

Una introducción a la simulación social como método de aprendizaje

Emilio Serrano, Universidad Politécnica de Madrid, España
Carlos Ángel Iglesias, Universidad Politécnica de Madrid, España

Resumen: La simulación social o SimSoc es un método para la exploración y entendimiento de procesos sociales mediante simulaciones en computadores. Una sociedad es un sistema complejo no lineal difícil de ser modelado y estudiado. Por ello, en los últimos años este paradigma ha ganado una gran importancia para el estudio de disciplinas tan dispares como Sociología, Biología, Física, Química, Ecología, y Economía. Este trabajo presenta la SimSoc, las ventajas que aporta en el aprendizaje interactivo, las principales metodologías de investigación en este paradigma, herramientas prácticas para construir estas simulaciones, modelos listos para ser utilizados en el aula, y casos concretos de simulaciones sociales emblemáticas como Sugarscape y el dilema del prisionero. Como se verá, incluso si se carece de conocimientos de programación, la SimSoc es una potente herramienta para comprender fenómenos complejos mediante la interacción con modelos simulados a los que se puede acceder desde un navegador web cualquiera.

Palabras clave: educación superior, tecnología en la educación, simulación social, inteligencia artificial

Abstract: The social simulation or SocSim is a method for the exploration and understanding of social processes through computer simulations. A society is a nonlinear complex system difficult to be modeled and studied. Therefore, in the last years, this paradigm has gained a great importance for the study of disciplines as diverse as sociology, biology, physics, chemistry, ecology, and economics. This paper presents the SocSim, the advantages of interactive learning, the main methodologies of research in this paradigm, practical tools to build these simulations, models ready for use in the classroom, and specific cases of emblematic social simulations as Sugarscape and the prisoner's dilemma. As explained, even without programming experience, SocSim is a powerful tool for understanding complex phenomena by interacting with simulated models which can be accessed from any web browser.

Keywords: Higher Education, Technologies in Education, Social Simulation, Artificial Intelligence

Introducción

Las ciencias sociales son ciencias que exploran aspectos de las sociedades humanas, tales como la sociología, la psicología, el derecho, la economía, la política, etcétera. Por otro lado, la simulación consiste en diseñar un modelo (una abstracción) de un sistema real y llevar a término experiencias con él (Gilbert & Troitzsch, 2005). Además, este modelo resultante debe ser más fácil de estudiar que el sistema real en alguna dimensión (como costo económico o reproducibilidad), ya que de lo contrario se experimentaría directamente sobre la realidad.

A partir de estas definiciones iniciales, podemos considerar la simulación social o SimSoc como la exploración y entendimiento de los procesos sociales mediante simulaciones en computadores (Gilbert & Troitzsch, 2005). Una sociedad es un sistema complejo no lineal, es decir, su comportamiento no puede expresarse como la suma del comportamiento de sus partes (Gilbert & Troitzsch, 2005). Y estos sistemas son muy difíciles de estudiar con modelos matemáticos basados en ecuaciones diferenciales, que es el enfoque clásico para el modelado de sociedades.

Con algunos matices, la SimSoc también es referenciada en la literatura especializada (por ejemplo, la revista “Journal of Artificial Societies and Social Simulation”) como Multi-agent based simulation (MABS), Agent-based social simulation (ABSS), y Agent-based modelling (ABM); siendo este último término el hegemónico en el área de la economía. Algunos recientes trabajos donde se hipotetiza que este paradigma podría evitar futuras crisis financieras (Farmer & Foley, 2009) son muestra del interés creciente por la SimSoc. A continuación se muestra y explica un



