer function matrix for multiple-input, multiple-output multivariable systems. The proposed technique determines a compensator so the multivariate coupled system can become a set of multiple uncoupled single-input, single-output systems. With this transformation, each input only controls one output, and each output is controlled by a single input, facilitating any control solution. The final developed technique uses a computational methodology to achieve the proposed objective. To demonstrate the efficiency of the implemented algorithm, an experimental validation is performed through mathematical simulations of different test plants.

Index Terms— Multivariable systems, uncoupled systems, transfer function matrix.

Resumen—En los sistemas de control multivariables se disponen de varias entradas y varias salidas, donde cada entrada controla más de una salida y cada salida es controlada por más de una entrada, lo cual se denomina acoplamiento, que presenta complicaciones matemáticas para su resolución, lo que dificulta el control de este tipo de sistemas. Este trabajo tiene como objetivo el desarrollo de una técnica de desacoplamiento por matriz función de transferencia para sistemas multivariables de múltiples entradas y salidas, que consiste en determinar un compensador, de tal forma que el sistema acoplado multivariable pueda convertirse en un desacoplado de varios sistemas de una entrada y una salida. Esto permite que cada entrada controle solamente una salida y que cada salida sea controlada por una entrada; lo que facilita la solución del control. El sistema propuesto utiliza una metodología computacional para lograr el objetivo planteado. Para demostrar la eficacia del algoritmo propuesto se realiza una validación experimental mediante simulaciones matemáticas de diferentes plantas de prueba.

Palabras Claves—Sistemas multivariable, sistemas desacoplados, matriz de función de transferencia.

E. Granizo, P. Ayala. Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador (e-mails: egranizo@espe.edu.ec y jpayala@espe.edu.ec).

M. Barragán. Facultad de Ingeniería Eléctrica y Electrónica, Escuela Politécnica Nacional, Quito, Ecuador (e-mail: marco.barragan@epn.edu.ec).

I. INTRODUCCIÓN

EL DESARROLLO de técnicas para el diseño de sistemas de control multivariable es de práctica común para sistemas complejos. Un método particular de diseño, es aquel que implica el uso de realimentación para conseguir estabilidad de sistemas de control de lazo cerrado [4]. Con este método es a menudo de interés conocer si es posible o no lograr que las entradas controlen a las salidas independientemente, esto es, que una sola entrada influya en una sola salida (i.e., el tener un sistema desacoplado). Ya conseguido el desacoplamiento de un sistema multivariable MIMO (Multiple Input - Multiple Output), puede ser analizado mediante las técnicas clásicas para sistemas SISO (Single Input - Single Output) [1].

La problemática de desacoplar un sistema MIMO mediante el proceso analítico algebraico [1], es muy complicado y al mismo tiempo se alarga, a la vez que se aumenta el número de entradas que es igual al número de salidas, y más por supuesto, si se aumenta también el orden del sistema [2].

Una de las formas para determinar el "desacoplamiento" de un sistema multivariable, es la técnica matriz función de transferencia mediante el análisis algebraico [3], que para sistemas de dos entradas y dos salidas, son aplicaciones de fácil resolución; sin embargo, para sistemas de mayor número de entradas y salidas se recomienda el empleo de algoritmos computacionales, debido a que los elementos de las matrices son fracciones de polinomios, y por consiguiente, las operaciones matriciales normalmente son aplicadas a elementos numéricos [6]. Este trabajo desarrolla e implementa un programa computacional en lenguaje Pascal que determina un compensador de lazo cerrado que permite que el sistema acoplado multivariable pueda convertirse en un sistema desacoplado.

Con lo indicado anteriormente, un sistema multivariable de orden n , con m entradas y m salidas, en el que se asume $m \leq n$, se define la matriz cuadrada función de transferencia $G(s)$, que relaciona el vector de entrada m dimensional $u(s)$ y el vector de salida m dimensional $y(s)$, de la siguiente forma:

$$y(s) = \underline{G}(s)\underline{u}(s), \quad (1)$$

donde para que el sistema multivariable sea desacoplado, la

matriz función de transferencia $\underline{G}(s)$ debe ser diagonal y no singular [2].

