Framework para desarrollos de robótica educativa en espacios virtuales

Gonzalo Zabala¹

¹ Centro de Altos Estudios en Tecnología Informática, Facultad de Tecnología Informática, Universidad Abierta Interamericana, CABA, Argentina.

gonzalo.zabala@uai.edu.ar

Abstract. *The presented thesis proposes to build a framework for the development of educational robotics virtual environments based on a 3D simulator that provides features, motivations, and resources similar to those found in the use of physical material, analyzing the impact of its use in comparison with traditional robotics kits.*

Resumen. *La tesis presentada propone construir un framework para el desarrollo de entornos virtuales de robótica educativa basados en un simulador 3D, que aporte características, motivaciones y recursos similares a los encontrados en el uso de material concreto, analizando el impacto de su uso en comparación con kits de robótica tradicionales.*

Keywords: pensamiento computacional, robótica, simulación, laboratorio virtual, programación visual

1. Introducción al campo de la investigación e identificación de los principales desafíos

El uso de robots en las aulas provee a los estudiantes un espacio de aprendizaje reflexivo y experimental. Cuando en equipo diseñan, construyen, programan y documentan el diseño de un robot, no sólo aprenden de tecnología, sino que también ponen en práctica habilidades fundamentales para el mundo que encontrarán cuando salgan de su educación formal. Por otra parte, el mercado laboral requiere cada día más de recursos humanos vinculados al desarrollo de tecnología, en campos como la programación, el diseño y las ingenierías. El estudio de la robótica puede ser un recurso para despertar vocaciones en estas áreas, además de aumentar la retención de estudiantes mujeres que están subrepresentadas en el campo de la tecnología [Miller & Nourbakhsh, 2016].

Por estos motivos, los robots se han convertido en un recurso didáctico valorado por los docentes de nivel primario y secundario. Sin embargo, aún es un recurso costoso para la mayoría de las escuelas y familias, y los docentes presentan algunos temores y rechazos para su incorporación en el aula [Reich-Stiebert & Eyssel, 2016]. Por otro lado, en contextos donde la educación necesariamente debe realizarse en forma remota, como el acaecido por la pandemia del COVID 19, el acceso a laboratorios con material físico se hace imposible.

Asimismo, a partir del aumento de poder de cómputo de las placas gráficas, han surgido con fuerza diferentes ambientes de simulación de robots: CoderZ, Webots, VRT, Gazebo, CoppeliaSim, entre otros. Lamentablemente, los que tienen por detrás un motor físico que representa lo más cabalmente posible al mundo real tienen alguno o algunos de los siguientes inconvenientes:

- Son muy costosos para nuestro país (valor dólar, alrededor de 50U\$\$ por estudiante).
- Necesitan de un equipamiento medianamente sofisticado por el alto nivel de procesamiento para simular la física.
- Son complejos de instalar y operar.

En síntesis, hasta el momento, no existe aún una solución al problema de la robótica educativa sin material concreto, que presente las siguientes características [Hughes, Shimizu, et al., 2019]:

- Acceso gratuito al software de modo que sea totalmente accesible para que lo utilicen los estudiantes.
- Que no posea altos requerimientos de cómputo.
- Facilidad de instalación para que el proceso no sea una barrera de entrada.
- Interfaz de uso intuitivo.
- Versatilidad, de modo que pueda utilizarse tanto para la enseñanza de los fundamentos de la robótica como para la experimentación de temas de mayor complejidad.

Un framework que permita el desarrollo de plataformas de simulación que cumplan con todas estas características, no sólo sería una herramienta fundamental en situaciones donde es necesaria la educación a distancia, sino que se convertiría en un recurso didáctico de gran utilidad por las siguientes razones:

1. Costo bajo o nulo.
2. Permite el trabajo con estudiantes que se encuentran alejados del acceso a los recursos materiales.
3. Permite el trabajo colaborativo entre estudiantes que se encuentran distanciados entre sí.
4. Sirve de entrada al mundo de la robótica educativa de una manera más sencilla que los kits materiales de bajo costo.

En función de lo presentado en el punto anterior, el problema que quiere resolver este trabajo de investigación es la ausencia de un simulador de robots apto para el ámbito educativo, que pueda suplantar el uso de kits físicos por cuestiones de imposibilidad de acceso o por costo.

