

DOCTORADOS DEL DCC

ALONSO GONZÁLEZ



El título de mi tesis es "Efficient Non-Interactive Zero-Knowledge Proofs". Las Zero-Knowledge proofs (ZKP) son protocolos que permiten demostrar que cierta proposición es verdadera sin revelar más información que el hecho que es verdadera. Un buen ejemplo de su utilidad puede ser el siguiente: una súper matemática, Peggy, demuestra que la Hipótesis de Riemann es verdadera y quiere contarle su hallazgo a su amigo Víctor, pero tiene miedo de que éste le robe la idea. Para eso hace una ZKP de la proposición "la Hipótesis de Riemann es verdadera" que tiene las siguientes garantías:

- a) Víctor sabe que con muy alta probabilidad Peggy no está mintiendo.
- b) Víctor sabe que la Hipótesis de Riemann es cierta, pero nada más. Es decir no sabe qué técnica ocupó Peggy para demostrarla, ni ninguna otra información no trivial.

Las ZKP han sido fundamentales en Criptografía y en otras áreas de ciencia de la computación.

Existen distintos tipos de ZKP, por ejemplo hay pruebas interactivas –donde Peggy y Víctor intercambian muchos mensajes– y no interactivas,

donde Peggy envía un único mensaje a Víctor. En mi tesis nos preocupamos de este tipo de ZKPs, que es muy útil para construir otros esquemas criptográficos inherentemente no interactivos (como firmas o encriptación).

Nuestros principales aportes son mejoras en la eficiencia de estas pruebas, enfocándonos principalmente en el tamaño de la prueba. Con estas mejoras logramos construir pruebas más eficientes de:

- **Correctitud de shuffles**, que son muy útiles en votación electrónica y en redes anónimas como ToR.
- **Range proofs**, también útiles en votación electrónica y comercio electrónico.
- **Firmas de anillo**, que permiten firmar a nombre de un grupo de usuarios sin revelar quién de los miembros del grupo fue el que firmó.

Antes de iniciar el doctorado obtuve la Ingeniería en Computación en el DCC de la Universidad de Chile, aunque también pasé por Ingeniería Eléctrica. De hecho, por esas cosas del destino empecé a tomar ramos de Computación, y cuando tomé Fundamentos de Ciencias de la Computación me di cuenta que esto era lo mío.

Después pasé casi dos años en el Centro de Modelamiento Matemático (CMM) trabajando en modelamiento geomecánico aplicado a minería. Aunque siempre mi objetivo era cursar el doctorado.

Hacer un doctorado es una experiencia difícil pero también bastante reconfortante. Lo más difícil es encontrar los problemas relevantes, lo que tiene que ver con entender qué es lo que le llama la atención a los científicos de tu área y qué es lo que te llama la atención a ti. Después viene la solución del problema en sí, que es cuando estás frente a la pizarra tratando de demostrar algo. Esta parte es difícil, pero súper entretenida. Lo mejor es cuando puedes gritar **¡Eureka!** Finalmente está el problema de cómo escribir un artículo, que es lo que más me cuesta. Es un desafío técnico –explicar de forma muy simple tus resultados– pero al mismo tiempo comunicacional, porque tienes que mostrar que tus resultados son útiles y relevantes.

Con mi profesor guía, Alejandro Hevia, tuvimos una relación bien horizontal y de buena onda. Con él empecé a trabajar en criptografía y aprendí lo fundamental. De todas formas, el doctorado tuvo sus partes difíciles y en algún momento incluso pensé en retirarme (aunque según me han contado, a todos les pasa). Durante estos momentos difíciles conocí a Carla Ràfols, mi coaguía, que por coincidencia estaba haciendo una visita en el CMM. Con Carla tuvimos afinidad desde el principio, y también nos entendimos bien a nivel técnico. Ella me dio el empujón que necesitaba, y con ayuda de Alejandro sacamos adelante el doctorado.

