

Modelo de Gestión de la Planificación del Desarrollo de Software como un Proceso Estocástico

Fernando Lucas Rivera Bernsdorff

Universidad Católica de Salta. Campo Castañares s/n, 4400, Salta. Argentina.

Universidad Nacional de Salta. Av. Bolivia 5150, 4400, Salta. Argentina.

rivera.fernando@gmail.com

Abstract. La simulación nos permite modelar, entender y visualizar procesos que son complejos y no lineales. Los procesos involucrados en el desarrollo de software cumplen con ambas características y son en esencia procesos estocásticos. En la ingeniería de software se busca gestionar los procesos de desarrollo de software, pero se hace muy difícil poder llegar a planes y estimaciones con precisión. Por este motivo surgen los modelos incrementales de desarrollo de software, los ciclos de vida y la ingeniería de software en general. En el presente trabajo se presenta un modelo y una solución que aplica simulación a la ingeniería de software en base a la información histórica asociada a la gestión de proyectos. Es decir que se busca una solución considerando el proceso de planificación de desarrollo de software como un proceso estocástico, con el propósito de planificar nuevos proyectos utilizando simulación.

Keywords: Ingeniería Software Simulación Estocástico Planificación

1 Introducción

La simulación nos permite modelar, entender y visualizar procesos que son complejos y no lineales [20] [21]. Los procesos involucrados en el desarrollo de software (DSW) cumplen con ambas características, son complejos y no lineales. Cuando hablamos de complejidad, hacemos referencia a situaciones donde un cambio puede tener efectos en cadena con consecuencias amplias y difíciles de controlar. En particular consideramos lo que Peter Senge denomina complejidad dinámica. [20] [21]

En la ingeniería de software (ISW) se busca gestionar y tener bajo control dichos procesos de desarrollo de software (PDS), pero debido al carácter dinámico y sistémico de los mismos, a su complejidad y los riesgos con su probabilidad de ocurrencia, se hace muy difícil poder llegar a planes y estimaciones con precisión [2] [12] [23]. Por este motivo surgen los modelos incrementales de DSW, los ciclos de vida y en última instancia se busca aumentar la capacidad para poder administrar mejor los PDS e ir madurando en dicha capacidad. [23]

Las técnicas de estimación son necesarias y más aún es necesario que las mismas busquen una mejora continua, se aproximen cada vez más a la realidad y contemplen

adfa, p. 1, 2011.

© Springer-Verlag Berlin Heidelberg 2011

los riesgos asociados. En general dichas técnicas de estimación tienen características estáticas [14], es decir que buscan basar la planificación de futuros proyectos y actividades tomando los datos del pasado pero considerando una estimación fija sin probabilidad de ocurrencia, lo cual lleva a que muchas veces se tomen promedios en las estimaciones, máximos, mínimos o algún otro valor fijo de referencia. El problema de gestionar los PDS considerando estimaciones basadas en promedios o valores con mayor probabilidad de ocurrencia es que se pierde información de la complejidad dinámica propia de dichos procesos. [14] [19]

2 El desarrollo de software como proceso estocástico

Una característica importante de dichos procesos es que no son determinísticos [5], lo cual significa que dadas ciertas variables de entrada no siempre se obtiene la misma salida [4] [6]. Estos procesos son estocásticos, es decir que dados los mismos datos de entrada puede haber más de un resultado, cada uno con una probabilidad de ocurrencia [4] [6]. Por ejemplo, para una actividad de un tipo dado en un proyecto, en distintas situaciones puede demandar distintas cantidades de tiempo y según los riesgos potenciales, dichos tiempos pueden variar aún más.

Es una tendencia simplificar dichas estimaciones tomando promedios, promedios ponderados, máximos, mínimos, últimas mediciones o tiempos fijos definidos por expertos como valores de referencia para realizar estimaciones, pero muchas veces dichas estimaciones están mejor representadas por distribuciones de probabilidad, ya que consideran los tiempos variables que pueden demandar dichas tareas y permite que no se pierda información valiosa debido a dichas simplificaciones.

La simulación de procesos de desarrollo de software (SPDS) posibilita ejecutar situaciones de la realidad en contextos controlados, pero con la capacidad de considerar el dinamismo, la complejidad y el carácter sistémico antes mencionado.

