

Soporte a la toma de decisiones y optimización en el proceso de planificación de un proyecto software basado en CMMI usando un modelo híbrido de simulación

Daniel Crespo, Mercedes Ruiz

Departamento de Lenguajes y Sistemas Informáticos
Escuela Superior de Ingeniería
C/ Chile, 1
11003 - Cádiz, España
dani.crespobernal@alum.uca.es, mercedes.ruiz@uca.es

Resumen La estimación de la duración de una tarea o de la cantidad de recursos necesarios en un proyecto software suele ser muy inexacta. De este hecho se desprende que una gestión eficaz del proyecto a menudo debe ser muy dinámica, es decir, puede ser necesario reasignar recursos, incorporar nuevo personal, cancelar tareas opcionales, etcétera, en respuesta a nueva información o a la revisión de las estimaciones. En este trabajo se propone un modelo híbrido de simulación aplicado en el ámbito del área de proceso de Planificación de Proyectos de CMMI que permite estudiar la evolución de un proyecto concreto a lo largo del tiempo así como apoyar la toma de decisiones de la alta gerencia y determinar los parámetros óptimos de coste y plazos de acuerdo con las necesidades de la dirección.

Keywords: Modelado y simulación, CMMI, planificación de proyectos, modelo híbrido

1. Introducción

Ahora, más que nunca, las empresas desean desarrollar productos y servicios de una manera óptima, rápida y barata. Al mismo tiempo, en el sector de la alta tecnología del siglo XXI, casi todas las organizaciones se han encontrado construyendo productos y servicios de complejidad creciente. Los problemas a los que estas organizaciones se enfrentan hoy en día requieren soluciones que requieren la implicación de toda la empresa y requieren un enfoque de integración. La gestión eficaz de los activos de la organización es fundamental para el éxito del negocio. En esencia, estas organizaciones son desarrolladoras de productos y servicios que necesitan una manera de manejar un enfoque integrado de sus actividades de desarrollo en el marco de la consecución de sus objetivos de negocio.

CMMI (*Capability Maturity Model Integration*) [1] es un modelo para la mejora de procesos que proporciona a las organizaciones los elementos esenciales

para desarrollar procesos eficaces. Se puede utilizar CMMI en las actividades de mejora de procesos como una colección de buenas prácticas, un marco para organizar y priorizar las actividades, un soporte para coordinar actividades multidisciplinares para construir adecuadamente un producto y/o un medio para alinear los objetivos de la mejora de procesos con los objetivos de negocio de la organización.

En este trabajo se propone un modelo de simulación híbrido que integra los enfoques de simulación de eventos discretos y simulación basada en agentes aplicado en el ámbito del área de proceso de Planificación de Proyectos (*PP, Project Planning*) y cuya finalidad es servir de apoyo al diseño y ejecución de cada una de las Prácticas Específicas (*SP, Specific Practices*) que componen el área mencionada anteriormente. El modelo se ha calibrado empleando datos históricos provenientes del repositorio de proyectos del ISBSG ver. 10 (*International Software Benchmarking Standards Group*) [13].

El trabajo se estructura de la siguiente manera: El apartado 2 ofrece una visión global del área de proceso PP de CMMI. En el apartado 3 se presentan las principales técnicas de gestión de proyectos y se referencian algunos trabajos actuales que estudian esta problemática. En el apartado 4 se describe el caso de estudio concreto en el que se aplica el modelo híbrido de simulación que se propone y se explican las características o aspectos principales del mismo. Asimismo, se describen los diferentes escenarios que se han simulado y se resumen los resultados obtenidos. Finalmente, el apartado 5 contiene las conclusiones y trabajos futuros.

2. Área de Proceso *Project Planning*

El propósito del área de proceso *Project Planning* consiste en establecer y mantener los planes que definen las actividades del proyecto. Este área incluye las siguientes actividades:

- Desarrollar el plan de proyecto.
- Interactuar con los *stakeholders* apropiadamente.
- Aprobar el plan.
- Realizar un mantenimiento del plan.