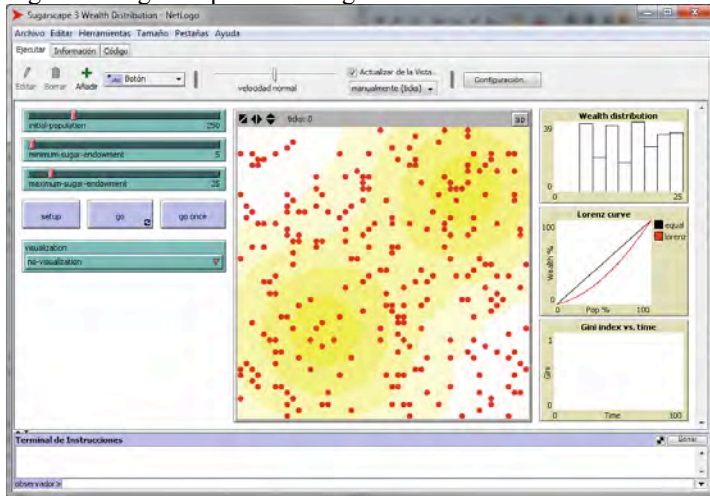
ejemplo ilustre de simulación social para entender mejor a que nos referimos con este método de investigación y aprendizaje.

Sugarscape

Sugarscape, introducida en *Growing Artificial Societies* (Epstein & Axtell, 1996), es probablemente la simulación social más conocida y referenciada (más de 4000 citas según google scholar). Utilizaremos este ilustre ejemplo para familiarizar al lector con una SimSoc concreta, de manera que pueda evaluar la validez de este paradigma para el aprendizaje de su área de conocimiento. La Figura 1 muestra una implementación online de Sugarscape en el framework de simulación social NetLogo (NetLogo, 2015).

En esta simulación, existe un entorno representado por una tabla bidimensional de 51x51 celdas (panel central en la figura). Cada celda puede contener una cantidad variable de azúcar que representa los recursos del entorno (marcado con amarillo en la figura, un color más oscuro indica más cantidad de azúcar). Finalmente, además del azúcar, cada celda puede contener a un “agente” que representa un individuo de la sociedad simulada (marcados con rojo en la figura). El agente habita en el entorno y periódicamente realiza acciones basadas en reglas que marcan cómo interactúa con el entorno y otros agentes.

Figura 1: SugarScape en NetLogo



Fuente: Epstein y Axtell, 1996.

Por ejemplo, en el caso de Figura 1, las reglas de los agentes han sido diseñadas para estudiar el reparto de riqueza. Más concretamente, los agentes se mueven a la celda adyacente a la que ocupan que tenga más azúcar para recolectarlo en cada instante de tiempo (o permanecen en su sitio si ninguna celda colindante ofrece más azúcar que la que ya ocupan). Además, cada unidad de tiempo hace a los agentes perder azúcar y en caso de que no dispongan de él, el agente muere y nace otro nuevo en una posición aleatoria. Estas sencillas reglas muestran que bajo una gran variedad de condiciones iniciales surge una distribución de la riqueza desigual: pocos agentes acumulan la mayoría del azúcar.

A partir de sencillas reglas como las explicadas, Epstein y Axtell (1996) exploraron aspectos sociales tan complejos como las migraciones estacionales, la reproducción sexual, la transmisión de enfermedades, la difusión de la cultura.

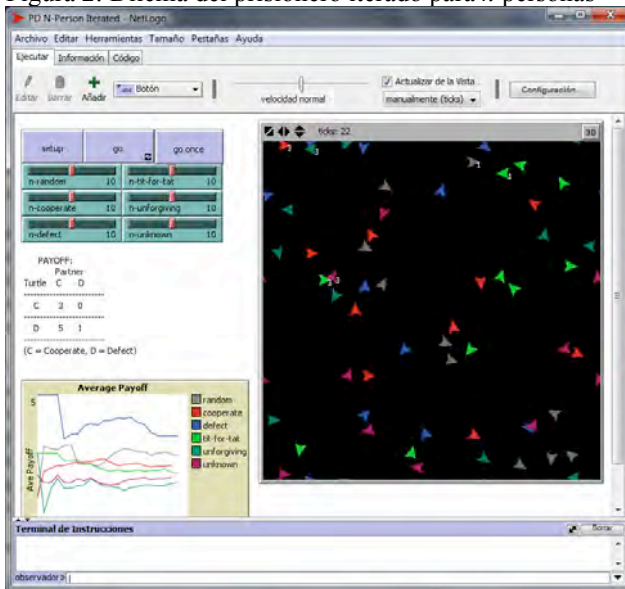
Utilidad y campos de aplicación de la simulación social

