

# Estudio y Desarrollo de Herramientas sobre Probadores de Teoremas

Claudia I. Inchaurredo<sup>1,2,3</sup>, Germán Montejano<sup>1</sup>, Lucília Camarão de Figueiredo<sup>3</sup>

<sup>1</sup>Facultad de Ciencias Físico Matemáticas y Naturales – Universidad Nacional de San Luis (Argentina)

<sup>2</sup>Dpto. Sistemas – Facultad de Tecnología y Cs. Aplicadas – Universidad Nacional de Catamarca  
Maximio Victoria 55 - C.P: 4700 - San Fernando del Valle de Catamarca (Argentina)

TEL. 03833- 435112

<sup>3</sup>Departamento de Ciências da Computação - Universidade Federal de Minas Gerais (Belo Horizonte-Brasil)

c\_inchaurredo@sistemas.frc.utn.edu.ar , gmonte@unsl.edu.ar, lucilia@dcc.ufmg.br

## Resumen

El aumento en el uso de las herramientas para la especificación y verificación de programas dentro de la industria de hardware y de software estaría revelando en el ámbito de las carreras de grado universitarias la necesidad de preparar a los estudiantes en el campo de las tecnologías de la verificación.

Esta preparación comenzaría con el uso de herramientas en la enseñanza de matemáticas y de razonamiento lógico.

Existen varias herramientas disponibles para este fin que ya datan de varios años de implementadas y se mantienen en constante evolución.

Para el presente trabajo se han seleccionado los Probadores de Teoremas, ya que se presupone que pueden constituir una herramienta particularmente efectiva para la enseñanza de matemática y lógica en los cursos introductorios.

Los sistemas de pruebas de teoremas, tanto interactivos como automáticos, han resultado exitosos en un número considerable de aplicaciones en los últimos años. Ejemplos de ello son Coq, Isabelle, HOL y PVS.

Sin embargo, aún persisten las dificultades para que sean adoptados por la mayoría de la personas. Es por ello que se considera la necesidad de estudiar y desarrollar más aún Herramientas Integradas para la Enseñanza en Carreras de Informática a través de los Probadores de Teoremas.

**Palabras Clave:** aprendizaje, ingeniería de software, probadores de teoremas.

## Contexto

El presente trabajo corresponde a un

aspecto de una Tesis de Maestría en Ingeniería de Software de la Universidad Nacional de San Luis, Facultad de Ciencias Físico Matemáticas y Naturales, y soportado parcialmente por el Proyecto Binacional enmarcado en el programa CAFB-BA para el fortalecimiento de posgrados entre la Universidad Nacional de San Luis (Argentina) y la Universidade Federal de Minas Gerais (Brasil).

## Introducción

Los Probadores de Teoremas (PT) se han empleado en investigación y por los profesionales pero no es habitual su uso en el área de la enseñanza en las carreras de grado, ya que se considera que son sistemas difíciles de aprender. En las carreras de posgrado su aplicación es más frecuente pero aún así, resultan complejos [13]. Entre los motivos que provocan este hecho se encuentran:

- Tienen una cantidad de tácticas complejas, que si bien permiten desarrollar pruebas muy complicadas, las nociones que encierran son difíciles de entender y de aplicar.
- En la mayoría de los casos, la interfaz que poseen es sencilla de utilizar para los expertos en la realización de pruebas complejas. En cambio, para los estudiantes de las carreras de grado que recién comienzan a realizar pruebas de teoremas sencillos de lógica dicha interfaz resulta muy complicada.
- En general, no soportan un aprendizaje de pruebas de una manera paso a paso, en forma incremental. Cada segmento de la prueba debe ser completado para poder pasar al siguiente, hecho que dificulta el aprendizaje por parte de los estudiantes.

- Los mensajes de error proporcionan información dentro del contexto general de una prueba, pero esta información es poco precisa o eficaz dentro del contexto específico de la prueba que se está realizando.
- Relacionado con lo anterior, los errores no son informados en el momento en que se cometen. El estudiante continúa la prueba sobre la base de haber cometido un error.
- No están centrados en la interpretación computacional de la lógica o aplicaciones a los lenguajes de programación.
- Muchas veces contienen pruebas o búsquedas automáticas de pruebas que si bien constituyen una ayuda interesante, no permiten analizar los detalles de la prueba implicada.
- La codificación de las variables ligadas complica la meta-teoría implicada en el probador de teoremas y en comparación con su empleo en una resolución en papel.
- Pueden provocar ansiedad y frustración en los estudiantes la primera vez que entran en contacto con ellos.

Para la investigación se examinaron varios PT que se encontraban disponibles, teniendo en cuenta ciertos criterios para su selección y análisis (Epgy, ETPS, SASyLF entre otros). Luego, de acuerdo a criterios más específicos de selección se eligieron cuatro de esos PT: Jape, ADN, ProofWeb y Pandora. Para verificar sus ventajas y desventajas se realizaron pruebas concretas en cada uno de ellos. Al concluir las pruebas, se pudieron establecer algunas comparaciones. Como producto del análisis y reflexión del camino recorrido se arribó a algunas conclusiones.