Del estudio del estado del arte en esta temática, no se ha encontrado algoritmos computacionales implementados en lenguajes de computación de alto nivel, que solucionen el desacoplamiento de sistemas multivariables.

El artículo tiene la siguiente estructura: en la sección II se trata brevemente la técnica de desacoplamiento mediante función de transferencia. En la sección III se expone la propuesta del algoritmo computacional de desacoplamiento de sistemas lineales invariantes en el tiempo (Linear Time-Invariant, LTI). En la sección IV se muestran los resultados que se obtienen al aplicar la propuesta del algoritmo computacional a sistemas MIMO. Finalmente en la sección V se resumen las principales conclusiones de trabajo de investigación

II. DESACOPLAMIENTO DE SISTEMAS MIMO POR MATRIZ FUNCIÓN DE TRANSFERENCIA

El desacoplamiento de sistemas MIMO por este método, consiste en obtener una matriz compensadora $\underline{G}_c(s)$ de dimensión $(n \times n)$ serie a la función de transferencia de paso directo $\underline{G}_p(s)$, tal que la matriz del sistema de lazo cerrado sea diagonal [2], como se puede observar en la Fig. 1.

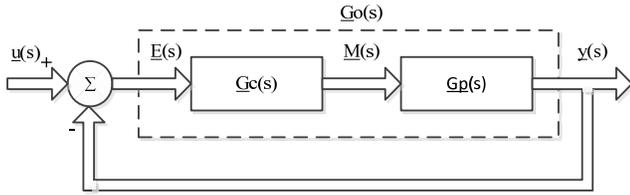


Fig. 1. Sistema multivariable de orden n con realimentación unitaria y compensador serie.

Donde $\underline{u}(s)$ es el vector de entrada n dimensional, $\underline{y}(s)$ es el vector de salida n dimensional, $\underline{E}(s)$ es el error entre el vector de entrada y el vector de salida, de orden n , $\underline{M}(s)$, es la nueva entrada de la planta $\underline{G}_p(s)$, $\underline{G}_o(s)$ es la matriz función de transferencia de paso directo cuando se añade un compensador y $\underline{G}_p(s)$ es la matriz función de transferencia de paso directo de dimensión $(n \times n)$ de una planta.

El proceso para obtener el compensador $\underline{G}_c(s)$ se muestra a continuación, se debe tener en cuenta que cada elemento de la matriz $\underline{G}_p(s)$ es un polinomio en el dominio de s y que I es la matriz identidad [2]:

$$\underline{G}_d(s) = \text{diag} [g_{d11}(s) \cdots g_{dnn}(s)], \quad (2)$$

$$\underline{G}_o(s) = \underline{G}_d(s) [I - \underline{G}_d(s)]^{-1}, \quad (3)$$

$$\underline{G}_c(s) = \underline{G}_p^{-1}(s) \underline{G}_o(s). \quad (4)$$

La matriz diagonal $\underline{G}_d(s)$ es conocida por imposición de diseño en el desacoplamiento de sistemas MIMO mediante función de transferencia, es decir, cada término de la diagonal de $\underline{G}_d(s)$ corresponde con las especificaciones de diseño para que el sistema cumpla con las características, tales como: estabilidad relativa, velocidad de respuesta, entre otras.