3 Simulación en el proceso de planificación del desarrollo de software

Actualmente no es una práctica común aplicar simulación en los procesos involucrados en la ISW y en la gestión de proyectos, debido a que las soluciones disponibles para simular no se encuentran integradas a herramienta de planificación de proyectos ni se basa en distribuciones de probabilidad sobre los datos históricos de planificaciones pasadas, lo cual hace que demande tiempo y esfuerzo, y no sea práctica su aplicación. [5] [12] [14]

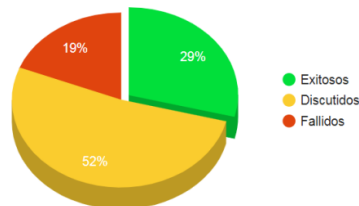


Fig. 1. Probabilidad de éxito de un proyecto de desarrollo de software. [9]

Como puede apreciarse en la figura 1 sigue siendo muy baja la probabilidad que un proyecto de software con cierta complejidad sea exitoso. Poder simular PDS podría ayudar a incrementar la probabilidad de éxito, considerando como uno de los criterios de éxito de un proyecto, que el mismo cumpla con los tiempos estimados. [9]

El presente trabajo presenta un modelo y una solución para incorporar la simulación a la planificación de proyectos de desarrollo de software (PPDS) y a la ISW en general, integrando los datos históricos de gestión de proyectos con la posibilidad de simular los PDS.

4 Antecedentes y actuales líneas de investigación

Aunque hay trabajos de investigación sobre simulación en ingeniería de software y proyectos de software, no hay herramientas que integren la gestión de proyectos y la simulación, como tampoco un modelo de gestión de proyectos que incluya la posibilidad de realizar simulaciones considerando la base de datos histórica de proyectos.

Se han detectado trabajos que aplican simulación en la PPDS [1] [7] [8] [10] [11] [13] [14] [15] [16] [17] [18] [19] [22], pero los mismos no se basan en el aprendizaje considerando la historia de tareas y los tiempos que las mismas demandaron, sino que se basan en procesos que utilizan probabilidades configurables, pero desconectadas de la información disponible y la experiencia de un equipo de DSW.

Por otro lado, se encuentran trabajos de investigación sobre simulación en la enseñanza de ISW y para la formación en la planificación de proyectos [10], lo cual es muy útil para poder entender la complejidad dinámica de los PDS. Se ha observado que la mayor cantidad de trabajos de investigación de simulación en ISW se centran en la enseñanza y los mismos son un buen aporte para el presente proyecto.

Por último, es importante destacar que no se han encontrado proyectos que integren la gestión de proyectos y su información histórica con la simulación de la planificación de nuevos proyectos.

5 Modelo para poder simular el proceso de desarrollo de software

Para poder simular un proyecto de desarrollo de software se debe contar con los siguientes elementos:

- Una taxonomía de tareas que permita utilizar los tiempos históricos en nuevas planificaciones de desarrollo de software, es decir tareas que tengan el mismo tipo de tarea en planificaciones pasadas y realizadas.
- Datos históricos de tiempos que demandaron los distintos tipos de tareas.
- Generación de las variables aleatorias que representan a cada tipo de tarea en base a los datos históricos.
- Definición de una planificación en cuanto a tareas a realizar. Cada tarea asignada a un tipo de tarea de acuerdo a una taxonomía definida. La dependencia entre tareas y el tipo de recurso que realizará cada tarea.

6 Taxonomía de tareas

Es fundamental definir una taxonomía de tareas para que la información histórica pueda ser utilizada correctamente en las simulaciones y con la mayor flexibilidad posible. Cada taxonomía soportará un modelo de DSW. En el presente trabajo se utilizará una clasificación de tareas para mostrar un caso aplicado al modelo y para poder ver como la simulación permite analizar información que no se encuentra disponible cuando la planificación se realiza con tiempos fijos, sin probabilidad de ocurrencia [14]. El tipo de tarea es un atributo clave dentro de los mencionados en [19].

Cada taxonomía debe garantizar que los tipos de tareas sean excluyentes, es decir, una misma tarea sólo puede pertenecer a un tipo de tareas. Para esto consideramos muy importante mantener una estructura jerárquica sobre dicha taxonomía y lo que menciona Antonelli, sobre adoptar una estructura de requerimientos que permita la correcta gestión de los mismos [3]. Es decir que como se modele la ingeniería de requerimientos será clave en la aplicación de dichos datos en las futuras simulaciones.

Aunque cada equipo de DSW podría definir su propia clasificación de tareas, para poder compartir información histórica de distintos proyectos, sería conveniente que distintos equipos adopten el mismo modelo de taxonomía, lo cual permite que distintos grupos de desarrollo puedan utilizar la información estadística de otros grupos de trabajo y así poder generar una base de conocimiento para enriquecer a todos los equipos que adopten los mismos estándares.

Una valiosa contribución a este proyecto han sido las propuestas realizadas por el Software Engineering Institute (SEI) buscando llevar más allá su modelo de madurez y capacidad con la propuesta de aplicar simulación [19]. Tomando como referencia dicho trabajo, se propone un modelo que permita implementar la simulación en esce-

narios reales, aplicar la simulación en la PPDS y de esta manera llevar dicha propuesta a la realidad.

