4^to Congreso Argentino de Ingeniería Aeronáutica

ANÁLISIS DE CONFIABILIDAD EN SOFTWARE DE AVIONICA

M. Marcos[a] **, G. Rodríguez**[a,b] **, M. Amor**[a,b] **y A. Cararo**[b]

[a]*Grupo Sistemas de Tiempo Real, Facultad de Ingeniería, Universidad Nacional de Río Cuarto, Ruta Nacional N° 36 Km 601 , 5800 Río Cuarto, Córdoba, Argentina. http://www.unrc.edu.ar*

[b]*Centro de Investigación y Desarrollo de Tecnologías Aeronáuticas, Área de Material Río Cuarto, Ruta Provincial N° 158 5805 Las Higueras, Córdoba, Argentina. http://www.faa.mil.ar*

Palabras claves: Confiabilidad, Mantenibilidad, Disponibilidad, Seguridad, Software, Aviónica

Resumen

Los sistemas basados en software tienen varias ventajas sobre los sistemas basados en hardware en términos de funcionalidad: costo, flexibilidad, facilidad de mantenimiento, reutilización, etc. Sin embargo, el software es propenso a fallas. Un software de seguridad crítica (como es el caso de un software de aviónica) mal escrito puede dar lugar a fallos catastróficos y situaciones de riesgo de vida. Por lo tanto, un software de seguridad crítica debe ser probado adecuadamente y determinada la probabilidad de ocurrencia de fallos del mismo.

La cuantificación de la fiabilidad del software se considera un asunto no resuelto totalmente, en particular en el ámbito de los sistemas embebidos, en la actualidad diferentes enfoques y modelos tienen supuestos y limitaciones que no son aceptables para aplicaciones de seguridad crítica. También, para construir software fiable, es necesario estudiar los factores que puedan afectar a la fiabilidad del software.

Uno de estos enfoques que más se aplica a entornos de seguridad crítica son los modelos de crecimiento de fiabilidad del software (SRGM, por sus siglas en inglés, Software Reliability Growth Models) que son utilizados para estimar la tasa de fallos y la fiabilidad de un sistema de software. La necesidad de este tipo de medidas se enmarca dentro del objetivo de cuantificar la confiabilidad del sistema-software.

El presente trabajo describe la aplicación del enfoque SRGM al mantenimiento de un software de aviónica teniendo en cuenta técnicas para cuantificar la confiabilidad y algunas acciones correctivas para mejorar este índice.

e-mail: mamor@ing.unrc.edu.ar

1. INTRODUCCIÓN

La actual generación de Sistemas de Aeronaves de la FAA cuenta, cada día más, con computadoras integradas que ofrecen soluciones de alta tecnología para diferentes funciones del sistema de aviónica. Estos sistemas de aviónica consisten de una computadora principal denominada Computadora de Misión (MC, por sus siglas en inglés, *Mission Computer*) la cual se comunica, a través de un bus digital, con otros equipos del avión tales como radares, centrales inerciales, equipos de comunicación, interfaces del piloto, etc., denominados LRU (por sus siglas en inglés, *Line Replaceable Unit*) a los fines de asistir al piloto durante la operación de la aeronave. El software embarcado de la MC es denominado Programa Operacional de Vuelo (OFP, por sus siglas en inglés, *Operational Flight Program*). Comprender profundamente el rol del OFP, nos ayudará a entender a fondo la funcionalidad del sistema, su misión y, particularmente, todo su sistema de aviónica integrada.

Los sistemas basados en software tienen varias ventajas sobre los sistemas basados en hardware en términos de funcionalidad: costo, flexibilidad, facilidad de mantenimiento, reutilización, etc. Sin embargo, el software es propenso a fallas. Un software de seguridad crítica (como es el caso de un OFP) mal escrito puede dar lugar a fallos catastróficos y situaciones de riesgo de vida. Por lo tanto, un software de seguridad crítica debe ser probado adecuadamente y determinada la probabilidad de ocurrencia de fallos del mismo.

