

Eryna: una herramienta de apoyo a la revolución de los videojuegos

Mariela Nogueira-Collazo, Carlos Cotta Porras, Antonio J. Fernández-Leiva

Universidad de Málaga, Andalucía Tech
{mnoqueira, ccottap, afdez}@1cc.uma.es

Resumen La industria del videojuego desde su inicio se ha caracterizado por un dinamismo constante, cada día son más los desarrolladores que se suman a la competencia y por otro lado, los usuarios se vuelven más expertos y exigentes. A la par de esto se ha fortalecido también la alianza entre la industria y las comunidades científicas de investigación que se sienten atraídas por este fascinante mundo. En este artículo se pretende hacer un resumen de los temas más abordados por los investigadores en los últimos años, que sin duda marcan los retos y las tendencias del futuro de los videojuegos; y basándonos en este estudio presentamos una herramienta que integrará varias de estas tendencias y facilitará el desarrollo de juegos avanzados.

Keywords: Inteligencia Artificial, Herramienta de desarrollo para videojuegos, Generación Automática de Contenidos

1. Introducción

El futuro nunca es fácil de predecir, pero cuando de ciencia se trata los pronósticos suelen ser fiables. Si echamos la vista atrás y observamos la evolución que han tenido los videojuegos, notaremos que hay dos factores que han marcado hitos en su historia: el hardware gráfico, y la Inteligencia Artificial (IA). Los primeros videojuegos carecían por completo de IA, no fue hasta el año 1970 que empezaron a aparecer los juegos de un solo jugador en los cuales el protagonista debía enfrentarse a personajes enemigos. Los primeros fueron *Qwak* y *Pursuit* ambos estaban soportados por una lógica sencilla. Los videojuegos basados en texto también comenzaron a conocerse y a pesar de emplear mecanismos muy básicos para la toma de decisiones, lograban controlar el entorno entre un humano y un jugador virtual. El avance del hardware y la llegada del microprocesador proporcionaron una mayor capacidad de cómputo a los programas y dio paso a la introducción de aleatoriedad en el comportamiento y movimiento de las entidades enemigas, dando lugar al enriquecimiento de las técnicas de movimientos basadas en patrones precalculados. Esta alianza entre hardware especializado y juegos más complejos que lo exploten se ha repetido hasta los días de hoy, ya contamos con dispositivos muy especializados y potentes, y también con juegos que retan a la inteligencia humana, sin embargo, nada parece ser suficiente, y para los desarrolladores los retos aún son grandes.

El juego, en todas sus modalidades, persigue un objetivo muy complejo, que es el de proporcionar entretenimiento y/o diversión al jugador humano, incluso los videojuegos serios [1], independientemente de sus fines educativos, instructivos, terapéuticos, etc., tienen que proporcionar su dosis de entretenimiento o de lo contrario serían rechazados por los usuarios. Y de esta sencilla razón surgen cuestiones muy interesantes: ¿Cómo medir la satisfacción del jugador? ¿Cómo desarrollar un videojuego que complazca a muchas personas que tienen gustos, culturas, habilidades, y expectativas diferentes? No son pocos los que se han implicado en responder a esto, psicólogos, matemáticos, informáticos, y muchos emprendedores fieles de diferentes culturas, subculturas y profesiones, han aportado respuestas, muchas de ellas las podemos encontrar si revisamos los resultados de la actividad científica-investigadora sobre videojuegos en los últimos años ([2], [3], [4], [5]). Las propuestas más relevantes apuestan por explotar técnicas de Inteligencia Computacional (IC) que doten a los oponentes virtuales de una IA personalizada según las necesidades y características de cada jugador, y que conviertan la partida sea una experiencia única y retadora.

El futuro es prometedor pero también supone un gran cambio en la concepción y desarrollo de los videojuegos, y con ello la transformación de las herramientas y metodologías que usamos para la creación de estos sistemas. En la siguiente sección se abordarán algunas de las tendencias fundamentales hacia las que se va encaminando el videojuego y se hará un análisis de las transformaciones que esto implicará en la industria actual. Luego daremos paso a la presentación de una herramienta que se está construyendo actualmente y tiene como objetivo principal apoyar la investigación y el desarrollo de videojuegos avanzados, y su aporte fundamental consiste en que integra por vez primera algunas de estas mencionadas tendencias.

