



***Facultad
de
Ciencias***

**Test de una plataforma escalable de Modelado
Basado en Agentes para la resolución de
problemas prácticos
(Test of an scalable ABM platform to address
practical problems)**

Trabajo de Fin de Grado
para acceder al

GRADO EN FÍSICA

Autor: Héctor Rodríguez

Directores: Jesús Marco

Co-Director: Luis Cabellos

Instituto de Física de Cantabria

Febrero - 2016

Resumen

La simulación basada en el modelado mediante agentes es una técnica que nos permite describir fenómenos de diferentes naturaleza (fenómenos físicos, sociales, biológicos...). En este estudio planteamos una simulación basada en el modelado de agentes para estudiar un problema real como es la eutrofización de las aguas del embalse de Cuerda del Pozo. En concreto se ha buscado obtener una estimación de la aportación de nutrientes por parte de las vacas a dicho embalse.

Para realizar la simulación basada en agentes se parte de una aplicación desarrollada en el Instituto de Física de Cantabria denominada "*Machanguitos*" la cual nos ha facilitado el proceso de desarrollo gracias a que nos permite describir el comportamiento de los agentes de manera sencilla. Además nos permite añadir capas GIS con las que los agentes interactúan a lo largo de la simulación. También nos permite hacer la extracción de datos relevantes de varias maneras.

Tras realizar una serie de simplificaciones, se ha conseguido simular 6300 agentes (vacas) moviéndose alrededor del embalse de Cuerda del Pozo durante 7 meses. Los agentes se desplazan y depositan abono sobre la orilla del embalse, cuyo nivel desciende en verano, abono que pasara a diluirse en el agua del embalse cuando su nivel vuelva a ascender. Según los datos aportados por un proyecto LIFE denominado ROEM+ que se realiza en este embalse, la concentración de nitritos y/o nitratos del embalse ronda los 0.10 mg/l, y además se estima que más del 50% de los nutrientes del embalse proviene de la ganadería.

A partir de la simulación realizada mediante agentes se puede estimar que el posible incremento en la concentración de nutrientes que aportarían las vacas al embalse de forma directa es del orden de 0.01 mg/l, un valor muy pequeño pero no despreciable comparado con el total.

Por último, se proponen nuevas mejoras que permitan mejorar esta estimación, teniendo en cuenta las simplificaciones que se han realizado para hacer posible la simulación.

Summary

Agent Based Modeling is a technique that allows us to describe phenomena of different nature (physical phenomena, social, biological ...). In this study we propose a simulation to study a real problem of eutrophication at the Cuerda del Pozo water reservoir. Specifically, we aim to obtain an estimate of the direct contribution of nutrients from the cows moving around this water reservoir.

This agent-based simulation is using an application developed at the Institute of Physics of Cantabria called "*Machanguitos*", which has facilitated the development process because it allows us to describe the behavior of agents easily. It also allows us to add GIS layers with which the agents interact throughout the simulation. It also allows us to extract relevant data in various ways.

After making a number of simplifications, we have managed to simulate 6300 agents (cows) moving around the Cuerda del Pozo for 7 months. The agents move and manure is deposited on the shore of the reservoir, whose level drops in summer, and it is diluted when the water level of the reservoir when goes up again. According to data provided by a LIFE project called ROEM + being performed in this reservoir, the concentration of nitrites and / or nitrates of the reservoir is around 0.10 mg / l, and it is estimated that over 50% of the nutrients in the reservoir comes from livestock.

From the simulation carried out by agents it can be estimated that the possible increase in nutrient concentration that would bring the cows to the reservoir directly is of the order of 0.01 mg / l, a very small but not negligible value compared to the total.

Finally, further improvements to improve this estimate, given the simplifications that have been made to enable the simulation, are proposed.

Prefacio

En primer lugar me gustaría agradecer a **Jesús** y a **Luis** el haberme dirigido el proyecto y permitirme empezar en el mundo de la simulación, así como el enseñarme de primera mano el cómo funciona una herramienta de simulación. También me gustaría agradecerles su actitud y esfuerzo por resolver la infinidad de dudas que me han ido surgiendo desde el primer día en el que se empezó a trabajar en el proyecto.

Por otra parte me gustaría agradecer a mis **padres** y **hermana** por creer incondicionalmente en mí y por insistir en que me acueste temprano aun sabiendo que no lo iba hacer. A **Laura** por ser mi compañera de vida y a mis amigos **Vea, Fran y Oscar** por estar siempre ahí. También me gustaría agradecer a **Pollito** su compañía en las largas noches de estudio.

Índice

Resumen	3
Prefacio	4
1 Introducción	6
1.1 Objetivo general: resolución de problemas prácticos utilizando una plataforma de simulación mediante modelado basado en agentes	6
1.2 Ejemplo práctico: Simulación de la aportación de nutrientes por parte del ganado en el estudio del problema de eutrofización del embalse de Cuerda del Pozo.	7
1.3 Planteamiento existente	10
1.4 Interés de explorar una solución basada en ABM	10
2 Plataformas basadas en Agentes	11
2.1 Introducción al Modelado Basado en Agentes (ABM)	11
2.2 Comparativa de Software ABM	11
2.3 Ejemplo de aplicación	12
2.4 Descripción del contexto de aplicación en este caso	12
2.5 Conclusiones	13
3 <i>Machanguitos</i>: Una implementación flexible y escalable de un modelo de agentes.	14
3.1 Introducción	14
3.2 Componentes	14
3.2.1 Procesos	14
3.2.2 Descripción de los agentes	16
3.2.3 Capas “GIS”	16
3.3 Conclusiones	16
4 Test de la plataforma para el caso de uso de CdP	17
4.1 El problema del impacto de la ganadería	17
4.2 Descripción de las cantidades de referencia	17
4.3 Implementación	18
4.4 Resultados y análisis de la variación de los resultados de acuerdo a los diferentes patrones de comportamiento de las vacas	23
4.5 Propuesta de mejoras en la integración	24
5 Conclusiones y Perspectivas	25
Referencias	25

1 Introducción

1.1 Objetivo general: resolución de problemas prácticos utilizando una plataforma de simulación mediante modelado basado en agentes

Las simulaciones juegan un papel muy importante en la resolución de los problemas reales en ciencia y tecnología a día de hoy [1], reduciendo en muchos casos las pruebas experimentales, y en algunas ocasiones cómo único medio de analizar el comportamiento de sistemas complejos. La simulación de un problema requiere en cualquier caso un planteamiento inicial del mismo, al nivel que nos permita la información disponible, que normalmente incluye las condiciones iniciales del sistema, los componentes del mismo, y una serie de escenarios a considerar. Además los datos experimentales sobre el problema deben permitir inicialmente contrastar la información sobre el sistema que debe proporcionarnos la simulación, guiar en muchos casos su desarrollo, y finalmente validarlo. Una vez preparada la simulación, esta nos permite extraer información relevante del sistema, estimar la importancia y el comportamiento de los diferentes componentes del mismo e incluso realizar predicciones sobre su evolución. Si es así, puede decirse que hemos construido un modelo del sistema implementado mediante una simulación. En el caso de sistemas complejos, tanto por su comportamiento como por constar de un gran número de componentes, la simulación puede ser uno de los métodos más efectivos de analizar un problema.

En este trabajo se estudia una forma de simulación muy utilizada en la resolución de ciertos problemas prácticos que es el modelado basado en agentes (ABM, del inglés Agent Based Model [2]).

Para introducir este tipo de modelado podemos considerar el problema práctico de analizar el tráfico en una ciudad. Si conocemos de modo aproximado el flujo de vehículos y la disposición y capacidad de las vías de circulación, en principio podríamos diseñar nuevas soluciones de control del tráfico, analizando el problema mediante las correspondientes ecuaciones. Pero otra opción que nos permite por un lado mayor flexibilidad en el análisis y por otro lado introducir mayor complejidad en el comportamiento individual de los componentes del problema (los vehículos y sus propietarios) es utilizar un modelo basado en agentes. Con el término agente nos referimos a entidades autónomas cuyo comportamiento individual, incluyendo la interacción con su entorno y eventualmente con otros agentes, puede modelarse con cierto grado de realismo. El comportamiento global del sistema emerge del comportamiento de todos los agentes mediante los cuales se describe el problema. Aunque el problema global es muy complejo, el problema de modelar el comportamiento de cada agente puede ser relativamente sencillo, en este ejemplo partiendo de los hábitos usuales (desplazamiento al trabajo, desplazamiento al centro comercial, desplazamiento a centros de ocio, etc.) o extraordinarios (asistencia a un evento en la ciudad, salida de vacaciones, etc.). Uno de los puntos clave es contar con algún tipo de información inicial sobre este comportamiento individual, que puede obtenerse mediante la observación de una muestra relevante de los individuos, o por el conocimiento previo del comportamiento esperado y su descripción estadística.

La forma de introducir este comportamiento individual, relativamente sencillo, en la simulación en un modelo basado en agentes es el primer problema que se estudia en este trabajo. La programación de este comportamiento en muchos de los paquetes de software existentes es relativamente compleja, y por ello se analiza una solución más sencilla aunque igualmente potente, utilizando una plataforma desarrollada en el Instituto de Física de Cantabria (IFCA), denominada *Machanguitos*[3].