Según el primer (y más citado) libro de texto sobre la simulación social (Gilbert & Troitzsch, 2005), la simulación social tiene las siguientes utilidades principales: comprender; predecir; sustituir capacidades humanas; entrenar; educar; y entretener. En lo referente al aprendizaje, destacan las facetas de comprensión y educación, si bien hay una clara relación entre distintas utilidades. La comprensión de fenómenos complejos es vital para la educación superior y una buena educación superior apoya la comprensión de fenómenos complejos.

Si bien en la introducción del artículo se ha presentado a la SimSoc como un método para explorar sociedades humanas, lo cierto es que este paradigma de investigación (y por lo tanto de aprendizaje) se ha convertido en uno de los medios preferidos para explorar sistemas complejos adaptativos en una gran variedad de disciplinas: física, biología, química, ecología, etcétera (Gilbert & Troitzsch, 2005). Ya se ha mencionado la plataforma NetLogo (Netlogo, 2015), este “framework” ofrece una biblioteca de modelos SimSoc (Netlogo models library, 2015) en una gran variedad de materias (arte, biología, sociología, ciencias de la computación, física, etcétera). Se puede acceder a estas simulaciones de manera offline tras instalar NetLogo en una computadora (con sistemas operativos Windows, Mac, o Linux), o bien de manera online mediante un navegador configurado para ejecutar Java Applets (seleccionando la opción “run this model in your browser”).

Por poner otro ejemplo de SimSoc popular, la Figura 2 muestra un problema clásico en teoría de juegos donde se estudian las estrategias en entornos competitivos: el dilema del prisionero (Irwin *et al.*, 2015). De manera muy resumida, el dilema del prisionero nos pone en el papel de un prisionero que tiene que decidir entre confesar su crimen y el de su compañero a la policía, o engañar a la policía. El resultado de esta decisión depende de la acción que tome nuestro compañero que también puede confesar o engañar. Si ambos confesamos, la pena es considerable (digamos 3 años). Si ambos engañamos, la pena es muy reducida (digamos 1 año). Si atendiésemos exclusivamente a la minimización del daño, siempre erigiríamos engañar, ya que se obtiene la menor de las penas. Sin embargo, siempre existe la tentación de confesar a la policía cuando el compañero engaña, en cuyo caso el confesor sale libre mientras que el que ha engañado obtiene una pena de 5 años. ¿Cuál es la mejor estrategia a largo plazo?, la respuesta depende de las estrategias del resto de agentes. Gracias al modelo de la figura es posible explorar estas y otras preguntas mientras que apoya el aprendizaje de este dilema y la problemática general en el campo de la teoría de juegos.

Figura 2: Dilema del prisionero iterado para n personas



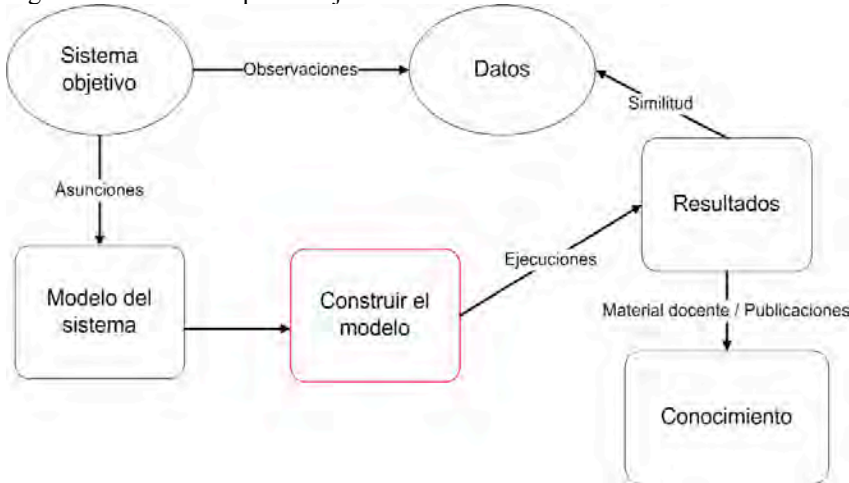
Fuente: Wilensky, 2002.