La planificación comienza con los requisitos que definen el producto y el proyecto e incluye la estimación de los atributos de las tareas y productos de trabajo, determinar los recursos necesarios, negociar los compromisos, elaborar un calendario e identificar y analizar los riesgos del proyecto.

El plan de proyecto proporciona la base para realizar y controlar las actividades del proyecto que se ocupan del compromiso obtenido con el cliente. Este plan necesita ser revisado a medida que el proyecto progresa para hacer frente a cambios en los requisitos y compromisos, estimaciones erróneas, acciones correctivas y cambios en el proceso.

3. Trabajos relacionados

La Tabla 1 muestra las principales técnicas que se utilizan para resolver los problemas de planificación de proyectos clasificadas en función de los tipos de problemas que ayudan a resolver.

Tabla 1. Principales técnicas de planificación de proyectos

Técnicas	Tipo de problema
Inteligencia artificial	Planificación colaborativa
Minería de datos	Identificación de factores de éxito
Programación lineal	Toma de decisiones
Simulación	Predicción y toma de decisiones
Algoritmos genéticos	Asignación de recursos

Entre los trabajos actuales que proponen aplicar estas técnicas en la planificación de proyectos pueden encontrarse los siguientes: En [9] se propone un nuevo enfoque del proceso de planificación de proyectos usando inteligencia artificial para gestionar de una manera unificada proyectos llevados a cabo mediante organizaciones separadas geográficamente. Las técnicas de minería de datos se aplican en [10] y [11] para analizar los factores de éxito de un proyecto software y estimar y mejorar la probabilidad de éxito en el mismo respectivamente. En [12] se propone un método para realizar la planificación de un proyecto usando programación lineal. Se han empleado técnicas de simulación en [7] para elaborar un agente de estimación basado en la ontología de CMMI centrado en el proceso PP y en [6] para adaptar un modelo de simulación de planificación de proyectos basado en agentes empleado para planificar actividades experimentales del Mars Rover de la NASA. Madachy propone en [15] un modelo de simulación dinámico para evaluar los riesgos de un proyecto software. En [4] se introducen técnicas de simulación para diseñar y documentar procesos con el fin de mejorar su comprensión y se describe como puede emplearse la simulación de procesos para realizar estimaciones de coste *bottom-up* en un proyecto. Con respecto al estudio de la asignación de recursos en proyectos software, en [16] se estudia esta problemática desde la perspectiva de proyectos software de mantenimiento masivo, en [17] se propone un algoritmo genético para llevar a cabo esta tarea y en [18] se usan técnicas de programación lineal para crear un algoritmo de asignación de recursos. Por último en [5] se sugieren posibles utilidades del uso de la simulación para alcanzar niveles más altos en CMMI.

El aporte principal de nuestro trabajo consiste en el empleo de un modelo de simulación híbrida que integra el enfoque de simulación de eventos discretos con la simulación basada en agentes para modelar partes concretas del proceso de ejecución de un proyecto software.

4. Caso de estudio

En este apartado se describe el caso de estudio en el que se enmarca el modelo híbrido de simulación propuesto y se explican las características del mismo. Asimismo, se describen los escenarios que se han simulado y se recogen y analizan los resultados obtenidos.

4.1. Descripción del Problema

Para el propósito de nuestro estudio hemos considerado una factoría de software certificada en el nivel 2 de CMMI que desarrolla software a medida para diferentes clientes. Dejando a un lado la Dirección y otros departamentos que no tienen influencia directa sobre el proceso en cuestión, la estructura organizativa de la empresa comprende los siguientes roles:

- Gerente: Encargado de supervisar el trabajo, controlar la producción y ejercer de enlace con los clientes.
- Coordinador: Mando encargado de planificar el trabajo de su equipo. Responsable directo de los analistas.
- Analista: Es el encargado de analizar y diseñar el software. Cada analista tiene uno o más programadores y testers a su cargo.
- Programador: Su misión consiste en desarrollar el software diseñado.
- Tester: Realiza las pruebas del software.