2. Principales tendencias de la IA aplicada a videojuegos

En el auge científico-investigativo de la Inteligencia Artificial aplicada a videojuegos han destacado tres temáticas que abren un amplio espectro para el aporte de algoritmos y técnicas novedosas. Se pretende obtener soluciones que generen partidas donde haya un equilibrio entre los retos que se le imponen al jugador y las particularidades de éste; donde los comportamientos de los jugadores virtuales no caigan en los extremos de la *estupidez artificial* [6] restándole realismo y diversión a la experiencia de juego; y que además todo esto se logre minimizando el esfuerzo humano empleado durante el proceso de desarrollo del software. Estas pretensiones nos son conocidas desde hace tiempo, la diferencia es que últimamente los esfuerzos por alcanzar estos retos han estado centrados en tendencias muy concretas, las cuales mencionaremos a continuación.

2.1. Juegos auto-adaptativos

Para lograr que un juego sea auto-adaptativo es necesario que el Documento de Diseño, el guión, el mecanismo de toma de decisiones, el manejo de los niveles,

la Interfaz de Usuario, y todos los otros artefactos que intervienen en el proceso de desarrollo; se elaboren con un enfoque genérico, que los haga ajustables a la amplia gama de preferencias que impone un grupo cualquiera de usuarios. El concepto de auto-adaptabilidad se refiere al hecho de que el software (en este caso, el juego) sea capaz de adaptarse o ajustarse automáticamente a las características del usuario, con el objetivo de brindarle una partida personalizada según sus preferencias y condiciones, que le resulte entretenida y esté a la altura de sus expectativas. Bajo este concepto, un mismo juego crearía una experiencia diferente para cada persona, pues estaría soportado por algoritmos que generarían muchas (o mejor, infinitas) combinaciones de todos los elementos ajustables.

Respecto a los mecanismos de ajustes se debe ser cuidadoso a la hora de implementarlos, pues por ejemplo, pudiera provocarse un cambio brusco en las reglas y frustrar al jugador, o conducir al juego hacia estados desconocidos que no estén soportados por los algoritmos de toma de decisiones usados, lo cual es posible cuando se emplean técnicas de Generación Automática de Contenidos[7] para generar infinitas combinaciones de los elementos ajustables. Otro aspecto importante es garantizar una retroalimentación del estado del jugador durante la partida y a partir de ella producir los ajustes necesarios, aumentando así la probabilidad de que los ajustes sean adecuados para el jugador. Ahora bien, la acción de “definir el estado de un jugador” es otra tarea complicada que no sólo abarca el análisis de los logs del juego, sino que incluye otros muchos factores que influyen en el comportamiento humano, como son las emociones.

Se sabe que los mecanismos que manejan los estados emocionales humanos son muy complejos pues muchas son las variables que determinan un cambio de ánimo, y buscar una generalidad no es simple debido a que cada persona tiene sus características y preferencias particulares. Actualmente se realiza el *IEEE Task Force on Player Satisfaction Modeling* [8] que tiene como objetivo incentivar la creación de propuestas para modelar y optimizar la medida de satisfacción que se puede percibir de un usuario mientras éste está jugando. Se han aportado resultados interesantes, de acuerdo con las taxonomías que muestran la evolución del tema [9] algunos trabajos se centran en definir un valor escalar que cuantifique el entretenimiento y otras en la elaboración de modelos que establezcan la relación de correspondencia entre las variables influyentes y el valor del entretenimiento ([10], [11], [12]). Como se podrá deducir, los procesos de modelado y evaluación de la satisfacción están estrechamente relacionados con el mecanismo de ajuste que se implemente; este último sería la etapa final en un intento por optimizar la satisfacción del jugador; sólo después de haber obtenido los modelos que identifiquen al jugador, y haber definido una medida para calcular su grado de entretenimiento, se podría usar esa información para adaptar o ajustar el juego a las características del usuario. Luego hay otros matices que siguen ampliando esta línea de investigación, por ejemplo: ¿Qué es más eficaz para obtener la retroalimentación del jugador: los modelos psicológi-

cos o los dispositivos biométricos? ¿Cuáles elementos de la partida son los más convenientes para tratarlos como “ajustables”? Estas y muchas otras son preguntas que forman parte de esta tendencia y aún siguen abiertas a la espera de soluciones.

