

MEJORAS EN LA TOMA DE DECISIONES EN PROYECTOS DE SOFTWARE. APLICACIÓN DE TÉCNICAS DE LÓGICA BORROSA

J. Aroba Páez ¹, I. Ramos Román ², J. Riquelme Santos ²

¹D. de Ingeniería Electrónica, Sistemas Informáticos y Automática. Universidad de Huelva. Ctra. Huelva-La Rabida s/n. Huelva, Spain. E-Mail : aroba@uhu.es

²D. de Lenguajes y Sistemas Informáticos. Universidad de Sevilla. Avda. Reina Mercedes s/n. Sevilla, Spain. E-Mail ([@lsi.us.es](mailto:isabel.ramos/riquelme))

Resumen

Los Simuladores de Proyectos de Desarrollo de Software basados en modelos dinámicos han supuesto un avance significativo frente a las técnicas tradicionales de estimación. Estos simuladores permiten conocer la evolución de un proyecto antes, durante y después de la ejecución del mismo. Pero su utilización en la estimación del proyecto antes de empezar la ejecución ha estado frenada por el gran número de atributos del proyecto que hay que conocer previamente. En este trabajo se presenta, en primer lugar, las diferentes mejoras que se han ido añadiendo al simulador desarrollado en nuestro departamento para facilitar al responsable de proyectos el uso de dichos simuladores y la estimación de los resultados deseados. En segundo lugar, se presenta una nueva mejora obtenida al utilizar técnicas de lógica borrosa a las bases de datos generadas por el simulador. En este último caso, el responsable del proyecto puede conocer, en función de las decisiones que tome, el nivel de cumplimiento de los objetivos del proyecto.

Palabras claves: Simulación, estimación de proyectos, técnicas de lógica borrosa.

1. INTRODUCCIÓN

La aparición, en los últimos años de los modelos dinámicos para Proyectos Software (PS) y de entornos de simulación potentes (Stella, Vensim, iThink, Powersim, etc.), han facilitado la aparición de los llamados *Simuladores de Proyectos de Software* (SPS) que ofrecen la posibilidad de simular el comportamiento de tales proyectos. Estos simuladores permiten a los directores de proyectos experimentar con diferentes políticas de gestión y dirección con un coste nulo, de forma que las decisiones que se tomen sean lo más acertadas posible.

Un SPS permite realizar un:

1. Análisis a priori del proyecto: Simulación del proyecto, antes de empezar el proceso de desarrollo, bajo diferentes políticas de gestión y/o diferentes estimaciones iniciales de recursos y bajo el empleo de distintas tecnologías de desarrollo.
2. Análisis post-mortem del proyecto: Simulación de proyectos que ya han finalizado, en los que interesa saber cómo podrían haberse mejorado los resultados obtenidos.

A partir del trabajo presentado en [RAMO99a] podemos decir que un SPS también permite realizar un:

3. Seguimiento del proyecto: Simulación del proyecto durante el proceso de desarrollo para ir adaptando las estimaciones del proyecto a la evolución real del mismo.

Es decir, los SPS nos permiten responder a cuestiones del tipo: *¿Qué ocurrirá si?* antes de comenzar, *¿Qué está ocurriendo?* durante la ejecución y *¿Qué hubiese ocurrido si?* una vez que ha finalizado (Figura 1).

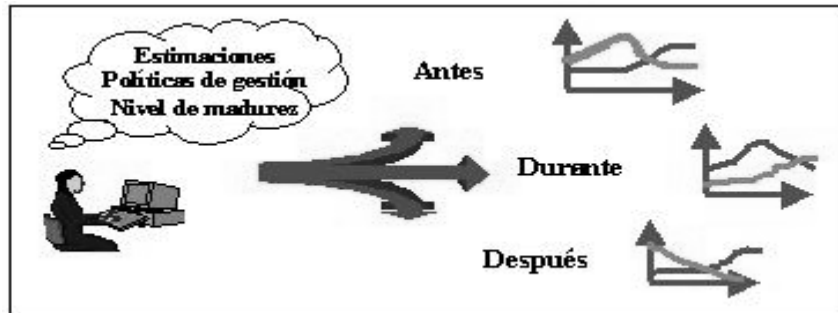


Figura 1: Posibilidades de los Simuladores de Proyectos Software.