El objetivo principal de este trabajo consiste en servir de base al diseño y construcción del área de proceso PP de CMMI mediante el estudio de la evolución en el tiempo de un proyecto concreto a través de la simulación de los procesos de planificación y ejecución del mismo. Para ello se estudia la problemática de determinar el número de recursos necesario para terminar el proyecto dentro de plazo, sin superar el presupuesto acordado y maximizando el margen de beneficio obtenido por la organización. Por otro lado, se analiza de cerca el proceso de asignación de recursos implementando diferentes políticas para llevarlo a cabo en función de las fechas de finalización establecidas para cada una de ellas, simulando un procedimiento de acciones correctivas en caso de detectar desviaciones significativas. Para ello, el subproceso de desarrollo del proyecto se simula mediante un modelo basado en agentes, obteniendo de esta manera una visión más realista y una estimación más acertada al modelar el comportamiento del equipo de proyecto desde el punto de vista de las personas.

El resultado final de la simulación vendrá dado a través de los siguientes indicadores:

- Distribución de esfuerzo de los recursos.
- Fecha de finalización estimada.
- Margen de beneficio obtenido.
- Coste estimado.
- Número estimado de defectos.

4.2. Construcción del modelo de simulación

Siguiendo la propuesta de Kellner para describir modelos de simulación [2] y la metodología de Martínez y Richardson [3], a continuación se describe el modelo de simulación construido. La implementación del modelo y las simulaciones se han realizado utilizando la herramienta de simulación AnylogicTM.

Propósito y ámbito del modelo El modelo propuesto pretende apoyar el diseño y construcción del área de proceso PP de CMMI proporcionando herramientas destinadas al soporte de cada una de las prácticas específicas propuestas en el área. El alcance del modelo abarca la ejecución de un proyecto concreto en una organización que implementa una o más áreas de proceso CMMI y entre las que se incluye la de Planificación de Proyectos.

Variables de salida Las principales variables que proporcionan información respecto del propósito del modelo son:

- *FechaFin*: Fecha estimada en la que finalizará el proyecto.
- *Esfuerzo*{ *Coordinadores*, *Analistas*, *Programadores*, *Testers*}: Número de horas empleado durante el proyecto por cada rol.
- *Coste*: Coste estimado del proyecto.
- *Margen de beneficio*: Estimación del beneficio económico obtenido.
- *Defectos*: Número estimado de defectos encontrados por el usuario durante el primer mes de uso del software.

Parámetros de entrada Los parámetros de entrada permiten configurar diferentes escenarios de simulación. Los utilizados en este estudio se han agrupado en las siguientes categorías:

- Parámetros de proyecto:
 - *Presupuesto*: Indica la cuantía del presupuesto acordado para el proyecto.
 - *Tamaño*: Tamaño funcional del software calculado en puntos de función ajustados.
 - *Plazo*: Se refiere al tiempo estimado para la ejecución completa del proyecto.
 - *Número de fases*: Indica el número de fases del proyecto cuando se sigue un ciclo de vida incremental.
 - *Inversión en software y hardware*: Cuantía del presupuesto para el software y hardware necesario.
 - *Número de* { *Coordinadores*, *Analistas*, *Programadores*, *Testers* }: Indica cuántos recursos se emplean de cada rol.
 - *Lenguaje de programación*: Especifica el tipo de lenguaje usado (3GL, 4GL, etcétera).
- Parámetros de tarea:
 - *Complejidad*: Complejidad estimada de la tarea (Baja, media o alta).