2.2. Videojuegos afectivos

Esta otra tendencia guarda mucha relación con la anterior pues ambas abordan el estudio del comportamiento humano centrado en las emociones, pero en esta segunda el objetivo principal es diferente.

El auge de la IA aplicada a videojuegos, y los numerosos intentos por simular el comportamiento humano, han abierto un prometedor camino donde se tratan las emociones como parte esencial de nuevas técnicas y algoritmos para la toma de decisiones y la simulación realista del entorno, explotando un nuevo campo de investigación llamado computación afectiva (CA). Fue Rosalind Picard quien, en 1995, introdujo el término CA que ella misma define como “el cómputo que relaciona, surge o influye en las emociones” [13]. En lo que a videojuegos respecta, una de las primeras formas usadas para incorporar emociones fue a través de la narrativa de la historia, siendo esta última una aliada por excelencia para generar situaciones que atrapen al jugador ya sea por los personajes, la presentación de todo tipo de conflictos, historias fantásticas o de la vida real, etc. Pero ya en nuestros días no nos conformamos con esto, y queremos que durante una partida no sólo sea el jugador humano el que reaccione emocionalmente (como un acto intrínseco de su naturaleza humana) sino que también los personajes virtuales (*Bots*) respondan ante las señales afectivas del humano y de los otros *Bots* con los que interaccionan.

Llegado a este punto de la explicación, toca volver a referirnos a la autoadaptabilidad pues para que realmente podamos considerar a un videojuego como afectivo, es necesario leer las distintas señales que el jugador lanza de forma imprevisible y a partir de ellas que el sistema decida qué “respuesta” afectiva debe ejecutar. Como vemos, se trata de un proceso de ajuste similar al que vimos anteriormente, la diferencia es que aquí el objetivo es crear un entorno emocional y manejar a los jugadores humanos y a los virtuales para que sientan y exterioricen determinadas emociones.

Algunas prácticas son tan simples como la de detectar un estado de tranquilidad extrema en el jugador humano (por ejemplo, mediante una lectura de su ritmo cardíaco o analizando el tiempo de inactividad que lleva en el juego), y proceder a disminuir la luz del entorno para crearle una atmósfera de ansiedad e intentar activarlo un poco. Otra variante emplea recursos del diseño gráfico para generar expresiones faciales que simulan emociones humanas y transmiten estímulos al usuario, en este caso se puede mencionar el videojuego *Fable* de la empresa Lionhead Studios que se caracteriza por poner especial interés en la IA de sus juegos, y que desde la versión 2 una de sus principales metas es la

generación de emociones realistas en los *Bots* que puedan ser percibidas por el usuario y generen una reacción en él.

2.3. Generación Automática de contenidos

Este concepto se refiere a la creación de contenidos para videojuegos de forma automática, mediante algoritmos de IA, y representa una de las líneas más revolucionarias en esta área. Es además una prueba irrefutable de que la aplicación de la IA en videojuegos va más allá del control tradicional de un *Bot* y trasciende a otros muchos aspectos del proceso de desarrollo y el diseño gráfico [14]. El objetivo principal de esta tendencia es darle el papel protagónico a las técnicas de IA para que se encarguen de crear y administrar los contenidos del juego, reconociendo a estos como los elementos fundamentales que hacen que un jugador se “mantenga enganchando” [7]. Se entiende como contenidos: la reglas del juego, la historia, los personajes, los entornos, la música, la interfaz de usuario, los niveles de dificultad, y todos los otros elementos estéticos, narrativos y lógicos que influyan en la experiencia de juego.