La idea intuitiva y natural recogida en un modelo dinámico para PS es que la evolución del proyecto y por tanto el cumplimiento de los objetivos depende de: la estimación inicial de los recursos necesarios para realizar el proyecto, de las políticas de gestión que se apliquen y de determinados aspectos relacionados con la propia organización de desarrollo (nivel de madurez o consolidación, experiencia, aspectos ambientales, disponibilidad de recursos, etc.).

La aparición, por tanto, de los SPS ha significado un avance significativo en la gestión y estimación de proyectos. Aún así, estos simuladores presentan un inconveniente importante que han frenado su utilización (a nivel empresarial): el número de atributos (tanto del proyecto como de la organización de desarrollo) que se deben conocer previamente.

Uno de nuestros primeros trabajos estuvo centrado en el desarrollo de un SPS [RAMO97] [RAMO98a] que permitiese simular el comportamiento de un proyecto utilizando la mitad de atributos que utiliza el modelo recogido en [ABDE91]. Pero aún nos encontrábamos con otro problema adicional, común al proceso que conlleva la toma de decisiones en cualquier proyecto: la diversidad y simultaneidad de políticas de gestión y factores que influyen sobre el proceso de desarrollo. Es decir, cuál es la combinación adecuada de los atributos para obtener unos objetivos concretos. Las técnicas de aprendizaje automático, nos permitió corregir el problema anterior [RAMO98b] [RAMO99b]. Estas técnicas, aplicadas a las bases de datos que genera el SPS, proporcionan al responsable del proyecto información sobre las políticas de gestión que tiene que aplicar para conseguir los objetivos, que previamente, se ha fijado. En este trabajo, presentamos un paso más en la línea de facilitar el uso de los SPS: la aplicación de técnicas de lógica borrosa.

En los siguientes apartados se describen, en primer lugar, las diferentes aportaciones que han facilitado el uso, no solo académico sino también empresarial, de los SPS. Y a

continuación, se presenta la información que podemos extraer al conjugar el uso de un SPS con las técnicas de lógica borrosa.

2. SIMULADORES DE PROYECTOS DE SOFTWARE

Las variables básicas que permiten conocer el comportamiento de un sistema dinámico, y por tanto el proceso de desarrollo de software que modelamos mediante dicho sistema, están definidas mediante ecuaciones diferenciales. El sistema incluye también una serie de atributos que nos permiten estudiar el comportamiento del mismo. Estos comportamientos vienen dados por las políticas de gestión que pueden ser aplicadas en los PDS, tanto las relacionadas con el entorno del proyecto (número de tareas, tiempo, coste, número de técnicos, complejidad del producto, etc.) como las relacionadas con la organización de desarrollo.

2.1 Situación inicial

Para simular el comportamiento de un PS mediante un SPS el responsable del proyecto debe seguir los siguientes pasos (Figura 2):

1. Definir los valores concretos que tomarán los atributos (parámetros y funciones) del proyecto.
2. Simular el proyecto mediante el SPS para obtener la evolución las variables que se desee analizar.
3. Analizar los resultados obtenidos y comprobar si estos resultados coinciden con los objetivos del proyecto (coste, calidad, tiempo, etc.).
4. Si los resultados obtenidos:
 - Coinciden con los objetivos del proyecto se conocerán los atributos, y por tanto las políticas de gestión, que nos permitirán obtener los resultados deseados en nuestro proyecto.
 - No coinciden con los objetivos del proyecto se deberá volver a redefinir el proyecto (paso 1) y se repetirán los pasos anteriores. Por tanto, se tendrá que simular el proyecto tantas veces como sea necesario hasta obtener los resultados deseados.