III. PROPUESTA DEL ALGORITMO COMPUTACIONAL DE DESACOPLAMIENTO

La propuesta del algoritmo computacional para el desacoplamiento de sistemas MIMO mediante función de transferencia, se resume en los siguientes aspectos: algoritmo computacional principal, diagrama de flujo resumido del algoritmo principal y las subrutinas utilizadas. En los siguientes literales se describe cada aspecto:

A. El algoritmo computacional principal

El algoritmo computacional principal propuesto para el desacoplamiento de sistemas MIMO mediante función de transferencia, se muestra a continuación:

- 1) Entrada del orden de la matriz $\underline{G}_p(s)$, que corresponde al número de entradas o salidas del sistema, que se define con $[n]$.
- 2) Entrada del mayor grado de los elementos de la planta $\underline{G}_p(s)$, luego de determinar el mínimo común múltiplo de sus denominadores, que se define con $[m]$.
- 3) Entrada del mayor grado del mínimo común múltiplo de los denominadores de $\underline{G}_p(s)$, que es el orden del sistema, y se define con $[t]$.
- 4) Entrada del mayor grado entre los numeradores y denominadores de los términos de la matriz diagonal $\underline{G}_d(s)$, que se define con $[m1]$.
- 5) Entrada de la matriz $\underline{G}_pL(s)$, de dimensión $(n \times n \times m)$ que es la matriz $\underline{G}_p(s)$ sin el mínimo común múltiplo de sus denominadores. m son planos que corresponden al grado de los polinomios de la matriz $\underline{G}_pL(s)$ en s .
- 6) Entrada del mínimo común denominador, $\delta(s)$, de orden t .
- 7) Entrada de la matriz diagonal $\underline{G}_d(s)$, que tiene dimensión $(n \times n \times m1)$. $m1$ es similar a m .
- 8) Determinar la matriz $\underline{G}_o(s)$, mediante:

$$\underline{G}_o(s) = \text{diag} \left[\frac{n_{11}(s)}{d_{11}(s) - n_{11}(s)} \cdots \frac{n_{nn}(s)}{d_{nn}(s) - n_{nn}(s)} \right],$$

donde:

Numerador $_i$ de $\underline{G}_o(s) = n_{ii}(s)$, $i = 1, 2, \dots, n$

Denominador $_i$ de $\underline{G}_o(s) = d_{ii}(s) - n_{ii}(s)$, $i = 1, 2, \dots, n$

- 9) Determinar la matriz $\underline{G}_pL^{-1}(s)$, mediante la subrutina: **MatrizInversaPolinomial**.
- 10) Determinar la matriz $\underline{G}_c(s)$, mediante el siguiente proceso:

$$\underline{G}_c(s) = \begin{bmatrix} \frac{\text{num}_{11}(s)}{\text{den}_1(s)} & & \frac{\text{num}_{1n}(s)}{\text{den}_n(s)} \\ & \cdots & \\ \frac{\text{num}_{n1}(s)}{\text{den}_1(s)} & & \frac{\text{num}_{nn}(s)}{\text{den}_n(s)} \end{bmatrix}$$

- 10.1) Calcular:

$$\text{nu}_j(s) = \delta(s) n_{jj}(s), \quad j = 1, 2, \dots, n$$

- 10.2) Calcular denominador de $\underline{G}_c(s)$,

$$\text{den}_j(s) = \text{dpl}(s) [d_{jj}(s) - n_{jj}(s)], \quad j = 1, 2, \dots, n.$$

Siendo: i = Fila de matriz.

j = Columna de matriz.

Donde $\text{dpl}(s)$, es mínimo común denominador de la

matriz $\underline{GpL}^{-1}(s)$

10.3) Calcular numerador de $\underline{Gc}(s)$,

$$num_{ij}(s) = nu_j(s) gl_{ij}(s), \quad i = 1, 2, \dots, n \\ , j = 1, 2, \dots, n$$

Donde $gl_{ij}(s)$, son los elementos de la matriz $\underline{GpL}^{-1}(s)$,
sin la relación: $\delta(s) / dpl(s)$

11) Obtener los compensadores PID,

$$P_k = num_{ij}(s) / den_j(s) \quad , i = 1, 2, \dots, n \\ , j = 1, 2, \dots, n \\ , k = -1, 0, 1, 2.$$

$$Gc_{ijk} = P_k \quad , i = 1, 2, \dots, n \\ , j = 1, 2, \dots, n \\ , k = -1, 0, 1, 2.$$

12) Imprimir el compensador serie $\underline{Gc}(s)$.