Cómo aprender con la simulación social

Ya se han explicado dos conocidos casos de SimSoc para disciplinas concretas, sociología y teoría de juegos. En esta sección se tratará el proceso general para la generación de conocimiento por medio de SimSoc.

La Figura 3 muestra el método de investigación y aprendizaje con SimSoc descrito por Gilbert and Troitzsch (Gilbert & Troitzsch, 2005) y posteriormente esquematizado por Drogoul et al. (2003).

Figura 3: Método de aprendizaje con SimSoc



Fuente: Drogoul et al., 2003.

En primer lugar existe ese sistema objetivo, esa parte de la realidad, de la que queremos aprender. En el ejemplo de Sugarscape descrito, el objetivo sería la distribución de riqueza en la economía. De nuestro conocimiento en este sistema objetivo, podemos realizar asunciones que intenten abstraer elementos de nuestro interés en un modelo. Siguiendo con el ejemplo, se identifica un entorno, recursos en el entorno (azúcar), y una visión parcial de los agentes (uno sólo sabe si hay más recursos a su alrededor pero no en todo el mundo). El sistema objetivo también permite obtener observaciones, datos que permitirán saber si la simulación se ajusta a la realidad. En el ejemplo, estos datos podrían ser indicadores de la desigualdad en un país como es el coeficiente de Gini (accesible en Google public data). Posteriormente, el modelo debe traducirse a un lenguaje entendible por los computadores, es decir, un lenguaje de programación. Las ejecuciones de la simulación, el modelo implementado, arrojan unos resultados que deben compararse con las observaciones (ya sean fruto de estudios sociológicos, datos empíricos obtenidos por medios como la sensorización de personas, u opiniones de expertos). Típicamente, el modelo no se ajustará a la realidad y hará falta refinar las asunciones, el modelo, y la implementación en un proceso iterativo. En este proceso iterativo, se construye conocimiento sobre el sistema objetivo.

Finalmente, una de las propiedades más aparecidas de la SimSoc es su capacidad para explorar escenarios “y si”, es decir, hipótesis planteadas sobre la simulación una vez se ha validado que su comportamiento es relativamente similar a las observaciones conocidas sobre el sistema objetivo. Por ejemplo, en el caso de Sugarscape, se puede plantear la hipótesis de qué ocurriría si asignásemos a toda la población una elevada riqueza inicial e igual para todos.

Sobre la necesidad de programar

La Figura 3 marca en rojo la fase de construcción del modelo. Uno de los elementos que pueden disuadir del uso de simulaciones sociales en el aula es la necesidad de dominar lenguajes de pro-

gramación que permitan implementar el modelo. Puesto que muchos educadores y estudiantes no tienen conocimientos de programación, pueden entender que la SimSoc no se adapta a sus perfiles. En esta sección se explora el papel de la programación en SimSoc.

En términos generales, saber programar beneficia profundamente el proceso de aprendizaje con SimSoc, ya que uno puede realizar cambios en la simulación de manera ágil para adaptarla a sus modelos teóricos y confrontar los resultados con las observaciones. Sin embargo, el proceso realmente valioso en el aprendizaje es la fase de modelado, donde se intenta formalizar un fenómeno de una disciplina concreta en un modelo de agentes que interactúan en un entorno con reglas simples. De esta manera, las contribuciones en SimSoc suelen darse en el campo de las ciencias sociales donde los expertos encargados de modelizar no suelen tener conocimientos de programación (la principal revista científica de SimSoc, “Journal of Artificial Societies and Social Simulation”, se encuentra indexada en el “JCR Social Science Edition”).