2. Solución propuesta

Respondiendo a los problemas no resueltos que se describieron en el punto anterior, la solución que se propone es la construcción de un framework para desarrollos online de robótica educativa, basada en un simulador 3D, que permita vivenciar una experiencia lo más similar posible a la realizada con material concreto. Este framework ofrecerá un espacio para la presentación de diferentes propuestas didácticas con diversas situaciones problema en distintos escenarios. Para resolver las situaciones, los estudiantes elegirán la arquitectura de robot adecuada y programarán su comportamiento con un lenguaje de programación visual y/o textual.

Con relación a las características del proyecto, en primer lugar, el uso de un simulador basado en un motor físico 3D se acerca a la experiencia de trabajar con kits físicos. Por otro lado, el costo material es muy bajo comparado con estos recursos. Además, la ejecución en un browser disminuye el requerimiento de equipamiento, lo que

baja aún más la necesidad de recursos. Por último, el desarrollo de una interfaz que simplifica el uso del motor 3D, el acompañamiento en la misma plataforma de propuestas didácticas y la incorporación de un lenguaje visual para la programación facilita la enseñanza de robótica en los niveles primario y secundario.

En síntesis, a continuación, se plantea el objetivo general y los específicos del presente proyecto.

Objetivo general

Desarrollar un framework para la creación de plataformas de simulación de robots en ambientes 3D con motor de física, orientado al uso en ámbitos educativos de nivel primario y secundario que permita un aprendizaje similar al de los materiales físicos.

Objetivos específicos

- Establecer los principales aspectos didácticos que debe permitir la implementación del framework.
- Determinar el modelo físico para el desarrollo del simulador.
- Desarrollar espacios virtuales y robots que operan en dichos espacios.
- Especificar un ambiente de programación con diferentes niveles de complejidad asociados a su poder expresivo.
- Determinar el impacto en estudiantes de una implementación del framework.

3. Trabajos relacionados

La robótica ha comenzado a tener cada vez más presencia en las aulas, y existen diversos estudios que analizan su impacto en el aprendizaje de los estudiantes en diversas áreas [Benitti & Spolaôr, 2017; Fagin & Merkle, 2002; García-Peñalvo et al., 2019; Lepuschitz et al., 2018; Merdan et al., 2016; Miller & Nourbakhsh, 2016; Theofanellis et al., 2020]. Sin embargo, el mundo de la simulación presenta otras características que obliga a realizar estudios propios, no sólo con respecto a la mejora en el aprendizaje, sino también en comparación con los laboratorios físicos [Eguchi & Shen, 2012; Fernandes, 2013; Guyot et al., 2011; Hughes, 2016; Infante Jiménez, 2014; Major, 2014].

Para comenzar, uno de los pioneros en esta área fue USARSim, un simulador de búsqueda y rescate en zonas urbanas que se utilizó para la competencia en la categoría Virtual Robots Competition de la Robocup a partir de 2006. Otra plataforma muy popular como base para diferentes proyectos de robótica en educación es Webots, un desarrollo open source de la firma Cyberbotics destinado a la construcción de simulaciones de robots móviles para la investigación. Sobre esta plataforma podemos encontrar diversas propuestas para la educación. Una de ellas es la presentada en [Guyot et al., 2011], donde se introduce un diseño curricular para diferentes niveles de complejidad usando el robot E-puck. En la misma línea surge el proyecto RPN, Robotic Programming Network, con diferentes cursos de robótica usando simuladores construidos sobre Webots [Casañ Núñez & Cervera, 2018; Núñez & Mateu, 2015]. En [Hughes, Aghababa, et al., 2019] se propone la construcción de una plataforma de rescate para Robocup Junior, no sólo sobre Webots, sino que la extienden a Gazebo y V-REP para poder comparar sus características y performance.

Por otra parte, otra plataforma de simulación de física y robótica desarrollada para educación es CoSpaces Edu. Está presente en Robocup Junior en Cospace Rescue

Challenge [Eguchi & Shen, 2012], y existen experiencias de vínculo de la plataforma con espacios curriculares [Hughes, 2016].

También cabe destacar la propuesta de S-Educ, un trabajo similar a la presente tesis, dado que desarrolla su propio espacio de simulación, acompañado de un conjunto de actividades para el aula, y un estudio de impacto en escuelas de Brasil [Fernandes, 2013].