7 Generación de variables aleatorias para cada tipo de tarea

Para poder implementar la solución propuesta se debe generar la aleatoriedad propia de los procesos estocásticos y utilizarla para modelar los PDS en base a variables aleatorias que representen los mismos.

En primer lugar generamos la aleatoriedad utilizando un método matemático que posea buenas propiedades estadísticas y a su vez sea eficiente. Para ello utilizaremos un generador congruencial lineal (linear congruencial generator) o GCL, introducidos por Lehmer. Se basan en el concepto matemático de números congruentes. Un número entero X es congruente con Y módulo n , si $X - Y$ es divisible por n , es decir, X e Y dan el mismo resto al ser divididos por $n \rightarrow X \equiv Y \pmod{n}$. [6]

Los mismos se generan en secuencia con la siguiente fórmula recursiva:

$$Z_i = (aZ_{i-1} + c) \pmod{m} \quad (1)$$

Para transformar la distribución uniforme en cada una de las distribuciones correspondientes a cada tipo de tarea definida en la taxonomía elegida, se utilizará el método de la transformada inversa. [4][6]

Este método se basa en la función inversa de la función de distribución de la variable aleatoria X . Dado que $F(X)$ está acotada entre 0 y 1, podemos generar valores en $U(0; 1)$ y sobre ellos aplicar $F^{-1}(X)$.

Se utiliza la función de distribución acumulada $F(x)$. Como $F(x)$ está definida entre 0 y 1, entonces se necesitan generar números aleatorios uniformemente distribuidos entre 0 y 1.

Si representamos un número aleatorio con R , entonces:

$$\begin{aligned} R &= F(x) \\ x &= F^{-1}(R) \end{aligned}$$

Este método habría que aplicarlo para cada tipo de tarea de la taxonomía definida para la gestión de proyectos. Buscando simplificar la generación de variables aleatorias, ya que deberíamos conocer cada $f(x)$ asociado a cada tipo de tarea y calcular sus integrales y luego las funciones inversas, en lugar de tomar variables continuas generaremos en forma dinámica una variable aleatoria discreta para cada categoría de tarea [6], lo cual se explica a continuación con un caso práctico, y con dicho caso luego se muestran los resultados de las simulaciones realizadas en el presente trabajo.

8 Generación de variables aleatorias para cada tipo de tarea

En lugar de buscar la función de distribución continua de cada tipo de tarea, el modelo utiliza variables discretas. Para garantizar que sea correcta y útil la distribución, el tamaño de cada intervalo dependerá de la longitud entre los extremos del dominio de la función, a los que llamaremos a y b, con un máximo número de intervalos, pero considerando que las muestras deben ser significativas.

Para poder generar simulaciones de casos reales se contó con la base histórica de gestión de proyectos de la empresa North Valley Software que fue compartida para el presente análisis, ya que para simular se necesita un gran número de datos históricos.

ID	Tipo de Tarea	Horas
1	Objeto Principal de Negocio	3,17
2	Objeto Principal de Negocio	2,58
3	Objeto Principal de Negocio	4,23
4	Objeto Principal de Negocio	4,36
⋮	⋮	⋮
48	Objeto Principal de Negocio	4,23
49	Objeto Principal de Negocio	3,96
50	Objeto Principal de Negocio	3,53

Table 1. Tiempos históricos de la muestra utilizada para generar la variable aleatoria.

En la tabla 1 se puede apreciar una muestra de los tiempos históricos que llevó el tipo de tarea seleccionado para explicar la generación de la variable aleatoria discreta, se toman sólo 50 datos históricos para una mejor comprensión, pero en la realidad la variable aleatoria se genera con todos los datos disponibles o se genera con los datos a partir de una fecha configurable.

count	50.000000
mean	4.264000
std	1.176493
min	2.500000
25%	3.530000
50%	4.130000
75%	4.847500
max	8.200000

Table 2. Propiedades del conjunto de datos de la muestra.

En este caso se puede apreciar que si se toma el tiempo estándar como el promedio que llevó dicho tipo de tarea en el pasado, el mismo es de 4,264 hs.

Se pueden observar en la tabla 2 los datos principales de la muestra. Dichos datos como la media, máximos y mínimos serán utilizados en las comparaciones entre distintos modelos y simulaciones.

En este punto es importante destacar que se puede detectar en el caso considerado, que existen grandes diferencias si se utiliza el promedio para hacer una planificación o se utiliza un enfoque optimista (mínimo) o pesimista (máximo). En el caso de considerar el promedio, el mismo es de 4,264 hs., en cambio si se considera el criterio optimista, que tomaría el valor mínimo de 2,5 hs. También se puede observar que es muy distinto el valor tomado en un criterio pesimista, es decir el máximo tiempo que llevo dicho tipo de tarea, el cual es de 8,2 hs.