Por tanto para disminuir el número de veces que se simulará el proyecto, el responsable del mismo deberá tener un nivel de conocimiento detallado de los valores concretos que tomarán los atributos del modelo, lo cual puede llegar a ser bastante difícil especialmente por dos motivos: primero, por el número de atributos que contiene un modelo dinámico y las múltiples combinaciones que se pueden establecer entre los mismos y segundo, porque en muchas ocasiones ni el propio responsable del proyecto conoce el valor concreto que tomarán determinados atributos aunque sí tendrá una idea clara del intervalo de valores dónde se moverá. Por ejemplo, posiblemente no sepa si la dedicación media de los técnicos en el proyecto será del 50% o del 24%, pero si sabrá que dicha dedicación estará entre un 20 y un 60%.

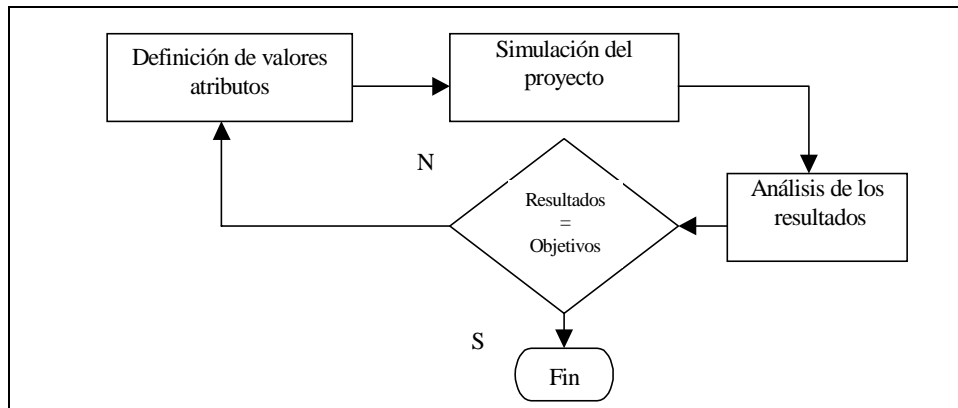


Figura 2: Pasos a seguir en la simulación de un proyecto utilizando un SPS.

Entre los pasos descritos anteriormente, sólo el paso 2 se realiza de manera automática. En el resto de los pasos debe actuar el responsable del proyecto.

2.2 Mejoras a la situación inicial

La aplicación de técnicas de aprendizaje automático, concretamente la herramienta C4.5 [QUINL93], nos permitió resolver el problema anterior. Estas técnicas nos permiten obtener reglas de gestión o de decisión¹. El conocimiento de estas reglas de gestión ayuda en la toma de decisiones para estimar automáticamente los resultados deseados por el responsable del proyecto (coste, tiempo de entrega, calidad, productividad, etc.). Además, la obtención de reglas de gestión permite al director del proyecto analizar cuáles son las políticas de gestión más significativas para conseguir los objetivos del PS.

En la Figura 3 se recoge el proceso seguido para la simulación de un PS y la obtención de reglas de gestión que permiten estimar dichos resultados u objetivos:

1. Definir:

- Los intervalos de valores que pueden tomar los atributos del proyecto.
 - Los objetivos del proyecto que se desean obtener. Por ejemplo, se desea que el coste final del proyecto no supere al inicial en más de un 20% y que el tiempo de entrega final no supere al inicial en más de un 10 %.
2. Simular el proyecto mediante el SPS. Al simular el proyecto, los atributos toman valores elegidos aleatoriamente dentro del intervalo de valores definido anteriormente.
 3. Generación automática de una base de datos. En cada simulación se obtiene un registro de la base de datos en los que se guardan los valores obtenidos para los atributos y los valores finales de las variables (obtenidos en la simulación) que nos

¹ Entendemos por regla de gestión o de decisión un conjunto de políticas de gestión (decisiones) que toma el director para cumplir con los objetivos del proyecto.

interese analizar (coste, tiempo, calidad, etc.). Cada registro de la base de datos representa un escenario posible del proyecto.

4. Aplicación de técnicas de aprendizaje automático. A partir de la base generada, el C4.5 aprende examinando la muestra de casos resueltos y proponiendo un conjunto de reglas para la toma de decisiones.
5. Obtención de reglas de gestión.