Desde un punto de vista económico, la aplicación de la Generación Automática de Contenidos (PCG, por la siglas en Inglés de *Procedural Content Generation*) supondría un ahorro considerable de presupuesto y de tiempo en el desarrollo de un juego. Pues los algoritmos especializados podrían generar los contenidos que hoy en día son creados por trabajadores (diseñadores, artistas gráficos, guionistas, músicos, etc.) a una velocidad mucho mayor y sin incurrir en gastos de presupuesto. Esto supone un cambio rotundo en la conceptualización del videojuego y abre las puertas hacia cuestiones como: ¿Quién evalúa la calidad de los contenidos generados? ¿Si un videojuego es infinito se aburrirá el jugador de él? ¿Es factible generar un juego completamente mediante PCG o sólo vale la pena hacerlo con algunos de los contenidos?

En este punto cabe preguntarse si estamos ya preparados para estos videojuegos futuristas, o al menos convencidos de que explotar estas tendencias nos permitirá obtener juegos mejores que los que tenemos hoy en día. Por el momento, no podemos responder con seguridad, tendremos que conformarnos con experimentar y darle una oportunidad al cambio.

En sintonía con estos tiempos de cambio hemos diseñado Eryna, que pretende ser una herramienta que facilite la investigación y el desarrollo de estas nuevas tendencias. Con ella se pondrán a disposición de los investigadores varios módulos que soportarán la aplicación y prueba de modelos computacionales (que serán instancias de las tendencias anteriores) cuyo objetivo será generar soluciones subjetivamente interesantes para un juego RTS que es parte de Eryna.

3. Descripción de la herramienta Eryna

No son pocos los juegos que han servido para apoyar el trabajo de los investigadores, la mayoría han sido empleados para probar mecanismos de IA que

optimizan el comportamiento de los jugadores virtuales. Por ejemplo tenemos a Orts, que es un juego de estrategia de tiempo real (RTS, por sus siglas en Inglés) que proporciona un programa cliente para la aplicación de técnicas de aprendizaje automatizado (Machine Learning Techniques) a su motor de juego [15]. También es muy típico el uso de videojuegos que son creados para un marco específico de competición y que fomentan la creación de técnicas de IA aplicada, como es el caso del “Google AI Challenge Contest”¹ cuyos juegos han sido objeto de muchas investigaciones ([16], [17], [18], [19]). Otro con el que hemos trabajado nosotros anteriormente es el *RobotWars*², un juego de tipo RTS que fue desarrollado en la Universidad de Málaga para probar modelos de IA basados en Coevolución, que se encargaban de generar y optimizar estrategias de juego victoriosas. Hay casos muy populares como es el Super Mario Bros que se emplea también como plataforma de competición para evaluar soluciones que persiguen optimizar el comportamiento inteligente de los *Bots* de este juego [20]; el StarCraft que ha sido usado para probar soluciones de IA y también mapas generados por algoritmos de PCG [21]; el TEMPO que es juego RTS bélico usado en disímiles investigaciones sobre la aplicación de modelos computacionales que generan mecanismos de IA [22].

Eryna es otra herramienta de apoyo a la investigación, similar a las que existen actualmente, con la distinción de que se han integrado en ella módulos que permitan la exploración de algoritmos de IA que estén en sintonía con las nuevas tendencias antes mencionadas; además se intenta suministrar facilidades a los investigadores para que no tengan que lidiar con problemas comunes como suelen ser la ausencia de un mecanismo que suministre los *logs* fácilmente para el posterior análisis de los resultados. Esta herramienta está actualmente en desarrollo, en la fase de implementación, lo que vamos a presentar en este artículo es su arquitectura mediante una vista lógica de sus módulos principales.

