

Método Lingüístico Difuso para el Análisis Dinámico de Riesgos en Modelos de Procesos de Software

Resumen– Así como el proceso de software es evolutivo y dinámico, la gestión de los riesgos también debe ser evolutiva y dinámica. En este artículo se propone un método lingüístico difuso para el análisis dinámico de riesgos que puede ser empleado en modelos de procesos de software iterativos y/o dirigidos por los riesgos. El método realiza procesos de computación con palabras utilizando el modelo computacional basado en 2-tuplas lingüísticas, lo que permite obtener resultados lingüísticos cercanos al lenguaje natural y de fácil interpretación.

Palabras clave-- Gestión de Riesgos, Proceso de Software, Toma de Decisión Dinámica, 2-Tupla Lingüística.

I. INTRODUCCIÓN

En general, las actividades estructurales del proceso de ingeniería de software se aplican en forma iterativa a medida que avanza el proyecto. Es decir, la comunicación, la planeación, el modelado, la construcción y el despliegue se ejecutan a través de cierto número de repeticiones del proyecto. Cada iteración produce un incremento del software en término de características y funcionalidades. Estas actividades estructurales del proceso de ingeniería de software son complementadas por actividades sombrija que también se aplican iterativamente y ayudan al equipo gestionar y controlar el avance, la calidad, el cambio y el riesgo [1].

Un riesgo nos da la medida de la probabilidad y la pérdida de un acontecimiento que puede impactar el proyecto, proceso, producto de software y/o a las personas que lo desarrollan [2]. La gestión de riesgos permite evaluar los acontecimientos que puedan afectar el resultado del proyecto o la calidad del producto, por lo que constituye uno de los factores clave de un proyecto de software exitoso [3-10].

Robert Charette [4] ofreció tres elementos conceptuales sobre el riesgo que podemos analizar desde la ingeniería de software:

- 1) “El riesgo se preocupa por los acontecimientos futuros”: Desde la planificación, es importante analizar qué puede ir mal y cómo hacerlo bien, como principio elemental según Pressman [1] para gestionar cualquier proyecto con éxito. El riesgo es un posible problema (su probabilidad de ocurrencia debe ser menor que 1) que debemos evitar y que podemos convertir en una oportunidad de mejora si logramos cambiar las circunstancias que lo causan.
- 2) “El riesgo involucra cambio”: Estos eventos adversos probables, pueden provocar cambios en los requisitos, los cronogramas, las tecnologías y el proceso de software en general. Se trata entonces de evaluar estos posibles impactos en todas las aristas del proyecto.

- 3) “El riesgo involucra elección y la incertidumbre que ella conlleva”: En este caso, si conocemos la probabilidad e impacto de los riesgos, el equipo debe elegir diferentes estrategias proactivas de gestión. Pero las elecciones entre tecnologías, modelos de calidad, entre otras relacionadas con el proceso de software, no son triviales. Se realizan en entornos donde la información es vaga, imprecisa y se valora un objeto “riesgo”, de naturaleza inherentemente difusa. Este tercer elemento es el más representativo y nos lleva entonces a la indispensable modelación del análisis de riesgos como un problema de toma de decisión multicriterio bajo incertidumbre, epicentro de este trabajo.

En la propia concepción del modelo espiral [1,11], con una estructura iterativa, se emplean los prototipos como enfoque para la de reducción de riesgos. Por consiguiente, el modelo espiral implica la evaluación de los riesgos técnicos en todas las etapas del proyecto con el objetivo de mitigarlos o minimizarlos antes de que se conviertan en un problema. Es precisamente esta estrecha relación con la experiencia y la experticia en el análisis de riesgo, un punto crítico en la implementación del modelo en espiral y otras técnicas de ingeniería de software, pues si algún riesgo importante no se descubre y gestiona adecuadamente, puede conducir al fracaso del proyecto de software [1].

Pero, volvamos a la esencia del modelo espiral. El modelo de desarrollo espiral [11] es un generador de modelo de proceso impulsado por el riesgo, que se usa para guiar la ingeniería concurrente con participantes múltiples de sistemas intensivos en software. Una de sus características distintivas, es el enfoque cíclico para el crecimiento incremental del grado de definición de un sistema y su implementación, mientras que disminuye su grado de riesgo.

Es esta característica distintiva la que nos plantea otra innegable y poco estudiada dimensión del riesgo, su carácter dinámico. El comportamiento de los riesgos en un proyecto de software puede variar con el tiempo. Nuevos riesgos pueden aparecer mientras riesgos secundarios pueden aparecer como consecuencia del tratamiento de los riesgos iniciales. Además, los gerentes de proyecto pueden decidir ignorar algunos riesgos considerados insignificantes inicialmente, pero determinadas circunstancias pueden cambiar la percepción sobre su impacto y probabilidad y convertirse así en riesgos de atención urgente. Este escenario exige realizar el análisis de riesgos en una forma dinámica para apoyar las decisiones en un proceso que realmente sea efectivo.