- *Tipo*: Indica el tipo de tarea (desarrollo, diseño, etcétera).
- *Prioridad*: La tarea puede tener una prioridad baja, media o alta. Este parámetro determina el orden de comienzo de las tareas en situaciones de compromiso.
- *Fecha de fin*: Indica la fecha tope de finalización de la tarea.
- *Fase*: En un proyecto con ciclo de vida incremental, indica la fase a la que pertenece.
- Parámetros de programador:
 - *Habilidades*: Se refiere a las competencias técnicas del programador, como el conocimiento de una determinada tecnología o lenguaje de programación.
 - *Influencia de trabajo en equipo*: Representa el estímulo recibido por un programador al trabajar en equipo.
 - *Tasa de adaptación*: Indica la facilidad o dificultad de un programador para adaptarse a nuevas tareas.
 - *Tasa de aprendizaje*: Indica la capacidad de un programador a la hora de recibir formación.
 - *Grado de compromiso*: Representa cuán comprometido está el programador con la organización.
 - *Proactividad*: Se refiere al grado de proactividad del programador.
 - *Habilidad con lenguajes de programación*: Indica el grado de habilidad del programador a la hora de escribir código fuente.
 - *Habilidad con la metodología de desarrollo empleada*: Representa el grado de habilidad y conocimiento de la metodología de desarrollo empleado.
 - *Coste por hora*: Simboliza el coste que supone una hora de programación.
- Parámetros de decisión:
 - *Política de asignación*: Configura la política de asignación de recursos a emplear.
 - *Ciclo de vida*: Indica el ciclo de vida que se va a emplear en el proyecto.

Abstracción del proceso El modelo propuesto, cuya arquitectura se encuentra representada en la Figura 1, consta de dos componentes principales: Un modelo de simulación de eventos discretos y un modelo de simulación basado en agentes. Para la parte del sistema modelada mediante eventos discretos se han usado los elementos típicos de este tipo de simulación existentes en la *Enterprise Library* de AnylogicTM: funciones, bloques (actividades, colas, selectores de rutas, recursos), caminos y entidades. Para el modelo basado en agentes además se han empleado gráficos de estado.

Modelo basado en eventos discretos. La simulación del proyecto comienza a partir de unos datos de entrada que lo caracterizan (especificaciones, presupuesto, etcétera). A lo largo del proceso de planificación el proyecto pasa por las diferentes fases consumiendo unos recursos determinados (personal o herramientas) durante un tiempo específico. Estos tiempos provienen de distribuciones triangulares aproximadas a los tiempos observados en la realidad. Una vez terminada

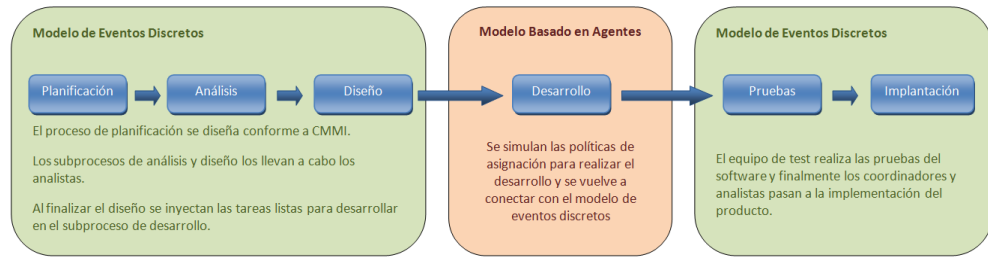


Figura 1. Arquitectura del modelo

la simulación del proceso de planificación, se simula la ejecución del proyecto a través de las diferentes actividades del ciclo de vida del mismo: análisis, diseño, desarrollo, pruebas e implantación.