Por otra parte, en el caso indicado del problema del tráfico en una ciudad un modelado basado en agentes requiere considerar la simulación de miles o incluso millones de agentes, correspondientes a los conductores de los vehículos. Aunque se considere un comportamiento individual relativamente sencillo, la simulación de un número tan elevado de agentes conlleva problemas prácticos computacionales importantes, desde su implementación en paralelo de la simulación, aprovechando el uso de múltiples procesadores, para que se realice en un tiempo mucho menor y permita obtener resultados en un plazo razonable (horas en vez de días o días en vez de meses), a la resolución de la interacción de los agentes con la descripción de su entorno. Este es el segundo punto que se analizará en detalle en este trabajo, aprovechando el despliegue del software de *Machanguitos* en el supercomputador ALTAMIRA de la Universidad de Cantabria.

En resumen, el objetivo general de este estudio es el análisis de cómo resolver problemas prácticos mediante simulación ABM, y en particular utilizando una plataforma sencilla que incluye el paquete de software denominado *Machanguitos* y su despliegue en el supercomputador ALTAMIRA.

Además, dentro del amplio rango de aplicación de estas técnicas en diferentes sectores como la biología, la economía o las ciencias sociales, se analizará cómo objetivo específico un ejemplo práctico, el cual se explica en la siguiente sección.

1.2 Ejemplo práctico: Simulación de la aportación de nutrientes por parte del ganado en el estudio del problema de eutrofización del embalse de Cuerda del Pozo.

El problema sobre el cual se trabaja en este estudio es una pequeña parte de un proyecto LIFE+ denominado ROEM+ (Gestión Avanzada de la Eutrofización de Aguas Superficiales en Territorio Rural de la Cuenca Hidrográfica del Duero)[4]. El IFCA participa en este proyecto en los sistemas de adquisición de datos y modelado a través de su colaboración con la empresa consultora medioambiental que lo ha diseñado, Ecohydros SL.

ROEM+ se presenta como una posible solución a un problema concreto que es la eutrofización de las aguas superficiales de la cuenca hidrográfica del Duero. Como primer paso se ha escogido el análisis del embalse de La Cuerda del Pozo (CdP), situado en la provincia de Soria. La idea es emplear técnicas de simulación predictiva y gestionar eficazmente la información en tiempo real extraída de una red de sensores repartidos por todo el embalse.

La eutrofización [5] de masas de agua como los embalses es un proceso provocado por una excesiva aportación de nutrientes (principalmente fósforo y nitrógeno) provenientes del entorno que da lugar a que muchos organismos fotosintéticos dispongan de materia prima en cantidades abundantes para proliferar. El incremento de la producción primaria aumenta la regeneración de estos nutrientes lo que incrementa la tasa de crecimiento de algas y de macrófitas. La primera fase de la eutrofización produce un aumento considerable del fitoplancton en la superficie lo que enturbia el agua e impide el acceso de luz a las capas inferiores. Debido a esto las algas más profundas mueren lo que produce un aumento de materia en descomposición sobre las que actúan las bacterias que se traduce en un aumento en el consumo de oxígeno. Cuando el oxígeno es escaso se desprenden además gases tóxicos debido al aumento de las fermentaciones, y todo ello implica una degradación del medio acuático y por tanto una disminución de la biodiversidad.

Las consecuencias de la eutrofización no se quedan aquí, el crecimiento de las algas macrofíticas puede interferir en la navegación, y también existen implicaciones sanitarias ya que se han relacionado desordenes gastrointestinales, respiratorios y dermatológicos causados por el contacto con estas aguas, llegando incluso a causar muertes por toxicidad.

Las aportaciones adicionales de nutrientes pueden tener diversos orígenes, destacando factores como los excrementos animales, o los vertidos de origen residual o industrial. Además el problema de la eutrofización en España no se da solo en casos aislados: cerca del 40% de los embalses/lagos españoles presentan predominancia de cianobacterias, principal indicador del proceso de eutrofización.

Con la finalidad de controlar y prever este fenómeno, en el proyecto ROEM+ se sugiere un sistema (ver figura 1) para la recolecta de datos a través de sensores, los cuales “alimentarán” las diferentes simulaciones para que estas tengan mejores resultados predictivos.

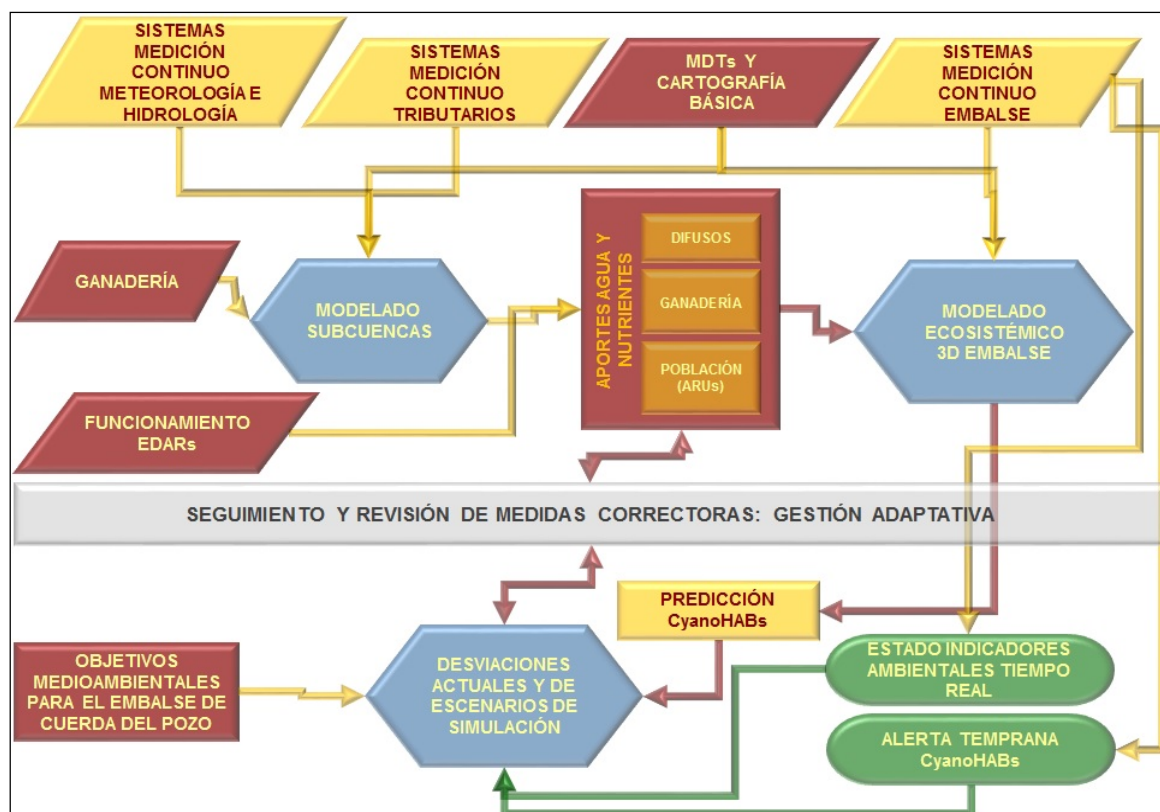


Figura 1. Modelo de la gestión integrada del embalse en el proyecto ROEM+. Se pueden distinguir tres partes: una parte de recolección de datos, otra de modelado y simulación y otra de corrección y predicción.

Este sistema[6] está siendo aplicado de manera íntegra y por primera vez al embalse de La Cuerda del Pozo al cual se le ha diagnosticado eutrofización tras varios estudios sobre la calidad de sus aguas.

El embalse de La Cuerda del Pozo (ver figura 2) está situado en el municipio de Vinuesa, en la provincia de Soria. Tiene una extensión de 2176 hectáreas con una longitud de 12 km y una profundidad de 10 a 36 metros. Este embalse fue terminado de construir en 1941 y es de tipo rural mediterráneo. Abastece de agua potable a Soria y parte de Valladolid permitiendo el riego de 26.000 hectáreas y además, tiene diferentes playas creadas en sus alrededores que se llenan de turistas en los veranos.

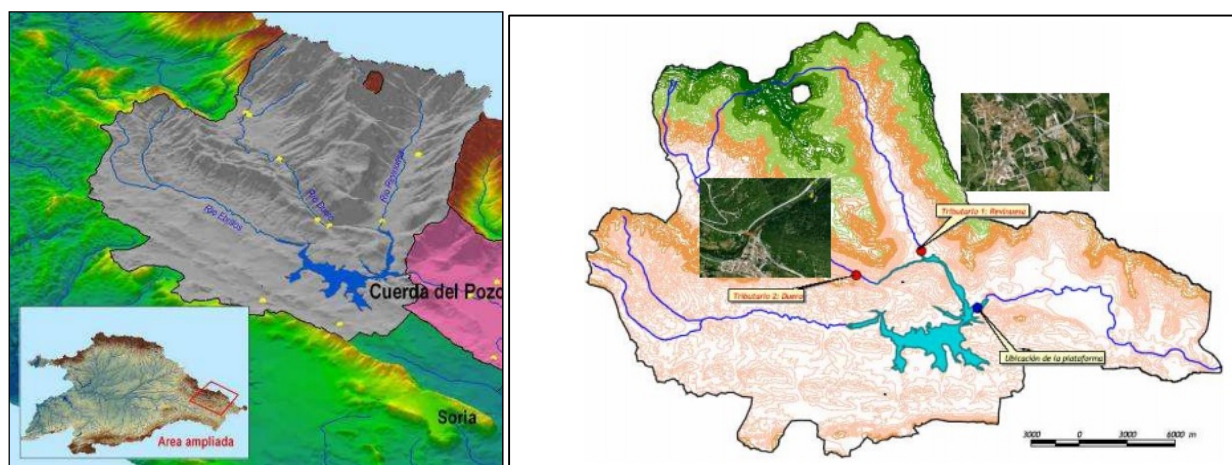


Figura 2. Izquierda: posición del embalse de La Cuerda del Pozo en la Cuenca Hidrográfica del Duero.

