

MODELO DE DATOS Y MODELO AERODINÁMICO 6-DOF DE AERONAVES PARA SIMULACION DE VUELO EN TIEMPO REAL

Javier E. Luiso^a, Pedro S. Giraudó^b, Eduardo N. Zapico^b y Horacio A. Abbate^a

^a*División Computación Gráfica y Visualización, Instituto de Investigaciones Científicas y Técnicas para la Defensa (CITEDEF), Villa Martelli, Buenos Aires, Argentina,
jluiso@citefa.gov.ar, habbate@citefa.gov.ar
<http://www.citefa.gov.ar>*

^b*Departamento de Aeronáutica, Universidad Nacional de Córdoba, Córdoba, Argentina,
pedrogiraudó@yahoo.com, ezapico@efn.uncor.edu
deptaeron@efn.uncor.edu*

Palabras clave: simulador de vuelo, modelo aerodinámico, motor de simulación

Resumen. El propósito de este trabajo es presentar el modelo de datos correspondiente a un modelo aerodinámico de seis grados de libertad desarrollado para simular el vuelo de una aeronave, con el objetivo de simular en tiempo real el comportamiento dinámico de vuelo. Así mismo se describe el funcionamiento del motor de simulación que opera sobre el modelo de datos presentado.

Como es usual, la aerodinámica de una aeronave es definida mediante un conjunto parámetros adimensionales conocidos como coeficientes aerodinámicos, que son utilizados para calcular las fuerzas y momentos de origen aerodinámico que actúan sobre la aeronave.

En particular los parámetros aerodinámicos son afectados por la velocidad de la aeronave, los ángulos relativos de la misma respecto del viento, la altura de vuelo y la deflexión de las superficies de control. Es por esto que para lograr una simulación que se aproxime a la realidad es necesario contar con un gran volumen de información. En el presente trabajo toda esa información es calculada mediante métodos analíticos. En su mayoría se ha utilizando el USAF Stability and Control Datcom (en versión digital). Este procedimiento probó ser rápido pero limitado debido a su incapacidad para estimar los efectos por deslizamiento de la aeronave.

El conjunto de datos aerodinámicos que forman parte del modelo fue generado mediante los métodos de análisis, y ha sido organizada de forma tal que del acceso y manejo de la misma sea de óptima performance durante la simulación en tiempo real dentro del motor de simulación. La solución adoptada y presentada en este artículo se basa en el diseño de un modelo de datos de estructura matricial que permite utilizar algoritmos de búsqueda rápidos y eficientes como también mecanismos de interpolación simples. Se eligió la configuración de la base de datos en matrices multidimensionales compuestas de coeficientes heterogéneos. Dependiendo del tipo de coeficientes, las matrices pueden tener dos, tres y hasta cuatro dimensiones. Se presenta el método de cálculo utilizado y los resultados obtenidos.

Como parte final del trabajo se presentan los resultados de ensayos realizados en el motor de simulación sobre el modelo aerodinámico de 6 grados de libertad del avión IA-63 Pampa.

1 INTRODUCCIÓN

La aerodinámica de cualquier aeronave es definida mediante un grupo de parámetros adimensionales conocidos como coeficientes aerodinámicos. Ellos dependen de la velocidad de vuelo (número de mach) y de los ángulos aerodinámicos α y β (ataque y deslizamiento). Estos coeficientes, son usados para calcular las fuerzas y momentos aerodinámicos que actúan sobre la aeronave. El modelo incluye coeficientes longitudinales, derivativas estáticas y dinámicas e incrementos de los mismos debido a la deflexión de los comandos de la aeronave.

El Motor de Simulación (Abbate et al, 2007; Luiso et al, 2008), que opera sobre el modelo de datos es un componente de software concebido y diseñado con el objetivo de proveer una herramienta de simulación de propósito general capaz de ejecutar algoritmos numéricos. Dichos algoritmos modelan el comportamiento físico de un determinado sistema de interés. Al ejecutar estos algoritmos el motor de simulación calcula el comportamiento físico de los mecanismos que forman parte del sistema en estudio.

El principal antecedente de nuestro sistema es la plataforma JSBSim desarrollada por Berndt (Berndt et al, 2004; Jackson et al, 2002), esta plataforma consiste en un motor de simulación, abierto, configurable y escalable dedicado a modelos dinámicos de vuelo; y el proyecto LaSRS de la NASA (Leslie, R. A, 1998).

Este trabajo se realiza en el marco del proyecto “Simulador de Vuelo” financiado con el subsidio PAE2004-22614 ANPCyT (Agencia Nacional de Promoción Científica y Tecnológica).

2 MODELO AERODINÁMICO DIGITAL

El modelo aerodinámico está basado en la aeronave IA 63 PAMPA, de fabricación nacional, cuyas características principales son las siguientes:

Avión Jet para entrenamiento básico y ataque liviano

Ala alta recta con diedro negativo.

Perfil alar Dornier DoA-7/-8 transónico avanzado.

Flaps Fowler simple ranura, plano de cola completamente móvil.

Cabina con configuración tandem y asiento trasero elevado.

Motor turbo fan Honeywell TFE-731-2C, con 15,57 kN de empuje

Máximo Mach operacional: 0.8, techo de servicio: 12900m, límite g: +6.0/-3.0.

2.1 Estructura de datos

Como se mencionó anteriormente, el comportamiento aerodinámico de la aeronave se describe mediante coeficientes aerodinámicos que se utilizan para calcular fuerzas y momentos y como consecuencia de ellos, aceleraciones lineales y angulares.

Para obtener un modelo aerodinámico realista de una aeronave es necesario contar con una gran cantidad de información. Por ejemplo para un modelo simple como el del ejemplo varios coeficientes son calculados para 14 ángulos de ataque, 4 números de Mach y 4 alturas, lo que hace un total de 224 valores diferentes por coeficiente. Otros coeficientes además son calculados para diferentes ángulos de deflexión de comandos por lo que el número final se multiplica. Como consecuencia de esto la base de datos generada para una sola aeronave es de un volumen importante lo que hace necesario contar con un buen diseño y organización capaz de facilitar los mecanismos de búsqueda y acceso (Zapico, E., 2008b, a).