13) Fin.

B. Diagrama de flujo resumido del algoritmo principal

El diagrama de flujo resumido del algoritmo principal de la técnica desacoplamiento matriz función de transferencia, se muestra en la Fig. 2.

C. Las subrutinas utilizadas

Las subrutinas utilizadas en este método de desacoplamiento de sistemas MIMO, se describen a continuación:

Fig. 2. Diagrama de flujo resumido del algoritmo principal de la técnica desacoplamiento matriz función de transferencia.

1. Producto Polinomio Polinomio

La subrutina **PRproductoPolinomioPolinomio**, multiplica dos polinomios en el dominio de s , los polinomios pueden tener diferente grado. Esta subrutina no utiliza subrutinas adicionales.

Secuencia de llamada

PRproductoPolinomioPolinomio(GradoP1,GradoP2, Polinomio1,Polinomio2,Prod).

Definición de símbolos

GradoP1 = Grado del *Polinomio1*.

GradoP2 = Grado del *Polinomio2*.

Polinomio1 = Coeficientes del *Polinomio1*, *Polinomio1*[0] término constante.

Polinomio2 = Coeficientes del *Polinomio2*, *Polinomio2*[0] término constante.

Prod = Coeficientes del polinomio del producto de los dos Polinomios.

2. Reducir Grado

La subrutina **ReducirGrado**, elimina los coeficientes de mayor grado que son cero, de un polinomio en el dominio de s . En el cálculo computacional se considera un valor de cero, si no excede un valor de error, que en este caso se toma en el orden de 1×10^{-6} . Esta subrutina no utiliza subrutinas adicionales.

Secuencia de llamada

ReducirGrado(IR, XY,IXY).

Definición de símbolos

IR = Grado del polinomio original *XY*.

XY = Coeficientes del polinomio original *XY*, *XY*[0] es el término constante.

IXY = Grado del polinomio resultante, eliminado los coeficientes de mayor grado que son cero.

3. División Polinomios

La subrutina **DivisionPolinomios**, divide dos polinomios en el dominio de s , los polinomios pueden tener diferente grado. Esta subrutina utiliza la subrutina **ReducirGrado**.

Secuencia de llamada

DivisionPolinomios(IX,X,IY,Y,IP,P).

Definición de Símbolos

IX = Grado del polinomio *X*.

X = Coeficientes del arreglo *X*.

IY = Grado del polinomio *Y*.

Y = Coeficientes del arreglo *Y*.

IP = Grado del polinomio *P* resultante de la división de los dos Polinomios.

P = Coeficientes del arreglo *P* resultante de la división de los dos Polinomios.

4. Matriz Inversa Polinomial

La subrutina **MatrizInversaPolinomial**, determina la inversión de una matriz polinomial en el dominio de s . La matriz inversa está compuesta por un polinomio denominador y una matriz tridimensional, donde cada uno de los planos de la matriz corresponde al respectivo grado polinomial. Esta

subrutina utiliza las subrutinas: **ProductoMatricesPolinomiales**, **PRproductoMatrizPolinomio** y **PRproductoPolinomioPolinomio**; las dos primeras subrutinas se describen en los siguientes numerales y la última subrutina está descrita en el primer numeral.

Secuencia de llamada

MatrizInversaPolinomial(GpL,Denominador).

Definición de símbolos

GpL = Elementos de la Matriz *GpL*, *GpL*[$i,j,0$] matriz constante.

Denominador = Coeficientes del Polinomio *Denominador*, *Denominador*[0] término constante.