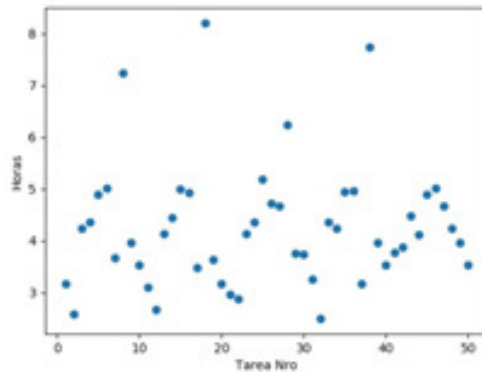


Fig. 2. Diagrama de dispersión de los datos históricos.

En la Figura 2 se visualiza el diagrama de dispersión donde los datos históricos se concentran entre 2,5 y 5 hs.

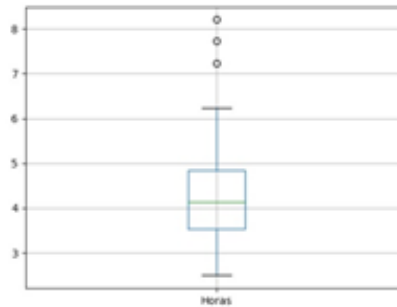


Fig. 3. Diagrama de caja de los datos históricos.

En la Figura 3 puede apreciarse el diagrama de cajas, donde resulta que se detectan 3 casos como valores atípicos. Los cuales se podrían considerar como outliers o valores fuera de rango.

Como mencionamos anteriormente la cantidad de intervalos que toma el modelo está asociada a la longitud del dominio de la muestra y el tamaño de la muestra, en este caso se consideran 15 intervalos, los intervalos siempre son de igual tamaño y cada intervalo tendrá su frecuencia asociada. No se divide por frecuencia sino que se divide por tamaño $(b-a) / 15$, siendo los valores b y a , los extremos del dominio.

En este caso puntual como se puede observar en la tabla 2, b es el máximo de los datos históricos igual a 8,2. Y a es el mínimo de los datos históricos igual a 2,5. Y como en este caso se toman 15 intervalos ya que el tamaño de la muestra lo permite, el tamaño de cada intervalo se obtiene de la siguiente manera:

$$(b - a) / 15 = (8,2 - 2,5) / 15 = 5,7 / 15 = 0,38$$

Es decir que cada intervalo será de 0,38 hs. El primer intervalo será de 2,5 a $2,5 + 0,38$, es decir de 2,5 a 2,88. El siguiente de 2,88 a $2,88 + 0,38$, es decir de 2,88 a 3,26 y así sucesivamente.

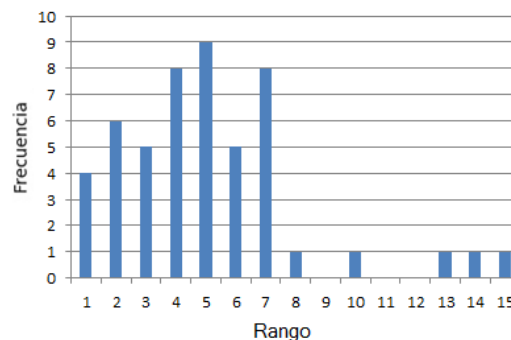


Fig. 4. Variable aleatoria discreta generada a partir de los datos históricos.

Con la variable aleatoria generada en base a la muestra de 50 casos de los datos históricos del tipo de tarea Objeto Principal del Negocio, se pueden realizar simulaciones de una PPDS.

Con la variable aleatoria generada, cuyo resultado se ve en la Figura 4 y generando de esta manera las variables aleatorias asociadas a cada tipo de tarea de la taxonomía utilizada, se pueden comparar los resultados de utilizar la simulación con los resultados de no utilizarla, es decir utilizando promedios, máximos (criterio pesimista) o mínimos (criterio optimista).

El modelo de la Figura 4 lo necesitamos representar en un modelo numérico que permita aplicar el algoritmo de simulación. Para ello se toma la cantidad de tiempos históricos del tipo de tarea considerado para generar la variable aleatoria y se los asigna a los rangos antes mencionados. Por ejemplo en el caso del primer rango se

dan 4 ocurrencias, es decir que de las 50 muestras históricas consideradas y expresadas en la Tabla 1, hay 4 casos en los que la tarea llevo entre 2,5 y 2,88 hs. que corresponde al primer rango de la variable aleatoria.