2.2 Modelo de la aeronave IA 63 PAMPA

Los coeficientes aerodinámicos que forman parte de la base de datos utilizada en el primer modelo digital desarrollado para esta aeronave son:

- Coeficientes longitudinales C_D , C_L , C_m , C_N , y C_A
- Derivativas estáticas $C_{L\alpha}$, $C_{m\alpha}$, $C_{y\beta}$, $C_{n\beta}$, $C_{l\beta}$
- Derivativas dinámicas C_{Lq} , C_{mq} , $C_{L\dot{\alpha}d}$, $C_{m\dot{\alpha}d}$, C_{lp} , C_{yp} , C_{nr} , C_{lr}
- Incrementos por deflexion de superficies de control $D(C_L)$, $D(C_m)$, $D(C_{Lmax})$, $D(C_{Dmin})$, $D(C_{DI})$, $D(C_l)$ (roll), $D(C_N)$ (yawing).
- Incrementos por efecto suelo.

Cada uno de estos coeficientes fueron calculados para valores de Mach de 0.2, 0.4, 0.6 y 0.8 y para valores de altura de 0, 5000, 10000 y 15000 metros.

La estructura final de los datos para cada coeficiente será en función de las dependencias sobre los ángulos aerodinámicos (α y β), la velocidad y altura. (ver [Figura 2](#) y [Figura 3](#))

El modelo de datos permite modificar la dependencia de un coeficiente en particular reduciendo la dimensión de la base de datos asociada. Esto es útil, por ejemplo para estudiar un caso particular del comportamiento de la aeronave a valores fijos de Mach y altura. Por ejemplo para el coeficiente de sustentación (C_L) la base de datos se reduce en dos dimensiones. (ver [Figura 1](#))

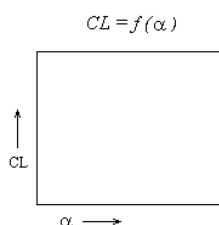


Figura 1: Base de datos de 2 dimensiones para el coeficiente C_L

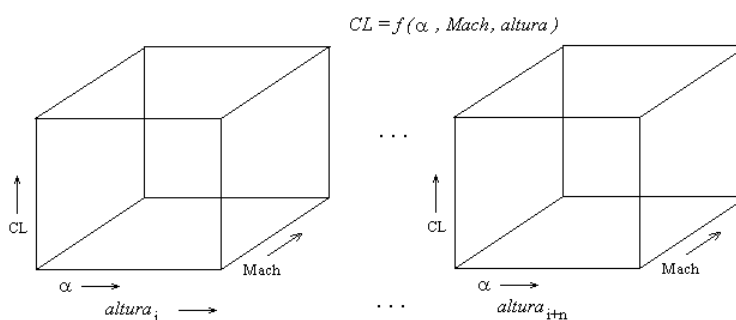


Figura 2: Base de datos de 3 dimensiones

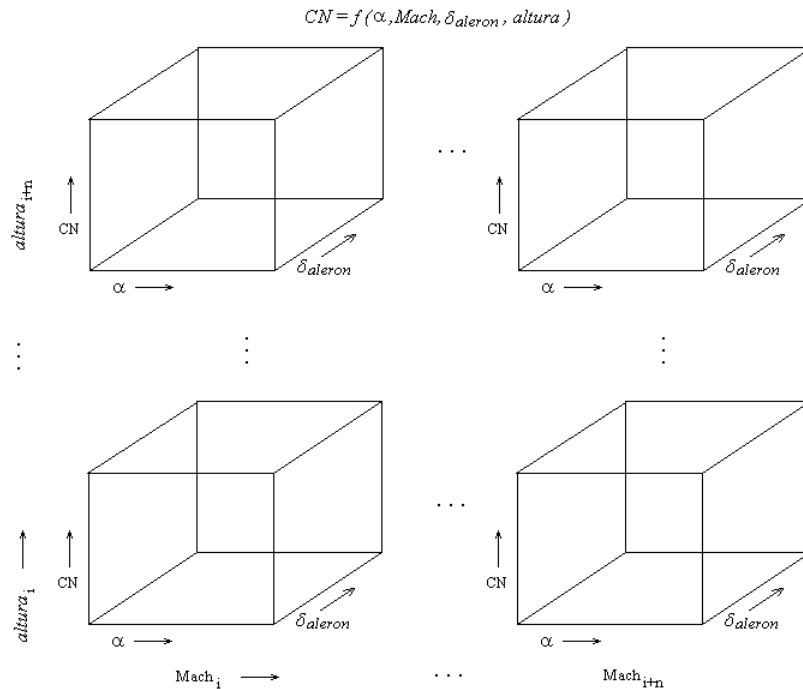


Figura 3: Base de datos de dimensión 4

2.3 Cálculo de fuerzas y momentos

Para una condición de vuelo determinada el modelo calcula los coeficientes aerodinámicos específicos aplicando funciones de interpolación cuya dimensión dependerá de las dimensiones de la base de datos utilizada.

Una vez interpolados todos los coeficientes las fuerzas y momentos son calculados construyendo en primer lugar los coeficientes aerodinámicos principales teniendo en cuenta las contribuciones de los efectos reflejados en las tablas de datos aerodinámicos:

- Coeficientes plano longitudinal (fuerza de resistencia, fuerza de sustentación y momento de cabeceo. (ver Eq. (1))
- Coeficientes plano lateral-direccional (fuerza lateral, momento de guiñada y momento de rolido. (ver Eq. (2))

$$\begin{bmatrix} CD \\ CL \\ CM \end{bmatrix} = \begin{bmatrix} CD_{basic} & 0 & 0 & CD_{\partial elev} \\ CL_{basic} & CL_q & CL_{d\alpha/dt} & CL_{\partial elev} \\ CM_{basic} & CM_q & CM_{d\alpha/dt} & CM_{\partial elev} \end{bmatrix} * \begin{bmatrix} 1 \\ Q.c \\ (d\alpha/dt).c \\ \partial elev \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} CY \\ CN \\ Cl \end{bmatrix} = \begin{bmatrix} CY_{\beta} & CY_p & 0 & 0 & CY_{\partial rud.} \\ CN_{\beta} & CN_p & CN_r & CN_{\partial ail.} & CN_{\partial rud.} \\ Cl_{\beta} & Cl_p & Cl_r & Cl_{\partial ail.} & Cl_{\partial rud.} \end{bmatrix} * \begin{bmatrix} \beta \\ P.b \\ R.b \\ \partial aler. \\ \partial rud. \end{bmatrix} \quad (2)$$

Luego se calculan las fuerzas (ver Eq. (3)) y los momentos (ver Eq. (4)).

$$\begin{bmatrix} D \\ L \\ Y \end{bmatrix} = \begin{bmatrix} CD \\ CL \\ CY \end{bmatrix} * \left[\frac{1}{2} \rho V^2 S \right] \quad (3)$$

$$\begin{bmatrix} l \\ M \\ N \end{bmatrix} = \begin{bmatrix} Cl.b \\ CM.c \\ CN.b \end{bmatrix} * \left[\frac{1}{2} \rho V^2 S \right] \quad (4)$$

Los valores P , Q y R representan las velocidades angulares, $(d\alpha/dt)$ es la derivada del ángulo de ataque, b , c y S son parámetros geométricos. ∂_{elev} , ∂_{aler} . y ∂_{rud} son las deflexiones de los comandos, ρ es la densidad del aire y V la velocidad de la aeronave respecto del aire.