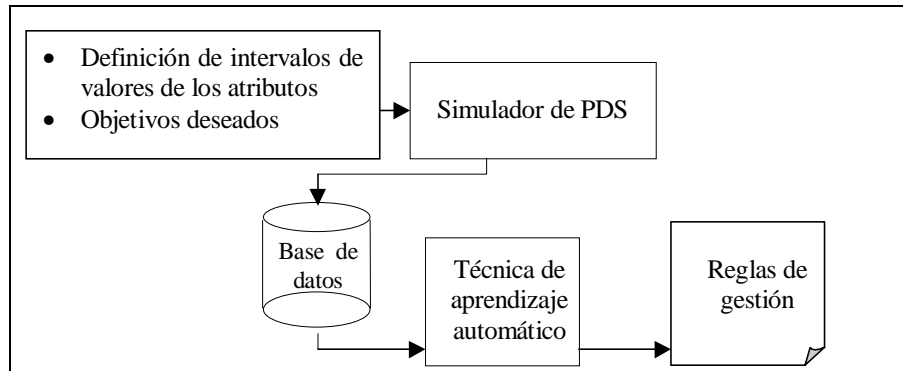


Figura 3: Pasos a seguir en la simulación de un proyecto utilizando un SPS y técnicas de aprendizaje automático (C4.5).

En este caso, el responsable del proyecto sólo debe realizar el primer paso, dónde además se facilita la definición de los atributos respecto del paso 1 del apartado anterior (figura 2), el resto de los pasos descritos se realizan automáticamente.

Las reglas de gestión obtenidas, del árbol jerárquico que proporciona el C4.5, son del tipo:

“Si la dedicación media de los técnicos es mayor del 50% y el tiempo medio de incorporación de los técnicos en el proyecto es menor de 5 días y la dedicación a formación es mayor del 5% entonces se cumplirán los objetivos deseados”

En definitiva, la utilización de una herramienta para la obtención de reglas de gestión permite a los responsables de PDS afrontar diferentes situaciones, cuando tienen que definir las políticas de gestión más adecuadas para optimizar los valores finales de determinadas variables por separado o bien conjuntamente. Una vez obtenidas las reglas de gestión será el responsable del proyecto el que decida qué regla o reglas son las más fáciles de aplicar en función del proyecto concreto y de la organización software en la que esté trabajando.

3. APLICACIÓN DE TÉCNICAS DE LÓGICA BORROSA

La aplicación de técnicas de lógica borrosa nos va a permitir añadir dos nuevas mejoras en el uso de un SPS:

1. La obtención de reglas de tipo cualitativo lo que facilita la interpretación de las reglas obtenidas.
2. El responsable del proyecto no tiene que definir previamente los resultados u objetivos deseados.

En la figura 4, se muestra los pasos a seguir para obtener este nuevo tipo de reglas. Se observa (paso 1) cómo el responsable del proyecto no tiene que definir los objetivos del proyecto. En este caso, la base de datos generada por el SPS es procesada por un algoritmo de fuzzy clustering que proporciona una información de tipo cualitativo. Esta información cualitativa que se obtiene a partir de los datos cuantitativos de los atributos del proyecto, es un conjunto de reglas de gestión borrosas en formato gráfico, que permiten al responsable interpretar los datos del proyecto de forma rápida y clara.

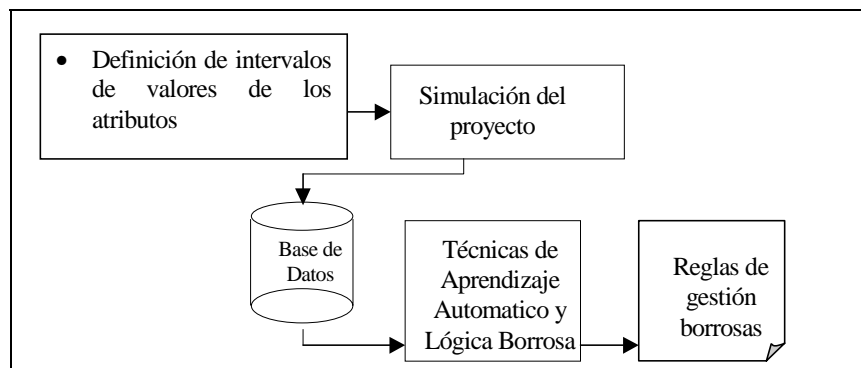


Figura 4: Pasos a seguir en la simulación de un proyecto utilizando un SPS y técnicas de lógica borrosa.