Como el software evoluciona a medida que el proceso avanza, el desarrollador y el cliente comprenden y reaccionan mejor ante los riesgos en cada nivel de evolución [1]. Así como

el proceso es evolutivo y dinámico, la gestión de los riesgos también debe ser evolutiva y dinámica. Para lograrlo, parece lógico aplicar algún enfoque de toma de decisión dinámica o multiperíodo [12-15] que permita evaluar y priorizar alternativas considerando su comportamiento en el tiempo.

Por otra parte, el enfoque típico para analizar los riesgos consiste en utilizar una tabla o matriz cruzada de probabilidad e impacto. Dependiendo de las preferencias de la entidad, se puede emplear una escala numérica o lingüística y a partir de ellas, combinaciones específicas de probabilidad e impacto que llevan a calificar un riesgo de importancia “alta”, “moderada” o “baja”. Para cada uno de los riesgos identificados, se evalúan la probabilidad y el impacto. La evaluación de la probabilidad de los riesgos estudia la probabilidad de ocurrencia de cada riesgo específico mientras que la evaluación del impacto de los riesgos estudia el efecto potencial de los mismos sobre un objetivo del proyecto, tal como el cronograma, el costo, la calidad o el desempeño [10].

Estas estimaciones de probabilidad e impacto se pueden obtener a partir de entrevistas o reuniones con participantes seleccionados por estar familiarizados con las categorías de riesgo incluidas en la agenda. Entre ellos se incluyen los miembros del equipo del proyecto y expertos que no pertenecen al proyecto.

El análisis cualitativo de riesgos es por lo general un medio rápido y económico de establecer prioridades para planificar la respuesta a los riesgos y sentar las bases para realizar el análisis cuantitativo, si fuera necesario. En algunos casos puede que no sea posible llevar a cabo el análisis cuantitativo de riesgos debido a la falta de datos suficientes o a la naturaleza incierta de estos. La disponibilidad de tiempo y presupuesto, así como la necesidad de declaraciones cualitativas o cuantitativas acerca de los riesgos y sus impactos, determinarán qué método o métodos emplear para un determinado proyecto. Por ejemplo, la disponibilidad de información histórica tiene fuerte influencia sobre el riesgo de estimación [1]. Cuando hay disponibles métricas de software exhaustivas para proyectos anteriores, pueden hacerse estimaciones con mayor precisión, así como establecerse calendarios para evitar las dificultades pasadas y el riesgo global se reduce. Si el ámbito del proyecto se comprende pobremente o si los requisitos del proyecto están sujetos a cambio, la incertidumbre y el riesgo en la estimación se vuelven peligrosamente altos.

En estos casos, donde es poco viable realizar estimaciones precisas debido a la falta de información histórica, o la no disponibilidad de medios y métodos rigurosos y confiables para la medición, es recomendable el uso de métodos cualitativos formales de toma de decisión basados en criterios de expertos.

El juicio de expertos constituye una entrada procedente de partes con sólidos conocimientos, respecto a las acciones a emprender en el caso de un riesgo específico y definido [10]. La experiencia puede ser proporcionada por cualquier grupo o persona con una formación especializada, conocimientos, habilidad, experiencia o capacitación en la elaboración de respuestas a los riesgos.

En este contexto de análisis de riesgo bajo incertidumbre, puede ser difícil para los expertos, calificar los riesgos utilizando valores precisos porque usualmente, este conocimiento es no preciso y presenta incertidumbre, por lo que el enfoque lingüístico difuso se puede utilizar para obtener una mejor solución. El enfoque lingüístico difuso facilita herramientas para modelar y gestionar la incertidumbre por medio de variables lingüísticas, proporcionando resultados confiables [16,17]. En la TDL es necesario llevar a cabo procesos de computación con palabras para operar con las valoraciones lingüísticas. Zadeh [18], señala dos situaciones que implican computar con palabras y no con números y que se manifiestan comúnmente en el proceso de software:

- 1) “Cuando la información disponible es demasiado imprecisa para justificar el uso de los números”: El riesgo es un acontecimiento que puede ocurrir o no, por lo que los datos sobre su probabilidad de ocurrencia e impactos, son imprecisos.
- 2) “Cuando hay una tolerancia a la imprecisión que puede ser explotada para conseguir robustez y dar un mayor acercamiento a la realidad”: Los métodos para el análisis de riesgos son eminentemente cualitativos y se busca que sus resultados puedan ser comprendidos por todos los involucrados. Por otra parte, sus resultados son los que justifican análisis cuantitativos posteriores que no siempre son más efectivos y pueden llevar a una “parálisis por análisis”.