3 MOTOR DE SIMULACIÓN

Sin importar las características del sistema a simular (ej.: avión volando, un vehículo terrestre desplazándose, barco navegando), éste se modela como una entidad que posee una entrada (parámetros externos que influyen sobre el sistema), una salida (parámetros afectados por el comportamiento del sistema) y una función de transferencia asociada. La función de transferencia del sistema define cómo se calcula en un ciclo de simulación el nuevo valor de la salida y el nuevo estado del sistema en función del estado corriente del sistema y del valor corriente de la entrada. Dependiendo de la complejidad del sistema, éste a su vez podrá estar conformado por módulos independientes interconectados entre si, llamados componentes (ver Figura 4). Para cada uno de estos componentes, que también son representados como entidades que poseen entradas, salidas y funciones de transferencia, existirá un modelo que simule su comportamiento. Toda esta estructura es especificada en archivos basados en XML (Berndt et al, 2004).

Durante la ejecución el Motor no sólo resuelve la simulación de cada uno de los modelos sino que, como conoce la forma en que los componentes se relacionan entre sí, determina el comportamiento del sistema completo. La evolución de la simulación se desarrolla en forma discreta en el tiempo, es decir se ejecutan ciclos periódicos de simulación y en cada uno se determina el aporte de cada componente a la simulación, y luego se integran los resultados obteniéndose el efecto total sobre el conjunto.

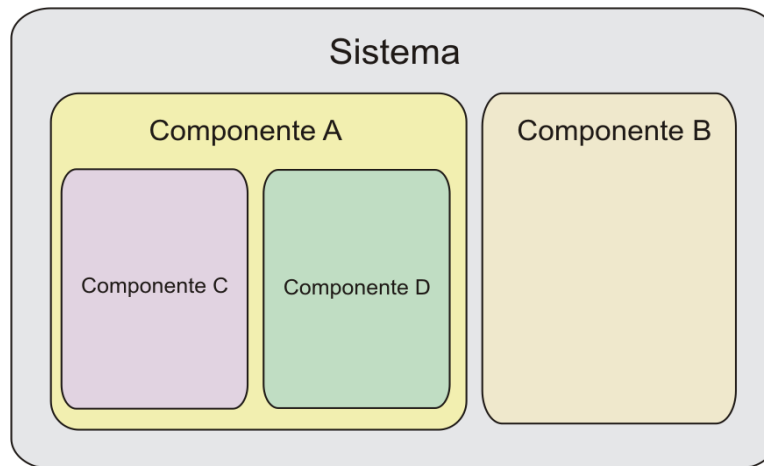


Figura 4. Sistema y sus componentes internos

3.1 Estructura

El Motor de Simulación ha sido estructurado en tres componentes principales: Biblioteca de componentes propios, el Interpretador XML y el módulo de Configuración y Control (ver Figura 5).

Tanto el Archivo de Especificación del Modelo como los Archivos de Inicialización de Componentes utilizan el formato XML. El Interpretador XML es el encargado de interpretar dichos archivos. En el primero de ellos se describe el sistema a simular declarando los componentes que lo constituyen y especificando las conexiones entre entradas y salidas de los componentes, y en los Archivos de Inicialización de Componentes se describen los conjuntos de datos requeridos por ciertos componentes para la implementación computacional de sus respectivas funciones de transferencia.

El módulo de Configuración y Control ofrece una interfaz que permite configurar parámetros propios de la simulación: establecer el paso de discretización, definir un tiempo de simulación y llevar adelante la simulación hasta que ese tiempo se cumpla, realizar simulaciones paso a paso, etc. Además brinda acceso al Estado de Sistema en cualquier momento, permitiendo incluso modificarlo sin tener que detener la simulación. Entendiéndose como Estado del Sistema al conjunto de magnitudes continuas y discretas, que caracterizan en cada momento la situación del mismo.

La Biblioteca de Componentes Básicos es un conjunto de componentes genéricos disponibles para ser aprovechados por el usuario en el diseño de modelos. En el proceso de describir el sistema como un conjunto de componentes interconectados, el usuario puede hacer uso de esta biblioteca simplificando el trabajo. También se dispone de un mecanismo simple y poderoso que permite el diseño e incorporación de nuevos componentes en función de las necesidades.

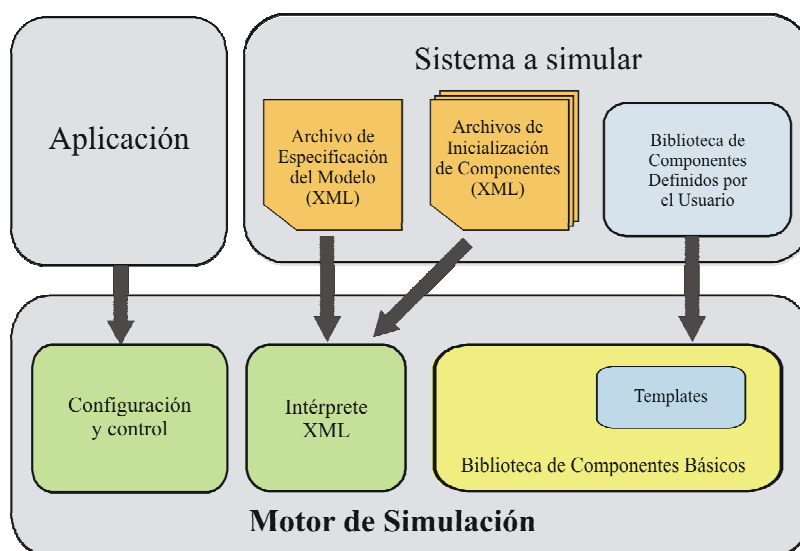


Figura 5. Estructura del Motor de Simulación

3.2 Componentes definidos por el usuario

La complejidad o especificidad de un subsistema puede implicar que no sea posible modelar su función de transferencia con los componentes básicos, o que el usuario desee emplear algún algoritmo numérico particular, en estos casos el usuario diseñará un componente propio.