Modelo basado en agentes. El modelo basado en agentes se emplea para simular el proceso de desarrollo del proyecto. Los agentes son los programadores y cada uno de ellos está descrito según los parámetros citados en la sección 4.2. Durante el desarrollo de un proyecto real, el tiempo estimado de una tarea rara vez coincide con el tiempo empleado en la misma. La realidad es que cada persona produce a un nivel diferente en función de ciertas destrezas y rasgos intrínsecos. En este trabajo se propone una estimación más realista del proceso de desarrollo de un producto software calculando la productividad esperada de cada recurso. La productividad es un factor que puede variar con el tiempo. Es habitual que al comenzar una tarea se produzca a un nivel más bajo debido al proceso de familiarización con los nuevos cometidos. También influye negativamente sobre la productividad la incorporación de nuevo personal al proyecto, debido a que parte del tiempo se invierte en poner al día y/o ayudar a los recién llegados. En la Tabla 2 se muestran las competencias contempladas en este trabajo y el peso asociado a cada una de ellas en el cálculo de la productividad en condiciones normales. Estas competencias son parametrizables y los pesos mostrados en la Tabla citada anteriormente son los que se han empleado para los experimentos descritos en la Sección 4.4. Los valores de los pesos deben calibrarse con la información histórica de la organización en la cual se va a emplear el modelo de simulación en cuestión.

En este trabajo se estudian dos políticas de asignación de recursos:

- Política 1: Mientras haya recursos libres se asignan a nuevas tareas del conjunto de tareas pendientes.
- Política 2: Si hay recursos libres, determinar si alguna tarea necesita un apoyo extra para terminar a tiempo (ver sección 4.4). En caso afirmativo asignar el recurso a la tarea en curso. Si no es necesario, se asigna a una nueva tarea del conjunto de tareas por realizar.

Tabla 2. Influencia de competencias sobre la productividad

Competencias genéricas (50 %)	Influencia trabajo equipo (40 %) Adaptación (30 %) Proactividad (20 %) Compromiso organización (10 %)
Competencias técnicas (50 %)	Habilidades específicas (35 %) Hab. lenguaje programación (40 %) Conocimiento metodología (25 %)

4.3. Verificación y validación del modelo

Todas las ecuaciones del modelo han sido verificadas mediante la ejecución de pruebas unitarias y de integración.

El modelo se ha validado mediante su calibración a partir de la información histórica proporcionada por el repositorio de proyectos del ISBSG ver. 10.

4.4. Simulaciones del modelo

Las simulaciones del modelo permiten estimar el número de defectos, plazo de ejecución y coste asociados al proyecto en cuestión. Se han realizado dos experimentos: El primero permite analizar los efectos de las diferentes políticas de asignación de recursos sobre la ejecución de las tareas; el segundo experimento está enfocado hacia la estimación del número óptimo de recursos necesarios para llevar a cabo el proyecto. A continuación se describe el caso de estudio considerado y se presentan y analizan los resultados obtenidos en los experimentos realizados.

Configuración de parámetros de entrada para el caso de estudio Las simulaciones del modelo se han realizado considerando que se ha ejecutado el subproceso de análisis del sistema del proyecto y se dispone de la información sobre cada una de las tareas a desarrollar.

Parámetros del proyecto

Tomando como referencia proyectos y equipos de factorías de software de tamaño medio, se han considerado los siguientes valores para los parámetros: *TamañoFuncional*: 500. *Tipo de Lenguaje*: 3GL. *numProgramadores*: 7. *numCoordinadores*: 2. *numAnalistas*: 2. *numTesters*: 5. *Plazo*: 50 días. *Presupuesto*: 50.000 €. *Número de fases*: 3.

Parámetros de las tareas

Los valores de los parámetros referentes a las tareas a desarrollar están almacenados en una base de datos de estimaciones. Esta base de datos es alimentada de forma manual por los analistas de la organización a raíz de la información obtenida durante el proceso de análisis del proyecto.

Parámetros de los programadores

Ya que el subproceso de desarrollo se ha definido mediante un modelo basado en agentes, es necesario proporcionar los parámetros que definen a los programadores de la organización (agentes del modelo). Éstos se encuentran recogidos en una base de datos destinada a tal efecto.