Los procesos de computación con palabras se desarrollan empleando modelos lingüísticos computacionales como el basado en la representación con 2-tupla lingüística [19], que permite obtener resultados comprensibles y cercanos al lenguaje humano.

En este artículo proponemos un método lingüístico para el análisis dinámico de riesgos que puede ser empleado en modelos de procesos de software iterativos y/o dirigidos por los riesgos. En la sección 2 se revisan los fundamentos teóricos y conceptuales del trabajo. La sección 3 presenta el método lingüístico difuso para el análisis dinámico de riesgos en procesos de software. La sección 4 ilustra a través de un ejemplo la aplicación del método a un problema sencillo y la sección 5 concluye el artículo.

II. PRELIMINARES

El método propuesto en esta contribución sigue el marco de decisión dinámico multicriterio propuesto en [12] basándose en el modelo de representación lingüístico 2-tupla [19]. En esta sección revisamos brevemente dichos modelos.

A. Sobre la Toma de Decisión Dinámica (TDD)

De manera general, en un problema de TDD la dimensión tiempo es considerada, es decir, se tienen en cuenta diferentes momentos de decisión; las alternativas no son fijas; los criterios no son fijos; y las comparaciones entre alternativas tienen en cuenta su evolución o comportamiento temporal [12,15].

En la mayoría de las soluciones de TDD, las preferencias son recogidas en múltiples periodos y utilizadas en su totalidad para

tomar una única decisión al final del proceso [13]. Como se muestra en la Fig. 1, el problema de TDD es descompuesto en múltiples problemas no dinámicos de toma de decisión, para resolver cada uno de ellos mediante los métodos clásicos de toma de decisión. Se tendrán múltiples procesos de agregación, tanto como periodos se consideren en el problema y una agregación final que tendrá en cuenta cada uno de estos resultados.

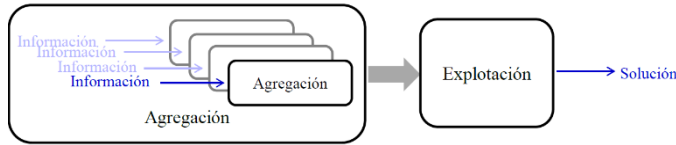


Fig. 3 Esquema de TDD.

Pero existe otro tipo de problemas donde se requiere una solución, bien en cada una de estas iteraciones, o bien al final de todas ellas, pero donde cada solución final o intermedia, debe tener en cuenta las soluciones anteriores [12,15]. Para tratar estas situaciones, y las anteriores, surge la propuesta de Campanella y Ribeiro [12], que para la resolución de un problema de TDD propone, inicialmente el cálculo de una evaluación no dinámica de cada alternativa en cada periodo a través de algún método convencional de toma de decisión. A partir del segundo momento de decisión, se calcula además una evaluación dinámica para cada alternativa empleando operadores de agregación asociativos que evitan guardar toda la información del problema de decisión, siendo necesario guardar únicamente la evaluación dinámica de cada alternativa en el periodo anterior y el actual. Además, se puede establecer una política de retención que module la información de las alternativas que serán recordadas de un periodo a otro.

Esta propuesta es flexible y adaptable a cualquier problema de TDD: los que requieren una única decisión final con enfoque multiperíodo; los que requieren diferentes y sucesivas decisiones donde la última estará influenciada por las anteriores; e incluso problemas de consenso. Por estos motivos el procedimiento general de TDD, que se muestra en la Fig. 2, parece lógico y adecuado para la solución de nuestro problema de análisis de riesgos dinámicos en tanto ofrece un enfoque flexible y general para la resolución de problemas de toma de decisión dinámica.



Fig. 2 Modelo general de TDD propuesto en [12]

A. Modelo lingüístico difuso de representación basado en 2-tuplas

El modelo lingüístico de representación basado en 2-tuplas fue introducido por Herrera y Martínez en [19] con el objetivo de mejorar la precisión de los resultados y facilitar los procesos de

computación con palabras tratando el dominio lingüístico como un dominio continuo, pero manteniendo la base lingüística (sintaxis y semántica). El modelado de la información lingüística está basado en el concepto de traslación simbólica y lo utiliza para representar la información lingüística mediante un par de valores, llamados 2-tupla lingüística (s_i, α) , donde El modelado de la información lingüística está basado en el concepto de traslación simbólica y lo utiliza para representar la información lingüística mediante un par de valores, llamados 2-tupla lingüística (s_i, α) , donde $s_i \in S = \{s_0, \dots, s_g\}$ es un término lingüístico y α es un valor numérico representando la traslación simbólica.