El Motor de Simulación provee un mecanismo simple y poderoso para que el usuario pueda diseñar e implementar componentes externos en función de las necesidades que surjan al modelar un sistema particular, cuando por ejemplo una función de transferencia no pueda ser modelada con los componentes básicos del motor.

El nuevo componente implementa una función de transferencia propia y particular definida por el usuario. Sus entradas y salidas pueden interconectarse con componentes propios del motor y/o otros componentes definidos por el usuario.

Todos los componentes que forman parte del modelo de un sistema, ya se que se trate de componentes propios del motor y/o componentes definidos por el usuario, durante la ejecución de dicho modelo pasan por las siguientes etapas:

Creación: a medida que el Intérprete XML procesa el archivo de especificación las distintas instancias de los componentes presentes en el sistema son creadas.

Interconexión: el archivo de especificación también contiene la información que determina el modo en que los distintos componentes se conectan entre sí. Una vez que todas las instancias han sido creadas la interconexión entre todas ellas es realizada.

Inicialización: antes de realizar el primer ciclo de simulación, el sistema en general y cada componente en particular es inicializado.

Ciclo de Simulación: en cada ciclo se ejecuta la función de transferencia del componente para determinar su aporte a la simulación.

4 IMPLEMENTACIÓN DEL CALCULO MATRICIAL DENTRO DEL MOTOR DE SIMULACIÓN

4.1 Tipos de datos multidimensionales

El motor de simulación posee un conjunto de componentes propios diseñados para el manejo de datos vectoriales, matriciales y de estructura cúbica (matriz multidimensional de dimensión 3). Un conjunto de herramientas que también forman parte del motor implementan mecanismos para la carga y acceso a los datos, de forma individual o como conjuntos de datos. También se implementan las operaciones aritméticas correspondientes.

4.2 Componentes del motor que resuelven la interpolación

Como se describe en el punto 2.3 los coeficientes aerodinámicos se calculan aplicando funciones de interpolación sobre los datos vectoriales, matriciales y multidimensionales (cúbicos) respectivamente.

El motor de simulación posee componentes propios para resolver interpolaciones sobre conjuntos de datos multidimensionales. Dichos componentes se utilizan para la resolver el cálculo matricial que requiere este modelo dinámico.

En el proceso de implementar el modelo de vuelo en el motor, los coeficientes aerodinámicos para el modelo IA 63 PAMPA fueron agrupados según la dependencia respecto de los ángulos aerodinámicos (α y β), la velocidad y altura. En función de esta dependencia será el tipo de interpolación necesaria para realizar el cálculo (ver Figuras 1, 2 y 3) y por lo tanto el tipo de componente propio del motor a emplear.

Al agrupar de este modo, los coeficientes aerodinámicos es posible utilizar un único componente para cada tipo de interpolación y poder realizar el cálculo aritmético de todos los coeficientes en un ciclo de simulación.

4.3 Interpolación 1D

Para el cálculo de un coeficiente dependiente solo de un parámetro, el cual requerirá una interpolación 1D se utilizan N datos organizados en un vector. Establecido el rango del parámetro de interpolación para el que están calculados los datos y el paso en entre ellos, el procedimiento de cálculo implica primero determinar a partir del valor del parámetro de interpolación las dos coordenadas consecutivas del vector a utilizar y luego previa normalización del parámetro, se realiza la interpolación lineal. (ver Figura 6, Eq. (5))

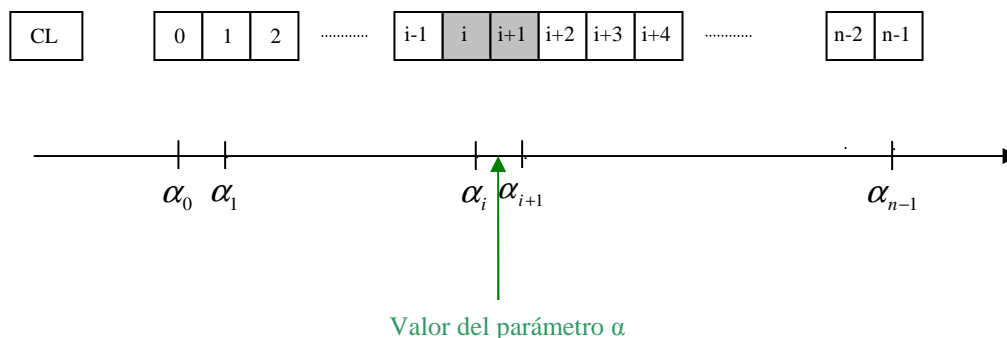


Figura 6: Interpolación lineal sobre un vector de datos

$$\alpha_{norm} = \frac{\alpha - \alpha_i}{\alpha_{i+1} - \alpha_i} ; \text{normalización del parámetro de interpolación} \quad (5)$$

$$cl = \alpha_{norm} CL[i] + (1 - \alpha_{norm}) CL[i + 1] ; \text{cálculo del coeficiente } cl$$

El componente ScalarInterpolator1D es un componente propio del motor que trabaja sobre una matriz de datos donde cada columna se corresponde con los N datos asociados a un coeficiente particular que serán utilizados para el cálculo de la interpolación de ese componente. De este modo en un ciclo de simulación se determina una única vez, a partir del valor del parámetro de interpolación, las dos coordenadas consecutivas del vector a utilizar para luego realizar el calculo de todos los coeficientes de una vez, calculando la interpolación lineal directamente sobre las filas de la matriz. (ver Figura 7, Eq (6))