Además, como se ha visto, existen bibliotecas de simulaciones sociales (Netlogo models library, 2015) listas para ser usadas en el aula y que permiten al estudiante realizar multitud de experimentos que apoyan el aprendizaje. Los ejemplos de la Figura 1 y la Figura 2 muestran dos ejemplos de simulaciones donde se pueden apreciar: controles para establecer los parámetros de la simulación; controles para ejecutar, pausar o parar la simulación; gráficas que resumen los resultados de la simulación; y un visor del entorno simulado. También destacan las pestañas “información” y “código”. En información, NetLogo ofrece datos de interés como: la motivación de la simulación, experimentos a realizar, cosas que el estudiante puede notar, etcétera.

Otra opción para “esquivar” la programación en SimSoc es el planteamiento de simulaciones participativas, donde el rol de los agentes es jugado por los estudiantes que ejecutan las reglas del modelo para ofrecer resultados. Además, el educador siempre puede buscar la colaboración de educadores en el área de las ciencias de la computación que, aunque no tengan conocimientos específicos en SimSoc, podrán programar de manera sencilla modelos establecidos por expertos en otros dominios en uno de los múltiples paquetes software para SimSoc.

Finalmente, otro elemento a tener en cuenta respecto a la programación en SimSoc es que suele ser bastante sencilla y no requiere estudios avanzados en informática. Por poner un ejemplo, la pestaña de “código” de la simulación Sugarscape explicada anteriormente, véase Figura 4Figura 1, contiene únicamente 162 líneas en lenguaje NetLogo.

Figura 4: SugarScape en NetLogo, pestaña de código

```

globals [
  gini-index-reserve
  torseiz-points
]

turtles-own [
  sugar          ;; the amount of sugar this turtle has
  metabolism    ;; the amount of sugar that each turtle loses each tick
  vision        ;; the distance that this turtle can see in the horizontal and vertical directions
  vision-points ;; the points that this turtle can see in relative to it's current position (based on vision)
  age           ;; the current age of this turtle (in ticks)
  max-age      ;; the age at which this turtle will die of natural causes
]

patches-own [
  psugar        ;; the amount of sugar on this patch
  max-psugar    ;; the maximum amount of sugar that can be on this patch
]

;;
;; Setup Procedures
;;

to setup
  if maximum-sugar-endowment <= minimum-sugar-endowment [
    user-message "Oops: the maximum-sugar-endowment must be larger than the minimum-sugar-endowment"
    stop
  ]
  clear-all
  create-turtles initial-population [ turtle-setup ]
  setup-patches
  update-torseiz-and-gini
  reset-ticks
end

to turtle-setup ;; turtle procedure
  set color red
  set shape 'circle'
  move-to one-of patches with [not any? other turtles-here]
  set sugar random-in-range minimum-sugar-endowment maximum-sugar-endowment
  set metabolism random-in-range 1 4
  set max-age random-in-range 60 100

```

Fuente: Epstein y Axtell, 1996.

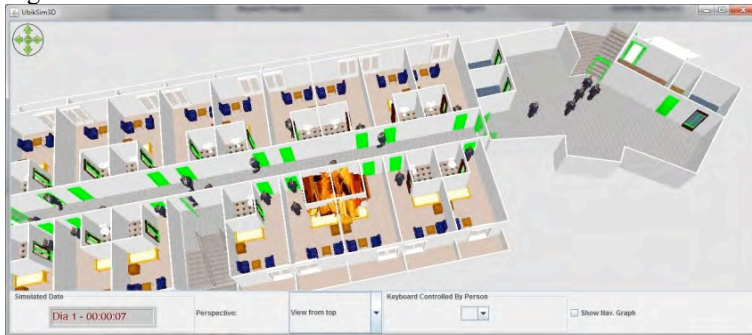
Este lenguaje de programación fue concebido para enseñar a programar a niños y para expertos de dominio sin experiencia en programación. En la actualidad, existen una gran variedad de proyectos destinados a enseñar programación para todos. La motivación es que programar enseña una serie de aptitudes transversales como es la capacidad de dividir un problema en partes más pequeñas, la capacidad de establecer un plan para llegar a un objetivo, o la concentración prolongada en el trabajo. Destacan los proyectos Scratch del Massachusetts Institute of Technology (MIT) (Scratch, 2015) y Code Studio, que a través de la conexión de bloques coloridos enseñan técnicas de programación directamente extrapolables a lenguajes generales como Python.