- $s_i \in S$ un término cuya semántica se define por una función de pertenencia difusa y la sintaxis elegida en función de las opciones que ofrece el Enfoque Lingüístico Difuso [17,18], y
- α es un valor numérico, traslación simbólica, que indica la traslación de la función de pertenencia difusa que representa el término más cercano $s_i \in S$ si s_i no coincide con la información lingüística computarizada. El valor de α se define entonces como [20].

$$\alpha = \begin{cases} [-0.5, 0.5] & \text{si } s_i \in S = \{s_0, \dots, s_{g-1}\} \\ [0, 0.5] & \text{si } s_i = s_0 \\ [-0.5, 0] & \text{si } s_i = s_g \end{cases}$$

La Fig. 3 muestra gráficamente los valores anteriores.

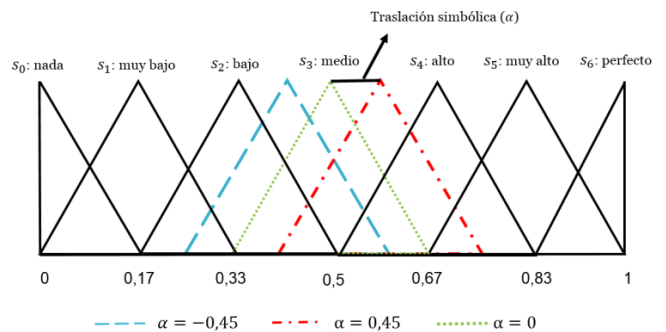


Fig. 3 Traslación simbólica y representación en 2-tupla lingüística.

Para operar con información lingüística, este modelo define la función $\Delta: [0, g] \rightarrow \tilde{S} = S \times [0.5, 0.5]$ que permite transformar un valor numérico contenido en el intervalo $[0, g]$ en la 2-tupla lingüística que representa la información equivalente:

$$\Delta(\beta) = \begin{cases} s_i, & i = \text{round}(\beta) \\ \alpha = \beta - i, & \alpha \in [0.5, 0.5] \end{cases}$$

donde round es el operador de redondeo, s_i es la etiqueta con índice más cercano a β .

β es biyectiva y $\Delta^{-1}: \tilde{S} \rightarrow [0, g]$ es definida mediante $\Delta^{-1}(s_i, \alpha) = i + \alpha$.

De esta forma la 2-tupla en \tilde{S} es identificada con el valor numérico en el intervalo $[0, g]$. La conversión de un término

lingüístico en un valor 2-tupla lingüístico consiste en añadir un valor 0 como traslación simbólica: $s_i \in S \Rightarrow (s_i, 0) \in \tilde{S}$. El modelo computacional lingüístico basado en las funciones de transformación Δ y Δ^{-1} incluye reglas de comparación, operadores de negación ($Neg(s_i) = s_j$ tal que $j = g - i$) y operadores de agregación para 2-tuplas, de los cuales los más utilizados son:

Media Aritmética extendida para 2-tupla lingüística [20]:

$$2TAM(X) = \Delta \left(\frac{1}{n} \sum_{j=1}^n \Delta^{-1}(s_i, \alpha)_j \right)$$

Media Ponderada extendida para 2-tupla lingüística [20]:

$$2TWM(X) = \Delta \left(\sum_{j=1}^n w_j \Delta^{-1}(s_i, \alpha)_j \right)$$

Donde es $X = \{(s_i, \alpha)_1, \dots, (s_n, \alpha)_n\}$ un conjunto de 2-tuplas y $W = (w_1, \dots, w_n)$ un vector de pesos tal que $w_j > 0$ y $\sum_{j=1}^n w_j = 1$.

III. UN MÉTODO LINGÜÍSTICO DIFUSO PARA EL ANÁLISIS DINÁMICO DE RIESGOS

El objetivo fundamental del método que se describe en esta sección, es apoyar la priorización de los riesgos en situaciones de decisión que requieren la obtención de una lista ordenada de riesgos durante diferentes momentos del ciclo de vida de un proyecto o una única lista de riesgos al final de todo un proceso evaluativo, pero siempre teniendo en cuenta el comportamiento actual y anterior de los riesgos.

$T = \{1, 2, \dots\}$ es el conjunto de iteraciones o momentos de evaluación considerados en el proceso de desarrollo. $R^t = \{r_i | i \in (1, \dots, m)\}$ es el conjunto de riesgos analizados en cada iteración $t \in T$. Cada riesgo es analizado de acuerdo a un conjunto de criterios $C^t = \{c_j | j \in (1, \dots, n)\}$ cuyos pesos son dados por el vector $W^t = \{w_j | j \in (1, \dots, n)\}$ con $w_j \in [0, 1]$, $\sum_{j=1}^n w_j = 1, \forall t \in T$. Normalmente los criterios analizados son probabilidad e impacto, pero el método es escalable y permite la incorporación de otros. El conjunto de expertos que evalúa los riesgos es $E^t = \{e_k | k \in (1, \dots, p)\}$. La preferencia emitida por el experto $e_k \in E^t$ sobre el criterio $c_j \in C^t$ para el riesgo $r_i \in R^t$ es representado por el término lingüístico x_{ijk}^t . El conjunto histórico de los riesgos en cada iteración es H^t .