Parámetros de decisión

Los valores de los parámetros por defecto de este grupo son los siguientes: *Política*: 1. *Ciclo de vida*: incremental.

Simulación de políticas de asignación de recursos

Asignación de recursos. Antes de comenzar el proceso de asignación de recursos, las tareas se disponen según la secuencia establecida en el plan de proyecto. La política de asignación de recursos número 1 busca el recurso más adecuado para llevar a cabo una determinada tarea. Para determinar la idoneidad de una asignación se busca un recurso entre cuyas habilidades se encuentre el tipo de tarea en cuestión. Si no se encuentra un recurso con las habilidades específicas requeridas, la tarea se asignará a cualquier recurso libre.

La política de asignación de recursos número 2 (ver sección 4.2) difiere de la primera en la asignación de personal adicional a una tarea siempre y cuando ésta no vaya a completarse a tiempo. Para determinar si una tarea finalizará en el plazo previsto se realiza periódicamente una estimación de la probabilidad de que ésto ocurra. Para ello se tiene en cuenta el rendimiento actual de la misma y se supone que se mantendrá de manera uniforme a lo largo del tiempo. Si como consecuencia de esta estimación se deduce que la tarea no podrá completarse en el plazo estimado, se asignará un nuevo recurso a la misma siempre y cuando dicha asignación no perjudique en demasía el rendimiento general y calidad del proyecto. De manera adicional, una tarea nunca tendrá más de 4 recursos asignados y la incorporación de un nuevo recurso a una tarea en curso supondrá un aumento de la probabilidad de obtener defectos en el software. Este incremento de probabilidad está parametrizado y para este experimento en concreto se ha fijado en 5%.

En este experimento se han simulado las dos políticas indicadas con la configuración de parámetros de entrada mencionada anteriormente. Los resultados se muestran en la Tabla 3.

Tabla 3. Resultados obtenidos tras la simulación

	Política 1	Política 2
Coste total (€)	43.306	39.246
Duración (Días)	46	42
Margen de beneficio (€)	6.693	10.753
Defectos	1	2

A la vista de los resultados, se observa que la política de asignación número dos produce mejores resultados que la política número uno en cuanto a coste y duración. Sin embargo, el número de defectos durante el primer mes de uso aumenta levemente. Se puede concluir que, siempre y cuando exista un control sobre la influencia del número de recursos asignados a una única tarea, es preferible usar esta estrategia a no ser que el cliente esté dispuesto a alargar el proyecto y asumir un coste sensiblemente mayor en beneficio de la calidad final del proyecto.

Experimento de optimización: Determinación del número de recursos óptimo para el proyecto La finalidad principal de este experimento consiste en determinar el número adecuado de recursos de cada rol a emplear durante el proyecto. Para ello se tendrá en cuenta la fecha límite de entrega del proyecto y se pretenderá maximizar el margen de beneficio obtenido. La configuración de parámetros para el experimento de optimización se describe a continuación:

- Número de simulaciones: 500.
- Parámetros de entrada: Los parámetros de entrada empleados son los que se describen en la sección 4.4.
- Parámetros de control del experimento de optimización: *numProgramadores* (varía entre 1 y 7), *numTesters* (varía entre 1 y 5), *numCoordinadores* (varía entre 1 y 2) y *numAnalistas* (varía entre 1 y 2).

La optimización parte de unos parámetros de control sugeridos por el coordinador del proyecto que pueden provenir de datos históricos de la organización u otro tipo de estimaciones. En la Tabla 4 puede observarse la diferencia entre los parámetros sugeridos y los obtenidos por el experimento. En la Figura 2 puede observarse la diferencia de plazos y beneficios usando los parámetros sugeridos a priori y los obtenidos por el experimento de optimización.

Para realizar este experimento se ha empleado el motor de optimización OptQuestTM[14] que es una implementación comercial de *Scatter Search*¹.