Rango	2,50-2,88	2,88-3,26	3,26-3,64	3,64-4,02
Ocurrencias	4	6	5	8
Prob	0,08	0,12	0,1	0,16
Acumulada	0,08	0,2	0,3	0,46
Rango	4,02-4,40	4,40-4,78	4,78-5,16	5,16-5,54
Ocurrencias	9	5	8	1
Prob	0,18	0,1	0,16	0,02
Acumulada	0,64	0,74	0,9	0,92
Rango	5,54-5,92	5,92-6,30	6,30-6,68	6,68-7,06
Ocurrencias	0	1	0	0
Prob	0	0,02	0	0
Acumulada	0,92	0,94	0,94	0,94
Rango	7,06-7,44	7,44-7,82	7,82-8,20	Total
Ocurrencias	1	1	1	50
Prob	0,02	0,02	0,02	1
Acumulada	0,96	0,98	1	1

Table 3. Probabilidades de los distintos rangos de la variable aleatoria, utilizadas para aplicar la transformada inversa discreta.

9 Generación de las variables aleatorias discretas para aplicar en la simulación

Como primer paso se generan los números aleatorios como se explicó anteriormente. Se toma un lote de los números aleatorios generados para mostrar los pasos que se siguen en el proceso de generación de la variable aleatoria y la simulación.

0,938326384515669	0,961610370921566	0,539794711788408	0,589487323128248	0,351559871254505
0,238001076010973	0,946090486249726	0,911679234654608	0,695398579592414	0,433663737152755
0,075314992415108	0,979384221072374	0,893353051454708	0,394486071163595	0,542754082341033
0,185900242820918	0,644742618207279	0,586619343314543	0,194337552643846	0,299320963300800
0,674733400351656	0,787925220178090	0,969016507813707	0,271908312873207	0,609621705673400
0,987736522995808	0,856701049474748	0,441225027266350	0,939248742181585	0,230625912280292

Table 4. Números aleatorios generados.

La asignación de rango se aplica por medio de la transformada inversa. En este caso se aplica la transformada inversa para variables aleatorias discretas. En el caso del primer número aleatorio generado 0,938326384515669 que representa la primera tarea del tipo de tarea Objeto Principal del Negocio simulada en la planificación. Dicho número se transforma en un rango de la variable discreta aplicando la inversa de

la acumulada. Es decir que en la tabla 3 se busca donde se ubica 0,938326384515669 en la acumulada y nos da que corresponde al rango 5,92-6,30.

Luego de generar las horas de las tareas por medio de los números aleatorios, las mismas se adjudican a uno de los 15 rangos según corresponda, lo cual dará como resultado el tiempo final para simular cada tarea, que es el tiempo que representa al rango.

Utilizando la transformación obtenida en los pasos anteriores para convertir la distribución uniforme de los números aleatorios a la distribución basada en la muestra de los datos históricos y graficada en la Figura 4, se obtienen como resultado los tiempos en las tareas a simular.

Cada uno de los números aleatorios de la tabla 4 se corresponde con cada una de las horas de la tabla 5 con el mismo orden, es decir que el primer número aleatorio generado 0,938326384515669 se corresponde con el tiempo de 6,11 hs. que lleva la tarea y a su vez, por considerar la variable aleatoria como discreta se la asigna al rango de tareas 5,92-6,30.

Hs	Rango	Hs	Rango	Hs	Rango	Hs	Rango	Hs	Rango
6,11	5,92-6,30	7,63	7,44-7,82	4,21	4,02-4,40	4,21	4,02-4,40	3,83	3,64-4,02
3,45	3,26-3,64	7,25	7,06-7,44	5,35	5,16-5,54	4,59	4,40-4,78	3,83	3,64-4,02
2,69	2,50-2,88	7,63	7,44-7,82	4,97	4,78-5,16	3,83	3,64-4,02	4,21	4,02-4,40
3,07	2,88-3,26	4,59	4,40-4,78	4,21	4,02-4,40	3,07	2,88-3,26	3,45	3,26-3,64
4,59	4,40-4,78	4,97	4,78-5,16	7,63	7,44-7,82	3,45	3,26-3,64	4,21	4,02-4,40
8,01	7,82-8,20	4,97	4,78-5,16	3,83	3,64-4,02	6,11	5,92-6,30	3,45	3,26-3,64

Table 5. Horas generadas por la transformación de los números aleatorios y la asignación de las mismas a los rangos de la variable aleatoria discreta.

La distribución obtenida es una de las generadas en la simulación y corresponde a los valores obtenidos y mostrados en la Tabla 5. y los mismos se basan en los números de la Tabla 4.