5. Producto Matriz Polinomio

Esta subrutina **PRproductoMatrizPolinomio** realiza la multiplicación de una matriz polinomial con un polinomio, en el dominio de s , cuyo resultado es también una matriz polinomial. Esta subrutina no utiliza subrutinas adicionales.

Secuencia de llamada

ProductoMatrizPolinomio(Fila,Columna,GradoM, GradoP,Matriz,Polinomio, Produc).

Definición de símbolos

Fila = Número de filas de la *Matriz*.

Columna = Número de columnas de la *Matriz*.

GradoM = Mayor grado de los elementos de la *Matriz* original.

GradoP = Grado del polinomio.

Matriz = Elementos de la *Matriz*, *Matriz*[$i,j,0$] matriz constante.

Polinomio = Coeficientes del *Polinomio*, *Polinomio*[0] término constante.

Produc = Elementos del resultado del producto Matriz-Polinomio.

6. Producto Matrices Polinomiales

La subrutina **PRproductoMatricesPolinomiales** multiplica dos matrices polinomiales en el dominio de s , las matrices pueden tener diferente grado. Esta subrutina no utiliza subrutinas adicionales.

Secuencia de llamada

PRproductoMatricesPolinomio(FilaM1,ColumnaM2, FilaColumna,GradoM1,GradoM2, MatrizM1,MatrizM2,Producto).

Definición de símbolos

FilaM1 = Número de filas de la *MatrizM1*.

ColumnaM2 = Número de columnas de la *MatrizM2*.

FilaColumna = Número de columnas de la *MatrizM1*, igual al número de filas de *MatrizM2*.

GradoM1 = Mayor grado de los elementos polinomiales de la *MatrizM1*.

GradoM2 = Mayor grado de los elementos polinomiales de la *MatrizM2*.

MatrizM1 = Elementos de la *Matriz1*, *MatrizM1*[$i,j,0$] matriz constante.

MatrizM2 = Elementos de la *Matriz2*, *MatrizM2*[$i,j,0$] matriz

constante.

Producto = Elementos del resultado del producto de matrices.

IV. RESULTADOS AL APLICAR EL ALGORITMO COMPUTACIONAL

El programa computacional propuesto es muy amigable y de fácil uso, porque está implementado en lenguaje Pascal en modo consola, con la ventaja que puede ser implementado en cualquier Sistema Operativo de Windows. El limitante de este programa consiste en que no se puede utilizar en el caso de matrices no invertibles de la planta.

La matriz polinomial de la planta $\underline{G}_p(s)$ debe estar expresada de la siguiente forma:

$$\underline{G}_p(s) = \left[\frac{1}{DELTA} \right] \underline{G}_p L,$$

donde $\underline{G}_p L$ es la matriz polinomial en el dominio de s del numerador de la planta $\underline{G}_p(s)$ y DELTA es el polinomio mínimo común múltiplo de los denominadores de los elementos de la planta $\underline{G}_p(s)$.

Para introducir los datos de la matriz $\underline{G}_p L$, debe tenerse en cuenta que esta matriz tiene tres dimensiones, donde la tercera dimensión representa las matrices o planos que corresponden al grado creciente de los polinomios en el dominio de s .

Se recomienda que los valores de las entradas de datos: grado de la matriz $\underline{G}_p(N)$, mayor grado de los elementos de la matriz \underline{G}_p una vez encontrado el mínimo común múltiplo de los denominadores (M), y mayor grado de los numeradores y denominadores de los elementos de la matriz diagonal $\underline{G}_d(M1)$; no sobrepasen del valor de 10, debido a que el ingreso de la información es demasiado extenso y los resultados no se presentan de forma clara.

A continuación se presenta un ejemplo de aplicación que tiene el fin de comprobar que el algoritmo computacional cumple con el método matemático. Cabe destacar que el algoritmo computacional puede resolver ejercicios de grado mayor a dos, porque el programa computacional puede aplicarse, no sólo a sistemas simples, sino también a sistemas complejos; cuyo desarrollo algebraico se tornaría largo y difícil, o inclusive impracticable.