### **Líneas de investigación y desarrollo**

Con el análisis de distintos PT como línea del trabajo, se pretendió abordar y desarrollar esencialmente:

- Una temática poco explorada en la enseñanza de lógica en las carreras de grado del área informática en nuestro país.

- Las características generales de los principales PT actuales en relación a su aplicación en carreras de grado.
- Las dificultades relacionadas con el aprendizaje y la utilización de los PT.
- La facilidad, utilidad y comodidad de su uso de cada uno de los probadores analizados.

Expresado de otra manera, se intentó concederles a los PT un espacio definido dentro de las asignaturas con contenidos de lógica.

### **Objetivos**

En el escenario descrito hasta aquí y con la intención de dar algunas respuestas a los interrogantes que han surgido en el mismo, se propusieron los siguientes objetivos para la investigación completa:

#### **Objetivo General**

- Determinar la factibilidad de la aplicación de los PT para la enseñanza de Lógica de Primer Orden en las carreras de grado del área informática.

#### **Objetivos Específicos**

- Buscar y conocer las características principales de los distintos tipos de PT.
- Analizar en profundidad aquellos PT que puedan resultar adecuados para la enseñanza de lógica en las carreras de grado.
- Proponer, si correspondiera, y con la debida fundamentación alguno de ellos para la enseñanza de lógica.

### **Resultados**

Cada PT tiene sus características particulares y han sido concebidos para lograr diferentes objetivos. Entre los motivos que llevaron a su selección, se pueden mencionar:

- Pertinencia con el tema de la investigación.

- Aparición o discusión del probador de teoremas en distintas publicaciones de relevancia.
- Acceso vía Internet al software del probador de teoremas en algunos casos.
- Acceso a la bibliografía de cada probador.

Todos los probadores que se han examinado han sido diseñados con algún objetivo didáctico o pedagógico.

### Formación de Recursos Humanos

Con este trabajo de Tesis se promueve una continuidad en la formación y fortalecimiento en investigación de su tesista con la guía de sus directores.

Dado su carácter de trabajo individual no se han incorporado becarios.

### Referencias

BRODA, K., MA, J., SINNADURAI, G., SUMMERS, A. (2007). Pandora: A Reasoning Toolbox using Natural Deduction Style. [Logic Journal of the IGPL, Volume 15, Number 4, Pp 293-304.](#)

JACOB, I. (2003). *Métodos formales en programación: ¿desmitificar para motivar?* En Actas de las IX Jornadas de Enseñanza Universitaria de la Informática (JENUI 2003). Ed. Thomson.

JAMNIK, M., KERBER, M., POLLET, M., BENZMUELLER, C. (2002). [LearnOmatic: System description.](#) In Voronkov, A., (ed.), 18th Conference on Automated Deduction, number 2392 in Lecture Notes in Artificial Intelligence, pages 150--155.

KALISZYK, C., WIEDIJK, F., HENDRIKS, M., van RAAMSDONK, F. (2008). *Teaching logic using a state-of-the-art proof assistant.* Proceedings of the Formal Methods in Computer Science Education 2008 workshop (FormEd'08).

Para la realización de pruebas se seleccionaron cuatro PT del grupo analizado. Uno de los motivos de esta selección fue la sencillez de la simbología y notación empleada en el probador de teoremas para el desarrollo de pruebas. Otro fue la rapidez y facilidad con la que un estudiante de un curso de lógica puede acceder al probador, conocer su manejo y utilizarlo. Sólo debe aprender unas pocas instrucciones o comandos.

LIANG, X., GE, F., YAN, L. (2007). *Some Logical Aspects of Mathematical Reasoning.* In R. Matuszewski and A. Zalewska, editors, [From Insight to Proof --- Festschrift in Honour of Andrzej Trybulec, Studies in Logic, Grammar, and Rhetoric 10\(23\), University of Bialystok.](#)

LLORENS, F., MIRA, S. (2000). *ADN (Asistente para Deducción Natural) (Natural Deduction Assistant)* First International Congress on Tools for Teaching Logic.

Página Web de Coq: <http://coq.inria.fr/>

Página Web de Isabelle: <http://www.cl.cam.ac.uk/research/hvg/Isabelle/>

Página Web de HOL: <http://hol.sourceforge.net/>

Página Web de PVS: <http://pvs.csl.sri.com/>

WING, J. (2000). *Weaving Formal Methods into the Undergraduate Computer Science Curriculum.* (Extended Abstract). In Proceedings of the 8th International Conference on Algebraic methodology and Software Technology (AMAST 2000)

YAKHNIS; R., FARRELL, J., SHULTZ, S. (1994). *Deriving programs using generic algorithms.* IBM Systems Journal, vol. 33, no. 1, pp. 158-181.