A continuación se describe el algoritmo de fuzzy clustering utilizado y se presentan los resultados obtenidos.

3.1 Descripción del algoritmo utilizado

El algoritmo de fuzzy clustering que se aplicará a los datos cuantitativos obtenidos tras la simulación del PS para obtener las reglas de gestión correspondientes, está basado en el algoritmo FCM (Fuzzy C-Means Method) [BEZD81]. Básicamente este algoritmo intenta conseguir clasificar n elementos $X_k \in X$ con $1 \leq k \leq n$, con P características cada uno, es decir, $X \subset \mathcal{R}^P$, en C clusters borrosos, asignando una función de pertenencia μ_{ik} :

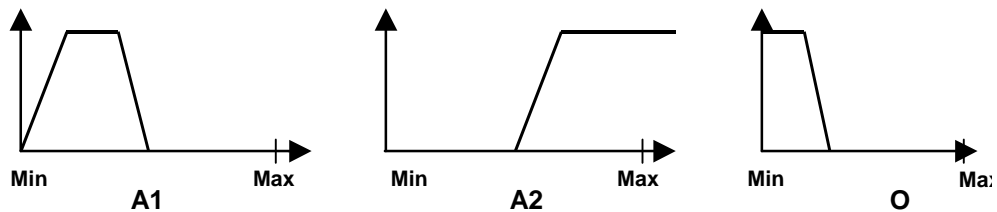
$$\mu_{ik} : X \rightarrow [0,1] \quad \forall i \quad 1 \leq i \leq c, \quad \forall k \quad 1 \leq k \leq n$$

El algoritmo de fuzzy clustering desarrollado sigue la metodología propuesta por Sugeno y Yasukawa [SUGE93] que se aplica a bases de datos cuantitativas con un solo parámetro de salida y que consiste básicamente en aplicar la partición borrosa FCM a la variable de salida Y . Una vez obtenida una partición del espacio de salida en clusters borrosos, se realiza una proyección de estos clusters sobre el espacio de entrada obteniendo un conjunto borroso en \mathcal{R}^n , que proyectado sobre cada eje asigna a cada característica un conjunto borroso.

Esta metodología ha sido modificada y ampliada, para que pueda ser aplicada a bases de datos con N variables de salida y proporcione un resultado más completo. Las mejoras que se han añadido a la metodología original son :

- El algoritmo puede trabajar con bases de datos cuantitativas, con N atributos de entrada y M de salida.
- Las diferentes variables del proyecto pueden ponderarse, asignándoles pesos para el cálculo de las distancias entre los puntos del espacio que se está particionando.
- Los conjuntos borrosos obtenidos son procesados por otro algoritmo, para obtener los trapecios de las reglas gráficas.
- La salida proporcionada por el algoritmo original, ha sido mejorada con un interface gráfico, que muestra el gráfico de las reglas de gestión obtenidas.

Las reglas de gestión borrosas obtenidas al aplicar a la base de datos generada por el SPS el algoritmo de fuzzy clustering, se representan de la siguiente forma :



En esta regla el conjunto borroso asignado a cada atributo del PS, es representado por un poliedro. En el eje X de cada conjunto borroso se representan los valores del atributo y en el eje Y el valor de pertenencia a un cluster. Esta regla de gestión borrosa se interpretaría de la siguiente forma:

Si A1 es bajo Y A2 es medio o alto Entonces O es muy bajo

3.2 Resultados obtenidos

En la figura 5 se muestra un conjunto de reglas de gestión borrosas obtenidas aplicando el algoritmo de fuzzy clustering, descrito anteriormente, a la base de datos cuantitativa obtenida como resultado de la simulación de un proyecto. En esta figura se muestran las reglas de gestión obtenidas a partir de una serie de datos cuantitativos con una sola variable de salida, en este caso el tiempo de entrega del proyecto (TIME). A partir de esta representación gráfica de las reglas, el responsable del proyecto tiene una información de tipo cualitativo, fácil de interpretar, que le permite tomar decisiones sobre el proyecto.