Sea el sistema MIMO de la Fig. 3, que está conformado por un sistema de dos entradas y dos salidas, con una matriz compensadora serie cuyos elementos son: gc_{11} , gc_{21} , gc_{12} y gc_{22} .

Se determina la matriz del compensador serie $\underline{G}_c(s)$ del sistema de la Figura 3, tal que la matriz de la función de transferencia de lazo cerrado $\underline{G}_d(s)$, debe ser diagonal por diseño para que el sistema sea desacoplado, es decir, el sistema tenga cada salida dependiente únicamente de una entrada:

$$\underline{G}_d(s) = \begin{bmatrix} \frac{1}{s+1} & 0 \\ 0 & \frac{1}{5s+1} \end{bmatrix} \quad (5)$$

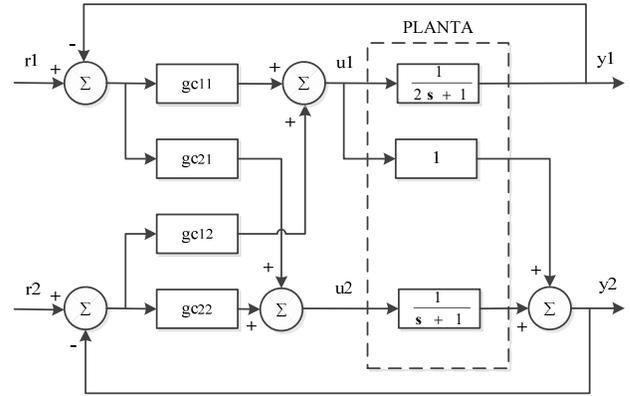


Fig. 3. Sistema de dos entradas y dos salidas, con compensador serie.

Por lo que al aplicar el método matemático descrito en [2], se obtiene la matriz del compensador serie $\underline{G}_c(s)$ añadido a la planta $\underline{G}_p(s)$:

$$\underline{G}_c(s) = \begin{bmatrix} 2 + s^{-1} & 0 \\ -2s - 3 - s^{-1} & 0.2 + 0.2s^{-1} \end{bmatrix} \quad (6)$$

Los elementos de la matriz (6) del compensador serie $\underline{G}_c(s)$ corresponden a los compensadores indicados en la Figura 3, los cuales tienen las estrategias de control proporcional, integral y derivativo (PID): $gc_{11}(s) = 2 + s^{-1}$ y $gc_{22}(s) = 0.2 + 0.2s^{-1}$ es control proporcional e integral; $gc_{21}(s) = -2s - 3 - s^{-1}$ es control proporcional, integral y derivativo; y $gc_{12}(s) = 0$ indica no hay compensador.

En el análisis realizado no se consideran perturbaciones externas, y además se producen cancelaciones en el numerador y el denominador de $\underline{G}_c(s)$ y $\underline{G}_p(s)$, entonces aunque se han conseguido los resultados deseados de ausencia de interacciones entre una entrada y varias salidas, se debe analizar cuando se producen perturbaciones en el sistema por fuerzas externas, porque el sistema podría hacerse "incontrolable" por dichas cancelaciones.

Por otro lado, al aplicar el algoritmo computacional propuesto en la sección III, se obtiene el compensador serie GC de la siguiente manera:

COMPENSADOR SERIE GC

Compensador PID (1,1)

Grado:-1 1.000

Grado: 0 2.000

Compensador PID (2,1)

Grado:-1 -1.000

Grado: 0 -3.000

Grado: 1 -2.000

Compensador PID (1,2)

Grado (0): 0 *No hay compensador

Compensador PID (2,2)