Derecha: puntos de monitorización instalados en el proyecto ROEM+.

En el caso del embalse de La Cuerda del Pozo se estima que la aportación de nutrientes al embalse tiene siete orígenes importantes: aportación de fracción sólida (erosión y sedimento), escorrentía urbana, fracción de nutrientes disuelta en aguas de escorrentía, precipitación directa al embalse, **ganadería**, núcleos de población y baño. La contribución de cada punto cambia en función de la época del año, pero se estima que la aportación por ganadería está en torno al 50% respecto a las otras fuentes (figura 3, presentada en [7]).



Figura 3. Aportaciones relativas de nutrientes al embalse de La Cuerda del Pozo en el periodo 2010-2014.

Dada la importancia de esta aportación de nutrientes provenientes de la ganadería, en este estudio se plantea como objetivo buscar una forma de mejorar su estimación.

La solución explorada es la simulación de los procesos relacionados con esta aportación de nutrientes al embalse de forma más detallada, comenzando por la ganadería extensiva de vacas y ovejas existente en la zona. Para ello se propone utilizar la técnica ABM modelando tanto vacas como ovejas mediante agentes, y explorando diferentes patrones de su comportamiento cuando se encuentran cerca de la orilla del embalse, para estimar la incertidumbre de una posible aportación directa de nutrientes al sistema. El interés de este estudio viene motivado por la observación directa de la presencia masiva de ganado en la misma orilla del embalse en los meses de verano, cómo se aprecia en la imagen adjunta (figura 4).



Figura 4. Fotografía tomada en la orilla del Embalse de Cuerda del Pozo en septiembre 2015.

1.3 Planteamiento existente

La solución utilizada a día de hoy para resolver este problema se basa en un conjunto de parámetros estimados con antelación. Para ello se utilizan valores promedios, obtenidos a partir de estadísticas globales y de la experiencia en estudios previos, para las diferentes variables del problema, para obtener mediante cálculos sencillos un valor aproximado de la variable global derivada. En nuestro caso las variables iniciales del problema serían el número de cabezas de ganado que hay en las cercanías del embalse de uno y otro tipo, la cantidad de residuos en términos de fósforo y nitrógeno que genera cada cabeza de ganado y el porcentaje de esta cantidad que alcanza las aguas superficiales por diferentes métodos (en particular el considerado en el planteamiento actual es la aportación por escorrentía).

Asignando valores a estas variables se puede estimar un valor aproximado de la aportación total de manera relativamente sencilla: se ha estimado que hay aproximadamente 6300 vacas y 10200 ovejas en las cercanías, y que aproximadamente cada una genera 90 g de fósforo al día. La fracción de nutriente que consigue alcanzar el embalse depende de la escorrentía, y se considera que para 30 mm de lluvia esta fracción es del 7%.

1.4 Interés de explorar una solución basada en ABM

Cómo se ha indicado, en este estudio se plantea la aplicación de un modelo basado en agentes al problema anterior. Se considera que puede ser una buena solución debido a que los ABM permiten la simulación de un gran número de agentes que tienen comportamientos diferentes y que se relacionan con los diferentes componentes de la simulación. Además, el comportamiento de los agentes puede cambiar, de tal manera que en un principio podemos definir el comportamiento de cada componente, pero también podríamos introducir nuevos componentes y nuevos comportamientos para intentar llegar a mejores resultados. También se podría dar el caso en el que el comportamiento de los agentes y sus relaciones sean dinámicas dentro de la simulación, es decir, que en función de cómo transcurre la simulación, el comportamiento de estos cambie. Desde este punto de vista una solución de este tipo podría aportar flexibilidad, escalabilidad y potencia a la hora de definir el problema.

En el caso concreto del problema de eutrofización en CdP, el modelado de las vacas y ovejas mediante agentes tiene interés principalmente para estimar una posible aportación directa al embalse. Como se ha dicho, es frecuente encontrar rebaños en la zona próxima al agua, especialmente en el caso de las vacas y en la época de verano, que es cuando mayor incidencia puede tener su aportación al proceso de eutrofización. Además de la posible aportación directa al agua del embalse si las vacas realizan su deposición mientras se refrescan dentro del agua, la zona del embalse que ocupan es usualmente zona inundable, de modo que cuando el nivel del embalse vuelve a subir incorpora directamente las deposiciones en esta zona, que como se puede apreciar en la fotografía tomada el pasado verano son muy abundantes.

El objetivo de esta simulación mediante ABM va a ser estimar una cota superior para esta aportación, que aunque inicialmente se estima no muy relevante dentro de la aportación global, no se puede evaluar de forma sencilla. Variando los parámetros de la simulación esperamos obtener una cota. Para ello deberemos fijar un modelo básico de comportamiento del ganado. Este modelo tiene que tener en cuenta las características de los agentes como su interacción con el resto de elementos de la simulación. En el capítulo 4 se describe con más detalle el modelo utilizado.

2 Plataformas basadas en Agentes

2.1 Introducción al Modelado Basado en Agentes (ABM)

El modelado basado en agentes es una técnica de simulación relativamente nueva la cual se ha empleado para resolución de problemas reales en diferentes campos de la ciencia. Esta técnica trata de modelar sistemas complejos mediante conjuntos de entidades totalmente autónomas llamados agentes.

Los agentes son los protagonistas de esta técnica de modelado y sus características hacen que los ABM destaquen en varios puntos respecto a otras técnicas de modelado destacando su flexibilidad a la hora de describir un sistema así como la capacidad de capturar comportamientos emergentes, fruto de la interacción entre los agentes.

Los agentes interactúan con el entorno y con otros agentes siendo estos capaces de tomar decisiones en base a unas reglas preestablecidas mediante programación. Estas reglas definen un estado interno del agente y pueden ser sencillas, y no cambiar a lo largo de la simulación, o bien complejas, las cuales hacen al agente flexible e incluso con capacidad de adaptarse a lo que ocurre en la simulación.

Otra característica importante que da versatilidad a ABM es la granularidad variable de los agentes: esto se traduce en que los agentes no tienen que representar un individuo sino que en función del sistema que se plantea estudiar pueden representar colectivos. Los propios agentes tienen características heterogéneas lo que permite obtener un amplio abanico de agentes partiendo de un mismo modelo, por ejemplo, en una simulación en el campo de las ciencias sociales se podría encontrar que el modelo de un individuo sea descrito por su clase social y su afinidad religiosa, dando como resultado agentes con diferentes preferencias o motivaciones.

Los comportamientos emergentes son el resultado de las interacciones de los individuos que participan en un sistema. Este tipo de comportamientos no se pueden estudiar como la aportación de todas las partes sino que se tiene que tener en cuenta las interacciones entre las partes. Algunos ejemplos de este tipo de comportamientos son los que ocurren en las bandadas de pájaros.

2.2 Comparativa de Software ABM

Existe un gran número de herramientas (*toolkits*) que permiten el desarrollo de simulaciones basadas en el modelado de agentes [8-17]. Estas aplicaciones están creadas para facilitar el desarrollo de simulaciones en diferentes sistemas. Así mismo vemos que gran cantidad de estas herramientas están orientadas a las ciencias sociales, también hay otras especializadas en simulaciones en los campos de la biología y física. También hay plataformas que están orientadas a la educación. Aunque muchas de estas herramientas son de libre uso, otras son de pago.

En un plano técnico cabe destacar los siguientes puntos en estas plataformas o herramientas:

- **Lenguaje de programación** con el que se ha desarrollado la herramienta (entre los que destacan lenguajes como Java, C/C++, Prolog y Python).
- **Estándar FIPA:** algunas de las plataformas están desarrolladas bajo los estándares de FIPA, una asociación que especifica cómo deben plantearse los sistemas basados en agentes.
- **Integración GIS y 3D:** Algunos toolkits permiten el uso de capas GIS y elementos 3D.

En la siguiente tabla se muestran algunas herramientas con sus especificaciones técnicas:

Nombre de la herramienta	Aplicaciones	Lenguaje de programación	FIPA	GIS	3D
<i>FLAME</i>	Uso en diferentes campos de la ciencia (genérico)	C	No	Sí	Sí
<i>JADE</i>	Uso genérico, basado en el estándar FIPA	Java	Sí	No	No
<i>Repast</i>	Ciencias sociales	C++	No	Sí	Sí
<i>NetLogo</i>	Física, biología, química y ciencias sociales	NetLogo	No	Sí	Sí
<i>Machanguitos</i>	Uso genérico	C++, Lua	No	Sí	No

Tabla 1. Algunas de las herramientas disponibles para la simulación basada en modelado de agentes.

2.3 Ejemplo de aplicación

Existe un gran número de proyectos que basan su estudio en el modelado basado en agentes aplicado a diferentes problemas.

Por ejemplo, en [18] se realiza un estudio de la gestión del agua con sociedades artificiales. En este estudio describen como han realizado una simulación que permita observar cómo evoluciona el abastecimiento de aguas urbanas y además, dan la opción al usuario de realizar cambios en la política de uso de estas aguas para observar cómo evoluciona la simulación respecto a otras políticas de uso.

Este estudio contempla diferentes actores como la distribución y evolución de las urbanizaciones en la zona simulada. Esta evolución se puede controlar modificando los precios de la vivienda en la simulación, que hará que la ciudad pueda crecer más o menos lenta con implicaciones obvias en el abastecimiento de agua.