El flujo de trabajo está compuesto por los pasos descritos a continuación:

Paso 1: Valoración de los riesgos por los expertos

En cada iteración $t \in T$, cada experto e_k emite su valoración sobre cada criterio para cada riesgo, mediante vectores de preferencias lingüísticas $(x_{ij1}^t, \dots, x_{ijp}^t)$.

Es importante destacar que debemos tener en cuenta un principio original del cálculo de la exposición al riesgo, pues si para un posible evento, su probabilidad o impacto son iguales a cero, la exposición debe ser igual a cero. Por otra parte, la probabilidad de un riesgo, recordemos, no puede ser igual a uno. El modelado lingüístico difuso respeta por naturaleza este principio al emplear variables lingüísticas con semántica difusa.

Paso 2: Cálculo del valor colectivo para cada criterio

El valor colectivo para cada criterio se nota x_{ij}^t y se calcula empleando el operador de agregación media aritmética (si todos los expertos tienen igual importancia) o la media ponderada (si deseamos diferenciar la importancia entre los expertos):

$$\begin{aligned} x_{ij}^t &= 2TAM(x_{ij1}^t, \dots, x_{ijp}^t) \\ x_{ij}^t &= 2TWM(x_{ij1}^t, \dots, x_{ijp}^t) \end{aligned}$$

Paso 3: Cálculo de la exposición estática para cada riesgo

A partir de los valores colectivos de cada criterio, se calcula la exposición estática para cada riesgo, x_i^t , empleando el operador de agregación media aritmética (si todos los criterios tienen igual importancia) o la media ponderada (si deseamos diferenciar la importancia entre los criterios).

$$\begin{aligned} x_i^t &= 2TAM(x_{i1}^t, \dots, x_{in}^t) \\ x_i^t &= 2TWM(x_{i1}^t, \dots, x_{in}^t) \end{aligned}$$

Paso 4: Cálculo de la evaluación dinámica del riesgo

En cada iteración, $t \in T$, la evaluación dinámica del riesgo, d_i^t , se calcula mediante la Ecuación:

$$d_i^t = \begin{cases} x_i^t & \text{if } r_i \in R^t \setminus H^{t-1} \\ Agg(x_i^t, d_i^{t-1}) & \text{if } r_i \in R^t \cap H^{t-1} \\ d_i^{t-1} & \text{if } r_i \in H^{t-1} \setminus R^t \end{cases}$$

El comportamiento del cálculo dependerá de las siguientes reglas:

1. Si el riesgo pertenece sólo al conjunto actual, su evaluación dinámica es igual a la exposición estática en esa iteración.
2. Si el riesgo pertenece a ambos conjuntos (histórico y actual), entonces su evaluación dinámica se obtiene mediante la agregación de la evaluación dinámica en el periodo anterior y la exposición estática en la iteración actual.
3. Si el riesgo pertenece sólo al conjunto histórico, su evaluación dinámica es la calculada en la iteración anterior.

En el método es importante el papel de las propiedades del operador de agregación Agg utilizado para el cálculo de las evaluaciones dinámicas. La asociatividad del operador evita que tengamos que guardar toda la información de todas las alternativas para todos los periodos. Además, el refuerzo [21-23] permite modelar distintas actitudes de los decisores al reforzar diferentes tendencias en las evaluaciones de los riesgos en las distintas iteraciones.

En nuestro caso deseamos un refuerzo total que normalmente se logra empleando operadores uninorma. Sin embargo, estos

operadores fueron creados para operar en el dominio [0,1] y no en el dominio lingüístico por lo que se realiza una extensión que permita operar con 2-tuplas lingüísticas, como se muestra en la siguiente ecuación:

$$Agg(x_i^t, d_i^{t-1}) = \Delta \left(\left(\frac{\frac{\Delta^{-1}(x_i^t)}{g} \cdot \frac{\Delta^{-1}(d_i^{t-1})}{g}}{\frac{\Delta^{-1}(x_i^t)}{g} \cdot \frac{\Delta^{-1}(d_i^{t-1})}{g} + 0.5 \cdot \left(1 - \frac{\Delta^{-1}(x_i^t)}{g} - \frac{\Delta^{-1}(d_i^{t-1})}{g}\right)} \right) \right)$$

Además, para la uninorma empleada se ha definido el elemento neutro $e = 0,5$ con el objetivo de modelar un refuerzo total en la agregación, de manera que:

- Cuando $\Delta(x_i^t) > 0.5$ y $\Delta(d_i^t) > 0.5$, habrá un refuerzo hacia arriba.
- Cuando $\Delta(x_i^t) < 0.5$ y $\Delta(d_i^t) < 0.5$, habrá un refuerzo hacia abajo.
- Cuando solo uno de los dos valores sea mayor que 0,5, habrá un refuerzo total.