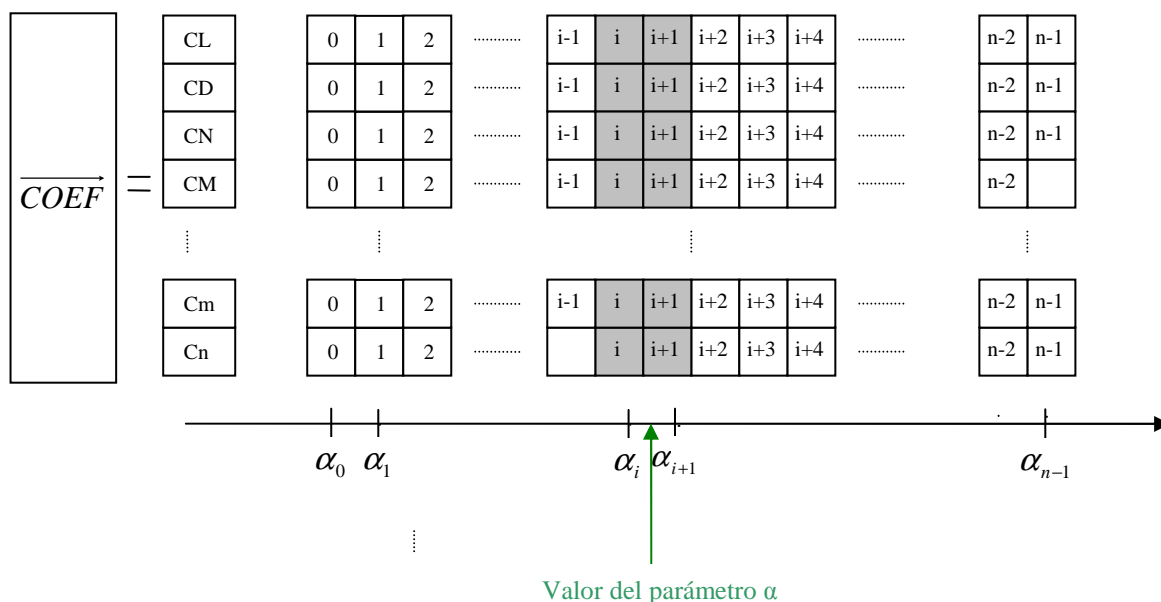


Figura 7: Interpolación lineal sobre múltiples vectores de datos

$$\alpha_{norm} = \frac{\alpha - \alpha_i}{\alpha_{i+1} - \alpha_i} ; \text{normalización del parámetro de interpolación} \quad (6)$$

$$\overline{coef} = \alpha_{norm} \overline{COEF}[i] + (1 - \alpha_{norm}) \overline{COEF}[i + 1] ; \text{cálculo de los coeficientes}$$

4.4 Interpolación 2D

Para el cálculo de un coeficiente dependiente de dos parámetros, se requerirá una interpolación 2D que utiliza un matriz de NxM datos. En este caso un valor particular, ambos parámetros determinan dos filas consecutivas y dos columnas también consecutivas a partir de las cuales se extraen los 4 datos a emplear para el cálculo del coeficiente. (ver Figura 8, Eq. (7))

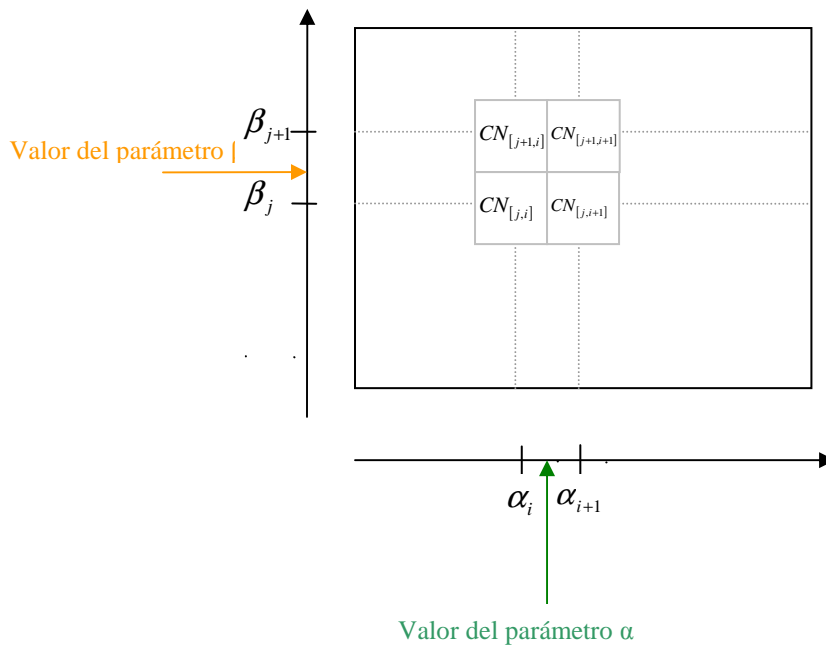


Figura 8: Interpolación lineal sobre una matriz de datos

$$\begin{aligned}
 \alpha_{norm} &= \frac{\alpha - \alpha_i}{\alpha_{i+1} - \alpha_i} ; \text{normalización de los parámetros de interpolación} \\
 \beta_{norm} &= \frac{\beta - \beta_j}{\beta_{j+1} - \beta_j} \\
 cn &= \alpha_{norm} \beta_{norm} CN[j, i] + (1 - \alpha_{norm}) \beta_{norm} CN[j, i+1] + \\
 &\quad \alpha_{norm} (1 - \beta_{norm}) CN[j+1, i] + \\
 &\quad (1 - \alpha_{norm})(1 - \beta_{norm}) CN[j+1, i+1] ; \text{cálculo del coeficiente } cn
 \end{aligned} \tag{7}$$

El componente ScalarInterpolator2D, otro componente propio del motor, trabaja sobre una matriz de datos de tres dimensiones (cubo de datos). Cada slice del cubo se corresponde con una matriz de datos de dimensiones NxM. Cada slice estará asociado a un coeficiente particular. De este modo en un ciclo de simulación se determina una única vez, a partir de los valores de los parámetros de interpolación, las dos filas y columnas a utilizar. Luego el cálculo de todos los coeficientes se realiza en un mismo paso, calculando la interpolación 2D en cada slice del cubo de datos. (ver [Figura 9](#), Eq. (8))