Grado:-1 0.200

Grado: 0 0.200

El resultado del programa computacional propuesto muestra la matriz del compensador serie GC por columnas, de la siguiente forma: $gc_{11}(s)$, $gc_{21}(s)$, $gc_{12}(s)$ y $gc_{22}(s)$; donde cada elemento es un polinomio en el dominio s que se le aproxima a un controlador PID, debido a que el objetivo de este método de desacoplamiento es tener un controlador que se implemente mediante un control Proporcional, Integral y Derivativo. Este resultado del compensador serie GC expresado en forma matricial, sería:

$$\underline{G}_C(s) = \begin{bmatrix} 2 + s^{-1} & 0 \\ -2s - 3 - s^{-1} & 0.2 + 0.2s^{-1} \end{bmatrix} \quad (7)$$

Al comparar las matrices (6) y (7), se puede comprobar que existe correspondencia entre el cálculo de la solución del método matemático y el resultado que se obtiene con el algoritmo propuesto. Se debe tener en cuenta que el programa computacional propuesto muestra los datos en punto flotante, con tres cifras decimales.

Por último, se debe tener en cuenta cuando un elemento de la matriz del compensador G_C puede ser inadecuado, debido a que no se ajusta a un controlador PID, ya que el grado del numerador es mucho menor al grado del denominador, en dos o más; por lo que el programa computacional mostrará el mensaje "No hay aproximación". También, hay que tener en cuenta cuando un elemento de la matriz G_C puede ser cero, lo que implica que no existe compensador, el mensaje que mostrará el programa es "No hay compensador". Además, el programa mostrará el mensaje "ERROR está mal dada la matriz diagonal G_d ", cuando la matriz diagonal G_d es singular y uno de sus elementos es igual a cero.

V. CONCLUSIONES

Para darle significado físico al resultado del compensador serie GC , cada elemento de la matriz es aproximado a un controlador PID, y debido a que las ecuaciones descritas son muy simples para desacoplar un sistema, permiten una fácil resolución algebraica para un sistema de dos entradas y dos salidas; sin embargo, ya que el proceso analítico es algebraico, el mismo que se puede alargar y complicar, a la vez que se aumenta el número de entradas, salidas y el orden del sistema; por lo que se hace necesario el contar con un algoritmo computacional como el propuesto. Esta técnica de desacoplamiento mediante un algoritmo computacional, solo sirve para sistemas lineales invariantes en el tiempo.

Cabe indicar que existen otros métodos de desacoplamiento de sistemas MIMO, tales como: realimentación de estado y realimentación de salida; los cuales serán presentados en próximos artículos.

RECONOCIMIENTOS

La presente investigación se realizó con el apoyo de la Escuela Politécnica Nacional y la Universidad de las Fuerzas Armadas ESPE. Un reconocimiento especial a los profesores: Dr. Diego Arcos Avilés y Dr. Vinicio Carrera Erazo, por su colaboración durante la redacción del artículo.

REFERENCIAS

- [1] Granizo, Evelio; "Desacoplamiento para Sistemas Continuos en el Tiempo"; Tesis; EPN; Quito; 1988.
- [2] Montgomerie G. A. and Nicholson H.; "Modern Approaches to Control System Desing"; Edited by Munro N; Published by The Institution of Electrical Engineers; London; 2015.
- [3] Chen, Chi Tsong; "Introduction to Linear System Theoryc"; Editorial, Holt Rinehart and Winston, Inc; United States of America; 1970.
- [4] Ogata, Katsuhiko; "Ingeniería de Control Moderno"; Cuarta Edición; Editorial Prentice - Hall Internacional; Buenos Aires; 2008.
- [5] Melsa, James L. and JONES, Stephen K.; "Computer Programs for Computational Assistang in the Study of Linear Control Theory"; Second Edition; Editorial McGraw - Hill Book Company; United States of America; 1973.
- [6] Frank, Ayres; "Matrices"; Colección Shaum; Editorial Mcgraw-Hill; México; 1992.
- [7] Granizo, Evelio; "Programación Estructurada en Seudo-códigos, Teoría Y Ejercicios"; Quinta Edición; Quito; 2017.