Paso 5: Priorización de riesgos

El orden en la lista de riesgos se obtiene de acuerdo al orden decreciente de las evaluaciones dinámicas y si es necesario se seleccionan los z-urgentes, que serán los de mayor evaluación dinámica y que por consiguiente son seleccionados para darle respuesta con mayor nivel de prioridad.

Paso 6: Actualización del conjunto histórico de riesgos

Por último, debe actualizarse el conjunto histórico de riesgos teniendo en cuenta una política de retención que, como mecanismo de memoria del proceso, se encarga de seleccionar los riesgos que serán recordados de una iteración a la próxima. De este modo, la política de retención puede ser vista como la regla que define los riesgos que pertenecen al conjunto histórico. Por ejemplo, pueden mantenerse todos los riesgos que sobrepasen cierto umbral en la evaluación o los k primeros según el orden obtenido en el paso anterior. Esta iniciativa puede ser útil en problemas a gran escala, donde se manejan grandes cantidades de riesgos. La valoración de cuándo un riesgo ha sido mitigado lo suficiente, es otra de las evaluaciones subjetivas que tienen lugar en los modelos dirigidos por los riesgos [24]. En los proyectos comunes, se sugiere mantener todos los riesgos que no se consideren eliminados o ampliamente mitigados.

IV.EJEMPLO ILUSTRATIVO

Con el objetivo de mostrar el funcionamiento del método lingüístico difuso para el análisis dinámico de riesgos, en esta sección se proporciona un ejemplo ilustrativo. El ejemplo ha sido adaptado del libro [1], para así facilitar su comprensión y relación con entornos de enseñanza y aprendizaje de la ingeniería de software.

Se supone entonces que contamos con una lista inicial de ocho riesgos identificados en la etapa de planificación y que en tres iteraciones diferentes analizaremos de acuerdo a ciertas preferencias colectivas.

El listado de riesgos y su presencia en cada iteración, se muestran en la tabla 1.

TABLA 1
RIESGO EVALUADOS EN CADA ITERACIÓN.

Riesgo	Declaración	t=1	t=2	t=3
R1	Estimación de tamaño puede ser significativamente baja	x	x	x
R2	Menos reuso que el planificado	x	x	x
R3	Fecha de entrega será apretada	x	x	x
R4	Cliente cambiará requisitos	x	x	x
R5	Tecnología no satisfará las expectativas	x		
R6	Falta de capacitación en herramientas	x	x	
R7	Personal inexperto	x	x	
R8	Alta rotación de personal	x		
R9	Mayor número de usuarios que el planificado		x	x
R10	Pérdida de fondos		x	
R11	Usuarios finales que se resisten al sistema			x

Remarcamos que para lograr mayor simplicidad ya se proveen las preferencias colectivas para cada criterio, por lo que no se recopilan aquí las preferencias de múltiples expertos. Los criterios de evaluación serán los convencionales, impacto y probabilidad y para la primera iteración, los valores que se presentan en la tabla 2 fueron adaptados de los originales en [1] mediante un proceso de normalización.

Las preferencias se expresan empleando el conjunto de términos lingüísticos con la representación difusa triangular que se muestra en la Fig. 4.

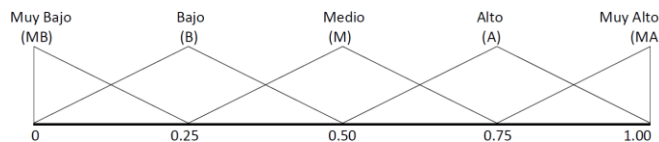


Fig. 4 Conjunto lingüístico de cinco términos simétricamente distribuidos.

Las tablas 2, 3 y 4 presentan las preferencias colectivas para cada criterio, así como la exposición estática y la evaluación dinámica para cada riesgo en cada iteración.

TABLA 2
VALORES LINGÜÍSTICOS DE IMPACTO, PROBABILIDAD Y EXPOSICIÓN AL RIESGO EN LA ITERACIÓN 1.

Riesgo	Impacto	Probabilidad	x_i^1
R1	(M,0)	(M,0.4)	(M,0.2)
R2	(M,0)	(A,-0.2)	(M,0.4)
R3	(M,0)	(M,0)	(M,0)
R4	(M,0)	(A,0.2)	(M,0.6)
R5	(B,0)	(B,0.2)	(B,0.1)
R6	(A,0)	(A,0.2)	(A,0.1)
R7	(M,0)	(B,0.2)	(M,-0.4)
R8	(M,0)	(M,0.4)	(M,0.2)

TABLA 3
VALORES LINGÜÍSTICOS DE IMPACTO, PROBABILIDAD, EXPOSICIÓN AL RIESGO Y EVALUACIÓN DINÁMICA EN LA ITERACIÓN 2.