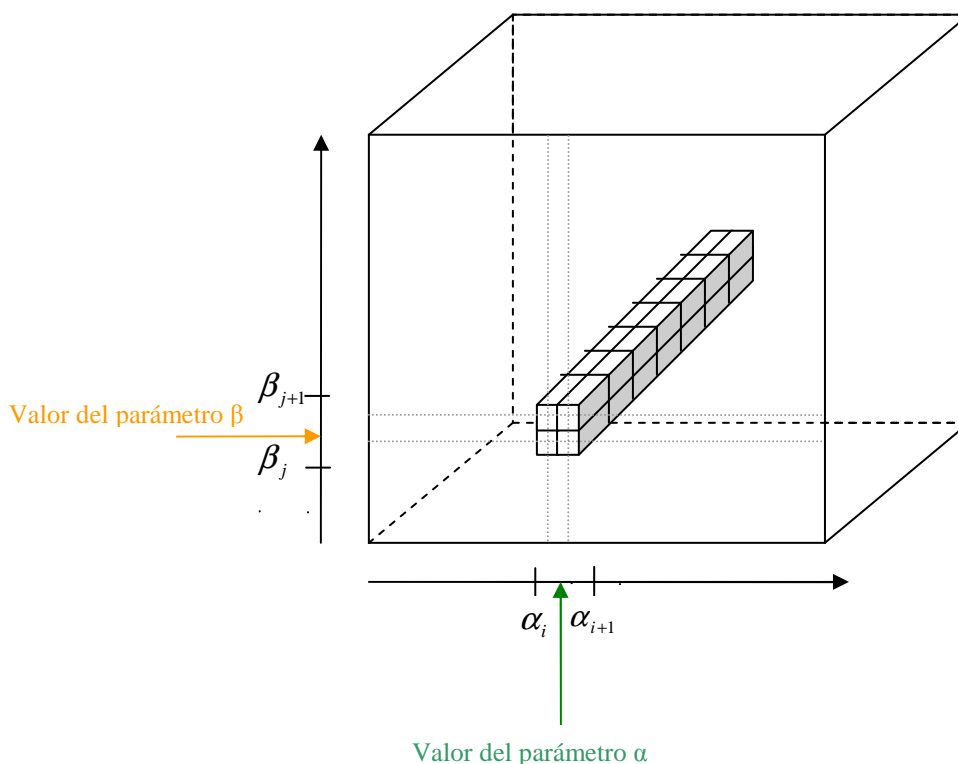


Figura 9: Interpolación lineal sobre múltiples matrices de datos

$$\alpha_{norm} = \frac{\alpha - \alpha_i}{\alpha_{i+1} - \alpha_i} ; \text{normalización de los parámetros de interpolación}$$

$$\beta_{norm} = \frac{\beta - \beta_j}{\beta_{j+1} - \beta_j}$$

$$\overline{coef} = \alpha_{norm} \beta_{norm} \overline{COEF}[j, i] + (1 - \alpha_{norm}) \beta_{norm} \overline{COEF}[j, i+1] + \alpha_{norm} (1 - \beta_{norm}) \overline{COEF}[j+1, i] + (1 - \alpha_{norm})(1 - \beta_{norm}) \overline{COEF}[j+1, i+1] ; \text{cálculo de los coeficientes} \quad (8)$$

El motor de simulación también posee componentes propios que implementan mecanismos de interpolación multidimensional los cuales son utilizados para realizar los cálculos de aquellos componentes que dependen de los ángulos aerodinámicos (α y β), la velocidad y altura (ver Figura 1).

Para completar el cálculo de las fuerzas y momentos en cada ciclo de simulación se diseñaron componentes cuya funciones de transferencia llevan adelante las operaciones matriciales correspondientes (ver Eq. (3) y (4)).

El esquema de componentes y sus interconexiones del bloque que realiza el cálculo de todos los coeficientes aerodinámicos y de las fueras y momentos puede verse más abajo. (ver Figura 10).

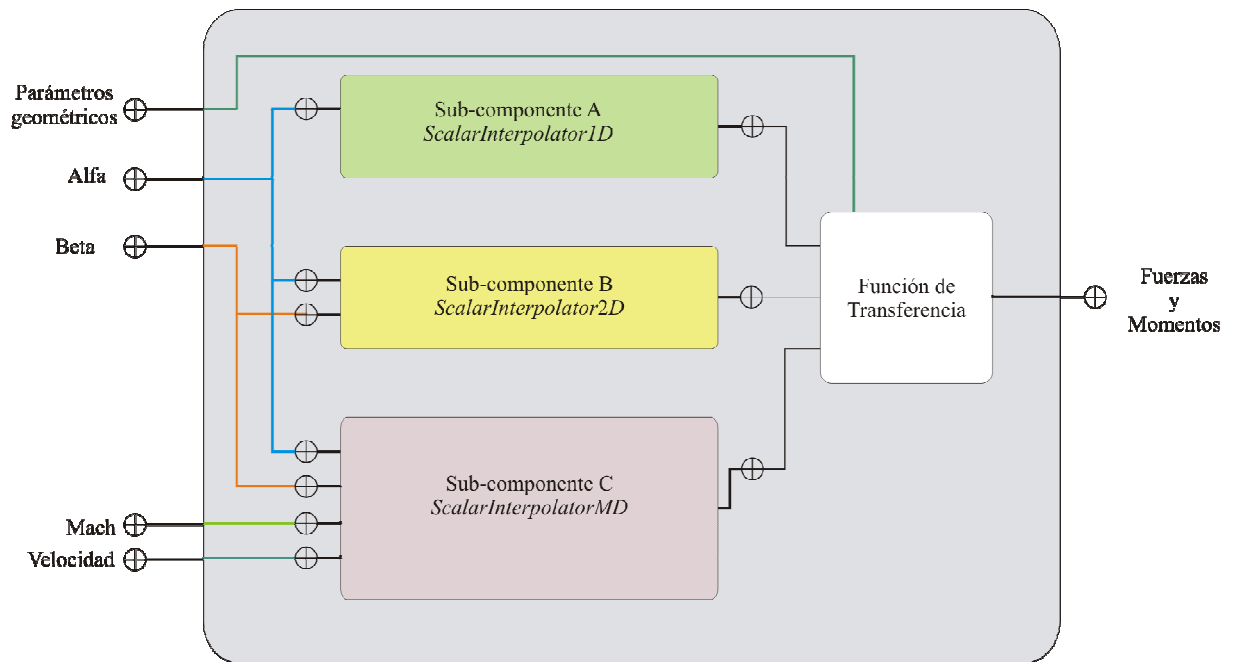


Figura 10. Interconexión de los componentes

5 ENSAYO Y COMPARACIÓN

Se llevó a cabo un paquete de ensayos sobre el modelo IA 63 PAMPA implementado para el motor de simulación, para determinar la respuesta a distintos estímulos en los comandos. En cada caso los resultados fueron comparados con la respuesta del modelo matemático implementado en GNU Octave. En cada caso la respuesta observada en ambas implementaciones fue similar, de acuerdo a lo esperado.

Se exponen a continuación dos de ellos. El primero que se muestra consistió en aplicar un doble pulso en el elevador (ver Figura 11) y se observó el efecto en el ángulo de actitud Tita (ver Figura 12).