También se tiene en cuenta diferentes parámetros de las familias (agentes) que componen la ciudad, como puede ser el número de integrantes, la edad media de la familia, los ingresos y ahorros, su clases social... al final este es el actor que decide sobre el gasto de agua que producen y sobre el movimiento de la unidad familiar. Por otra parte, también se tiene en cuenta factores climatológicos y el número de depósitos de agua o de plantas de desalinización del agua.

Implementando todos estos elementos bajo una misma simulación consiguen estimar el agua disponible en lugares como la región metropolitana de Barcelona.

2.4 Descripción del contexto de aplicación en este caso

Si se deseara estudiar el problema planteado en la sección 1.2 con un modelado basado en agentes, tendríamos dos modelos de agentes, uno que describiría el comportamiento de las ovejas y otro que describiría el comportamiento de las vacas en las cercanías del embalse. Este comportamiento se podría describir mediante programación y se reduciría al movimiento de los agentes en función del entorno y la deposición del abono sobre el terreno.

Para comprobar que el agente no está, por ejemplo, en mitad del embalse, se necesita una *toolkit* que permita la integración con capas GIS, para añadir relieve a la simulación y hacerla más realista. Por otra parte, sería necesario que la herramienta permita cambios en las capas GIS, ya que al ser un embalse el nivel del agua variará en función de la época del año.

En este caso, la información que se desea extraer (la cantidad de P y N que el embalse absorbe) no nos la dan los agentes directamente, por lo que se necesitará un sistema para almacenar los datos relacionados con las deposiciones de los agentes.

El número de agentes que se pretende simular, incluyendo tanto vacas como ovejas, es cercano a los 17000, un número relativamente alto pero posiblemente no sea necesario una plataforma de alto rendimiento para simularlo. Dado que el impacto esperado correspondiente a la aportación de las vacas se espera que sea mucho mayor que el la aportación de las ovejas, y la metodología a seguir es la misma, se ha optado por modelar por el momento solamente la aportación de las vacas.

2.5 Conclusiones

Los ABM son modelos compuesto por agentes que además de interactuar con el entorno también lo hacen entre si, lo que permite no solo el estudio de los agentes como individuos si no de sus relaciones (fenómenos emergentes). La descripción de estos agentes es flexible gracias a su granularidad y la posibilidad de definir un amplio espectro de agentes partiendo de sus atributos. Su comportamiento puede variar a lo largo de la simulación lo que dota a los agentes de cierta capacidad de adaptación/aprendizaje.

Esta técnica de modelado puede ser de gran utilidad en estudios de fenómenos de emergencia (ya que otras técnicas no son capaces de simular estos fenómenos). También en sistemas en los que los agentes tiene un comportamiento que evoluciona con lo que ocurre en la simulación, ya sea su comportamiento con el medio o con otros agentes.

El ABM es útil en diferentes campos de la ciencia y se pueden encontrar diversas herramientas especializadas en los diferentes campos. Para el caso propuesto en la sección 1.2 se ha decidido utilizar el *toolkit* Machanguitos por las razones que se explican en el apartado 3.

3 Machanguitos: Una implementación flexible y escalable de un modelo de agentes.

3.1 Introducción

Machanguitos [3] es el *toolkit* que se ha elegido para la simulación del problema propuesto. Como se indicaba en la tabla 1 está programado en C++ y Lua, y ha sido desarrollado por Luis Cabellos (IFCA). Dispone de la funcionalidad de capas GIS y además cumple con todos los requisitos propuestos en el apartado 2.4 en el que se describe una posible solución basada en ABM al problema del estudio.

A pesar de que en el problema que nos concierne no se necesite es una herramienta escalable pensada para ejecutarse en sistemas de alto rendimiento (HPC). Este rendimiento se consigue debido a que Machanguitos es una aplicación concurrente de simulaciones de agentes que divide la ejecución en múltiples procesos comunicados mediante Message Passing Interface (MPI).

3.2 Componentes

3.2.1 Procesos

A la hora de ejecutar Machanguitos es posible elegir la cantidad de procesos que se desean iniciar para ejecutar la simulación, sin embargo, es necesario ejecutar tres procesos como mínimo para que la simulación pueda ejecutarse. De estos procesos el primero será el master de la ejecución (Driver Process), el segundo proceso se encarga de las capas GIS y otros datos compartidos (Data Process) y el tercer proceso y sucesivos se encargan de ejecutar la lógica de los agentes (Client Process). Se puede ver cómo están relacionados los diferentes procesos en la figura 5.

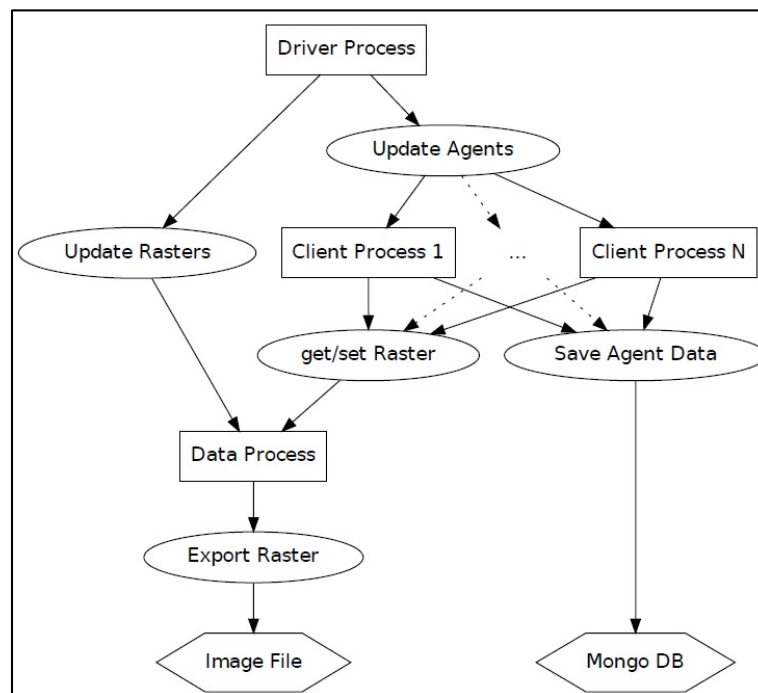


Figura 5. Esquema relacional entre los procesos que se ejecutan al utilizar Machanguitos

3.2.1.1 Proceso Driver

Todos los procesos se inician en el mismo momento pero el proceso Driver es el que gestiona como debe ejecutarse la simulación. Las simulaciones en *Machanguitos* vienen definidas por un fichero de configuración (ver ejemplo, tabla 4) en lenguaje **Lua** en el que se definen diferentes parámetros que el proceso Driver lee y ejecuta. En este fichero se definen:

- **Las capas de datos de la simulación.** El proceso *Driver* envía los mensajes necesarios para que el proceso *Data* cargue (de un fichero externo) o cree las diferentes capas de la simulación. Los agentes son capaces de interactuar con estas capas y puede servir como capas de consulta (en las que el agente extrae información de la capa) o de almacenamiento de datos fáciles de extraer.
- **Los agentes de la simulación.** El proceso *Driver* divide la carga de los agentes entre los procesos *Client* y manda mensajes necesarios para que se creen en memoria en estos procesos.
- **Parámetros de simulación.** Como puede ser el tiempo y el número de iteraciones a ejecutar.

Como se puede ver en la **figura 6**, una vez que se han cargado en memoria los diferentes recursos en los procesos correspondientes el proceso *Driver* se encargará de marcar la pauta de la simulación indicando a los procesos *Clients* el comienzo de cada iteración, esperando a que los clientes ejecuten una iteración, actualizando las capas de datos y repitiendo este proceso el número de veces que se ha definido en el fichero de configuración hasta completar la simulación.

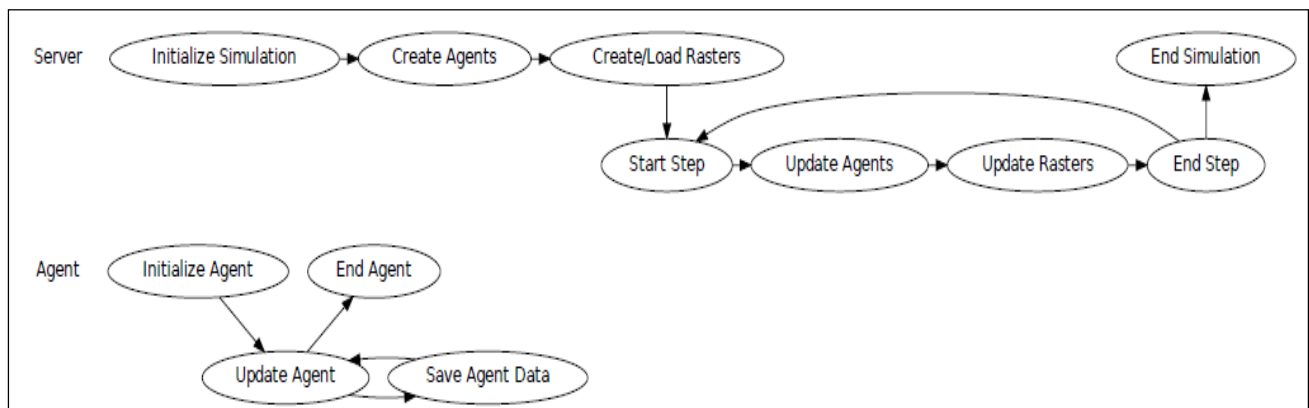


Figura 6. Arriba: Representación esquemática de una simulación completa ejecutada con *Machanguitos*.
Abajo: Funcionamiento de un agente exportando información en cada iteración (*Save Agent Data*).