Riesgo	Impacto	Probabilidad	x_i^2	d_i^2
R1	(B,0)	(M,0)	(M,-0.5)	(M,-0.31)
R2	(B,0)	(M,0)	(M,-0.5)	(M,-0.11)
R3	(M,0)	(M,0)	(M,0)	(M,0)
R4	(A,0)	(A,0)	(A,0)	(A,0.39)
R6	(M,0)	(A,0.20)	(A,-0.40)	(A,0.46)
R7	(B,0)	(B,0.20)	(B,0.10)	(B,-0.19)
R8	(A,0)	(B,0.20)	(M,0.10)	(M,0.10)
R10	(B,0)	(M,-0.40)	(B,0.30)	(B,0.30)

TABLA 4
VALORES LINGÜÍSTICOS DE IMPACTO, PROBABILIDAD, EXPOSICIÓN AL RIESGO Y EVALUACIÓN DINÁMICA EN LA ITERACIÓN 3.

Riesgo	Impacto	Probabilidad	x_i^3	d_i^3
R1	(B,0)	(B,0)	(B,0)	(B,-0.21)
R2	(A,0)	(A,0)	(A,0)	(A,-0.08)
R3	(B,0)	(B,0)	(B,0)	(B,0)
R4	(B,0)	(MA,0)	(A,-0.50)	(A,-0.39)
R9	(A,0)	(A,0)	(A,0)	(A,0.07)
R11	(A,0)	(M,-0.40)	(M,0.30)	(M,-0.05)

Algunas observaciones pueden ilustrar la eficacia del método en la modelación y análisis de la evolución de la exposición al riesgo.

- El uso del modelo 2-tupla nos permite hacer diferenciaciones en los resultados a través de la traslación simbólica, pero sin perder la interpretabilidad que nos sugiere la obtención de una palabra como resultado.
- Para la iteración 2, R1 y R2 obtienen igual exposición estática según los modelos tradicionales, sin embargo, el valor de R2 en la iteración 1 fue mayor, lo que implica que teniendo en cuenta su evolución, este riesgo es más urgente. Esta lógica es la que se obtiene con un valor mayor de la evaluación dinámica para R2.
- Otro comportamiento que logramos modelar es el de las tendencias en la evaluación. R4 aumenta su exposición con valores por encima de “Medio” por lo que su evaluación dinámica en la iteración 2, es mayor que sus valores de exposición en las iteraciones 1 y 2: se realiza un refuerzo hacia arriba en la evaluación final. Sin embargo, R7 no solo disminuye su exposición, sino que en ambos casos tiene valores por debajo de “Medio”, lo que indica que puede ser un riesgo a ignorar en próximas iteraciones, debido a que parece lógico que en la medida que avanza el desarrollo del proyecto, el equipo adquiere mayor experticia en el empleo de las tecnologías y el propio desarrollo del software. Entonces en este caso, nuestro método realiza un refuerzo hacia abajo de la evaluación dinámica.
- El refuerzo total puede comprobarse con el resultado de la evaluación dinámica para R2. En la iteración 2 su evaluación dinámica está por debajo de “Medio” pero en la iteración 3 aumenta la exposición hasta un valor

por encima, en este caso el resultado está entre los dos valores.

Finalmente, las listas ordenadas de riesgos para la priorización se presentan a continuación:

- Iteración 1: R6>R4>R2>R1=R8>R3>R7>R5
- Iteración 2: R7>R10>R1>R2>R3>R9>R4>R6
- Iteración 2: R1>R3>R11>R4>R2>R9

V.Conclusiones

El análisis de riesgos en los modelos de proceso de software es un aspecto de medular importancia debido a que pueden facilitar que el equipo de desarrollo se centre, en cada iteración, en resolver los requisitos y deficiencias realmente significativas. Los métodos tradicionales de análisis de riesgos presuponen que el listado de riesgos es estático durante el desarrollo del proyecto de software y proponen escalas y combinaciones cualitativas fijas. Para solucionar estas limitaciones, en este artículo se ha presentado un método lingüístico difuso para el análisis dinámico de riesgos en modelos de procesos de software guiados por los riesgos.

El método propuesto está basado en la toma de decisión dinámica por lo que permite tener en cuenta las variaciones en la exposición a los riesgos del proyecto de software, ya sea por la aparición de nuevos riesgos o porque lógicamente y eficazmente, el equipo ha logrado mitigar algunos. Una uninorma extendida para 2-tupla lingüística es utilizada como operador en el cálculo de la evaluación dinámica, por lo que no es necesario guardar toda la información del problema de decisión a lo largo del tiempo.

Además, se modela la incertidumbre de la información sobre los riesgos mediante el modelo de representación difuso basado en 2-tuplas lingüísticas. Esto permite realizar procesos de computación con palabras utilizando el modelo computacional basado en 2-tuplas lingüísticas y con ello, obtener resultados lingüísticos cercanos al lenguaje natural y de fácil interpretación.