Asimismo, [Hughes, Shimizu, et al., 2019] realiza una revisión de diferentes simuladores de rescate para educación, considerando requerimientos similares a los definidos en este proyecto de investigación.

Con relación al traslado de servidores de los simuladores a la nube, en [Cervera et al., 2017] se presenta un desarrollo en este sentido, que finalmente se tradujo en la creación del sitio The Construct [Learn Robotics from Zero; Robotics & ROS Online Courses, 2020]. Allí se ofrecen diversos cursos de robótica sobre simuladores con estas características.

Con relación a la metodología utilizada para la realización de esta investigación, podemos dividir el análisis en dos. Por un lado, la que se seleccionó para llevar a cabo la plataforma, y por el otro la usada para determinar la efectividad del uso del recurso didáctico en relación con los laboratorios físicos.

Con relación al estudio del impacto del uso de simuladores y laboratorios virtuales en las aulas, [Major, 2014] analiza su efectividad para el aprendizaje en programación. En [Hughes, 2016] se presenta una experiencia realizada con CoSpace en 500 escuelas durante 3 años. Basados en la misma herramienta, en [Eguchi & Shen, 2012] se realiza un estudio entre 78 participantes de la Robocup 2011. En [Casañ Núñez & Cervera, 2018] se analiza el impacto de una herramienta creada en el marco de la iniciativa “The Robot Programming Network”, aunque no es un ambiente físico en 3D como el del presente trabajo. Junto con el desarrollo del simulador, en [Fernandes, 2013] se realiza un experimento con 60 usuarios entre profesores, estudiantes y adultos sin conocimientos en robótica. Finalmente, en [Guyot et al., 2011] se desarrolla una propuesta curricular sobre Webots, que es evaluada en 64 estudiantes de escuela media y en la universidad donde se llevó a cabo la investigación.

4. Grado de innovación de la propuesta

La propuesta más innovadora de este trabajo es la de proporcionar un framework que permite combinar en una única plataforma lo que hasta ahora sólo se encontraba de manera parcial en las propuestas anteriores del mismo tipo. Estas características son:

- a) Integración de actividades de aula para nivel primario y secundario en la misma plataforma: una interfaz que combine las secuencias didácticas junto con el simulador facilita la tarea del docente y el estudiante. Esta integración la podemos encontrar, implementada de otra manera a la de esta tesis, en S-Educ [Fernandes, 2013] y en la herramienta Robotbenchmark [robotbenchmark, 2020].
- b) Simulador 3D con física y de uso sencillo: simuladores como Webots, Gazebo y CoppeliaSim (V-REP) tienen una muy buena representación 3D con física. Sin embargo, no poseen una capa que aisle la complejidad inherente a la riqueza de estos simuladores.

- c) Libre, gratuito y de código abierto: la gratuidad permite que el alcance de la herramienta llegue a los que habitualmente no podrían adquirir kits de robótica física. Además, el código abierto abre la oportunidad de trabajos futuros a la comunidad académica.
- d) Diferentes tipos de robots y con posibilidad de modificar sensores y actuadores: para acercar la plataforma a los kits del mundo físico, debería tener la posibilidad de ofrecer componentes mecánicos y electrónicos para la construcción del robot, o por lo menos facilitar la importación de construcciones 3D. Esto aumenta la complejidad de la herramienta, esencialmente en los aspectos de detección de colisiones. En el presente proyecto se renunció a esta característica, pero se ofrece un set de robots y la activación o no de diferentes sensores y actuadores. De esta manera, el estudiante debe analizar los requerimientos de los problemas para definir la arquitectura del robot necesaria para su solución.
- e) Ejecución en la nube: trasladar el problema de requerimientos a un servidor, ofreciendo acceso al simulador mediante un browser, permite que estudiantes y docentes de todo el mundo puedan acceder al recurso sin necesidad de equipamientos de alto costo. Sobre este mismo principio se construyó Robot Programming Network [Núñez & Mateu, 2015], Constructsim [Cervera et al., 2017] y RobotBenchmark [robotbenchmark, 2020].
- f) Concurrencia de robots sobre un mismo escenario: ciertos desafíos de competencia o colaborativos exigen la presencia de más de un robot. Por otro lado, esta característica presenta desafíos de análisis y programación distintos a los ambientes estáticos, dado que el contexto en el que el robot opera cambia permanentemente y de forma impredecible. Los simuladores de base mencionados en el punto b) tienen esta posibilidad. S-Educ [Fernandes, 2013] y USARSim [Carpin et al., 2007] también permiten escenarios con múltiples robots.
- g) Programación visual y textual: Las herramientas visuales de programación son de gran utilidad para la introducción de conceptos de pensamiento computacional en los estudiantes de primaria [Sáez-López et al., 2016]. Además, son más motivadoras que las textuales [Tsukamoto et al., 2016]. Al mismo tiempo, los estudiantes perciben que la programación visual es menos poderosa y auténtica que la basada en texto [Weintrop & Wilensky, 2015], además de que la industria utiliza esta última forma de programación. Por este motivo, tener un lenguaje de programación visual que genere el código en texto, al mismo tiempo que pueda programarse completamente en modo textual, permite la combinación de las mejores características de los dos paradigmas.