Actualmente me encuentro realizando un postdoc en el "Laboratoire de l'Informatique du Parallélisme" de la ENS de Lyon.

MAÍRA MARQUES



Llegué a Chile hace nueve años para volver a estudiar, para cambiar mi mirada de cómo funciona la computación en teoría, y cómo hacer mejor las cosas. En esos nueve años logré finalizar mi magíster y ahora mi doctorado en el DCC de la Universidad de Chile. Mi tema de magíster y doctorado tuvo que ver con ingeniería de software, en particular, cómo enseñarla de una manera más cercana a la realidad de esta industria, algo que tengo muy presente después de haber trabajado ocho años en ella.

Tradicionalmente, la ingeniería de software se ha enseñado con clases expositivas. Sin embargo, esta disciplina requiere mucho más que sólo teoría. Con el fin de tratar de entender el estado del arte en este ámbito, en mi tesis de doctorado realicé una extensa revisión bibliográfica. Además, se llevó a cabo un estudio de la enseñanza de ingeniería de software en las principales universidades de Chile. Los resultados obtenidos indican que hace poco tiempo que estas instituciones han comenzado a enseñar ingeniería de software de una manera teórico-práctica, involucrando a los estudiantes en experiencias de desarrollo de software ya sea en cursos

basados en proyectos, como en cursos prácticos de fin de carrera (*capstone*). Ambos tipos de cursos tienen objetivos distintos; los primeros son generalmente apoyados por procesos más rigurosos, mientras que los últimos son frecuentemente abordados con estrategias de desarrollo ágiles.

Se han propuesto varias estrategias de instrucción y de uso de procesos de ingeniería de software para cursos *capstone*, pero muy pocos están disponibles para cursos basados en proyectos. Además, aún los procesos más rigurosamente reportados en la literatura no incluyen suficiente detalle para que instructores y estudiantes involucrados puedan reproducirlos en cursos basados en proyectos. Es con el objetivo de llenar este vacío que mi tesis concibe y propone EduProcess: un proceso de desarrollo de software prescriptivo que puede ser utilizado en los cursos de ingeniería de software basados en proyectos que toman parte de programas de computación de pregrado.

Con relación al uso de este proceso, mi tesis de doctorado mostró que EduProcess permite que las experiencias prácticas pueden ser reproducidas sin demandar un esfuerzo adicional considerable, y además que permite ayudar a producir resultados positivos en proyectos de software. Todo eso con vista a que el método fuera usado para apoyar a pequeños equipos de software que trabajan de manera distribuida la mayor parte del tiempo y tienen instancias esporádicas de sincronización de sus trabajos. EduProcess incluye dos subprocesos: uno principalmente a cargo de los estudiantes (e involucra a los usuarios y clientes), y el otro a cargo del equipo instruccional del curso y sirve de apoyo a la experiencia de enseñanza-aprendizaje.

La validación del proceso se realizó con un estudio de caso, y los resultados obtenidos indican que EduProcess ayuda a aumentar la coordinación de los equipos, el sentido de pertenencia al equipo de los estudiantes y su efectividad, pero no necesariamente aumenta su productividad. Además, permite a los equipos hacer un diagnóstico más preciso de su proyecto y priorizar mejor las tareas de modo de maximizar su tasa de éxito. EduProcess hace una contribución al avance del estado del arte en el dominio de la educación de ingeniería de software, y específicamente en el proceso de enseñanza-aprendizaje de ingeniería de software en cursos basados en proyectos.