El método extiende formalmente los modelos tradicionales de análisis de riesgos para el tratamiento de las valoraciones de múltiples expertos. Por otra parte, permite, de manera flexible, incorporar no solo el análisis basado en criterios tradicionales como el impacto y la probabilidad, sino que también pueden considerarse otros en correspondencia con las necesidades del proyecto y la importancia de cada criterio en el resultado final.

El desarrollo de un ejemplo ilustró el funcionamiento de método y nos llevó a plantearnos como trabajos futuros, la aplicación en desarrollos y problemas reales de mayor complejidad ingenieril y en la naturaleza de los riesgos y con una mayor cantidad de iteraciones, dinamismo e incertidumbre.

REFERENCIAS

- [1] Pressman, R.: Ingeniería del Software. Un Enfoque Práctico. 7ma edición, McGraw-Hill (2010)
- [2] Zulueta, Y.; Despaigne, E.; Hernández, A.: La gestión de riesgos en la producción de software y la formación de profesionales de la informática: experiencias de una universidad cubana. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software, vol. 5, núm. 3, octubre, pp. 6-20 Asociación de Técnicos de Informática, Madrid, España (2009)

- [3] Boehm, B.: Software Risk Management: Principles and Practices, IEEE Software, p.32-41 (1991).
- [4] Charette, R. N.: Software Engineering Risk Analysis and Management, McGraw-Hill/ Intertext (1989).
- [5] CMMI: CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development, V1.1, Tech. Rep. CMU/SEI-2002-TR-004, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, 2002
- [6] Dorofee, A.; J. Walker, et al. Continuous Risk Management Guidebook. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, p. 187 (1996).
- [7] IEC. Part 3—Guide to Risk Analysis of Technological Systems. En: CEI/IEC 300–3–9:1995 Risk Management. Geneva, International Electrotechnical Commission (1995).
- [8] IEEE: Standard 1540–2001: Standard for Software Life Cycle Processes–Risk Management. IEEE Standards Association (2001).
- [9] McConnell S.: Rapid Development, Microsoft Press (1996).
- [10] PMI: Guía de los Fundamentos para la dirección de proyectos (Guía del PMBOK). Pennsylvania, EE.UU. Project Management Institute, Inc., p. 502 (2009).
- [11] Boehm, B.: The Spiral Model as a Tool for Evolutionary Software Acquisition, CrossTalk, (2001), disponible en www.stsc.hill.af.mil/crosstalk/2001/05/boehm.html.
- [12] Campanella, G. y Ribeiro, R.A.: A framework for dynamic multiple-criteria decision making, Decision Support Systems 52 (2011), no. 1, 52-60.
- [13] Xu, Z.S.: On multi-period multi-attribute decision making, Knowledge-Based Systems 21 (2008), no. 2, 164-171
- [14] Zhu, Q., Li, H. y Yu, M.: Dynamic multiattribute decision making based on advantage retention degree, Journal of Information & Computational Science 10 (2013), no. 04, 1105-1119.
- [15] Zulueta, Y., Martínez, J., Bello, R. y Martínez, L.: A discrete time variable index for supporting Dynamic Multi-Criteria Decision Making: International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, Volumen 22, No. 1, pp1–22, 2014.
- [16] Bonissone, P.P.: A fuzzy sets based linguistic approach: Theory and applications, Proceedings of the 12th Conference on Winter Simulation (Piscataway, NJ, USA), WSC '80, IEEE Press, 1980, pp. 99-111
- [17] Zadeh, L.: The concept of a linguistic variable and its application to approximate reasoning-I, Information Sciences 8 (1975), no. 3, 199-249
- [18] Zadeh, L.: Fuzzy logic = computing with words, IEEE Transactions on Fuzzy Systems 4 (1996), no. 2, 103-111.
- [19] Herrera, F. and Martínez, L.: A 2-tuple fuzzy linguistic representation model for computing with words, IEEE Transactions on Fuzzy Systems 8 (2000), no. 6, 746-752.
- [20] Martínez L., Rodríguez R.M., Herrera F.: The 2-tuple Linguistic Model. Computing with Words in Decision Making, 1st edn. Springer International Publishing, New York (2015).
- [21] Yager, R.R. and Rybalov, A.: Uninorm aggregation operators, Fuzzy Sets and Systems 80 (1996), no. 1, 111-120.
- [22] Yager, R.R. and Rybalov, A.: Full reinforcement operators in aggregation techniques, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 28 (1998), no. 6, 757-769.
- [23] Yager, R.R. and Rybalov, A.: Bipolar aggregation using the uninorms, Fuzzy Optimization and Decision Making 10 (2011), no. 1, 59-70.
- [24] Fairbanks, G.: Just Enough Software Architecture: A Risk-Driven Approach. Marshal and Brainerd (2010), disponible en rhinoresearch.com.