La cuantificación de la fiabilidad del software se considera un asunto no resuelto totalmente, en la actualidad diferentes enfoques y modelos tienen supuestos y limitaciones que no son aceptables para aplicaciones de seguridad. También, para construir software fiable, es necesario estudiar los factores que puedan afectar a la fiabilidad del software [1, 2].

Uno de estos enfoques que más se aplica a entornos de seguridad crítica son los modelos de crecimiento de fiabilidad del software (SRGM, por sus siglas en inglés, *Software Reliability Growth Models*) que son utilizados para estimar la tasa de fallos y la fiabilidad de un sistema de software. La necesidad de este tipo de medidas se enmarca dentro del objetivo de cuantificar la confiabilidad del sistema-software [3, 4].

En el mantenimiento de un OFP cuantificar la confiabilidad del mismo es esencial para programar el ciclo de mantenimiento del mismo [5]. Actualmente la Fuerza Aérea Argentina (FAA), a través del Centro de Investigación y Desarrollo de Tecnologías Aeronáuticas (CITeA), cuenta con la capacidad de mantenimiento correctivo y perfectivo sobre los OFP de sus aeronaves a los efectos de corregir fallas descubiertas posterior a la entrega del sistema y realizar posibles mejoras propuestas, todo ello en el ámbito de las Reglamentaciones de Aeronavegabilidad Militar (RAM) [6].

El presente trabajo describe la problemática actual de confiabilidad de software en el ámbito del mantenimiento de un OFP en la órbita del CITeA, se analizan técnicas para cuantificar la confiabilidad y algunas acciones correctivas para mejorar este índice.

2. CONFIABILIDAD DE UN SOFTWARE

2.1. Conceptos básicos

La confiabilidad (*dependability*) no se mide directamente, sino a través de sus atributos, los cuales, expresados desde una perspectiva RAMS son: fiabilidad (*reliability*), disponibilidad (*availability*), mantenibilidad (*maintainability*) y seguridad (*safety*).

Una medida de confiabilidad importante es la tasa de fallos (número de fallos por unidad de tiempo) que sirve para evaluar la frecuencia de fallos de un sistema tal y como es percibida por el usuario. En hardware, la fluctuación normal de la tasa de fallos suele ser la conocida curva de la bañera, ver Figura 1.

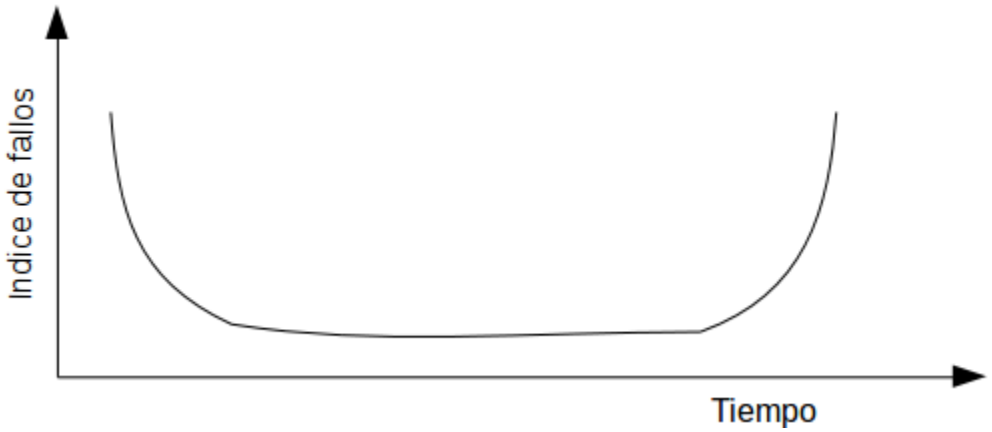

Figura 1. Curva de la bañera hardware.

Sin embargo, el comportamiento de la tasa de fallos para sistemas de software es diferente, ya que cuando el sistema se encuentra disponible para el usuario se puede considerar que la tasa de fallos permanece constante en el tiempo. Ahora bien, durante el periodo de desarrollo de un sistema de software este comportamiento no es así, ya que la tasa fallos no es constante en el tiempo. De hecho, durante la fase de pruebas la tasa de fallos debería ser decreciente y aproximadamente constante durante su fase de vida útil, ver Figura 2.