Tabla 4. Comparación entre los parámetros sugeridos y los parámetros obtenidos tras la experimentación

	Parámetros sugeridos	Parámetros óptimos
numCoordinadores	2	1
numAnalistas	2	2
numProgramadores	7	6
numTesters	4	2

¹ Es un método metaheurístico para resolver problemas de optimización que encuentra automáticamente los mejores parámetros de un modelo teniendo en cuenta una serie de restricciones previamente especificadas.

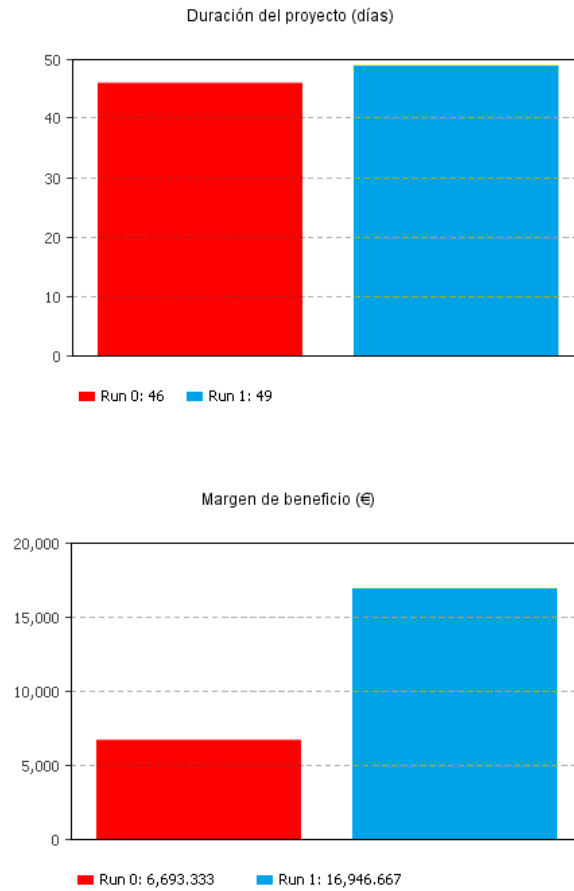


Figura 2. Comparación entre ejecución con parámetros estimados a priori (Run 0) y ejecución con parámetros óptimos (Run 1)

5. Conclusiones y trabajos futuros

En este trabajo se presenta un conjunto de los resultados de una iniciativa de investigación que se centra en la utilización de las técnicas de modelado y simulación en diferentes áreas de proceso de CMMI. En el ámbito del proceso PP de CMMI se ha desarrollado un modelo híbrido de simulación que sirve de soporte para la mayoría de las prácticas específicas propias del área destacando la fijación de plazos, la estimación de esfuerzo, la determinación del coste total del proyecto así como la determinación del número de recursos necesario para llevarlo a cabo.

En el caso de estudio considerado en este trabajo, se obtienen las siguientes conclusiones:

- Una política de asignación destinada al apoyo puntual de tareas susceptibles de no acabar a tiempo produce mejores resultados en cuanto a coste y plazo, pero puede ocasionar pérdidas de calidad en el producto final.
- Pequeñas desviaciones en la determinación del número óptimo de recursos para un proyecto suponen importantes incrementos en los costes del mismo.

La finalidad principal de nuestros próximos trabajos es la siguiente:

- Validar el modelo en empresas del entorno.
- Incorporar al modelo el área de proceso de Seguimiento y Control del Proyecto (PMC, *Project Monitoring and Control*) de CMMI.
- Simular diferentes alternativas en la construcción de áreas de proceso concretas con el fin de determinar cuál es la que mejor se adapta a la organización estudiando su comportamiento a lo largo del tiempo.
- Permitir soporte para la construcción de áreas de proceso de CMMI nivel 4 simulando una gestión cuantitativa de los proyectos.