3.1. Descripción lógica del sistema

A continuación se presenta la Vista Lógica de la Arquitectura del Sistema, en ella se muestran los componentes principales del sistema y su interacción, los diagramas han sido distinguidos entre sí por las funcionalidades que le brindan al usuario, el primero de ellos (Figura 1) se refiere al módulo de Eryna como *Plataforma Interactiva de prueba y optimización*, y el segundo (Figura 2) muestra la arquitectura de Eryna como *Entorno de Competición de IA online*. La Figura 1 ofrece la primera vista del sistema, entre sus componentes se puede ver una versión no gráfica del juego (en la figura se nombra “Game_Engine_Local”) que se usará para lanzar partidas mono o multijugador entre los *Bots* que han sido implementados por un usuario (investigador) usando para ello el componente (“IA_Engine”). El “Game_Engine_Local” es la base de este módulo, funciona como un juego RTS sobre el cual se evaluará el desempeño de todos los recursos

¹ <http://aichallenge.org/>

² <http://wp.me/p2cObl-60>

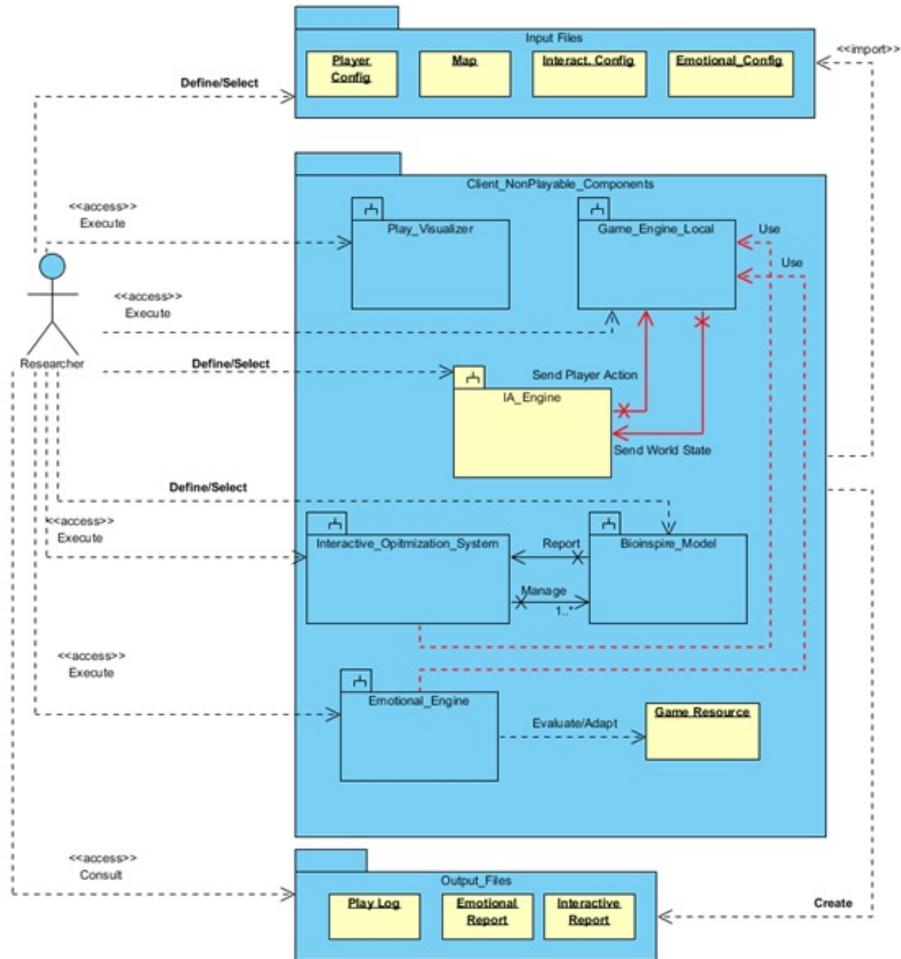


Figura 1. Vista lógica de la Plataforma interactiva de prueba y optimización

y modelos que se usen en la herramienta, todos los módulos que necesiten conocer cómo funciona un modelo determinado usarán este *game-engine* para eso. También se integrará un subsistema de Optimización Interactiva cuyo objetivo será el de optimizar el funcionamiento de modelos bioinspirados que sean suministrados por el usuario y que tendrán como objetivo la generación de soluciones subjetivamente interesantes para el juego RTS. Los modelos afectivos también tendrán su espacio, pues habrá un módulo que se encargará de evaluar (basado en alguno de los modelos definidos para esto en la literatura) el comportamiento emocional que genera en el juego RTS determinado recurso proporcionado por el usuario. Por último, es importante mencionar que siempre que el investigador emplee la herramienta como plataforma de prueba y evaluación le serán emitidos ficheros de logs sobre el funcionamiento de su recurso que le permitirán hacer un análisis serio de los resultados a posteriori. El objetivo general de este módulo es permitir a los investigadores que prueben u optimicen recursos del juego que hayan sido creados por algoritmos especializados para eso.