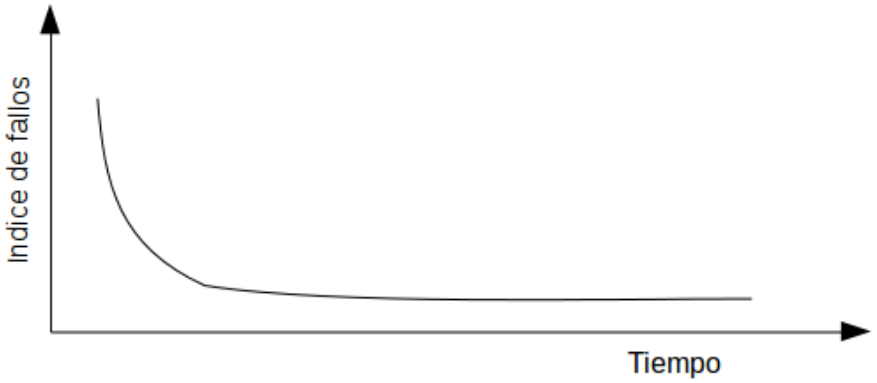

Figura 2. Curva de la bañera “ideal” software.

Considerando el mantenimiento como un proceso continuo durante su vida útil, cada vez que en el ciclo de vida del software sale una nueva versión (sea por mejoras o correcciones) ocurre también que se pueden introducir nuevos errores. Entonces la nueva curva incluyendo las incorporaciones de nuevas versiones queda como se muestra en la Figura 3.

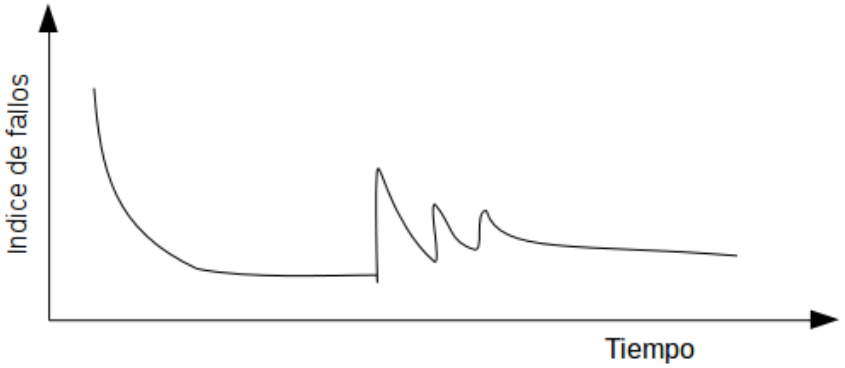

Figura 3. Curva de la bañera “real” software.

2.2. Predicción del número de fallos de un software

Aplicando técnicas de SRGM [7] durante la fase de pruebas del desarrollo de un sistema de software, se podrá predecir el número de fallos que tendrá un sistema en su fase operativa, de forma que se dispondrá de una útil herramienta de decisión para:

- Establecer si el conjunto de pruebas al que ha sido sometido el sistema ha resultado suficiente, de forma que se pueda autorizar su paso a la fase de operación.
- Reducir la posibilidad de que ocurra una incidencia grave durante la operación.
- Estimar la tasa de fallos del sistema.
- Estimar el momento, durante la fase de pruebas, en que se alcanzan los objetivos.

Los modelos de crecimiento de fiabilidad se clasifican según el tipo de variable aleatoria bajo estudio como Tiempo entre Fallos (TBF, por sus siglas en inglés, *Time Between Failures*) o Número de Fallos observados por unidad de tiempo (FC, por sus siglas en inglés, *Failure Count*).

La calidad de las predicciones del modelo de fiabilidad de un sistema se ve afectada por dos tipos de incertidumbres sobre el comportamiento de los fallos detectados:

- No se conoce el efecto que sobre la fiabilidad tendrá la reparación de un error (de forma que se pueden introducir nuevos errores en el software).
- Se desconoce el efecto del entorno operacional, esto es, no se sabe cuándo ni qué inducirá a la detección de un fallo.