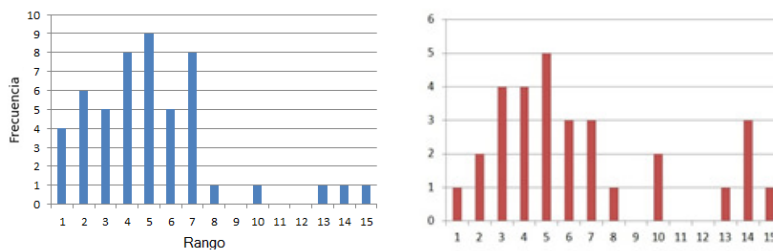


Fig. 5. Comparación de la variable aleatoria discreta generada a partir de los datos históricos y la distribución de los datos obtenidos en una de las simulaciones realizadas

La distribución que se visualiza en la Figura 5 en rojo, es una de las tantas generadas en la simulación y corresponde a los números aleatorios generados, que pueden verse en la Tabla 4, y que fueron transformados según la distribución de probabilidad del tipo de tarea como puede apreciarse en la Tabla 5.

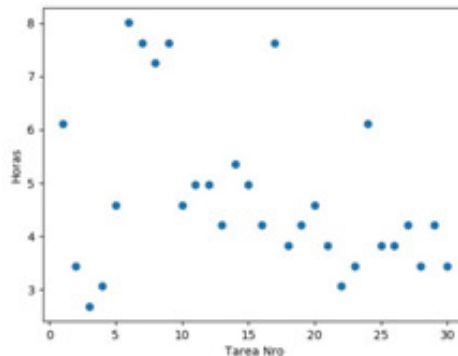


Fig. 6. Dispersión de los datos obtenidos en una de las simulaciones realizadas

En la figura 6 se observa la dispersión para compararla con la dispersión de los datos históricos. Algunos datos pueden parecer fuera de rango, pero la consideración de los mismos implica incluir los riesgos, ya que son datos reales y confirmados. Es decir que hay riesgo que este tipo de tarea demande un tiempo de 8,2, más allá que el promedio sea de 4,264.

10 Modelo subyacente para poder simular en base a los datos históricos representados estocásticamente

Una vez generadas las variables aleatorias que representan cada tipo de tarea en base a los datos históricos, lo cual se realiza en forma dinámica como fue explicado anteriormente; se aplica la simulación en una nueva planificación de desarrollo de software representada con la siguiente información:

- Tareas a realizar con sus duraciones en tiempo. Cada tarea asignada a un tipo de tarea de acuerdo a la taxonomía de tareas definida.
- Las dependencias entre las tareas.
- El tipo de recurso que realizará cada tarea.
- Los recursos disponibles por tipo de recurso y las cantidades de dichos recursos en cada intervalo de tiempo del proyecto.

Esta información es la que se suele llevar al momento de planificar un proyecto de desarrollo de software. En este caso la información se integra con los datos históricos de proyectos, lo cual permite realizar simulaciones sin requerir un trabajo adicional.

11 Aplicación de la simulación a un caso concreto y resultados obtenidos

Uno de los casos en los que se aplicó el modelo de simulación propuesto corresponde a un proyecto con 94 tareas finales de la base de datos de gestión de proyectos antes mencionada. De las 94 tareas del proyecto, el tipo de tarea OPN – Objeto Principal del Negocio, utilizado para explicar la generación de las variables aleatorias, corresponde a 30 de las 94 tareas del proyecto.

Un punto destacado para evaluar en este proyecto y por eso se lo eligió, es que la empresa tuvo una demora significativa en los tiempos del mismo y la simulación permitió obtener información útil en cuanto a los riesgos del proyecto, que podría haber mostrado y anticipado que el proyecto podía requerir más tiempo para su finalización. Este análisis es “ex post” ya que se realiza cuando el proyecto se encuentra finalizado.

Incluso considerando sólo la simulación del tipo de tarea Objeto Principal del Negocio para facilitar la comprensión, se puede observar la diferencia de considerar o no simulación en la planificación del proceso de desarrollo de software. Considerando la simulación de las 30 tareas cuyos tiempos simulados son los de la tabla 5. Podemos observar que el tiempo total es de 143,4 hs. y si calculamos el promedio, es de 4,78 hs. muy distinto al promedio histórico de 4,264 (Tabla 2).

Para ver si era un caso aislado se generaron varios flujos de números para realizar simulaciones y se detectó que gran parte están cercanos al promedio, pero un porcentaje no menor (cercano al 20%) da una generación de números que al transformarlos según lo planteado anteriormente y calculando el promedio tienen una diferencia de entre el 10% y 11% con el promedio de la muestra histórica.

Como se tiene registro de los datos históricos, se consultó el motivo del retraso real del proyecto en su momento y el mismo estaba dado porque un integrante del proyecto dejó el mismo a las 2 semanas del inicio y se necesitó una adaptación a los estándares y al framework de desarrollo de la empresa que hizo que todas las tareas demanden más de lo previsto. Consultando los datos históricos, muchos de los tiempos altos en la tarea considerada se relacionan con este tipo de situaciones. Es decir que no era un hecho aislado y se podía y debía considerar para planificar.