En la Figura 2 se muestra la segunda vista, que consta de una aplicación web donde estará accesible toda la información y los archivos necesarios para el uso de la herramienta. La web también dará acceso directo al juego RTS, los usuarios podrán subir *Bots* al servidor y lanzar una partida donde estos jugadores virtuales compitan entre ellos, y/o contra jugadores humanos; esta versión del juego sí tendrá una interfaz gráfica. También estará disponible un ranking general que contenga la posición de los *Bots* que son evaluados usando este servicio. El videojuego RTS que se usará en la herramienta, se está construyendo específicamente para este fin, se trata de un juego multijugador, donde cada jugador virtual o humano compite con los demás para ser el único ganador. La mecánica del juego es la de un clásico juego de entorno espacial en el que los usuarios conquistan planetas; pero en este caso la lógica se complica pues habrá que manejar un conjunto de recursos (como sucede típicamente en los juegos de estrategia). El motor lógico del juego desencadena automáticamente un grupo de “reglas evolutivas” que se encargan de controlar la producción/variación de los recursos naturales en el entorno sin que medie la intervención del jugador. Las posibles acciones que puede ejecutar un jugador durante la partida son: enviar unidades entre mundos; y asignar misiones en un mundo determinado para apoyar determinado objetivo que desee el jugador (por ejemplo: construir armas, producir alimentos). Se pretende hacer un juego flexible, que esté parametrizado y le permita al investigador ajustar diferentes opciones para crear muchos niveles de dificultad en las partidas; los mapas serán ficheros con una nomenclatura simple de generar; incluso las reglas del juego están siendo definidas en función de parámetros que también podrán ser alterados o ajustados por el usuario si este lo cree necesario.

En estos momentos el juego y la herramienta están siendo implementados, siguiendo la estructura general aquí presentada, en pocos meses se lanzará la herramienta y estará al alcance de los investigadores que deseen emplearla. En la siguiente sección concluimos este artículo haciendo un resumen general de