A continuación, presentamos un cuadro comparativo con las propuestas similares ya existentes, tomando como categorías de comparación las que acabamos de mencionar.

Tabla 1. Comparación entre plataformas de robótica simulada

	Framework propuesto	Virtual Robotics Toolkit	CoderZ	Webots	Gazebo	S-Educ	Robobenchmark	CoppeliaSim	Eyesim VR
Gratuito	x			x	x	x	x	x	x
Código abierto	x			x	x	?	x	x	
Uso sencillo	x	x	x			x	x		
Bajos requerimientos	x		x						
Orientado a educación	x	x	x			x	x		x
Integración de actividades didácticas	x		x			x	x		
Diferentes robots	x	x	x	x	x		x	x	
Modificación de actuadores y sensor	x			x	x	x		x	
Ejecución en la nube	x		x	x	x		x	x	
Concurrencia de robots	x	x		x	x	x		x	x
Programación visual	x		x	x	x			x	
Programación textual	x		x	x	x	x	x	x	x

5. Esbozo de la metodología y resultados obtenidos hasta la fecha

La metodología seleccionada para el estudio consiste en los siguientes pasos:

- Desarrollo de un mapeo sistemático para el estado del arte.
- Selección de motor de física 3D para la elaboración del framework.
- Construcción de ambientes de simulación en el motor seleccionado.
- Ensamblaje con el entorno de programación visual y textual.
- Montaje de la plataforma en servidores.
- Testeo y evaluación con grupos experimentales y de control.

El mapeo sistemático ya fue realizado, obteniendo información sobre los autores, instituciones, congresos y journals más involucrados con la temática, y una lista de experiencias de robótica con simuladores en diversos niveles educativos con sus resultados y conclusiones. También se analizaron los diversos motores físicos utilizados en educación para la robótica simulada.

A partir de este estudio, y experimentando con los diferentes motores mencionados, se seleccionó a Webots como base para el desarrollo del framework. En colaboración con los institutos de educación técnica, se elaboró un simulador de Sumo robótico en Webots, y adaptamos otro de fútbol. Con ellos organizamos campeonatos virtuales locales y latinoamericanos, analizando mediante encuestas las ventajas y desventajas que encontraron los participantes en relación con la robótica física.

Actualmente se están ampliando los ambientes de simulación para una curva de aprendizaje más suave, trasladando a la virtualidad desafíos clásicos de la robótica educativa.

6. Trabajo futuro

En el primer semestre de 2023, se realizará el ensamble con el entorno de programación visual y textual ya desarrollado. Al mismo tiempo, se comenzará la experiencia en servidores, y el desarrollo de la interfaz web que permita la operación del ambiente de simulación. En forma paralela, se plantearán las características y metodología del estudio de campo que permitirá la comparación entre los logros obtenidos con el uso del simulador desarrollado en comparación con los materiales tradicionales.

En el segundo semestre de 2023 y el primer semestre de 2024 se llevará a cabo dicho estudio de campo con estudiantes de secundaria. Para ello, se dictará un curso cuyo objetivo es que aprendan conceptos básicos de robótica y programación de robots. Durante el estudio, el grupo experimental trabajará con la plataforma desarrollada, mientras que el grupo de control utilizará kits educativos de robótica física.