Existen más de un centenar de modelos, pero muchos de ellos no han podido ser probados en entornos operacionales con datos reales. Los modelos FC han resultado muy exitosos a la hora de modelar sistemas reales y además abordan ambas incertidumbres de una forma más realista que los modelos TBF.

Los modelos de crecimiento de fiabilidad son modelos paramétricos que están definidos por el valor de ciertos parámetros, cada uno de los cuales tiene un significado concreto. Por ejemplo, el número esperado de fallos para el modelo de Schneidewind viene expresado por la siguiente ecuación (1):

$$m(i) = \frac{a}{b}\left(1 - exp(-bi)\right) \tag{1}$$

donde:

a: número de fallos al comienzo de las pruebas
b: tasa de fallos por unidad de tiempo

Estos parámetros se pueden estimar dinámicamente mediante métodos de inferencia estadística, como por ejemplo por Máxima Verosimilitud o Estimación Bayesiana de más reciente aplicación, los cuales se nutren de los datos reales recogidos a lo largo de la fase de prueba.

Es importante señalar que los datos recogidos están íntimamente vinculados a la estrategia de pruebas que se utilice. Existen diferentes métodos para generar casos de prueba y distintas aproximaciones para detectar los fallos como por ejemplo el uso de perfiles operacionales los cuales ofrecen una aproximación estadística con el objetivo de simular el uso más típico del sistema. El uso de una u otra estrategia de prueba provocará alteraciones en la frecuencia y severidad de los fallos observados, lo que indica que los modelos elegidos deberán ser también adecuados a esta estrategia.

2.3. Confiabilidad en sistemas aerotransportados

En la industria de la aviación, existen estándares específicos, aplicados a sistemas de aviónica, que implementan rigurosas prácticas el desarrollo atendiendo a requisitos de seguridad y con ello de confiabilidad.

Los estándares, que han sido aprobados por las autoridades de certificación en todo el mundo, incluyendo Argentina a través de su Administración Nacional de Aviación Civil, se centran en determinados objetivos y no son preceptivos en la naturaleza. La SAE ARP 4754 es la guía para el desarrollo de sistemas y aeronaves [8], la SAE ARP 4761 se refiere a los métodos que conducen al proceso de evaluación de riesgos y seguridad en los sistemas y equipos del avión [9], la RTCA-254 hacen a las prácticas para la certificación de un equipo de hardware aerotransportado [10] y finalmente la RTCA DO-178C trata sobre los aspectos de certificación en el software de sistema y equipos de vuelo [11, 12]. Las autoridades de aviación a nivel internacional exigen la

conformidad con estos estándares para el proceso de desarrollo de software verificando su aplicación para la cualificación de un software de vuelo que opera en un dominio aeronáutico.

Una vez definida la funcionalidad básica del avión y ha sido correspondientemente documentado su diseño conceptual, se realiza un proceso de evaluación de riesgo y peligros conocido como FHA (por sus siglas en inglés, *Functional Hazard Assessment*). El FHA identifica y clasifica las condiciones de fallos asociados con las funciones del avión y combinaciones de funciones.

Nivel	Severidad	Descripción
A	Catastrófica	La condición de fallo puede causar múltiples muertes, por lo general con la pérdida del avión.
B	Peligroso	La condición de fallo no tiene un gran impacto negativo sobre la seguridad o el rendimiento, o reduce la capacidad de la tripulación pueda manejar el avión debido a la dificultad física o una mayor carga de trabajo, o que cause lesiones graves o mortales entre los pasajeros.
C	Mayor	La condición de fallo no reduce significativamente el margen de seguridad o aumenta significativamente la carga de trabajo de la tripulación. Puede dar lugar a malestar pasajero (o incluso lesiones de menor importancia).
D	Menor	La condición de fallo no reduce ligeramente el margen de seguridad o ligeramente aumenta la carga de trabajo de la tripulación. Los ejemplos pueden incluir causando molestia para los pasajeros o de un cambio de plan de vuelo de rutina.
E	Sin Efecto	La condición de fallo no tiene ningún impacto en la seguridad, la operación de la aeronave, la tripulación o carga de trabajo.