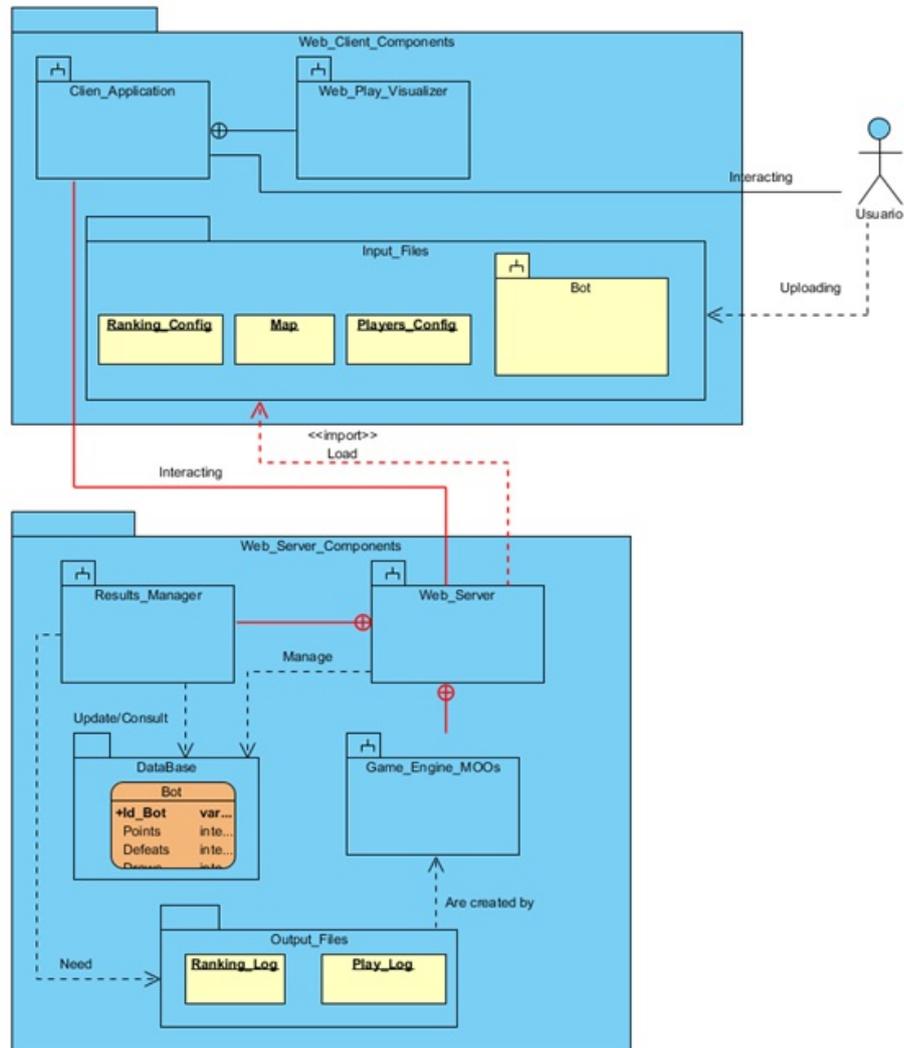


Figura 2. Vista lógica del entorno de competición online

las ideas en él expuestas. Para ver más información sobre el prototipo del juego y otros detalles técnicos pueden remitirse a su sitio web ³

4. Discusiones finales

Como se ha dicho anteriormente, estamos inmersos en el desarrollo de esta herramienta, el objetivo principal para el que fue concebida es el de apoyar la creación de algoritmos que aporten soluciones de IA para videojuegos. En ella hemos querido hacerle un espacio a las tendencias mencionadas al inicio, con la intención de que el juego sea una plataforma de prueba para diferentes técnicas que persiguen un objetivo en común: el de aumentar la satisfacción del jugador en las partidas. El aporte principal de Eryna es que integra varias técnicas sobre un mismo entorno de prueba, que es además intrínsecamente complejo (por ser un juego RTS). Aún queda mucho trabajo por hacer, incluso después de haber terminado la herramienta, habrá que seguir trabajando en su ampliación y flexibilidad para asimilar nuevas líneas.

La creación de videojuegos avanzados no sería posible sin la utilización de herramientas que permitan al desarrollador experimentar y probar sus propuestas, así que el quehacer investigativo de las comunidades afines a este tema debe mantenerse activo, para no perder posición en la frontera del conocimiento. Las herramientas ya están evolucionando, luego le corresponderá a la industria actualizarse, los procesos de creación y hasta los roles dentro del equipo de desarrollo de los juegos se van a revolucionar, la alianza entre investigación y desarrollo vuelve a imponer tendencias y parece que esta vez llegarán para quedarse por bastante tiempo.

5. Agradecimientos

Este trabajo está financiado parcialmente por los proyectos ANYSELF (TIN2011-28627-C04-01)⁴ del MICINN y P10-TIC-6083 (DNEMESIS)⁵ de la Junta de Andalucía, y por la Universidad de Málaga. Campus de Excelencia Internacional Andalucía Tech.

Referencias

1. Tarja Susi, Mikael Johannesson, and Per Backlund. Serious games : An overview. Technical Report HS- IKI -TR-07-001, University of Skövde, School of Humanities and Informatics, 2007.
2. Raúl Lara-Cabrera, Carlos Cotta, and Antonio J. Fernández-Leiva. A review of computational intelligence in rts games. In *FOCI*, pages 114–121. IEEE, 2013.