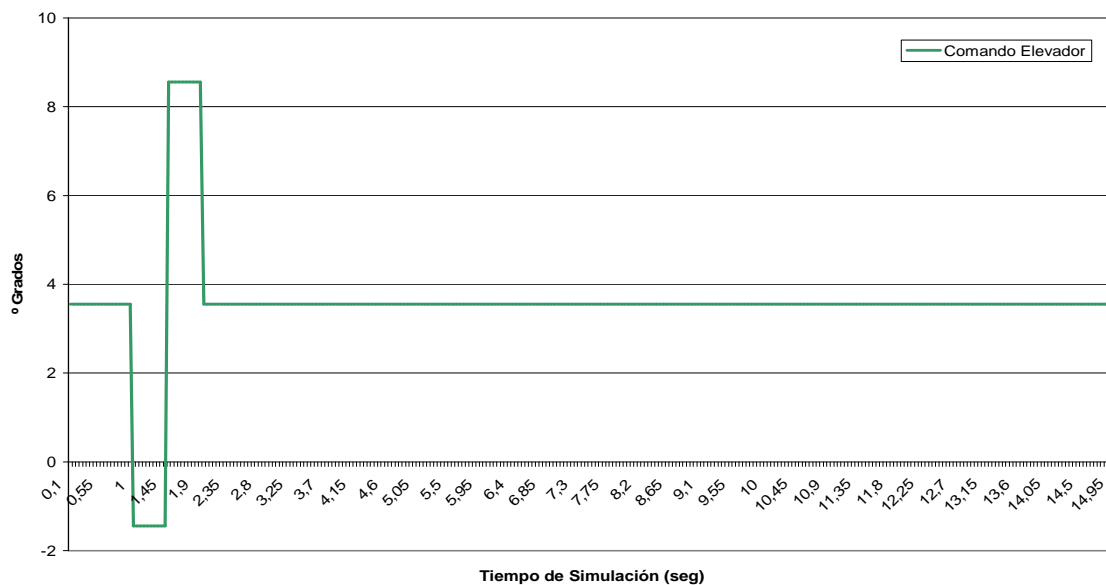


Figura 11. Evolución del comando Elevador

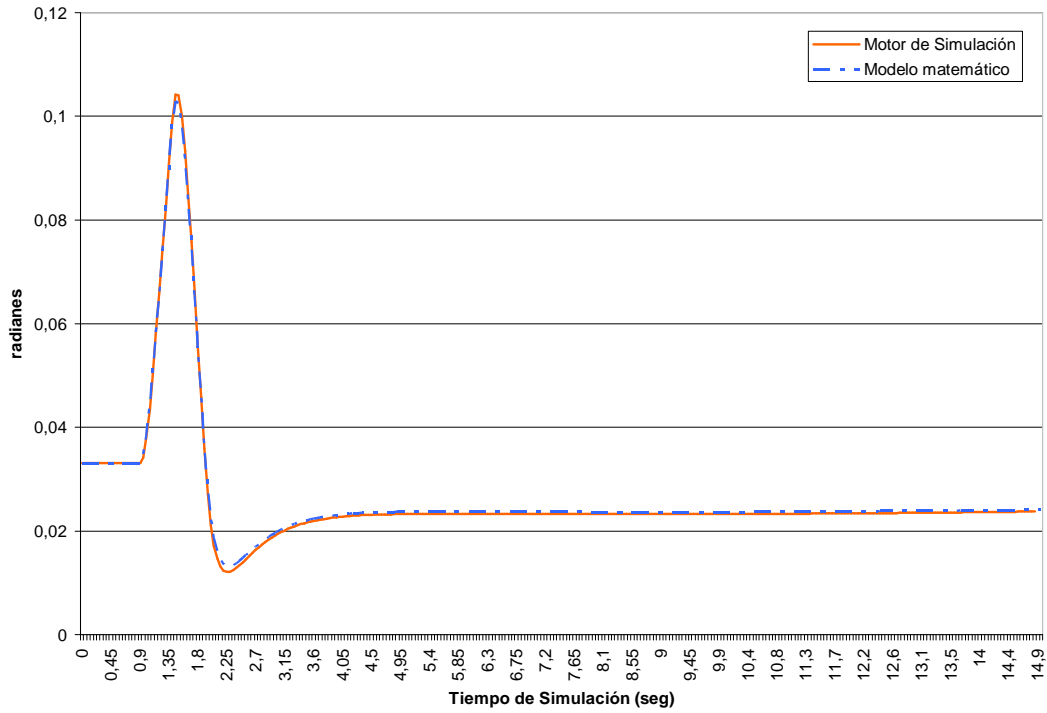


Figura 12. Efecto en el ángulo de actitud Tita.

Para el segundo ensayo se aplicó un doble pulso en los alerones (ver [Figura 13](#)) y en este caso se observó el efecto en el ángulo de rolido Psi (ver [Figura 14](#))

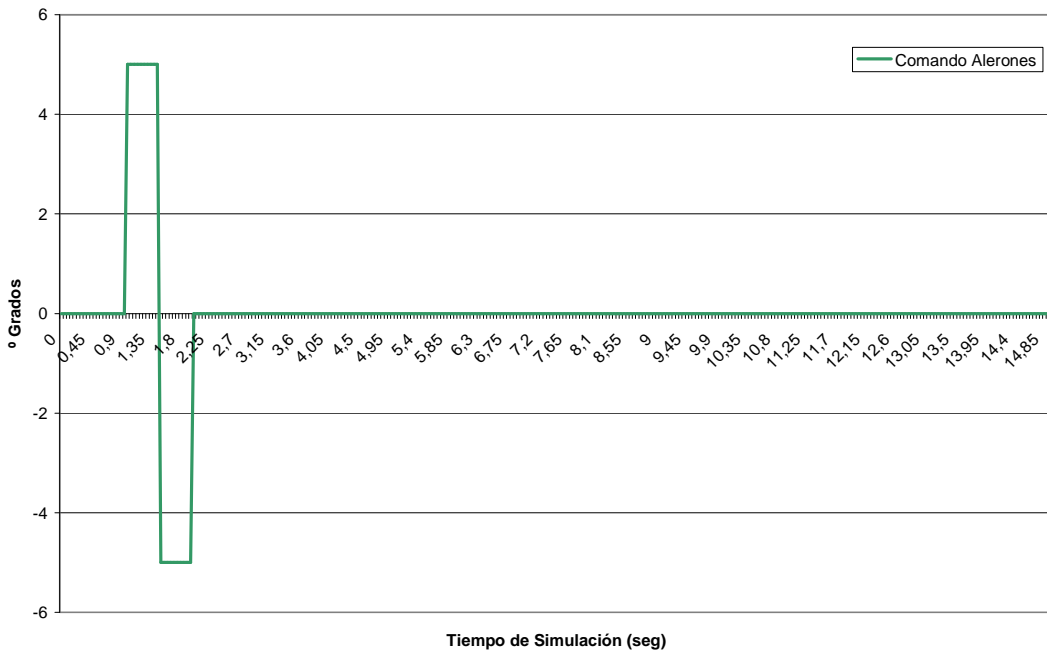


Figura 13. Evolución del comando Alerones.