Tabla 1: DAL definidos en RTCA DO-178C

La Tabla 1, muestra la clasificación del estándar RTCA DO-178C de los niveles denominados DAL (por sus siglas en inglés, *Design Assurance Level*) para el diseño, de acuerdo al impacto que tiene la condición de fallo en la reducción de la capacidad de un vuelo seguro.

En un sistema de software, un error de código, en una función por ejemplo, puede atravesar las fronteras del software, del sistema y conducir a una condición de fallo en la aeronave como representa la Figura 4.

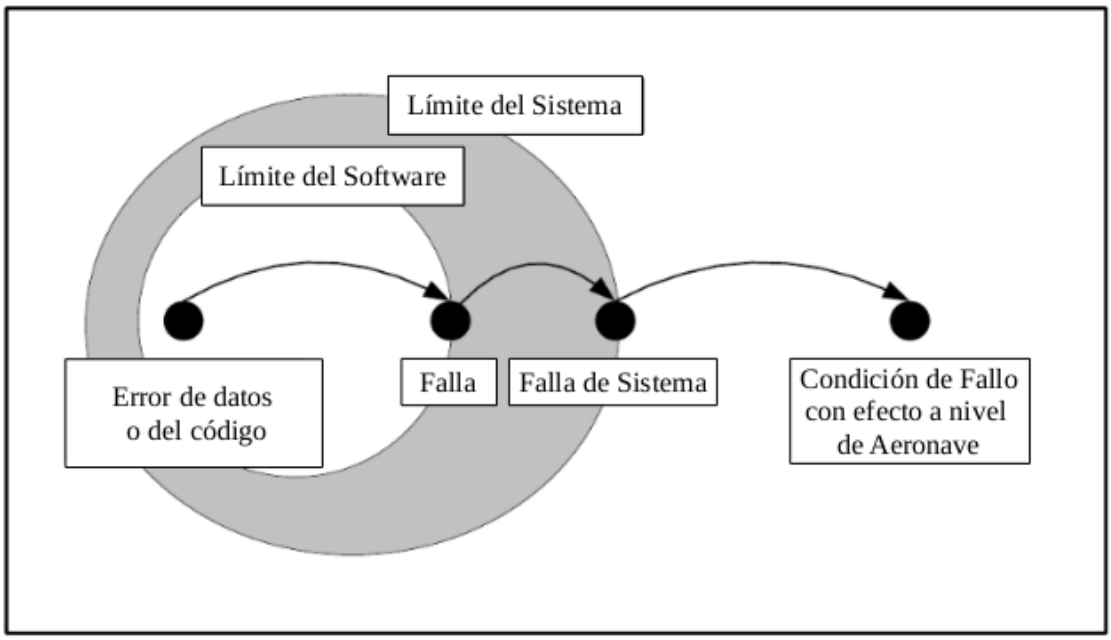

Figura 4. Propagación de falla en un sistema de software

En el ámbito aeronáutico, el requisito DAL de un OFP es de nivel A y debe diseñarse con los mecanismos de tolerancia a fallos correspondientes para garantizar que la probabilidad de ocurrencia de una condición de fallo sea de 1×10^{-9} por hora de vuelo [13, 14].

3. CASO DE ESTUDIO SOBRE EL PROCESO DE MANTENIMIENTO DEL OFP

3.1. Descripción de caso de estudio

Como caso de estudio se puso en práctica la implementación de un nuevo proceso de compilación en el OFP de un sistema de aviónica de la FAA. El nuevo proceso consistió en el ajuste del proceso de gestión de la configuración y el cambio del compilador. Cuando el nuevo OFP fue compilado, fue evaluado a través de pruebas de regresión, en un Banco de Ensayos de acuerdo a un procedimiento que implicaba intervalos regulares de testing de 15 días.

En el proceso de evaluación del nuevo OFP se sucedieron fallas que fueron corregidas cíclicamente al final de cada intervalo arrojando la siguiente distribución de fallas en el tiempo como se muestra en la Tabla N°2.