Agradecimientos. Esta investigación está parcialmente financiada por el Ministerio de Ciencia e Innovación de España y por los fondos europeos FEDER mediante los proyectos TIN2007-67843-C06-04 y TIN2010-20057-C03-03.

Referencias

1. CMMI Product Team, *CMMI for Development, Version 1.2*, Software Engineering Institute, Carnegie Mellon University, pp. 3-15, pp. 327-252, 2006.
2. Kellner, M.I., Madachy, R.J., Raffo, D., *Software process simulation modeling: Why? What? How?*, J. Syst. Software, 46(2-3): 91-105, 1999.
3. Martínez, I.J., Richardson, G.P., *Best Practices in System Dynamics Modeling*, Proceedings of the 19th International Conference of the Systems Dynamics Society. Atlanta, USA, 2001.
4. Raffo, D.M., Wakeland, W., *Moving Up the CMMI Capability and Maturity Levels Using Simulation*, Software Engineering Institute, Carnegie Mellon University, pp. 20-31, pp. 77-94, 2008.
5. Miller, M.J., Pulgar-Vidal F., Ferrin, D.M., *Achieving Higher Levels of CMMI Maturity Using Simulation*, Proceedings of the 2002 Winter Simulation Conference, pp.1473-1478, 2002.
6. Joslin, D., Poole, W., *Agent-based simulation for software project planning*, Proceedings of the 2005 Winter Simulation Conference, pp. 1059-1065, 2005.
7. Lee, C.S, Wang M.H, Yan Z.R., Lo, C.F., Chuang, H.H., Lin, Y.C., *Intelligent Estimation Agent Based on CMMI Ontology for Project Planning*, IEEE International Conference on Systems, Man and Cybernetics, pp. 228-233, 2008.
8. Lee, C.S, Wang M.H, Chen, J.J, Hsu, C.Y, *Ontology-based Intelligent Decision Support Agent for CMMI Project Monitoring and Control*, International Journal of Approximate Reasoning, vol. 48, no. 1, pp. 62-76, 2008.

9. Drabble, B., *Artificial intelligence for project planning*, Future Developments in Projects Management Systems, IEE Colloquium, pp. 1-5, 1995.
10. Yousef, A.H., Gamal, A., Warda, A., Mahmoud, M., *Software Projects Success Factors Identification using Data Mining*, Computer Engineering and Systems, The 2006 International Conference, pp. 447-451, 2006.
11. Prasad, A., Arsiwala, J., Singh, P.P., *Estimating and improving the probability of success of a software project by analysing the factors involved using data mining*, Artificial Intelligence and Education (ICAIE), 2010 International Conference, pp. 391-393, 2010.
12. Yanzhang W., Shengju H., *A multiple criteria 0-1 linear programming method of project planning and investment decision making*, Innovation in Technology Management - The Key to Global Leadership. PICMET '97: Portland International Conference on Management and Technology, p. 394, 1997.
13. ISBSG, *ISBSG Project Repository Release 10*, <http://www.isbsg.org>.
14. OptTek Systems, Inc., *OptQuest*, <http://www.opttek.com/Products/OptQuest.html>.
15. Madachy, R., *A software project dynamics model for process cost, schedule and risk assessment*, Department of Industrial and Systems Engineering, USC, Ph.D Dissertation, pp. 33-41, 1994.
16. Antoniol, G., Di Penta, M., Harman, M., *Search-Based Techniques for Optimizing Software Project Resource Allocation*, Genetic and Evolutionary Computation - GECCO 2004, pp. 1-11, 2004.
17. Shan, X., Jiang, G., Huang, T., *The Optimization Research on the Human Resource Allocation Planning in Software Projects*, Management and Service Science (MASS), 2010 International Conference, pp. 1-4, 2010.
18. Sebt, M.H., Akrami, A., Banki, M.T., Shahhosseini, V., *Optimized Allocation of Expert Human Resources to Project*, Modelling & Simulation, 2009. AMS '09. Third Asia International Conference, pp. 1-6, 2009.