7. Referencias

- Benitti, F. B. V., & Spolaôr, N. (2017). How Have Robots Supported STEM Teaching? En M. S. Khine (Ed.), *Robotics in STEM Education: Redesigning the Learning Experience* (pp. 103–129). Springer International Publishing. https://doi.org/10.1007/978-3-319-57786-9_5
- Casañ Núñez, G. A., & Cervera, E. (2018). *The Experience of the Robot Programming Network Initiative*.
- Cervera, E., Casañ, G., & Tellez, R. (2017). Cloud Simulations for RoboCup. *Robot World Cup*, 180–189.
- Eguchi, A., & Shen, J. (2012). Student learning experience through CoSpace educational robotics. *Society for Information Technology & Teacher Education International Conference*, 19–24.
- Fagin, B. S., & Merkle, L. (2002). Quantitative analysis of the effects of robots on introductory Computer Science education. *Journal on Educational Resources in Computing*, 2(4), 2–es. <https://doi.org/10.1145/949257.949259>
- Fernandes, C. da C. (2013). *S-Educ: Um Simulador de Ambiente de Robótica Educacional em Plataforma Virtual*. <https://repositorio.ufrn.br/jspui/handle/123456789/15464>
- García-Peñalvo, F. J., Conde, M. Á., Gonçalves, J., & Lima, J. (2019). Computational thinking and robotics in education. *Proceedings of the Seventh International Conference on Technological Ecosystems for Enhancing Multiculturality*, 2–5.
- Guyot, L., Heiniger, N., Michel, O., & Rohrer, F. (2011). Teaching robotics with an open curriculum based on the e-puck robot, simulations and competitions. *Proceedings of the 2nd International Conference on Robotics in Education*. Vienna, Austria.
- Hughes, J. (2016). Robotic rescue simulation for computing teaching in the UK: A case study. *2016 IEEE Global Engineering Education Conference (EDUCON)*, 1051–

1055.

Hughes, J., Aghababa, F. P., Kabiri, A., Shimizu, M., & Visser, A. (2019). *A proposal on more advanced Robot Rescue Simulation challenges for Robotics Education*★.

Hughes, J., Shimizu, M., & Visser, A. (2019). A review of robot rescue simulation platforms for robotics education. *Robot World Cup*, 86–98.

Learn Robotics from Zero; Robotics & ROS Online Courses. (2020). The Construct. <https://www.theconstructsim.com/>

Lepuschitz, W., Merdan, M., Koppensteiner, G., Balogh, R., & Obdržálek, D. (2018). *Robotics in Education: Methods and Applications for Teaching and Learning* (Vol. 829). Springer.

Major, L. (2014). *An empirical investigation into the effectiveness of a robot simulator as a tool to support the learning of introductory programming*. [PhD Thesis]. Keele University.

Merdan, M., Lepuschitz, W., Koppensteiner, G., & Balogh, R. (2016). *Robotics in education: Research and practices for robotics in STEM education* (Vol. 457). Springer.

Miller, D. P., & Nourbakhsh, I. (2016). Robotics for education. En *Springer handbook of robotics* (pp. 2115–2134). Springer.

Núñez, G. A. C., & Mateu, E. C. (2015). RPN: Aprendizaje de la programación de robots mediante bloques en un entorno 3D. *ReVisión*, 8(3), 8.

Reich-Stiebert, N., & Eyssel, F. (2016). Robots in the classroom: What teachers think about teaching and learning with education robots. *International Conference on Social Robotics*, 671–680.

Robotbenchmark. (2020). <https://robotbenchmark.net/>

Sáez-López, J.-M., Román-González, M., & Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using “Scratch” in five schools. *Computers & Education*, 97, 129–141.

Theofanellis, T., Voulgari, E., & Tsolakis, S. (2020). Educational Robotics and Computational Thinking Development. En *Handbook of Research on Tools for Teaching Computational Thinking in P-12 Education* (pp. 306–334). IGI Global.

Tsukamoto, H., Takemura, Y., Oomori, Y., Ikeda, I., Nagumo, H., Monden, A., & Matsumoto, K. (2016). Textual vs. Visual programming languages in programming education for primary schoolchildren. *2016 IEEE Frontiers in Education Conference (FIE)*, 1–7.

Weintrop, D., & Wilensky, U. (2015). To block or not to block, that is the question: Students’ perceptions of blocks-based programming. *Proceedings of the 14th international conference on interaction design and children*, 199–208.