Cantidad de fallas	Periodo (15 días)
11	1
8	1
9	1
5	1
2	1
1	1
0	1
0	1
0	1
0	1

Tabla 2: DAL definidos en RTCA DO-178C

Para el conjunto de datos observados existe la posibilidad de utilizar diferentes modelos para realizar las estimaciones. De entre todos los posibles modelos es necesario elegir aquel que mejor “modela” el sistema real bajo prueba. Para este fin se utilizan una serie de parámetros que comparan algunas características de los modelos tales como bondad de ajuste, capacidad predictiva, etc. Como soporte para este análisis se utilizó SMERFS (por sus siglas en inglés, *Statistical Modeling and Estimation of Software Reliability Functions*) [15], un programa para estimar y predecir la fiabilidad del software durante la fase de prueba. La idea básica detrás de la mayoría de estos modelos es la aplicación de una distribución de Poisson cuyos parámetros tomen diferentes valores para diferentes modelos [16].

En la Figura 5 se comparan los datos observados (puntos verdes) con 3 modelos estimados con distintas ponderación de parámetros, entre ellas, la distribución No-Homogénea de Poisson (NHP, en esta gráfica representada en color violeta).

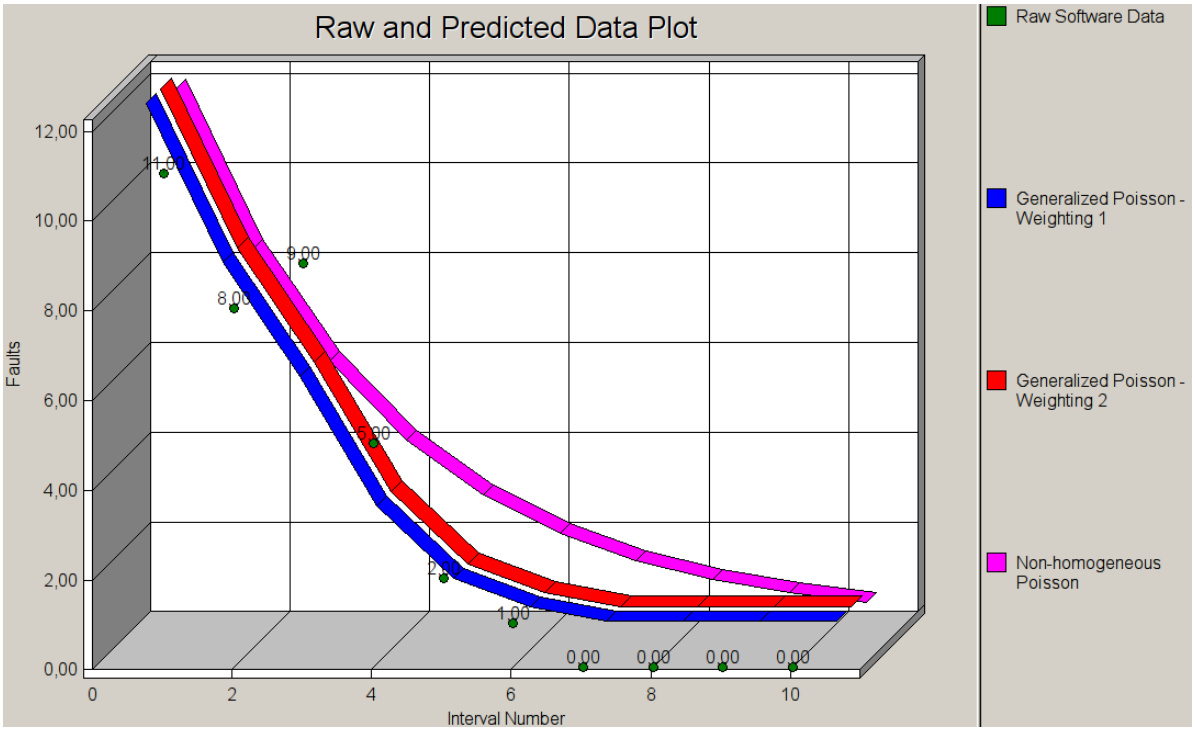

Figura 5. Comparación de los datos observados con los estimados por los modelos

Debido a la naturaleza de los datos sólo se pudieron ejecutar modelos cuya variable estudiada era el FC (esto es contabilizarán el número de fallos por intervalo), ya que fueron descartados por imprecisos los datos observados que aportaban el tiempo entre las ocurrencias de los fallos (TBF).