³ <http://eryna.lcc.uma.es>

⁴ <http://anyself.wordpress.com/>

⁵ <http://dnemesis.lcc.uma.es/wordpress/>

3. Julian Togelius, Georgios N. Yannakakis, Kenneth O. Stanley, and Cameron Browne. Search-based procedural content generation: A taxonomy and survey. *IEEE Trans. Comput. Intellig. and AI in Games*, 3(3):172–186, 2011.
4. Santiago Ontañón, Gabriel Synnaeve, Alberto Uriarte, Florian Richoux, David Churchill, and Mike Preuss. A survey of real-time strategy game ai research and competition in starcraft. *IEEE Trans. Comput. Intellig. and AI in Games*, 5(4):293–311, 2013.
5. Georgios N. Yannakakis. How to model and augment player satisfaction: a review. In *The 1st Workshop on Child, Computer and Interaction, WOCCI 2008*, page 21. ISCA, 2008.
6. Lars Lidén. Artificial stupidity: The art of intentional mistakes. *AI Game Programming Wisdom*, 2:41–48, 2003.
7. Mark Hendriks, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup. Procedural content generation for games: A survey. *TOMCCAP*, 9(1):1, 2013.
8. Official wiki of IEEE Task Force on Player Satisfaction Modeling. <http://gameai.itu.dk/psm>. 2011.
9. G. Yannakakis and J. Hallam. Capturing player enjoyment in computer games. *Advanced Intelligent Paradigms in Computer Games*, pages 175–201, 2007.
10. N. Baba and H. Handa. Utilization of evolutionary algorithms to increase excitement of the commons game. pages 1–10, 2006.
11. H.P. Martínez, K. Hullett, and G.N. Yannakakis. Extending neuro-evolutionary preference learning through player modeling. In *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*, pages 313–320. IEEE, 2010.
12. S. Tognetti, M. Garbarino, A. Bonarini, and M. Matteucci. Modeling enjoyment preference from physiological responses in a car racing game. In *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*, pages 321–328. IEEE, 2010.
13. Rosalind W Picard. *Affective computing*. MIT Media Laboratory, 1995.
14. Simon M. Lucas. Computational Intelligence and AI in Games: A New IEEE Transactions. *IEEE Trans. Comput. Intellig. and AI in Games*, 1(1):1–3, 2009.
15. Michael Buro and Timothy Furtak. Rts games and real-time ai research. In *Proceedings of the Behavior Representation in Modeling and Simulation Conference (BRIMS)*, volume 6370, 2004.
16. Antonio J. Fernández-Leiva Mariela Nogueira Collazo, Carlos Cotta. Virtual player design using self-learning via competitive coevolutionary algorithms. *Natural Computing*, pages 1–16, 2014.
17. Antonio Fernández-Ares, Antonio Miguel Mora, Juan Julián Merelo Guervós, Pablo García-Sánchez, and Carlos Fernandes. Optimizing player behavior in a real-time strategy game using evolutionary algorithms. In *Proceedings of IEEE Congress on Evolutionary Computation, CEC2011*, pages 2017–2024, 2011.
18. Antonio Miguel Mora, Antonio Fernández-Ares, Juan J. Merelo Guervós, Pablo García-Sánchez, and Carlos M. Fernandes. Effect of Noisy Fitness in Real-Time Strategy Games Player Behaviour Optimisation Using Evolutionary Algorithms. *J. Comput. Sci. Technol.*, 27(5):1007–1023, 2012.
19. Raúl Lara-Cabrera, Carlos Cotta, and Antonio J. Fernández-Leiva. A procedural balanced map generator with self-adaptive complexity for the real-time strategy game planet wars. In *Applications of Evolutionary Computation - 16th European Conference, EvoApplications 2013, Proceedings*, pages 274–283, 2013.
20. Julian Togelius, Noor Shaker, Sergey Karakovskiy, and Georgios N. Yannakakis. The mario ai championship 2009-2012. *AI Magazine*, 34(3):89–92, 2013.

21. Julian Togelius, Mike Preuss, Nicola Beume, Simon Wessing, Johan Hagelbäck, and Georgios N. Yannakakis. Multiobjective exploration of the starcraft map space. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games, CIG 2010*, pages 265–272. IEEE, 2010.
22. Phillipa Avery and Sushil J. Louis. Coevolving team tactics for a real-time strategy game. In *IEEE Congress on Evolutionary Computation*, pages 1–8, 2010.