

Seguridad aérea e informática

En el pasado mes de mayo diversos medios de comunicación publicaron lo siguiente: 'Un fallo en el software del ordenador que controla los motores fue la causa del accidente que el pasado día 9 sufrió un avión A400M en Sevilla y que provocó la muerte de cuatro tripulantes y heridas a otros dos, según los primeros resultados de la investigación. Eso explica que el avión de transporte militar perdiera empuje poco después de despegar en un vuelo de prueba y que se precipitara a pesar de contar con cuatro motores, que no pueden fallar simultáneamente' (El País, 19 mayo 2015). Según dicha investigación interna, uno de los errores estuvo en los ordenadores que transmiten las órdenes a los propulsores que pudieron transmitir órdenes contradictorias y provocar que los pilotos no pudieran controlar la aeronave (ABC, 19 mayo 2015).

Después de leer estas noticias surge la pregunta ¿es seguro el software?

En una aeronave moderna la mayoría de los sistemas de control se basan en la utilización de pequeños computadores 'empotrados', diseñados para realizar funciones muy concretas y cuya actuación se monitoriza por medio de programas software. Esto tiene consecuencias prácticas como: abaratamiento de los costes de producción, versatilidad para realizar mejoras sin más que modificando algún programa, cambio radical en cuanto a la comunicación piloto-avión, y la importancia extrema que adquiere la fiabilidad de los programas.

La informatización está provocando que la comunicación piloto-avión (interfaz hombre-máquina) cambie radicalmente. De tener volantes, palancas, ruedas y otras piezas mecánicas se pasa a interactuar a través de pantallas táctiles, ratones, joysticks, etc. Para los pilotos supone un cambio muy profundo de manipulación, aunque las interfaces estén dotadas de elementos adicionales para darles apariencia de dispositivos mecánicos.

Sin duda en aviones altamente informatizados, para algunas operaciones o frente a un fallo, el piloto echará en falta la existencia de elementos mecánicos con los que interactuar. Este problema se soluciona con un adecuado entrenamiento de los pilotos y con las nuevas generaciones de los mismos que llegarán acostumbradas desde la infancia a una comunicación hombre-máquina más virtual que mecánica. También sería muy oportuno poder pasar en cualquier momento de un entorno de funcionamiento automático a control manual, no dejando en ninguna situación imponente al piloto, permitiéndole utilizar su conocimiento, afianzado en la experiencia, para monitorizar eventualmente la conducción del avión.

Está ampliamente probado que el control por software es sumamente eficiente reuniendo ventajas indudables. No obs-

ALBERTO PRIETO
MIEMBRO DE LA ACADEMIA DE CIENCIAS
MATEMÁTICAS, FÍSICO-QUÍMICAS Y NATURALES
DE GRANADA.
CATEDRÁTICO DE LA UNIVERSIDAD DE GRANADA

No hay que alarmarse: en Ciencia y en Ingeniería cuando se identifica un problema se busca y, generalmente, se encuentra la solución



Cabina o 'cockpit' de un Airbus A320. :: EFE/MARIUS PALMEN

tante, la realización de programas requiere un conocimiento completo y exhaustivo del proceso a informatizar, ya que el programador debe prever todos los casos posibles para que el programa reaccione correctamente ante cualquier contingencia. Esto es especialmente importante en los sistemas críticos, que se definen como aquellos en los que un fallo puede dar lugar a pérdidas humanas o pérdidas ambientales o económicas significativas. Este es el caso de los sistemas de control de una aeronave: en las especificaciones del diseño se debe prever cualquier situación posible (o incluso, se podría decir 'imposible') para que los ingenieros informáticos las

consideren en sus programas para que el sistema reaccione oportunamente.

Otro problema susceptible de presentarse son los errores de programación. En la década de los setenta un satélite artificial se perdió en el espacio, suponiendo la pérdida de millones de dólares, debido a un sencillo error de programación consistente en la falta de un espacio entre dos caracteres de una instrucción. Los errores mecánicos o electrónicos de un sistema son relativamente fáciles de percibir, por ser más evidentes o visibles, pero las instrucciones de un programa son intangibles siendo mucho más sutil detectar errores.

Pero, después de lo dicho, no hay que alarmarse: en Ciencia y en Ingeniería cuando se identifica un problema se busca y, generalmente, se encuentra la solución. Así, el problema del error de programación antes citado se solucionó satisfactoriamente con el desarrollo de nuevos lenguajes de programación

(denominados 'fuertemente tipificados'), como el lenguaje Ada. También existe una disciplina dentro de la informática denominada Ingeniería del Software que trata de la producción de programas y aplicaciones de forma sistemática, utilizando muchas de las técnicas utilizadas y ampliamente probadas en ingeniería para el diseño, fabricación y mantenimiento de productos industriales. Especial énfasis se hace en la calidad del producto final (en nuestro caso software) teniendo en cuenta tres factores: confiabilidad o aptitud del sistema para proporcionar servicios tal y como se han especificado en la fase previa al diseño, inocuidad o garantía del sistema para funcionar sin fallos (que en algunas aplicaciones pueden ser catastróficos), y seguridad o habilidad del sistema para protegerse así mismo frente a intrusiones ya sean accidentales o deliberadas.

Además, los diseños se someten a dos procesos: uno de verificación, por el que se hace un chequeo de si el sistema construido se ajusta fielmente a las especificaciones, y otro de validación para comprobar si el sistema se ajusta a las necesidades y expectativas del cliente. Estos son exhaustivos en el caso de sistemas críticos.

En definitiva, como cualquier construcción de ingeniería, no se puede asegurar al cien por cien el funcionamiento perfecto del software, pero existe una metodología (Ingeniería del Software) ampliamente estudiada y conocida por los ingenieros informáticos para producir programas de muy alta calidad, teniendo una especial consideración con los sistemas críticos. Aun así, siempre debería preverse que un experto pudiese ser capaz de imponerse al sistema automático, en tanto que es extremadamente complejo predecir absolutamente todo.