3.2. Resultados obtenidos del análisis de confiabilidad

Si bien a partir del intervalo 6 existen diferencias entre los modelos propuestos y los datos relevados se observa que al inicio del intervalo 9, el modelo NHP muestra una aproximación asintótica más pronunciada a los valores obtenidos de referencia con lo cual se asume una estimación más precisa del crecimiento de la confiabilidad en el tiempo respecto de los otros modelos.

La Figura 6 muestra la estimación de los parámetros y las predicciones de fallas para el modelo NHP.

En la misma se observa que para un próximo período de prueba, de un intervalo (15 días), el número de fallas esperado es de 0,34. La misma lectura, pero respecto a los intervalos de prueba en lugar del número de fallas, se puede observar en la misma Figura 6 donde el período de falla esperado hasta la ocurrencia de una (1) falla, es de 6 intervalos de pruebas.

Este indicador de 6 intervalos estimado a través del modelo NHP fue el utilizado como tiempo de referencia para las pruebas funcionales del OFP.

Los procesos de calidad de CITeA están condicionados por diferentes estándares de ingeniería como el DO-178C para desarrollo de software, DO-254 para desarrollo de hardware, etc. Particularmente en el caso del software, el estándar exige 5 diferentes planes: el de desarrollo, el de manejo de la configuración, el de verificación, el de certificación y el de calidad.

La estimación del modelo establece un indicador del tiempo de pruebas que se manifiesta directamente en la actualización de estos planes para poder ser ejecutado.

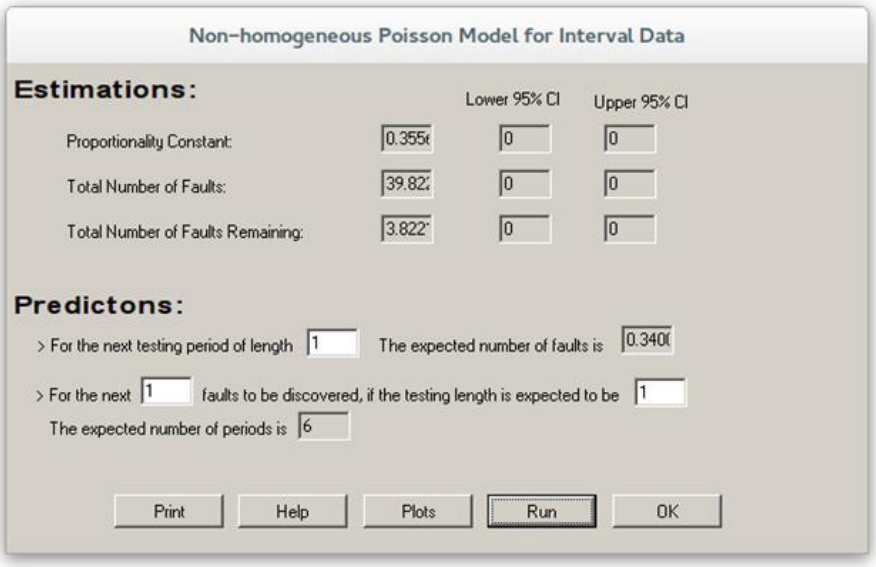

Figura 6. Estimación de confiabilidad basado en el Modelo de Poisson No-Homogéneo

4. CONCLUSIONES Y TRABAJO FUTURO

Las técnicas de crecimiento de fiabilidad del software, se utilizan para estimar la tasa de fallos y la fiabilidad de un sistema de software durante la fase de pruebas de su desarrollo. Además, se podrá predecir el número de fallos que tendrá un sistema en la fase de operación, de forma que se dispondrá de una útil herramienta de decisión para:

- Determinar en que DAL está el sistema.
- Establecer si el conjunto de pruebas al que ha sido sometido el sistema ha resultado suficiente para cumplir con el DAL asignado y autorizar su paso a la fase de operación.
- Reducir la posibilidad de que ocurra una incidencia grave durante su operación.
- Estimar el momento, durante la fase de pruebas, en que se alcanzan los objetivos de fiabilidad del sistema.