Experiencias en la educación mediante simulación social

En 2012 se impartió la primera asignatura en simulación social basada en agentes en España como una asignatura optativa de la titulación de ingeniero de Telecomunicación en la Universidad Politécnica de Madrid.

Los estudiantes usaron simulación social para estudiar problemas concretos relacionados con la titulación como la propagación de un virus en una red de computadores, la difusión de información en una red social, o la evacuación de un edificio ante una emergencia guiando a los ocupantes con smartphones (ver caso en Figura 5).

Figura 5: Simulación de un incendio en un hotel



Fuente: Serrano *et al.*, 2014.

La asignatura tuvo una gran concurrencia dado su carácter optativo, más de veinte alumnos; una valoración positiva generalizada del alumnado; y un alto porcentaje de aprobados (superior al 90%). Durante las prácticas, los estudiantes demostraron un aprendizaje profundo de los problemas planteados. Por poner un ejemplo que ilustra muy bien la calidad del aprendizaje obtenido, algunas de las estrategias de evacuación planteadas por estudiantes y los experimentos que apoyaban su validez fueron publicadas en un artículo en el primer cuartil del JCR edición de ciencias (Serrano *et al.*, 2014).

Finalmente mencionar que, en el contexto del proyecto Mosi-Agil, cuatro grupos de investigación de la comunidad de Madrid en España trabajan para el desarrollo de sistemas informáticos para modelado social de la inteligencia ambiental aplicada a grandes instalaciones. Uno de los objetivos del proyecto es la difusión de la simulación social como método de investigación y docencia.

Conclusión

La simulación social (SimSoc) ofrece una potente herramienta para la docencia y el aprendizaje de sistemas complejos mediante la exploración de simulaciones computacionales. En el enfoque más completo, estudiantes y docentes deben embarcarse en un proceso de investigación que pasa por la modelización de un sistema objetivo, la obtención de datos de este sistema, la modelización de una abstracción del sistema destinada a estudiarlo con sencillez, la programación del modelo, la experimentación con la simulación resultante, y la comparación de los resultados de la sociedad de agentes

simulada con los datos del sistema real. Sin embargo, también se presentan valiosos enfoques que constituyen un subconjunto del principal. Este es el caso del uso de bibliotecas de simulaciones que están listas para ser usadas en el aula, y que contienen la información necesaria para plantear experimentos con ellas. En cualquiera de los enfoques propuestos (investigación completa, uso de bibliotecas, simulaciones participativas, colaboración con expertos en ciencias de la computación, etcétera), el fin último es que el alumno construya su conocimiento en la materia apoyado por la experimentación y visualización de simulaciones.

Agradecimientos

Este trabajo de investigación es apoyado por el ministerio español de economía y competitividad bajo el proyecto de investigación y desarrollo CALISTA (TEC2012-32457); por el ministerio español de industria, energía y turismo bajo el proyecto de investigación y desarrollo BigMarket (TSI-100102-2013-80); y por la comunidad autónoma de Madrid a través del programa MOSI-AGIL-CM (S2013/ICE-3019).