Al momento en que la empresa realizó el proyecto, consideraba un tiempo de 4 hs para planificar la tarea en cuestión, es decir un tiempo fijo y el cual era menor que el promedio general histórico que es de 4,264 hs y más tiempo que el promedio si no se consideran en el mismo los tiempos que aparentemente podían ser valores extremos o outliers, que no lo son ya que se comprobó que son valores históricos reales.

La simulación realizada tiene una diferencia significativa si se toma un enfoque pesimista considerando el máximo de la tabla 2 de 8,2 hs. con un total de 246 hs. para

las 30 tareas, es decir que si el caso es simulado con el enfoque pesimista da con un tiempo 92,30 % mayor al promedio y si se evalúa el enfoque pesimista con el caso simulado el mismo da un 71,55% superior.

Algo similar sucede si se utiliza un enfoque optimista tomando el mínimo de la tabla 2 de 2,5 hs. con un total de 75 hs. Si se planifica considerando el promedio da un 70,56% superior al enfoque optimista y el caso simulado da un 91,2% superior al optimista.

En el caso de considerar el promedio de 4,264 hs., aunque gran parte de las simulaciones se asemejan a dicho número, un número no menor genera resultados distintos. Si se considera el promedio para planificar da un total de 127,92 hs. con un 12,10% de diferencia del plan simulado.

Para una mejor comprensión se explicó en detalle la simulación considerando sólo uno de los tipos de tareas del proyecto, la cual es una de las más utilizadas en el proyecto y la empresa. Aplicando simulación a todo el proyecto se observaron resultados similares, pero amplificadas debido a la mayor duración del mismo.

El poder generar escenarios con los enfoques pesimistas, optimistas y otros casos simulados utilizando las variables aleatorias en base a la información histórica permite visualizar y considerar posibles riesgos y tiempos que puede llevar un proyecto. Esta información permite adelantarse a los diferentes resultados posibles y actuar en consecuencia considerando la PPDS como un proceso estocástico y no como un proceso determinístico.

Analizando los resultados y comparándolos con los datos históricos de proyectos que la empresa suministró, se confirma que disponer de dicha información puede mejorar la planificación y en caso de tomar riesgos, asumirlos sabiendo que escenarios se podrían dar.

Las simulaciones del proyecto completo consideran escenarios posibles con duraciones cercanas a los 4 meses, las cuales se pueden comparar con el enfoque optimista que da cerca de 1,7 meses (37,5 días), con el pesimista de 5,6 meses, considerando meses de 22 días hábiles, y con 2,9 meses de duración si se considera el promedio para planificar. Es decir que gracias a las simulaciones se pudo observar que el tiempo estándar de 4 hs. que consideraban, hacía que tomen un valor menor al tiempo real, incluso se pudo visualizar que si se considera el promedio, el riesgo de no poder cumplir con el proyecto es muy alto. Un nivel de confianza adecuado lo daban escenarios con una duración total del proyecto superior en un 20 a 25% a los tiempos que ellos consideraban. También permitió detectar, que incluso con dicha planificación, hay riesgos que se deben considerar. Es decir que todo resultado siempre tiene una probabilidad de ocurrencia.

El resultado que más se destacó, es que teniendo los datos históricos de planificaciones anteriores y los datos actuales de planificación, detallados en el modelo de planificación subyacente, se podían generar simulaciones y distintos escenarios sin trabajo adicional.

12 Conclusiones

En el presente trabajo se describe un modelo para generar en forma dinámica las variables aleatorias que representan a cada uno de los tipos de tareas definidas en la taxonomía adoptada por una organización en su PPDS. Dichas variables aleatorias se generan a partir de datos históricos de proyectos y las mismas permiten simular nuevas PPDS.

Estas simulaciones permiten visualizar las planificaciones con la complejidad dinámica propia de los PDS, es decir que se ven como procesos estocásticos en lugar de determinísticos.

Como resultado, integrando la gestión de proyectos con simulaciones, luego de varias corridas se puede observar la dinámica y visualizar escenarios como pueden ser máximos y mínimos u otros escenarios posibles realizando un análisis de sensibilidad. Para poder realizar dichas simulaciones, se definió como base un modelo de gestión de proyectos y una taxonomía de tareas.

Al analizar porque no se utiliza la simulación en la PPDS se detecta que el mayor motivo es el trabajo y tiempo adicional que llevaría aplicar simulación y esto es porque no se dispone de una herramienta que integre los datos históricos para poder simular las nuevas planificaciones que se desean analizar, de forma tal que simular en base a datos históricos reales no requiera ninguna tarea adicional a las personas que gestionan proyectos de desarrollo de software

Los resultados obtenidos permiten visualizar la información relevante que ofrece generar escenarios de simulación en los PPDS. Por otro lado, la información analizada sobre casos de simulación confirma la relevancia de la información obtenida en procesos de simulación, pero también manifiestan que el tiempo y esfuerzo que lleva realizar simulaciones con datos históricos no facilita su utilización. Se confirma la importancia y relevancia de una herramienta que integre la gestión de proyectos y su información histórica con la posibilidad de realizar simulaciones con dichos datos, para facilitar la generación de distintos escenarios, o como alternativa poder importar dichos datos históricos para generar las variables aleatorias en las herramientas de simulación.