Por ejemplo, en la tercera (R3) regla se puede comprobar que:

"El tiempo de entrega del proyecto (TIME) toma valores altos cuando la dedicación de los técnicos en el proyecto (DEDIC) es muy baja, el tiempo medio de adaptación de los

técnicos nuevos (ADEDE) es bajo, el tiempo medio de la incorporación de los técnicos al proyecto (INCDE) es medio-alto y el tiempo medio en la salida de los técnicos del proyecto (OUTDE) es alto".

El resto de las reglas indican los intervalos de valores que deberían tomar los atributos para que el tiempo de desarrollo fuese muy bajo (R1), bajo (R2) y medio (R4). Este tipo de reglas de gestión también se pueden obtener para bases de datos cuantitativas con N-variables de salida de forma simultánea, de forma que el tipo de información que se le proporciona al responsable del proyecto es mucho más completa, y le facilitará la toma de decisiones.

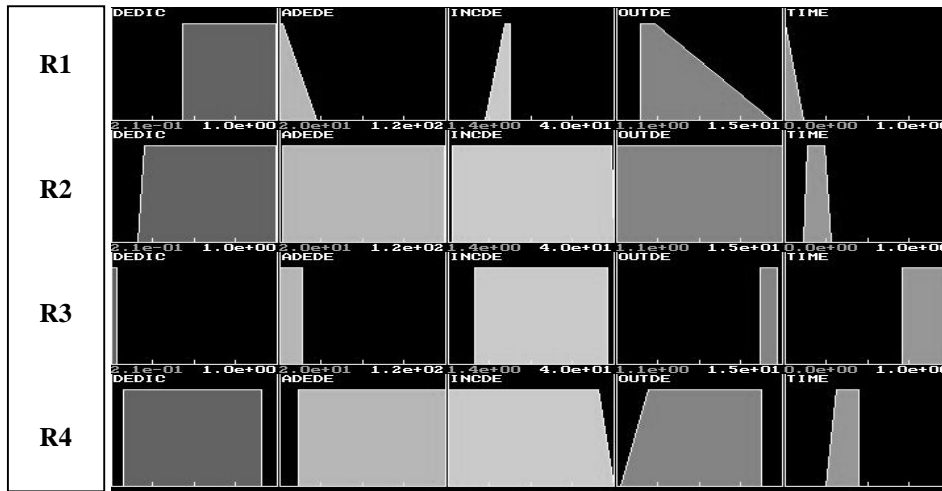


Figura 5: Ejemplo de reglas de gestión borrosas con una variable de salida (TIME)

Como se ha comentado anteriormente, con la ampliación que se ha realizado al algoritmo de fuzzy clustering descrito se pueden utilizar tantos atributos de entradas como variables de salida que se desee. En la figura 6, se muestran las reglas de gestión borrosas obtenidas aplicando el algoritmo a la base de datos cuantitativa generada en la simulación del proyecto con seis atributos y con dos variables de salida: Coste y Tiempo.

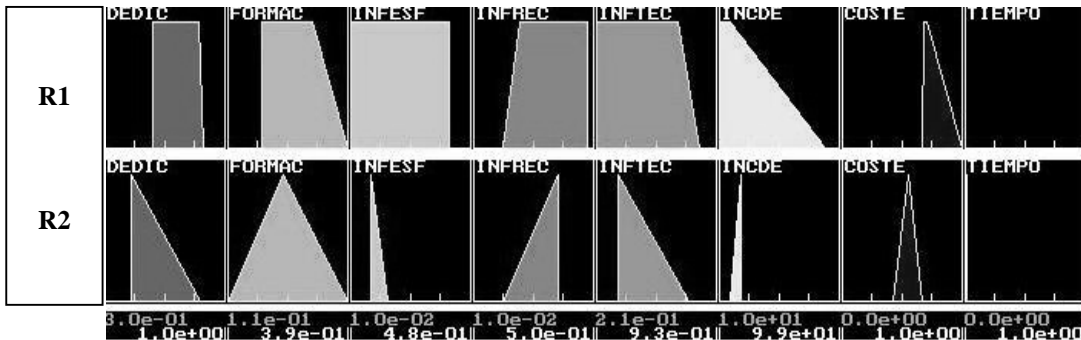


Figura 6: Ejemplo de reglas de gestión borrosas con dos variables de salida simultáneamente (Coste y Tiempo).