3.2.1.2 Proceso Data

El proceso *Data* tiene en memoria los datos de las capas GIS de la simulación y es el proceso que consultan los agentes (desde los procesos *Client*) para comprobar el estado de una capa determinada y modificar sus valores si así se ha definido en la descripción de los agentes.

3.2.1.3 Procesos Client

Los procesos *Client* tienen en memoria los agentes de la simulación y ejecutan su función de actualización cada vez que el proceso Driver marca el comienzo de una nueva iteración. El número de procesos *Client* que se ejecutan en una simulación puede variar y en principio se podría decir que cuantos más procesos paralelos se ejecuten menor va a ser el tiempo que tarde en finalizar la ejecución. Como se discute más adelante, esta mejora de tiempo depende de diversos factores. Como ya se ha comentado el proceso *Client* se comunica con el proceso *Data* para almacenar información en las capas GIS, pero además tiene la posibilidad de exportar a una base de datos no relacional (mongo DB) cualquier variable que los agentes hayan utilizado en cada iteración (ver **figura 5**).

3.2.2 Descripción de los agentes

Una de las principales ventajas de *Machanguitos* es la sencillez con la que se puede describir un agente. Para hacer esto *Machanguitos* añade una capa de abstracción haciendo que el lenguaje de programación con el que se describan los agentes no sea C sino Lua, un lenguaje de scripting que tiene una sintaxis relativamente sencilla y familiar para cualquiera con nociones básicas de programación.

El fichero Lua describe todo lo relativo al agente, desde su estado inicial a la evolución en cada paso de la simulación. Obviamente se puede acceder al resto de capas (capas GIS y rasters) desde este punto.

La tabla 2 muestra un ejemplo de descripción de un agente básico usando el lenguaje Lua.

```
function Agent:init()
  io.write( "*** cat init\n" )
  self.val = false;
  self.cname = "micho" .. math.random(100)
end
function Agent:update( delta )
  if self.val then
    io.write( "*** cat " .. self.cname .. " = " .. delta .. " TT\n" )
  else
    io.write( "*** cat " .. self.cname .. " = " .. delta .. " FF\n" )
  end
  self.cname = "micho" .. math.random(100)
  self.val = not self.val
end
```

Tabla 2. Ejemplo de fichero Lua describiendo el comportamiento de un agente.

Además una vez descrito el comportamiento del agente individual hay que declarar el número de agentes que se desean invocar.

3.2.3 Capas “GIS”

Las capas GIS nos permiten añadir capas en las que se describe un "relieve". Desde el punto de vista del agente se darán casos en los que el agente no pueda acceder a un punto del mapa debido a la pendiente del terreno o debido a que sea un punto inaccesible (como por ejemplo el fondo del embalse). Una característica importante de *Machanguitos* es que dispone de capas GIS en un punto intermedio entre capas dinámicas y estáticas ya que se tiene la posibilidad de cambiar entre capas cuando se desee.

3.3 Conclusiones

Machanguitos es una aplicación que facilita la simulación basada en agentes gracias a que permite describir con relativa facilidad el comportamiento de los agentes. Además pone a disposición del usuario la posibilidad de comunicar los agentes con capas GIS, haciendo que estos puedan consultar o modificar valores de dichas capas. También expone otra forma de exportar los datos a través de una base de datos no relacional (MongoDB). Se entiende que *Machanguitos* ha sido planteada para ser una aplicación escalable ya que, como vemos en la **figura 5**, divide los procesos *Client* para mejorar el rendimiento de la simulación.

Machanguitos reúne todas las necesidades recogidas en el apartado 2.4 y además facilita, desde un punto de vista técnico, el planteamiento de la simulación, por estas razones se considera que puede ser de gran utilidad a la hora de simular el problema propuesto.

4 Test de la plataforma para el caso de uso de CdP

4.1 El problema del impacto de la ganadería

Como se mostraba en la **figura 3**, la aportación de nutrientes provenientes de la ganadería al embalse de la Cuerda del Pozo ronda el 50% del total, lo cual supone un grave problema de cara al proceso de eutrofización del embalse debido al exceso de nutrientes en el mismo. En concreto, el problema reside en que una parte importante del abono depositado a diario por las vacas y ovejas del entorno del embalse acaba llegando a este. Por esta razón una modelización debería describir el movimiento de la ganadería alrededor del embalse y tener en cuenta en qué posición se ha realizado deposición de abono, para poder calcular si esta llegará al agua del embalse. De esta forma se podría estimar la cantidad de fósforo y nitrógeno que la ganadería aporta al embalse de forma más precisa que a través de una parametrización.

A la hora de abordar este trabajo no se cuenta con datos suficientes que permitan realizar esta simulación de forma completa, ya que ello requeriría la monitorización al menos de una muestra de ganado a lo largo de todo el año para conocer su actividad, y la determinación con cierta precisión de la situación de vacas y en el entorno del embalse. Por ello y dado que el objetivo es mostrar la posible utilidad de los sistemas ABM para abordar problemas complejos, se ha optado por **restringir el análisis a una posible contribución extra a la ya conocida: la que corresponde a las deposiciones directas de las vacas en la orilla del embalse**, que se integran en el mismo cuando el agua vuelve a sumergir la zona en que se han realizado.

4.2 Descripción de las cantidades de referencia

Existen un gran número de variables que se pueden tener en cuenta a la hora de realizar la simulación. En este estudio se ha tratado de tener en cuenta las más relevantes en cuanto al problema final se refiere, es decir el aporte final de Nitrógeno (N) y Fósforo (P) al agua del embalse.

Por ello partimos de la cantidad diaria de estiércol que genera una vaca [19] que es de 64 kg en promedio; de esa cantidad 306 g (aproximadamente el **0.47%**) son de nitrógeno y 40 g son de fósforo (aproximadamente el **0.06%**).

Nuestro objetivo es simular el impacto a lo largo del ciclo anual de las deposiciones directas de las vacas en la orilla del embalse, y para ello hemos considerado un periodo de simulación que se extiende desde el 1 de Abril al 31 de Octubre de 2014. La razón principal de elegir este periodo es por las condiciones de temperatura del mismo, que favorecen en principio la presencia del ganado en las orillas del embalse (ver tabla adjunta).

Mes	Año	Temperatura Máxima/°C	Temperatura mínima/°C	Cota Embalse/hm ³
Abril	2014	16.2	5.5	218,2
Mayo	2014	16.8	5.7	234,7
Junio	2014	21.6	9.1	221,1
Julio	2014	24.0	11.6	198,4
Agosto	2014	25.1	11.8	172,0
Septiembre	2014	23.0	11.4	145,3
Octubre	2014	20.0	8.7	126,6
Noviembre	2014	11.0	4.2	125,2
Diciembre	2014	8.5	0.2	150,0
Enero	2015	8.5	-1.0	163,5
Febrero	2015	4.9	-0.5	173,5
Marzo	2015	11.9	1.8	192,8
Abril	2015	17.0	4.3	215,7

Tabla 3. Datos relevantes para la simulación a lo largo de un año: temperaturas máxima y mínima mensual, y cota superior del embalse (fuente: estación DORII en CdP).

Cómo muestra igualmente la tabla adjunta, el nivel (cota) del embalse entre Abril de 2014 y Abril de 2015 decreció hasta Noviembre, para incrementarse posteriormente, por lo que se ha calculado el aporte directo de forma simplificada sumando las deposiciones efectuadas por el ganado en la orilla del embalse que es accesible al mismo en verano, pero queda posteriormente sumergida de nuevo, incorporando los nutrientes directamente al agua del embalse. No se ha tenido en cuenta el importante proceso de mineralización de estos nutrientes, ya que el objetivo del trabajo es solamente estimar la viabilidad y posible interés del método.

La simulación se realiza por tanto desde el 1 de Abril del año 2014 al 30 de Octubre del 2014, pero se ha simplificado considerando una duración mensual fija de 30 días, es decir un periodo total de 210 días. El paso temporal de simulación propuesto ha sido de 10 minutos, lo que permite evaluar el rendimiento del software *Machanguitos*: cada escenario simulado requiere un total de **30240** iteraciones por agente.

4.3 Implementación

A la hora de la implementación se han tenido que simplificar varios puntos en la simulación. Como se ha comentado en el punto anterior se dispone de los mapas de las orillas del embalse, que se considera que cambian con periodicidad mensual, de acuerdo a las cotas indicadas en la tabla anterior, por lo que en la simulación se pueden apreciar “saltos” en los que la cota de agua pasa de un valor a otro de manera escalonada, como se aprecia en la figura siguiente:

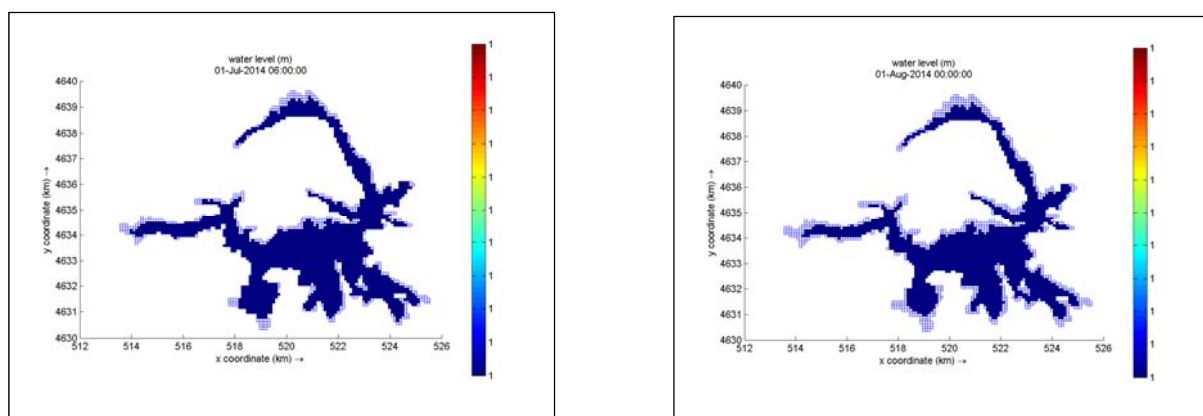


Figura 7. La figura de la izquierda muestra la cota de agua (azul oscuro plano) en el mes de Julio; la de la derecha muestra la cota de agua (azul oscuro plano) en Agosto.