13 Referencias

1. Gong, Haojie; Liu, Bohan; Shao, Dong: "A Simulation Model of Kanban Software Process". 24th Asia-Pacific Software Engineering Conference, 2017.
2. Project Management Institute: A guide to the Project Management Body of Knowledge. Newtown Square, PA, 2017
3. Antonelli, Leandro.; Rossi, Gustavo; Oliveros, Alejandro: A "Collaborative Approach to Describe the Domain Language through the Language Extended Lexicon". *Journal of Object Technology* 15 (3), 3:1-27, 2016.
4. Law, Averill M.: *Simulation modeling and analysis*. Mc Graw-Hill., 2015.
5. Loper, Margaret L.: *Modeling and Simulation in the Systems Engineering Life Cycle: Core Concepts and Accompanying Lectures*. Simulation Foundations, Methods and Applications. Springer, 2015.
6. Banks, Jerry: *Discrete-event system simulation*. Pearson Education, 2014.
7. D. Baia, C. de Lucena, P. Alencar, D. Cowan, P. Bommel, C. Valadares and T. Oliceira, "A Multiagent-Based Simulation Model to Support Management Decision Making in Software Development," David R. Cheriton School of Computer Science, University of Waterloo., Tech. Rep., Jan 2014.
8. D. Baia, C. de Lucena, P. Alencar, R. Rocha, C. Valadares, and D. Cowan, "MultiAgent-Based Simulation in Software Project Management: Scope Management Representantion and Visualization," David R. Cheriton School of Computer Science, University of Waterloo., Tech. Rep., Nov 2014.
9. STANDISH GROUP, "CHAOS MANIFESTO 2013," The Standish Group International, 2013.
10. Breno Bernard Nicolau de França, Guilherme Horta Travassos: "Reporting guidelines for simulation-based studies in software engineering". *Proceedings of the EASE 2012 - Published by the IET*, 2012.
11. R. Turner, R. J. Madachy, D. Ingold, and J. A. Lane, "Modeling processes in systems engineering," in *2012 International Conference on Software and System Process, ICSSP 2012, Zurich, Switzerland, June 2-3, 2012*, 2012, pp. 23–27.
12. A. Budzier, "Why Your IT Project may be Riskier than You Think" *Harvard Business Review*, 2012.
13. Dan Houston, "Research and Practice Reciprocity in Software Process Simulation", *IEEE, 2012. ICSSP 2012, Zurich, Switzerland*, 978-1-4673-2352-9/12. Pp. 219-220
14. Magennis, Troy: "Forecasting and Simulating Software Development Projects. Effective Modeling of Kanban & Scrum Projects using Monte-carlo Simulation". *FocusedObjective.com*, 2011.
15. Cao, L.; Ramesh, B; Abdel-Hamid, T. K.: "Modeling dynamics in agile software development," *ACM Trans. Management Inf. Syst.*, vol. 1, no. 1, pp. 5:1–5:26., 2010.
16. R. Agarwal and D. Umphress, "A Flexible Model for Simulation of Software Development Process," *Proceedings of the 48th Annual Southeast Regional Conference*, 2010.
17. R. Cherif and P. Davidsson, "Software Development Pprocess Simulation: Multi Agent-Based Simulation versus System Dynamics," in *Multi-Agent-Based Simulation X*, vol. 5683, no. 7. Berlin, Springer, 2010, pp. 73–85.
18. Dapeng Liu, Qing Wang, Junchao Xiao, "The Role of Software Process Simulation Modeling in Software Risk Management: a Systematic Review". *Third International Symposium on Empirical Software Engineering and Measurement*. IEEE, 2009.
19. Raffo, David M.; Wakeland, Wayne: *Moving up the CMMI Capability and Maturity Levels Using Simulation*. Software Engineering Institute, Carnegie Mellon., 2008.

20. R. Madachy, *Software Process Dynamics*. Piscataway, NJ: IEEE Press, 2008.
21. Senge, Peter M.: *The Fifth Discipline: The Art & Practice of The Learning Organization*. Doubleday, USA, 2006.
22. T. Williams, "Assessing and moving on from the dominant project management discourse in the light of project overruns," *IEEE Transactions on Engineering Management*, vol. 52, pp. 497-508, November 2005.
23. McConnell, Steve. *Software Project Survival Guide*. Microsoft Press., 1998.