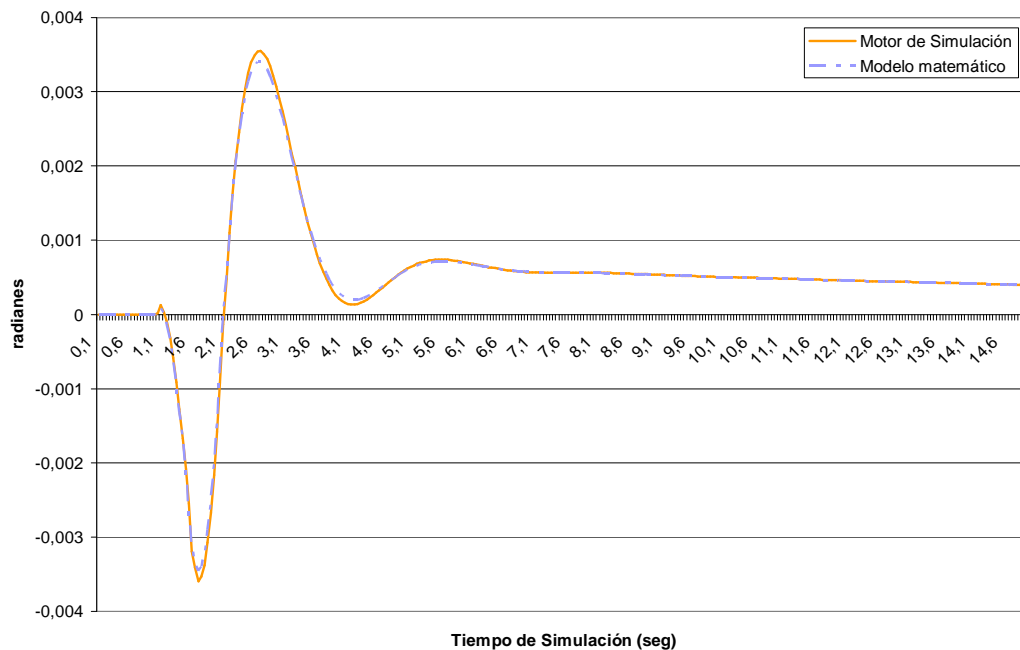


Figura 14. Efecto en el ángulo de rolido Psi.

6 GLOSARIO

Coefficientes:

- CD = resistencia
- CL = sustentación
- CM = momento de cabeceo
- CY= fuerza lateral
- CN= momento de guiñada
- Cl= momento de rolido

Derivativas:

- $CL\alpha$ = sustentación respecto del ángulo de ataque
- $CM\alpha$ = momento de cabeceo respecto del ángulo de ataque
- $CY\beta$ = fuerza lateral respecto del ángulo de deslizamiento
- $CN\beta$ = momento de guiñada respecto del ángulo de deslizamiento
- $Cl\beta$ = momento de rolido respecto del ángulo de deslizamiento
- CLq = sustentación respecto de la velocidad de cabeceo
- CMq = momento de cabeceo respecto de la velocidad de cabeceo
- $CLd\alpha/dt$ = sustentación respecto de la derivada del ángulo de ataque en el tiempo
- $CMd\alpha/dt$ = momento de cabeceo respecto de la derivada del ángulo de ataque en el tiempo
- Clp = momento de rolido respecto de la velocidad de rolido
- Cyp = fuerza lateral respecto de la velocidad de rolido
- CNp = momento de guiñada respecto de la velocidad de rolido
- CNr = momento de guiñada respecto de la velocidad de guiñada
- Clr = momento de rolido respecto de la velocidad de guiñada

Sub-índices:

basic. = contribución debido a la configuración limpia; efectos de primer orden
δelev., *δaler.*, *δrud.* = contribución por deflexión de comando de elevador, alerón y timón de dirección respectivamente.
norm. = valor normalizado.

7 CONCLUSIONES

Se presentó un modelo de datos correspondiente a un modelo aerodinámico de la aeronave IA 63 PAMPA de seis grados de libertad desarrollado con el objetivo de simular el vuelo de una aeronave como también su implementación en el motor de simulación. En especial se expuso la solución adoptada para el cálculo de los coeficientes aerodinámicos y el modo en que dicha solución se modeló en el motor de simulación.

Se realizaron ensayos con el objetivo de contrastar la respuesta de la implementación en el motor de simulación con la respuesta obtenida en la implementación del modelo matemático en GNU Octave. Los resultados muestran un comportamiento similar

REFERENCIAS

- Abbate, H.A., Galán, M.V., y Luiso, J.E., Motor de Simulación de propósitos generales. *4º Jornadas de Informática Industrial JII 2007 – Jornadas Argentinas de Informática e Investigación Operativa JAIIO 2007*, Agosto 2007, Mar del Plata, Argentina
- Berndt, Jon S., JSBSim: An Open Source Flight Dynamics Model in C++. *AIAA Modeling and Simulation Technologies*. Conference and Exhibit, August 2004, Providence, Rhode Island
- Jackson, E. B., Hildreth, B. L., Flight Dynamic Model Exchange Using XML, *Modeling & Simulation Technologies Conference*, Paper Number AIAA-2002-4482, August, 2002.
- Leslie, R. A., Geyer, D. W., Kevin Cunningham Patricia C. Glaab, P. Sean Kenney, Michael M. Madden, *LaSRS++ An object-oriented framework for real time simulation of aircraft*. AIAA-98-4529
- Luiso, J.E., Galán, M.V., Giraudo, P.S., Zapico, E., y Abbate, H.A., Motor de Simulación para Simuladores de Vuelo. *Primer Congreso Argentino de Ingeniería Aeronáutica 2008 CAIA 2008*, La Plata, Argentina.
- Zapico, E., Giraudo, P.S., Abbate, H.A., A Tensor 6-DOF Flight Dynamics Model for a Low-Cost, Real-Time Flight Simulator. *AIAA Modeling and Simulation Technologies Conference and Exhibit*. 2008a, Honolulu, Hawaii
- Zapico, E., Giraudo, P.S., Abbate, H.A., Un modelo de dinámica del vuelo para un simulador de vuelo en tiempo real. *Primer Congreso Argentino de Ingeniería Aeronáutica 2008 CAIA 2008b*, La Plata, Argentina.