A medida que el tiempo simulado avanza las capas van cambiando al cambiar de mes, pero no de forma gradual, sino escalonada.

Para facilitar la incorporación de estas capas a la simulación, se ha utilizado un editor de imagen. Se ha buscado utilizar solo el canal de grises y se ha dado un tamaño fijo de 700px de ancho por 700px de alto a todas las imágenes. Transformando de esta manera la imagen se consigue que el programa sea capaz de leer adecuadamente la capa. En la siguiente figura se puede observar el resultado de este proceso.

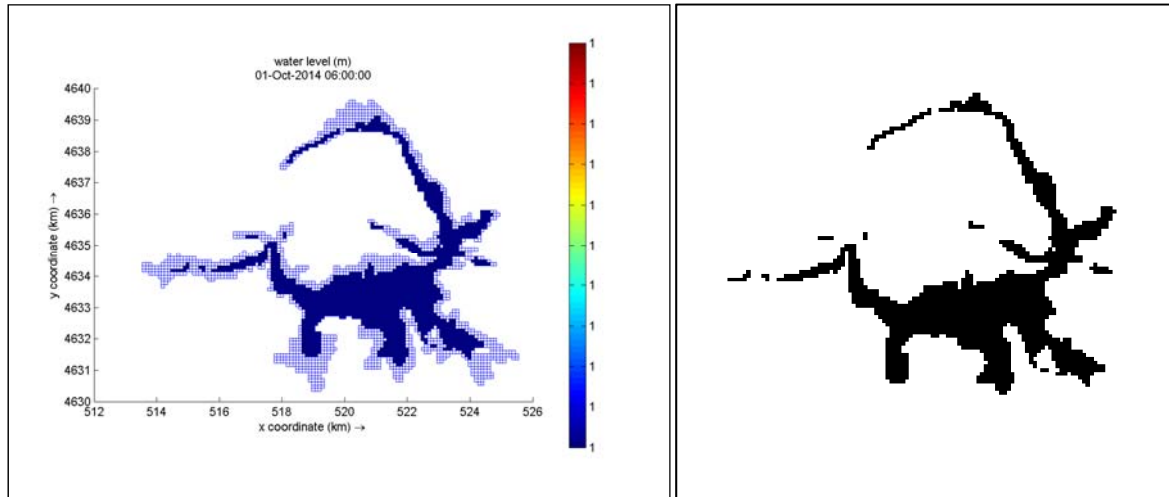


Figura 8. La figura de la izquierda muestra la cota de agua (azul oscuro plano) en el mes de Octubre. La figura de la derecha es misma de la izquierda tras haber definido las orillas.

A la hora de la simulación los agentes comprobarán los puntos de la figura para determinar el límite de su movimiento por la orilla del embalse.

Estas imágenes son cargadas en una capa que tendrá asociada una función de actualización (archivo .lua), la cual gestionará cuando debe cargarse la imagen de cada mes.

Además de esta capa, se ha utilizado otra capa adicional para la salida de datos que es la capa de abonado. Esta capa está definida en la configuración inicial de la simulación pero además, se la puede asociar un archivo .lua, el cual permite modificar a lo largo de la simulación los valores de la propia capa, independientemente de los agentes. En la capa del abono, tendremos un valor por cada pixel que hace referencia a la cantidad de abono depositada por los agentes. En un principio y para este estudio solo los agentes son capaces de modificar el valor de los pixeles de esta capa, pero se podría desarrollar una función de actualización con el paso del tiempo, por ejemplo debido a procesos de mineralización. Además de las capas de extracción de datos existe una capa auxiliar que nos permite llevar la cuenta del tiempo dentro de la simulación.

Existe un fichero de configuración en el que se declaran todas las funcionalidades hasta ahora comentadas, y además se pueden declarar los parámetros de conexión de una base de datos y los diferentes agentes.

El comportamiento de los agentes se declara también en archivos .lua (un fichero por cada tipo de agente) definiéndose una función de inicialización y otra de actualización.

En el caso de las vacas, en la función de inicialización del agente se define una posición inicial y un parámetro m_a que describe la cantidad de abono depositada en cada iteración por las vacas. Se proponen hasta 10 posiciones diferentes iniciales para los agentes, simulando así la situación de diferentes establos alrededor del embalse. Cada uno de ellos está situado a menos de 1 km. del embalse, de tal forma que las vacas llegan a la orilla del embalse en cuestión de horas.

```
Function initialize()
config.setLogLevel("info")
config.setvars({
    starttime = 0.0,
    endtime = 24 * 30 * 7,
    iters = 6 * 24 * 30 * 7,
    randomseed = 3165,
})
```

```

data.createLayer("daily",
{
    x0 = 0.0,
    x1 = 10000.0,
    y0 = 0.0,
    y1 = 10000.0,
    w = 700,
    h = 700,
    default = 0,
})
data.setLayerUpdate("daily", "dayupdate")
data.createLayer("manure",
{
    x0 = 0.0,
    x1 = 10000.0,
    y0 = 0.0,
    y1 = 10000.0,
    w = 700,
    h = 700,
    default = 0,
    layers = 1,
})
data.loadLayer("area", "1.png",
{
    x0 = 0.0,
    x1 = 10000.0,
    y0 = 0.0,
    y1 = 10000.0,
})
data.setLayerUpdate("area", "areaupdate")
config.addAgent("cow", 6300)
end

function endIteration(num)
    if num % 300 == 0 then
        raster.manure.save("export/output"..num..".png");
    end
end
end

```

Tabla 4. Configuración inicial de la simulación en donde se declara el número y tipo de agentes que se desea simular, el tiempo de simulación y el número de iteraciones a realizar. También se declaran las diferentes capas que se utilizarán.

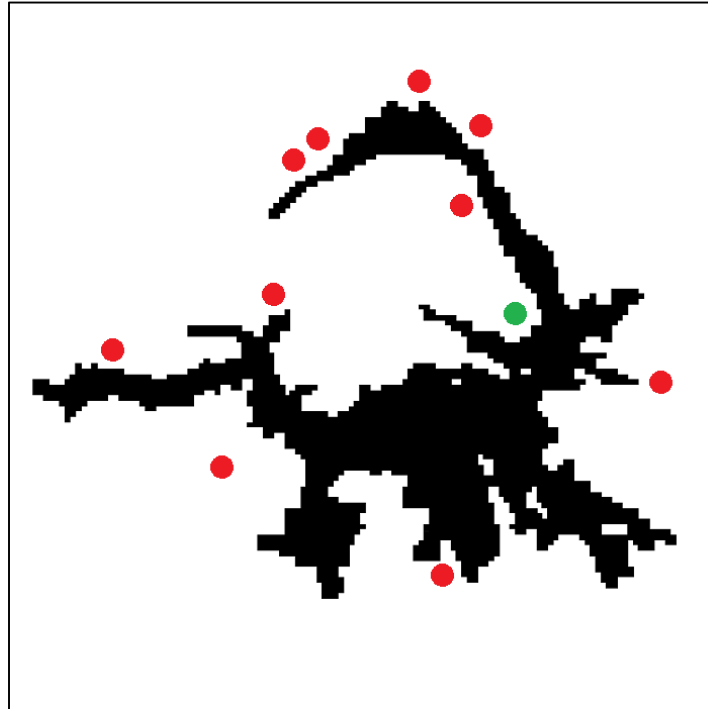


Figura 9. Mapa del embalse de Cuerda del Pozo tras ser tratado para su uso en la simulación. Los puntos rojos son la situación de los establos representados en la simulación. El punto verde representa un lugar de preferencia al que los agentes tenderán a ir, que se corresponde con el observado in-situ.

La función de actualización es un poco más compleja ya que existen varios condicionales. El primer condicional comprueba la hora del día de tal manera que si la hora simulada no está entre las 8 de la mañana y 8 de la tarde el agente no se moverá, simulando que la vaca está estabulada o durmiendo. En caso contrario pasaremos a comprobar el siguiente condicional que es el que compara un número aleatorio con uno de los parámetros de la simulación. Dicho parámetro p_m es un porcentaje que intenta reproducir la probabilidad de que la vaca se mueva o no, este parámetro se introduce en la simulación debido a que las vacas tienden a estar gran cantidad de tiempo acostadas en la misma posición. Si esta condición se cumple se invoca a un método llamado “*checkGis*” que se explica a continuación:

- “*checkGIS*”: Este método comprueba la posición actual del agente (x,y) y le añade un valor para simular el movimiento (x+dx, y+dy), el nuevo valor es temporal y con él se comprueba, con ayuda de la capa GIS, si dicho punto es accesible para el agente. En caso afirmativo invoca a un segundo método llamado “*depositManure*”. Se ha añadido una preferencia de movimiento de tal forma que las vacas se mueven ligeramente más rápido cuando se aproximan a la coordenada (7300,4580). Esta coordenada corresponde al punto de color verde que aparece en la figura 9, que a su vez corresponde con la posición en la que se tomó la imagen de la figura 4, donde existe mayor concentración de vacas.
- “*depositManure*”: Este método se ocupa de añadir la cantidad de abono definida en la inicialización del agente a la capa de abono.