En el CITeA se aplicaron técnicas de crecimiento de fiabilidad en un nuevo proceso de compilación del OFP de un sistema de aviónica de la FAA durante la fase de pruebas de integración de sistemas, previamente a la puesta en operación.

De los resultados manifiestos en el caso de estudio presentado se concluye que una técnica de crecimiento de fiabilidad permite estimar cuánto tiempo deberá estar sometido el sistema de software en “fase de pruebas” para lograr una confiabilidad especificada acorde a un DAL determinado.

Para cada software corresponde su propio análisis de confiabilidad que puede concluir en diferentes modelos de estimación.

Estas técnicas permiten también ser aplicadas a nivel de sistema. Avanzar en ese sentido, le otorgaría al presente trabajo un significativo aporte a los efectos de complementar todos los aspectos de confiabilidad necesarios para cumplir los exigentes estándares de seguridad aplicados en sistemas embarcados de alta criticidad.

REFERENCIAS

[1] Babu, A. *“Software Reliability in Safety Critical Supervision and Control of Nuclear Reactors”*. Indira Gandhi Centre for Atomic Research, Kalpakkam. Phd Thesis. 2014.

[2] V. L. Foreman, F. M. Favarò and J. H. Saleh, *"Analysis of software contributions to military aviation and drone mishaps,"* 2014 Reliability and Maintainability Symposium, Colorado Springs, CO, 2014, pp. 1-6. doi: 10.1109/RAMS.2014.6798450.

[3] Yépez, A., Redondo López D. *“Análisis de Fiabilidad de Sistemas Aplicando Técnicas de Crecimiento de Fiabilidad de Software”*, Métodos y Tecnología de Sistemas y Procesos (MTP), RPM-AEMES, VOL. 4, No 3, Septiembre 2007.

[4] Yao Yiping and Cheng Jun, *"Study on combined software and hardware reliability growth model for avionics system,"* Industrial Technology, 1994., Proceedings of the IEEE International Conference on, Guangzhou, 1994, pp.338-341. doi: 10.1109/ICIT.1994.467099.

[5] Satterthwaite, Charles P. *"Maintaining an Operational Flight Program"*. Wright Laboratory, Wright-Patterson AFB. Technical Memorandum. 1992.

[6] PC 14-05 (RAM) VR 01/15 Reglamento de Aeronavegabilidad Militar. http://www.fuerzas-armadas.mil.ar/Dependencias-DIGAMC-Normas-Vigentes.aspx.

[7] Wood A. *"Software Reliability Growth Models"*, Tandem Computers. Technical Report 96.1. 1996.

[8] SAE ARP 4754A, Guidelines for Development of Civil Aircraft and Systems Software Considerations in Airborne Systems. 2010.

[9] SAE ARP 4761, Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment. 1996.

[10] RTCA DO-254, Design Assurance Guidance For Airborne Electronic Hardware, dated April 19, 2000.

[11] RTCA DO-178C, Software Considerations in Airborne Systems and Equipment Certification, dated December 13, 2011.

[12] Won Keun Youn, Seung Bum Hong, Kyung Ryoon Oh, Oh Sung Ahn, *"Software certification of safety-critical avionic systems: DO-178C and its impacts"*, Aerospace and Electronic Systems Magazine IEEE, vol. 30, pp. 4-13, 2015, ISSN 0885-8985.

[13] McDermid JA, Kelly TP. Software in Safety-Critical Systems: Achievements and Prediction. Nuclear Future. 2006.

[14] Rierson, L. *"Developing Safety-Critical Software: A Practical Guide for Aviation Software and DO-178C Compliance"*. New York: CRC Press. 610 p. 2013.

[15] SMERFS (Statistical Modeling and Estimation of Software Reliability Functions. https://www.slingcode.com/smerfs/

[16] M. Razeef , N. Mohsin. *"Software Reliability Growth Models: Overview and Applications"*, Journal of Emerging Trends in Computing and Information Sciences, vol. 3, NO. 9, SEP 2012, ISSN 2079-8407.