Por ejemplo, la segunda (R2) regla indica que:

"El coste toma valores medios y el tiempo toma valores muy bajos, simultáneamente, cuando la dedicación media (DEDIC) es media-baja, la dedicación a formación (FORMAC) es media, la infraestimación del esfuerzo necesario (INFESF) es baja, la infraestimación del tamaño del proyecto (INFREC) es media-baja, el número de técnicos que comienza el proyecto (INFTEC) es medio-alto y el tiempo medio de incorporación de los técnicos en el proyecto (INCDE) es bajo".

4. CONCLUSIONES

En este trabajo se han presentado las ventajas que presentan los SPS, basados en modelos dinámicos, frente a las técnicas tradicionales de estimación (la mayoría de carácter estático) y los inconvenientes que, desde nuestro punto de vista, presentaban la utilización de dichos simuladores. Estos inconvenientes estaban derivados del gran número de atributos, tanto del proyecto como de la organización de desarrollo, que previamente se deben de conocer para utilizar el simulador.

La utilización conjunta de un SPS y de técnicas de aprendizaje automático, como el C4.5, para facilitar la toma de decisiones (mediante reglas de gestión cuantitativas) en el complejo proceso de desarrollo de software nos permitió dar un paso importante en nuestro objetivo inicial: facilitar el uso de los SPS. Siguiendo con esa línea, actualmente estamos adaptando y aplicando técnicas de lógica borrosa para obtener información de tipo cualitativo sobre los valores de los atributos de nuestro proyecto y los resultados obtenidos. En este caso, el responsable del proyecto partiendo de la definición inicial de los intervalos de valores que tomarán los atributos sobre los que tiene mayor incertidumbre, puede conocer el nivel de cumplimiento de los objetivos del proyecto.

AGRADECIMIENTOS

Este trabajo se ha desarrollado con el apoyo del proyecto CICYT N° TIC99-0351.

6. BIBLIOGRAFÍA

- [ABDE91] Abdel-Hamid T., Madnick S.; "Software Project Dynamics: an integrated approach". Prentice-Hall, 1991.
- [BEZD81] Bezdek J.C.; "Pattern Recognition with fuzzy objective function algorithm". New York : Plenum press., 1981.
- [QUIL93] Quinlan J.R.; "C4.5 Programs for Machine Learning". Morgan Kaufmann Publ. San Francisco, 1993.
- [RAMO97] Ramos I., Ruiz M.; "Modelo Dinámico simplificado para la gestión de proyectos software". II Jornadas de Ingeniería del Software. Pág.: 146-157. San Sebastián, 1997.
- [RAMO98a] Ramos I., Ruiz M.; "A Reduced Dynamic Model to Make Estimations in the Initial Stages of a Software Development Project". In: Hawkings C et al (ed), INSPIRE III. Process Improvement through Training and Education, Pp.:172-185. London, 1998.

- [RAMO98b] Ramos I., Riquelme J.C.; "Obtención de reglas de gestión para proyectos de desarrollo de software". III Jornadas de Ingeniería del Software. Pág.: 387-398. Murcia, 1998.
- [RAMO99a] Ramos I., Ruiz M.; "Monitorización y seguimiento estratégico de Proyectos de Desarrollo de Software". XV Congreso Nacional de Ingeniería de Proyectos. EA04. León, 1999.
- [RAMO99b] Ramos I., Riquelme J.C.; "The Dynamic Models for Software Development Projects and the Machine Learning Techniques". International Conference on product focussed software process improvement. Pp.: 560-574. Oulu (Finland), 1999.
- [SUGE93] Sugeno M., Yasukawa T.; "A Fuzzy-Logic Based approach to qualitative Modeling". IEEE Transactions on Fuzzy Systems, Vol.1.Pp.: 7-31, 1993.