El código escrito en *Lua* del comportamiento descrito anteriormente se puede ver en la tabla 5.

```

function Agent:init()
    self.stable = math.random(10);

    if (self.stable == 1) then
        self.x = 4229;
        self.y = 2013;
    end
    if (self.stable == 2) then
        self.x = 4520;
        self.y = 1980;
    end
    if (self.stable == 3) then
        self.x = 5757;
        self.y = 1214;
    end
    if (self.stable == 4) then
        self.x = 6414;
        self.y = 1657;
    end
    if (self.stable == 5) then
        self.x = 6586;
        self.y = 2757;
    end
    if (self.stable == 6) then
        self.x = 8929;
        self.y = 5400;
    end
    if (self.stable == 7) then
        self.x = 5928;
        self.y = 7985;
    end
    if (self.stable == 8) then
        self.x = 3786;
        self.y = 4200;
    end
    if (self.stable == 9) then
        self.x = 3042;
        self.y = 6571;
    end
    if (self.stable == 10) then
        self.x = 1443;
        self.y = 5157;
    end
    self.x = self.x + math.random(5);
    self.y = self.y + math.random(5);

    self.changeDir = true;
    self.counterDir = 0;

    self.grassToManure = 1.0;
    self.cowLaziness = 0.5;
end

```

```

function Agent:update(delta)
    local rstDaily = raster.daily;
    local hour = rstDaily:get(0, 0, 0);
    if (hour > 8 and hour < 20) then
        local movementPercent = math.random();
        if (movementPercent > self.cowLaziness) then
            self:checkGis(delta);
        end
    end
end
function Agent:checkGis(delta)
    if self.changeDir == true then
        local favouriteX = 510;
        local favouriteY = 320;
        local randX = math.random();
        if (self.x < favouriteX) then
            randX = randX + 0.25;
        end
        local randY = math.random();
        if (self.y < favouriteY) then
            randY = randY + 0.25;
        end
        if (randX >= 0.5) then
            self.dirX = 1; else self.dirX = -1;
        end
        if (randY >= 0.5) then
            self.dirY = 1; else self.dirY = -1;
        end
        self.changeDir = false;
    else
        self.counterDir = self.counterDir + 1;
        if ( self.counterDir > 5 + math.random()*10 ) then
            self.counterDir = 0;
            self.changeDir = true;
        end
    end
    self.dx = self.dirX*((50.0)*delta);
    self.dy = self.dirY*((50.0)*delta);
    tempX = self.x + self.dx;   tempY = self.y + self.dy;
    local rstArea = raster.area;
    local area = rstArea:get( 0, tempX, tempY);
    if ( area > 0 and tempX < 10000 and tempX > 0 and tempY
    < 10000 and tempY > 0 )then
        self.x = tempX; self.y = tempY;
        self:depositManure(delta);
    end
end
function Agent:depositManure(delta)
    local rstManure = raster.manure;
    local inc = self.grassToManure;
    rstManure:increment( 0, self.x, self.y, inc );
end

```

Tabla 5. Código Lua para describir el comportamiento de las vacas

Una vez terminada la simulación se necesita un proceso de extracción de datos. Se ha empleado un script escrito en C# para comparar los valores de la imagen GIS de Abril (tras haberla tratado), con la capa de abono. A medida que se compara pixel por pixel se realiza un sumatorio de los valores de la capa de abono que coincidan con los pixeles de agua (negro) en la capa de Abril, es decir, se suman el valor de los pixeles que estarían inundados. El número (m_{abono}) que se espera de este proceso es la cantidad (en kg) de abono depositada por los agentes, esta cantidad permite a su vez calcular la cantidad de fósforo (el 0.06%) y nitrógeno (el 0.47%) según lo que se ha explicado en el primer párrafo de la sección 4.2.

En la simulación se ha estudiado el cambio en los resultados de acuerdo con la variación de un parámetro que afecta directamente a la simulación, este parámetro es la probabilidad que los agentes tienen de moverse por el terreno (p_m). Se han elegido los valores 0.5, 0.6, 0.7, 0.8 y 0.9, cuanto mayor sea el valor de este parámetro las vacas se volverán menos activas, moviéndose menos. Se han realizado un total de 5 simulaciones diferentes con los diferentes valores del parámetro p_m , cada simulación se ha repetido 3 veces con diferentes *randomseed* (parámetro que nos asegura de que simulación dispone números aleatorios totalmente diferentes), para al final obtener un valor medio de la cantidad de abono depositada ($\langle m_{\text{abono}} \rangle$).

4.4 Resultados y análisis de la variación de los resultados de acuerdo a los diferentes patrones de comportamiento de las vacas

Como se ha comentado anteriormente los agentes interaccionan con las diferentes capas de la simulación, estas capas, si se desea, son guardadas en formato PNG el cual nos permite ver a simple vista el estado de un capa en un tiempo dado. Gracias a esto se pueden realizar una serie de comprobaciones que permiten saber si la simulación funciona correctamente. En la figura 10 se muestra la evolución temporal de la capa de abono perteneciente a una de las simulaciones realizadas para diferentes valores del parámetro p_m (50 y 90).

En ella se puede observar como en la figura 10.a los agentes llegan sustancialmente más lejos respecto a los diferentes establos. Además, en ambas figuras (10.a y 10.b) se puede ver la silueta del embalse lo que muestra cómo los agentes llegan a la orilla del mismo sin que se consideren las deposiciones directas.

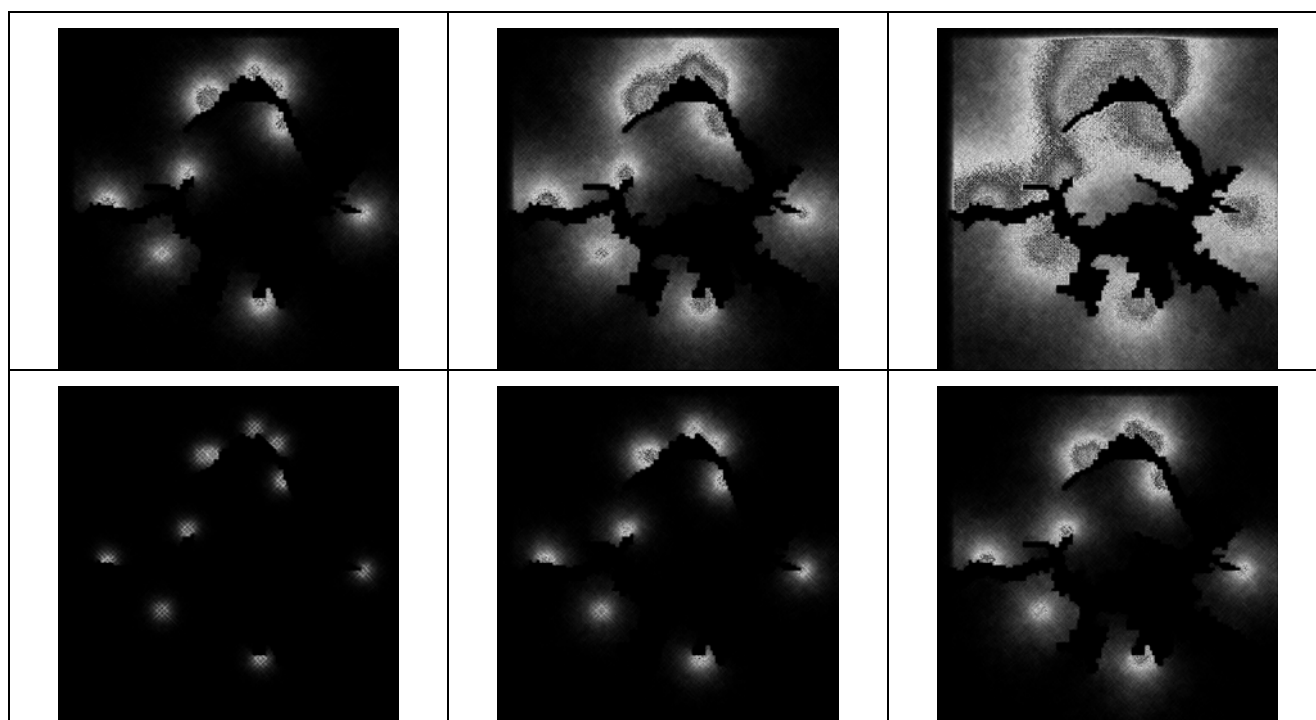


Figura 10. La figura superior (a) y la inferior (b) muestran la capa de abono para diferentes etapas de la simulación cuando el parámetro p_m vale 0.5 y 0.9 respectivamente. En ambas figuras se muestran 3 imágenes que representan 1, 3 y 7 meses tras el inicio de la simulación.

En la tabla siguiente se muestra el resultado de las tres simulaciones realizadas para cada uno de los cinco valores del parámetro de movilidad de las vacas (p_m).

p_m	$m_{1abono}(kg)$	$m_{2abono}(kg)$	$m_{3abono}(kg)$	$\langle m_{abono} \rangle (kg)$
0,5	8470	8420	8407	8433±39
0,6	7198	6845	7299	7114±275
0,7	5422	5433	5221	5359±138
0,8	3614	3789	3614	3672±117
0,9	2031	2079	2016	2042±38

Tabla 5: Estimación obtenida del abono depositado por las vacas en la orilla del embalse que queda sumergida, en kg, para diferentes valores del parámetro de movilidad.