Durante el doctorado en el DCC tuve la oportunidad de participar en importantes conferencias internacionales de mi área: ICER, FIE, SIGCSE, ICSE y ITICSE, y además participar en conferencias de mujeres que trabajan en computación como Grace Hopper Conference y LATiNiTY. En SIGCSE pude representar al DCC en el Graduate Student Research Competition y gané el primer premio. El DCC me dio la oportunidad de conocer a las personas más importantes dentro de mi área, me mostró la cercanía que un profesor puede tener con sus alumnos, además de los importantes lazos de amistad que se pueden formar durante los años del doctorado. El DCC para mí es mi casa, es donde aprendí a mirar las cosas con otros ojos, con ojos de curiosidad, con ojos de querer saber más. En este momento estoy dictando clases para primer año en la Universidad de Chile,

la Universidad de O'Higgins, y la Universidad de los Andes. Me encanta dictar clases, pero la investigación de cómo podemos enseñar mejor computación y de cómo mejorar los procesos de desarrollo de software es lo que más me apasiona y lo que quiero seguir haciendo. En este

momento estoy buscando alguna parte donde pueda seguir haciendo ese trabajo, donde pueda contribuir con mi granito de arena a mejorar la enseñanza de computación en Chile y quién sabe en el mundo.

JUANPABLOSANDOVAL



Durante mi doctorado me dediqué a investigar a fondo los cambios de código fuente que afectan el rendimiento de un programa. Hoy en día, los desarrolladores de software están sacando constantemente nuevas versiones de sus programas. Sin embargo, los cambios que realizan en estas nuevas versiones pueden afectar negativamente su rendimiento. Por ejemplo, una nueva versión puede ser mucho más lenta que versiones anteriores. Probar si una nueva versión afectará el rendimiento es una tarea muy costosa que normalmente requiere ejecutar varios puntos de referencia (lo cual puede tardar horas, incluso días).

Junto a mi profesor guía Alexandre Bergel, realizamos un estudio empírico sobre una variedad de programas, con el fin de evaluar cómo el rendimiento de un programa evoluciona en el tiempo, a medida que es modificado. Guiados por este estudio, propusimos "Horizontal Profiling", una técnica de muestreo que permite inferir si una nueva versión afectará negativamente el rendimiento del programa. El objetivo de Horizontal Profiling es reducir la sobrecarga que requiere monitorear el rendimiento de cada versión, ejecutando los puntos de referencia solo en las versiones que contengan cambios costosos de código fuente.

Determinar cuándo una versión podría afectar el rendimiento sin ejecutar el programa, no fue una tarea fácil. Pero después de muchos intentos pudimos diseñar un modelo de costo adecuado para predecir una posible falla en el rendimiento. Gracias a "Horizontal Profiling" los desarrolladores pueden identificar una gran porción de las regresiones de rendimiento, sin tener que ejecutar costosos puntos de referencia en todas las versiones.

Mi experiencia en el DCC, fue muy buena. Estoy muy agradecido con el DCC, la Universidad de Chile, CONICYT y el Gobierno chileno por darme la oportunidad de realizar mis estudios de doctorado. Todo este tiempo he sentido el apoyo de todos: los compañeros de clase, los profesores, las diferentes autoridades del Departamento de Ciencias de la Computación y la Facultad. Siento que el DCC no solo me ayudó a crecer académicamente sino también como persona. También pude disfrutar de muchas actividades extracurriculares que me permitieron aprender mucho de la cultura chilena.

Después de finalizar mi doctorado, realicé un postdoc en LAM-Research, California, Estados Unidos. Donde fui invitado principalmente para evaluar las técnicas y herramientas que desarrollé durante mi doctorado en un entorno industrial. Esto representó una excelente oportunidad para aplicar los resultados de investigación en una empresa donde el rendimiento y la calidad del software son muy importantes.

Actualmente, estoy dictando clases en la Universidad Mayor de San Simón, Bolivia, donde realice mis estudios de pregrado. Así mismo, continúo trabajando en varios proyectos de investigación con mi profesor guía Alexandre Bergel. Por otro lado, junto con una fundación sin fines de lucro, también me dedico a dar clases de robótica y computación a estudiantes de colegio. Mi objetivo siempre fue volver a mi país Bolivia y poder contribuir en diferentes áreas, sobre todo en investigación. ■