REFERENCIAS

- Iglesias, C. A. (Coord.). (2015). Asignatura de simulación social basada en agentes. Disponible en: <http://www.gsi.dit.upm.es/es/noticias/1-gsi/183-curso-de-libre-eleccion-simulacion-social-basada-en-agentes.html>. Último acceso 20 de Julio de 2015.
- Code Studio. Disponible en: <https://studio.code.org/>. Último acceso 20 de Julio de 2015.
- Wikipedia. (2015). *Comparativa de software para el modelado basado en agentes*. Disponible en: <https://goo.gl/mjDjrr>. Último acceso 20 de Julio de 2015.
- Google Public Data. Disponible en: <http://www.google.com/publicdata/directory>. Último acceso 20 de Julio de 2015.
- Mosi-agil, modelado social de inteligencia ambiental aplicado a grandes instalaciones. Disponible en: <https://www.gsi.dit.upm.es/mosi/>. Último acceso 20 de Julio de 2015.
- Netlogo. Disponible en: <http://ccl.northwestern.edu/netlogo>. Último acceso 20 de Julio de 2015.
- Netlogo Models Library. Disponible en: <http://ccl.northwestern.edu/netlogo/models/>. Último acceso 20 de Julio de 2015.
- Scratch. <https://scratch.mit.edu/>. Último acceso 20 de Julio de 2015.
- Drogoul, A., Vanbergue, D. y Meurisse, T. (2003). Multi-agent based simulation: Where are the agents? In: *Proceedings of the 3rd International Conference on Multi-agent-based Simulation II, MABS'02*, (pp. 1-15). Berlín, Alemania: Springer-Verlag.
- Epstein, J. M. y Axtell, R. (1996). *Growing Artificial Societies: Social Science from the Bottom Up*. Washington, Estados Unidos: The Brookings Institution.
- Farmer, J. D. y Foley, D. (2009). The economy needs agent-based modelling. *Nature*, 460(7256), 685–686.
- Gilbert, N. y Troitzsch, K. G (2005). *Simulation for the Social Scientist*. Reino Unido: McGraw Hill.
- Irwin, K., Edwards, K. y Tamburello, J. A. (2015). Gender, trust and cooperation in environmental social dilemmas. *Social Science Research*, 50, 328-342.
- Serrano, E., Poveda, G. y Garijo, M. (2014). Towards a holistic framework for the evaluation of emergency plans in indoor environments. *Sensors*, 14(3):4513.

SOBRE LOS AUTORES

Emilio Serrano: Profesor ayudante doctor del departamento de Inteligencia Artificial en la Universidad Politécnica de Madrid. Obtuvo el título de Ingeniero en Informática por la Universidad de Murcia con una mención honorífica a la Excelencia Académica en 2006. Después de una estancia como investigador asociado en la Universidad de Edimburgo durante un año, recibió el grado de doctor con mención europea en 2011. Su tesis doctoral recibió el premio extraordinario de doctorado en 2012. El Dr. Serrano es coautor de más de 40 publicaciones en congresos y revistas internacionales. También ha participado en varios programas de financiación europea y nacional para proyectos de investigación. Sus principales intereses de investigación incluyen: aprendizaje automático avanzado; análisis de redes sociales; simulación social basada en agentes; inteligencia ambiental; y, especialmente, la interacción entre estos campos.

Carlos A. Iglesias Fernández: Doctor Ingeniero de Telecomunicación por la Universidad Politécnica de Madrid (UPM), es Profesor Titular de la ETSI de Telecomunicación de la UPM (ETSIT). Actualmente dirige el grupo de sistemas inteligentes (GSI). El Dr. Iglesias ha sido Subdirector de I+D de Grupo Gesfor y Director de Innovación de Germinus XXI. También ha trabajado como profesor en las universidades de Valladolid y Pontificia de Comillas de Madrid. Desde 2013, ocupa los cargos de: Adjunto al Director para Alumnos; y, Responsable de Calidad el Centro y Coordinador de la Oficina de Prácticas. Respecto a la producción científica, tiene más de 40 artículos de investigación publicados en congresos, libros y revistas nacionales e internacionales. También participa activamente en proyectos de investigación nacionales y europeos, destacando su trabajo como coordinador técnico de los proyectos europeos: FP6 Romulus; FP7 Omelette; y, FP7 EuroSentiment.