Cómo se observa en la tabla anterior, las vacas depositan hasta un total de 8500 kg de abono en la orilla, que correspondería a unos 40 kg de N. Teniendo en cuenta un volumen embalsado de agua de más de 150 Hm³, el incremento correspondiente esperable en la concentración de nitratos y/o nitritos estaría en un máximo de 0.012 mg/l, un valor pequeño, pero no despreciable, ya que los valores típicos del embalse están en el entorno de los 0.10 mg/l.

Debe tenerse en cuenta que el efecto del abono depositado en las orillas inundables es mayor que el que corresponde al depositado en otras zonas cercanas al embalse, que también se incorporaba a la estimación a partir del cálculo de la escorrentía. Por tanto el valor estimado de 0.012 mg/l podría considerarse en principio una cota superior de incertidumbre a este efecto de aportación directa al embalse.

4.5 Propuesta de mejoras en la integración

Existen varios puntos a mejorar en la integración realizada. Se podría decir que uno destaca sobre el resto y es el hecho de que tanto el fósforo como el nitrógeno del abono sufren procesos de mineralización al cabo del tiempo como se explica en diferentes estudios [20]. Para integrar esta mejora en el programa actual habría que utilizar una función de actualización para la capa de abono en la cual se podría definir la evolución de los valores del nitrógeno y fósforo "disponibles" en el abono en función del tiempo. Por otra parte, el comportamiento de los agentes se podría mejorar desde varias perspectivas.

Según detallan varios artículos [21] existe una relación entre el tiempo que pasan las vacas en las cercanías o incluso dentro del agua con la temperatura ambiente mínima del día. Se puede estimar que pueden pasar 2 horas diarias cuando la temperatura mínima es de 20 °C, 1 hora cuando la temperatura mínima es de 15 °C, ½ hora cuando la temperatura esta entre 5 °C y 15 °C. Esta mejora podría plantearse de varias maneras, una de ella sería integrar en la capa "daily" las condiciones ambientales externas para representar factores cómo la temperatura, viento, radiación solar o cubrimiento por nubes, lo cual no solo nos permitiría realizar esta mejora, sino que nos permitiría introducir estos otros fenómenos.

Otro factor a mejorar es el posicionamiento de los agentes. En la simulación realizada para este estudio se reparten inicialmente los diferentes agentes en las proximidades de 10 puntos en las cercanías del embalse, que se corresponderían con las estabulaciones, y el comportamiento de los agentes cuando llega la noche es no moverse ni depositar abono. Una posible mejora sería implementar un movimiento de vuelta al punto inicial cada día para simular la vuelta de las vacas al establo. Por último, las vacas se mueven usualmente de forma conjunta, en rebaño, y por ello podría simularse conjuntamente el movimiento de los mismos, cómo un solo agente. Igualmente el parámetro de movilidad, p_m , podría acotarse de forma más realista realizando una observación detallada del comportamiento de las vacas in-situ.

Por último, otro punto de mejora al modelo del embalse podría ser incorporar el impacto de la ganadería ovina, mediante los agentes correspondientes.

5 Conclusiones y Perspectivas

Gracias a la simulación ABM se ha conseguido dar una estimación de la incertidumbre en cuanto a la aportación de nitrógeno y fósforo por parte de la ganadería que se encuentra en las orillas del embalse.

Para describir el modelo mediante agentes, se han empleado una serie de simplificaciones con el fin de poder incorporar los diferentes elementos del problema a la simulación. Estas simplificaciones nos permiten describir el movimiento de los agentes por el terreno de una manera sencilla, y también nos permiten estimar la cantidad de abono depositada por los agentes fácilmente.

En un futuro se podría realizar un estudio del comportamiento de las vacas, para poder obtener un factor de movilidad (p_m) más realista. También se podría añadir un nuevo tipo de agente que describa a las ovejas las cuales también hacen su aportación de nitrógeno y fósforo al embalse. Estas mejoras junto con las mejoras propuestas de incorporar un proceso de mineralización al abono depositado, y harían que la simulación describiese mejor el efecto de los agentes haciendo que los resultados fuesen más cercanos al valor real y que dicha simulación permitiera mejorar los estudios relativos al embalse de Cuerda del Pozo.

Desde el punto de vista del desarrollo del proyecto, se ha utilizado el software Machanguitos, el cual ha facilitado el desarrollo debido a que su uso es bastante intuitivo. Se considera intuitivo ya que dispone de las funciones necesarias para una simulación con agentes y además estas están muy bien separadas entre sí, de tal forma que a la hora de desarrollar siempre se tiene claro que se está modificando. Por otra parte Lua es un lenguaje muy sencillo de utilizar siempre y cuando se tenga experiencia a la hora de programar. A pesar de esto, se considera que Machanguitos podría mejorar la gestión del tiempo de simulación a la hora del desarrollo, ya que no existe un valor del tiempo común a todos los elementos de la simulación (agentes y capas) sino que cada uno tiene que llevar la cuenta del tiempo haciendo en muchos casos repetir código innecesario. Esta sugerencia se va a tener en cuenta en futuras versiones del código en desarrollo.

Referencias

- [1] Ver por ejemplo "Science in the Age of Computer Simulation", E. Winsberg, U. Chicago Press Books, ISBN 9780226902043, Octubre 2010
- [2] Una introducción puede verse en https://en.wikipedia.org/wiki/Agent-based_model
- [3] *Machanguitos*, paquete desarrollado por Luis Cabellos, ver <https://ifca.github.io/machanguitos/>
- [4] Proyecto LIFE+ ROEM+, ver <http://www.roemplus-life.eu>.
- [5] Moreta Pozo J.C, Oleas M. "La Eutrofización de los lagos y sus consecuencias" Ibarra, Universidad Técnica del Norte; 20008, <http://repositorio.utn.edu.ec/bitstream/123456789/720/2/06%20NUT%20099%20TESIS.pdf>
- [6] López González L, López Fidalgo A, Monteoliva Herreras A, Monná A. "Gestión Avanzada de la Eutrofización de Aguas Superficiales a escala integral de Cuenca Hidrográfica". Congreso Nacional del Medio Ambiente; 2012; Instituto Tecnológico de Galicia; 2012. P. 4-8.
- [7] Monteoliva A. presentado en Taller ROEM+, Sesión II, Explotación de Datos, Vinuesa 15 Mayo 2015.
- [8] Macal. C, North. M. Center for Complex Adaptive Agent Systems Simulation (CAS2) Decision & Information Sciences Division [Proceedings of the 2005 Winter Simulation Conference M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.] Argonne National Laboratory; Argonne, IL 60439, U.S.A. 2005; [Consultado 12 Julio 2015]. Disponible en: <http://www.informs-sim.org/wsc05papers/002.pdf>.
- [9] OpenABM & CoMSES. Frequently Asked Questions. [Consultado 20 sep 2015]. Disponible en: <https://www.openabm.org/faq-page#t780n3730>.

-
- [10] Bonabeau E. Agent-based modeling: Methods and techniques for simulating human systems. PNAS (Cambridge). Mayo 2002 [Consultado 23 sep 2015]; Vol 99(3): 8. Disponible en: http://www.pnas.org/content/99/suppl_3/7280.full.
- [11] Macal C, North M. Tutorial on agent-based modelling and simulation. Palgrave Macmillan, Journal of simulation. 2009 Nov [Consultado 30 sep 2015]; 4: 151-162. Disponible en: <http://www.palgrave-journals.com/jos/journal/v4/n3/full/jos20103a.html>.
- [12] Serenko A, Detlor B. Agent toolkits: A general overview of the market and an assessment of instructor satisfaction with utilizing toolkits in the classroom. MGD School of Business, McMaster University. Hamilton (Ontario); 2002 Julio [Consultado 1 oct 2015]. Disponible en: [http://www.aserenko.com/papers/Agent Toolkits Working Paper.pdf](http://www.aserenko.com/papers/Agent%20Toolkits%20Working%20Paper.pdf)
- [13] Poslad S, Buckle P, Hadingham R. The FIPA-OS agent platform: Open Source for Open Standards; [Citado 5 oct 2015]. Disponible en: <http://fipa-os.sourceforge.net/docs/papers/FIPAOS.pdf>
- [14] The Foundation for Intelligent Physical Agents. [Consultado 5 oct 2015]. Disponible en: <http://www.fipa.org/>
- [15] Flexible Large-scale Agent Modelling Environment. [Consultado 5 oct 2015]. Disponible en: <http://www.flame.ac.uk/>
- [16] Netlogo. [Consultado 10 oct 2015]. Disponible en: <https://ccl.northwestern.edu/netlogo/>
- [17] The Repast Suite. [Consultado 12 oct 2015]. Disponible en: <http://repast.sourceforge.net/>
- [18] López-Paredes, A., Saurí, D. & Galán, J. M. (2005). Urban water management with artificial societies of agents: The FIRMABAR simulator. Simulation 81(3), 189-199.
- [19] Gregorio Salcedo Díaz, "Manual de mejora ambiental de las explotaciones lecheras de Cantabria" n7; Gobierno de Cantabria. 2010.
- [20] Joann K. Whalen, et al., "Nitrogen and phosphorus mineralization potentials of soils receiving repeated anual cattle manure applications", *Biology and Fertility of Soils* 01/2001; 34(5):334-341.
- [21] Shannon B. Brown, et al., "Predicting time cattle spend in streams to quantify direct deposition of manure. 2014, Agriculture and Biosystem Engineering Publications, Digital Repository Iowa State University.